

# A Teleoperation System with Impedance Control and Disturbance Observer for Robot-Assisted Rehabilitation

Teng Li, *Graduate Student Member, IEEE*

**Abstract**—Physical movement therapy is a crucial method of rehabilitation aimed at reinstating mobility among patients facing motor dysfunction due to neurological conditions or accidents. Such therapy is usually featured as patient-specific, repetitive, and labor-intensive. The conventional method, where therapists collaborate with patients to conduct repetitive physical training, proves strenuous due to these characteristics. The concept of robot-assisted rehabilitation, assisting therapists with robotic systems, has gained substantial popularity. However, building such systems presents challenges, such as diverse task demands, uncertainties in dynamic models, and safety issues. To address these concerns, in this paper, we proposed a bilateral teleoperation system for rehabilitation. The control scheme of the system is designed as an integrated framework of impedance control and disturbance observer where the former can ensure compliant human-robot interaction without the need for force sensors while the latter can compensate for dynamic uncertainties when only a roughly identified dynamic model is available. Furthermore, the scheme allows free switching between tracking tasks and physical human-robot interaction (*pHRI*). The presented system can execute a wide array of pre-defined trajectories with varying patterns, adaptable to diverse needs. Moreover, the system can capture therapists' demonstrations, replaying them as many times as necessary. The effectiveness of the teleoperation system is experimentally evaluated and demonstrated.

**Index Terms**—Teleoperation, robot-assisted rehabilitation, movement therapy, impedance control, disturbance observer.

## I. INTRODUCTION

Paralysis is a side effect of stroke that requires physical therapy for rehabilitation, without which the stroke survivor may lose motor control on some body parts in the long term. Patients with paralysis suffer from an inability to move some body parts, reduced mobility, restricted range of motion, and muscle stiffness [1]. During conventional physical therapy, the therapist needs to help the patient to move the affected body part, *e.g.*, arm, repetitively to regain mobility [1]. This repetitive movement therapy training is labor-intensive and strenuous for the therapist but necessary for the patient. Beyond stroke, patients with motor dysfunction due to other

This research is supported in part by the Canada Foundation for Innovation (CFI) under grants LOF 28241 and JELF 35916, in part by the Government of Alberta under grants IAE RCP-12-021 and EDT RCP-17-019-SEG, in part by the Government of Alberta's grant to Centre for Autonomous Systems in Strengthening Future Communities (RCP-19-001-MIF), in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada under grants RTI-2018-00681, RGPIN-2019-04662, and RGPAS-2019-00106, and in part by the Edmonton Civic Employee Charitable Assistance Fund.

T. Li is with the Department of Electrical and Computer Engineering, Faculty of Engineering, University of Alberta, Edmonton T6G 1H9, Alberta, Canada. E-mail: teng4@ualberta.ca

conditions (*e.g.*, cerebral palsy, Parkinson's disease) or injuries on body parts (*e.g.*, fall, car accident), may also need physical therapy which largely increases its demand. Robot-assisted rehabilitation is a promising solution where a robot can be deployed to assist the therapist in helping the patient perform movement therapy practices repetitively with a certain degree of accuracy [1], [2].

There is an abundance of work that has been done for robot-assisted rehabilitation over the past three decades while many of them involved only one robot manipulator [3]–[6]. Some new techniques can be integrated into the system to enhance its capability. For example, Ocampo and Tavakoli built a visual-haptic colocation system for robot-assisted rehabilitation exercises by involving a 2D augmented reality (AR) display [2]. By providing spatial AR visual feedback together with haptic feedback, the system enabled better task performance in terms of task completion time.

Learning from demonstration (LfD) is a popular option for the robot to learn and help with repetitive rehabilitative and assistive practices. Najafi *et al.* proposed a framework using a potential field function with a velocity field controller to learn and produce the therapist's assistance in robot-assisted rehabilitation where the therapist just needs to demonstrate once [1]. The potential field function is used to model and reproduce the therapist's motion and assistive force, while the velocity field controller is designed to compensate for and regulate the patient's deviation.

In recent years, teleoperation systems have gained increasing interest due to the additional advantages of allowing in-home environment therapy and avoiding physical contact for better hygienic conditions. Sharifi *et al.* proposed an approach for therapist-patient collaboration in a telerobotic system with time delays [7]. The approach refers to a bilateral impedance control strategy that can filter out the patient's involuntary hand tremor. Their telerobotic system allows the patients to move their limbs on the master robot side while being assisted as much as needed by the online assistive force provided by the therapist on the second robot side. Later, Sharifi *et al.* employed a similar strategy for assist-as-needed tele-rehabilitation [6]. With this strategy, the system can minimize the therapist's movements while providing a better perception to the therapist about the patient's issues and recovery status in motor control ability. As an additional benefit of the strategy, the patient can receive assistance from two sources, *i.e.*, the adjustable impedance model, and the exerted force by the therapist.

Safety and compliance are the paramount concerns when developing a robotic system for rehabilitation, either for collaboration or teleoperation. Impedance control is a commonly adopted control scheme considering that it can provide compliant robot behavior thus ensuring safety. Impedance control has its intrinsic property of compliance which makes it more suitable for human-robot interaction [8], [9]. A robot with an impedance controller can be programmed to be soft (compliant) or rigid (non-compliant) as necessary, which ensures a safe human-robot interaction [10]. Moreover, the measurement of the interaction force is not needed for impedance control. In brief, the sensor-free and compliance properties make impedance control a popular choice for human-robot interaction scenarios. Another common problem in building such a robotic system is that the dynamics of the robot might be unknown or only a roughly identified dynamic model be available. An integrated framework of combining impedance control with disturbance observer has been proposed in our previous work [9]. By involving a disturbance observer to compensate for the modeling error and other uncertainties, an accurate impedance control can be achieved even with an inaccurate dynamic model.

Due to the diverse motor abilities among the patients, each individual patient requires individualized physical therapy for rehabilitation. Even for the same patient, different therapy programs are also required at different rehabilitation phases. For example, a patient may not be able to move the limb at all at the very beginning, which means that passive movement is needed with the help of the therapist. Gradually, the patient can voluntarily move the limb in a small area, and in this phase, assist-as-needed rehabilitation may be a better option [6]. When the patient is able to move the limb around relatively freely, more complicated moving trajectories or tasks can be designed for the patient to further enhance the rehabilitation.

To better meet these requirements, a bilateral teleoperation system is proposed for robot-assisted rehabilitation in this paper, which integrates several features, including pre-defined trajectory tracking, LfD, and force feedback, into the same teleoperation system under the same control architecture. The designed control architecture, a main contribution of this work, enables free switching between pre-defined trajectory tracking mode and physical human-robot interaction (*pHRI*) mode. With the control architecture, the proposed teleoperation system has the following detailed features,

- (1) Force feedback is provided on the master robot (therapist) side. With this feature, the therapist can better understand the status of the patient on the second robot side.
- (2) A large variety of pre-defined trajectories with different patterns can be programmed thus the therapist can select the ones that are most suitable to individual patients.
- (3) The therapist on the master robot side can design customized trajectories for individual patients as needed.
- (4) Similar to the concept of LfD, the customized trajectory demonstrated by the therapist can be recorded, and then it can be replayed by the system on its own as many times as needed without losing reproduction fidelity.

The remainder of this paper is organized as follows: Section

II is devoted to introducing the general methods including dynamics, impedance control, and disturbance observer. Section III presents the details of the proposed teleoperation system, including the master/second robot kinematics, dynamics, and force feedback rendering. Section IV presents experiments and corresponding results for evaluating the system. Section VI gives the concluding remarks.

## II. METHODS

### A. Robot Dynamics and Impedance Control

A general dynamic model for an  $n$ -degree-of-freedom (DOF) rigid robot [11] can be given by

$$\underbrace{\mathbf{M}(\mathbf{q})}_{\hat{\mathbf{M}}+\Delta\mathbf{M}} \ddot{\mathbf{q}} + \underbrace{\mathbf{S}(\mathbf{q}, \dot{\mathbf{q}})}_{\hat{\mathbf{S}}+\Delta\mathbf{S}} \dot{\mathbf{q}} + \underbrace{\mathbf{g}(\mathbf{q})}_{\hat{\mathbf{g}}+\Delta\mathbf{g}} + \boldsymbol{\tau}_{\text{fric}}(\dot{\mathbf{q}}) = \boldsymbol{\tau} + \underbrace{\boldsymbol{\tau}_{\text{ext}}}_{\mathbf{J}^T \mathbf{F}_{\text{ext}}} \quad (1)$$

where  $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$  are the actual joint angle, velocity, and acceleration, respectively,  $\mathbf{M} \in \mathbb{R}^{n \times n}$  denotes the inherent inertia matrix,  $\mathbf{S} \in \mathbb{R}^{n \times n}$  denotes a matrix of the Coriolis and centrifugal forces,  $\mathbf{g} \in \mathbb{R}^n$  represents the gravity vector.  $\hat{\mathbf{M}}, \hat{\mathbf{S}}, \hat{\mathbf{g}}$  represent their estimates, while  $\Delta\mathbf{M}, \Delta\mathbf{S}, \Delta\mathbf{g}$  are the corresponding estimated errors.  $\boldsymbol{\tau}_{\text{fric}} \in \mathbb{R}^n$  is joint friction,  $\boldsymbol{\tau} \in \mathbb{R}^n$  is the commanded joint torque vector,  $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^n$  is the torque caused by the external force,  $\mathbf{F}_{\text{ext}} \in \mathbb{R}^6$  is the external force in Cartesian space.

A desired impedance model [8]–[10] for robot-environment interaction can be expressed as

$$\mathbf{F}_{\text{imp}} = \mathbf{M}_{\text{imp}}(\ddot{\mathbf{x}} - \ddot{\mathbf{x}}_d) + (\mathbf{S}_x + \mathbf{D}_{\text{imp}})(\dot{\mathbf{x}} - \dot{\mathbf{x}}_d) + \mathbf{K}_{\text{imp}}(\mathbf{x} - \mathbf{x}_d) \quad (2)$$

where  $\mathbf{M}_{\text{imp}}, \mathbf{D}_{\text{imp}}, \mathbf{K}_{\text{imp}}$  are user-designed matrices for inertia, damping, and stiffness, respectively. Note that  $\mathbf{x}_d, \dot{\mathbf{x}}_d, \ddot{\mathbf{x}}_d$  are the desired position, velocity, and acceleration, respectively in Cartesian space, while  $\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}}$  are the actual ones.  $\mathbf{S}_x$  is the Coriolis and centrifugal matrix of the robot in Cartesian space and  $\mathbf{S}_x = \mathbf{J}^{-T} \mathbf{S} \mathbf{J}^{-1} - \mathbf{M}_x \dot{\mathbf{J}} \mathbf{J}^{-1}$ , where  $\mathbf{M}_x = \mathbf{J}^{-T} \mathbf{M} \mathbf{J}^{-1}$  is the inherent inertia of the robot in Cartesian space [12].

To avoid the measurement of external forces [9], the designed inertia matrix can be set as the inherent inertia matrix of the robot, *i.e.*,  $\mathbf{M}_{\text{imp}} = \mathbf{M}_x$ . Then, to reach (2) as the closed-loop dynamics governing the robot-environment interaction in an ideal scenario of no model errors and no joint friction, setting with  $\mathbf{F}_{\text{imp}} = \mathbf{F}_{\text{ext}}$ , the impedance control law can be given by

$$\boldsymbol{\tau} = \mathbf{M} \mathbf{J}^{-1}(\ddot{\mathbf{x}}_d - \mathbf{J} \mathbf{J}^{-1} \dot{\mathbf{x}}_d) + \mathbf{S} \mathbf{J}^{-1} \dot{\mathbf{x}}_d + \mathbf{g} + \mathbf{J}^T [\mathbf{D}_{\text{imp}}(\dot{\mathbf{x}}_d - \dot{\mathbf{x}}) + \mathbf{K}_{\text{imp}}(\mathbf{x}_d - \mathbf{x})] \quad (3)$$

Note that when implementing the impedance controller (3) in practice, the estimates  $\hat{\mathbf{M}}, \hat{\mathbf{S}}, \hat{\mathbf{g}}$  will be used for the calculation since an accurate model of a physical robot is usually not available.

For robot end-effector (EE) moving to a fix point, *i.e.*, set-point regulation, it has  $\ddot{\mathbf{x}}_d = \mathbf{0}$ ,  $\dot{\mathbf{x}}_d = \mathbf{0}$ . Then, the impedance control law (3) can be simplified to (4), which is also known as task-space proportional-derivative (PD) controller with gravity compensation.

$$\boldsymbol{\tau} = \mathbf{J}^T [\mathbf{K}_{\text{imp}}(\mathbf{x}_d - \mathbf{x}) - \mathbf{D}_{\text{imp}} \dot{\mathbf{x}}] + \mathbf{g} \quad (4)$$

Based on (4), when the fixed point is set to be time-varying points of the real-time Cartesian position, *i.e.*,  $\mathbf{x}_d = \mathbf{x}$ , it can be further reduced to be

$$\tau = -\mathbf{J}^T \mathbf{D}_{imp} \dot{\mathbf{x}} + \mathbf{g} \quad (5)$$

in which mode, physical human-robot interaction (*pHRI*) is enabled, and the user can move the robot EE freely in its workspace. Note that in this *pHRI* mode, it is possible to involve force feedback if needed, then the controller with force feedback will be given by

$$\tau = -\mathbf{J}^T \mathbf{D}_{imp} \dot{\mathbf{x}} + \mathbf{g} + \tau_{ff} \quad (6)$$

where the  $\tau_{ff}$  is the rendered force feedback which will be introduced in detail later. It is worth mentioning that for a backdrivable haptic device, it is defaulted in *pHRI* mode without needing a controller like (5), and the operator feels the intrinsic inertia of the robot mechanism. In other words, only the force feedback term  $\tau_{ff}$  in (6) is needed for a haptic device if force feedback rendering is needed, while the damper term is applicable but not necessary. If the operator implements the whole controller (6) on a haptic device for *pHRI* with force feedback rendering, the operator will feel an extra damper force (due to the damper term) in addition to the intrinsic inertia from the robot mechanism, but the damper force is usually small since robot EE held by the operator is usually in a low-speed movement. The advantage of the controller (6) is that it is a general form for robotic control thus applicable to all robotic systems other than haptic devices. Furthermore, it can realize a seamless transition from trajectory-tracking tasks to *pHRI* due to the fact that (5) (*pHRI* mode) is automatically reduced from (3) (trajectory-tracking mode) under some *pHRI*-specific conditions.

For the teleoperation system proposed in this paper, as illustrated in Fig.1, both the master robot and the second robot are implemented with an impedance controller based on (3) which are expressed as

$$\begin{cases} \tau_{M,imp} \\ = \hat{\mathbf{M}} \mathbf{J}^{-1} (\ddot{\mathbf{x}}_{M,d} - \hat{\mathbf{J}} \mathbf{J}^{-1} \dot{\mathbf{x}}_{M,d}) + \hat{\mathbf{S}} \mathbf{J}^{-1} \dot{\mathbf{x}}_{M,d} + \hat{\mathbf{g}}_M + \tau_{ff} \\ + \mathbf{J}^T [\mathbf{D}_{M,imp} (\dot{\mathbf{x}}_{M,d} - \dot{\mathbf{x}}_M) + \mathbf{K}_{M,imp} (\mathbf{x}_{M,d} - \mathbf{x}_M)] \\ \tau_{S,imp} \\ = \hat{\mathbf{M}} \mathbf{J}^{-1} (\ddot{\mathbf{x}}_{S,d} - \hat{\mathbf{J}} \mathbf{J}^{-1} \dot{\mathbf{x}}_{S,d}) + \hat{\mathbf{S}} \mathbf{J}^{-1} \dot{\mathbf{x}}_{S,d} + \hat{\mathbf{g}}_S \\ + \mathbf{J}^T [\mathbf{D}_{S,imp} (\dot{\mathbf{x}}_{S,d} - \dot{\mathbf{x}}_S) + \mathbf{K}_{S,imp} (\mathbf{x}_{S,d} - \mathbf{x}_S)] \end{cases} \quad (7)$$

where the subscript  $M$  and  $S$  stand for “Master” and “Second”, respectively, and subscript  $imp$  stands for “impedance”, and subscript  $d$  stands for “desired”. It should be noted that the force feedback  $\tau_{ff}$  is only set for the master robot in *pHRI* mode.

## B. Force Feedback and Rendering

The proposed teleoperation system, as shown in Fig.1, is set as bilateral, which means that force feedback is rendered and delivered to the operator on the master robot side when the second robot interacts with the surrounding environment. In this work, a virtual spring force will be rendered as the force feedback based on position error between the two robots. To

this end, the real-time position of the second robot is sent back to the master robot via user datagram protocol (UDP) for force rendering. The position error between the master robot EE and the second robot EE is calculated for generating a virtual spring force which can be expressed as

$$\tau_{ff} = \mathbf{J}^T \mathbf{F}_{ff} = \mathbf{J}^T \mathbf{K}_{ff} (\mathbf{x}_S - \mathbf{x}_M) \quad (8)$$

where  $(\mathbf{x}_S - \mathbf{x}_M)$  is the position error between the master robot EE and the second robot EE, and  $\mathbf{K}_{ff}$  is the virtual spring stiffness coefficient which can be tuned to make the virtual spring stiffer or softer.

## C. Disturbance and Disturbance Observer

By collecting all the disturbances together, the dynamic model (1) of a robot without considering force feedback ( $\tau_{ff} = 0$ ) can be re-written as

$$\hat{\mathbf{M}} \ddot{\mathbf{q}} + \hat{\mathbf{S}} \dot{\mathbf{q}} + \hat{\mathbf{g}} = \tau + \underbrace{[\tau_{fric} + (\Delta \mathbf{M} \ddot{\mathbf{q}} + \Delta \mathbf{S} \dot{\mathbf{q}} + \Delta \mathbf{g})]}_{\tau_{disturb}} \quad (9)$$

where  $\tau_{disturb}$  denotes the lumped uncertainties that usually include three main aspects, *i.e.*, the model error ( $\Delta \mathbf{M} \ddot{\mathbf{q}} + \Delta \mathbf{S} \dot{\mathbf{q}} + \Delta \mathbf{g}$ ), the joint friction  $\tau_{fric}$ , and the external disturbances  $\tau_{ext}$ , where the last aspect may involve constant disturbance and/or time-varying disturbance. The constant disturbance may be a constant payload attached to the robot end-effector (EE) or body, while time-varying disturbance may be robot-environment interaction forces such as human-applied forces during human-robot interaction. A disturbance observer usually estimates the lumped uncertainties  $\tau_{disturb}$  [13], but cannot discriminate any single component when more than one component exists.

The main uncertainties of the teleoperation system (in free motion mode) constructed in this paper will include the model error and the inaccurate joint friction which will be estimated and compensated for by using a disturbance observer. A variety of observers are available for use [14]–[17], such as generalized momentum observer (GMO) [17], extended state observer (ESO) [18], nonlinear disturbance observer (NDOB) [13], and disturbance Kalman filter (DKF) method [19], [20].

In this paper, a nonlinear disturbance observer (NDOB) is employed considering that the NDOB has the advantage of estimating the nonlinearities in the dynamics. An adapted NDOB design based on [13] can be expressed as

$$\begin{cases} \mathbf{L}_{obs} = \mathbf{Y}_{obs} \mathbf{M}_{obs}^{-1} \\ \mathbf{p} = \mathbf{Y}_{obs} \dot{\mathbf{q}} \\ \dot{\mathbf{z}} = -\mathbf{L}_{obs} \mathbf{z} + \mathbf{L}_{obs} (\hat{\mathbf{S}} \dot{\mathbf{q}} + \hat{\mathbf{g}} - \tau - \mathbf{p}) \\ \tau_{NDOB} = \mathbf{z} + \mathbf{p} \end{cases} \quad (10)$$

where  $\mathbf{L}_{obs} \in \mathbb{R}^{n \times n}$  is the observer gain matrix,  $\mathbf{Y}_{obs} \in \mathbb{R}^{n \times n}$  is a constant invertible matrix needs to be designed,  $\mathbf{M}_{obs}$  is designed to be a symmetric and positive definite matrix in practice and thus invertible,  $\mathbf{z}$  is an auxiliary variable,  $\mathbf{p}$  is an auxiliary vector determined from  $\mathbf{Y}_{obs}$ ,  $\tau_{NDOB}$  is the estimated lumped uncertainties via the NDOB observer. For the full designing procedures of this NDOB and complete theoretical analysis, refer to [13].

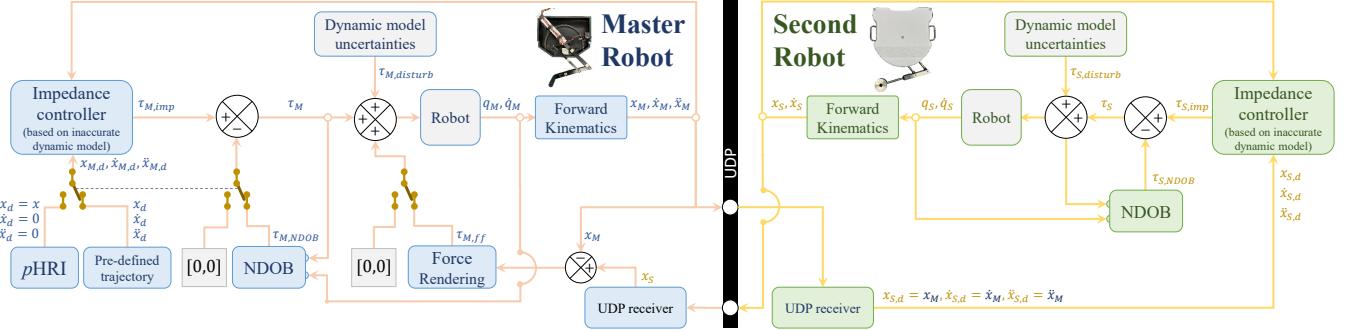


Fig. 1: Block diagram of control schemes for the proposed teleoperation system. The dashed line means a linked switch. The output of NDOB  $\tau_{M,NDOB}$  and  $\tau_{S,NDOB}$  are the estimation of the lumped uncertainties  $\tau_{M,disturb}$  and  $\tau_{S,disturb}$ , respectively.

A teleoperation system integrating impedance control with NDOB is proposed for robot-assisted rehabilitation. The block diagram of the proposed teleoperation system is illustrated in Fig.1.

### III. ROBOTIC SYSTEMS FOR TELEOPERATION

The experimental setup is shown in Fig. 2. A 2-DOF planar upper-limb rehabilitation robot 1.0 (in black color, Quanser Inc., Markham, ON, Canada), and a 2-DOF planar upper-limb rehabilitation robot 2.0 (in white color, Quanser Inc., Markham, ON, Canada) are used for constructing the teleoperation system. The Rehab robot 1.0 (black) is used as the master robot while the Rehab robot 2.0 (white) is used as the second robot. To build virtual models of the two robots, the kinematic model and dynamic model of the two robots are reconstructed based on [21]–[23]. It should be noted that the Rehab robot 2.0 (white) is an upgraded version of the Rehab robot 1.0 (black), so they have the same form of kinematics and dynamics (with different values for the parameters). All the experiments are conducted using MATLAB/Simulink (version R2017a, MathWorks Inc., Natick, MA, USA) with Quarc real-time control software (Quanser Inc., Markham, ON, Canada), which is running on two computers with a 3.33 GHz Intel(R) Core(TM) 2 i5 CPU and a Windows 7 Enterprise 64-bit operating system. The control rate of the robots is 1,000 Hz, while the sampling rate for acquiring data is set as 500 Hz. The bilateral communications between the two robots are realized by UDP at a rate of 1,000 Hz.

#### A. Robot Kinematics

A thorough exploration of the kinematics and dynamics of the 2-DOF rehabilitation robot 1.0 (black) can be found in [22], [23]. Additionally, a Denavit–Hartenberg (DH) table for this robot is provided in Table I. The corresponding frame definitions are illustrated in Fig. 2. Note that the DH table and frame definitions are derived based on the Rehab robot 1.0 (black), but they are also applied to the Rehab robot 2.0 (white).

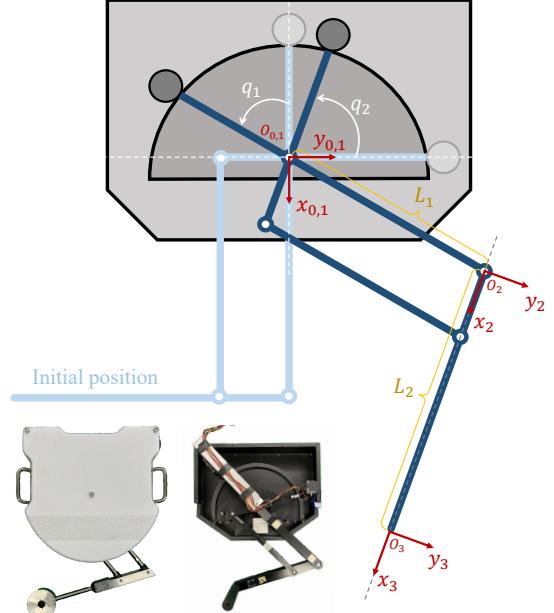


Fig. 2: Schematic of the 2-DOF planar upper-limb rehabilitation robot (black, white) and frame attachment to each joint. Frame  $\{0\}$  is the base frame while frame  $\{3\}$  is the end-effector (EE) frame.  $L_1, L_2$  are link lengths.  $q_1, q_2$  are joint angle variables.

TABLE I: Denavit–Hartenberg (DH) parameters for the 2-DOF Quanser Rehab robot 1.0 (black) and Rehab robot 2.0 (white) kinematic chain (for the homogeneous transform in the modified convention).

no.	Joint	$a(m)$	$\alpha(rad)$	$d(m)$	$\theta(rad)$
1	Joint 1	0	0	0	$q_1$
2	Joint 2	$L_1$	0	0	$-\frac{\pi}{2} - q_1 + q_2$
3	Joint 3 (EE)	$L_2$	0	0	0

According to the DH parameters in Table I and the frames determined in Fig. 2, the homogeneous transformation matrix  $\mathbf{T}$  from EE frame {3} to base frame {0} can be obtained as

$$\mathbf{T} = \begin{bmatrix} \sin(q_2), \cos(q_2), 0, L_1 \cos(q_1) + L_2 \sin(q_2) \\ -\cos(q_2), \sin(q_2), 0, L_1 \sin(q_1) - L_2 \cos(q_2) \\ 0, 0, 1, 0 \\ 0, 0, 0, 1 \end{bmatrix} \quad (11)$$

where  $q_1, q_2$  are joint angles,  $L_1, L_2$  are link lengths.

The Jacobian matrix  $\mathbf{J}$  can be expressed [22], [23] by

$$\mathbf{J} = \begin{bmatrix} -L_1 \sin(q_1), L_2 \cos(q_2) \\ L_1 \cos(q_1), L_2 \sin(q_2) \end{bmatrix} \quad (12)$$

where  $q_1, q_2$  are joint angles,  $L_1, L_2$  are link lengths.

Based on the transformation matrix (11), the forward kinematics [22], [23] can be written as

$$\begin{cases} x = L_1 \cos(q_1) + L_2 \sin(q_2) \\ y = L_1 \sin(q_1) - L_2 \cos(q_2) \end{cases} \quad (13)$$

For the Rehab robot 1.0 (black), the link lengths are  $L_1 = 0.254$  m,  $L_2 = 0.2667$  m. The joint limits restricted by the mechanical structure [22] are

$$\begin{cases} -55^\circ \leq q_1 \leq 90^\circ \\ 0^\circ \leq q_2 \leq 145^\circ \\ 35^\circ \leq q_1 - q_2 + 90^\circ \leq 145^\circ \end{cases} \quad (14)$$

For the Rehab robot 2.0 (white), the link lengths are  $L_1 = 0.340$  m,  $L_2 = 0.375$  m. The joint limits restricted by the mechanical structure are

$$\begin{cases} -86^\circ \leq q_1 \leq 132^\circ \\ -49^\circ \leq q_2 \leq 154^\circ \\ 35^\circ \leq q_1 - q_2 + 90^\circ \leq 145^\circ \end{cases} \quad (15)$$

The inverse kinematics is obtained [22] as

$$\begin{cases} q_1 = \text{acos}\left(\frac{x^2+y^2+L_1^2-L_2^2}{2L_1\sqrt{x^2+y^2}}\right) + \text{atan2}(y, x) \\ q_2 = q_1 + \text{acos}\left(\frac{L_1^2+L_2^2-x^2-y^2}{2L_1L_2}\right) - 90^\circ \end{cases} \quad (16)$$

The workspace of the master robot and the second robot are illustrated in Fig. 3. Note that the workspace of the master robot (the red color area) is shifted +0.1 m along the  $x$ -axis for a better overlap with the workspace of the second robot.

### B. Robot Identified-Dynamics

According to [22], the identified inertia matrix for the Rehab robot 1.0 (black) is given by

$$\hat{\mathbf{M}} = \begin{bmatrix} \alpha_1, -\frac{1}{2}\alpha_2 \sin(q_1 - q_2) \\ -\frac{1}{2}\alpha_2 \sin(q_1 - q_2), \alpha_3 \end{bmatrix} \quad (17)$$

where  $\alpha_1=0.06929$ ,  $\alpha_2=0.04217$ ,  $\alpha_3=0.04416$  are experimentally identified dynamic coefficients in the inertia matrix.

The identified Coriolis and centrifugal forces related matrix for the Rehab robot 1.0 (black) [22] is given by

$$\hat{\mathbf{S}} = \begin{bmatrix} 0, \frac{1}{2}\alpha_2 \cos(q_1 - q_2)\dot{q}_2 \\ \frac{1}{2}\alpha_2 \cos(q_1 - q_2)\dot{q}_1, 0 \end{bmatrix} \quad (18)$$

where  $\alpha_2=0.04217$  is experimentally identified.

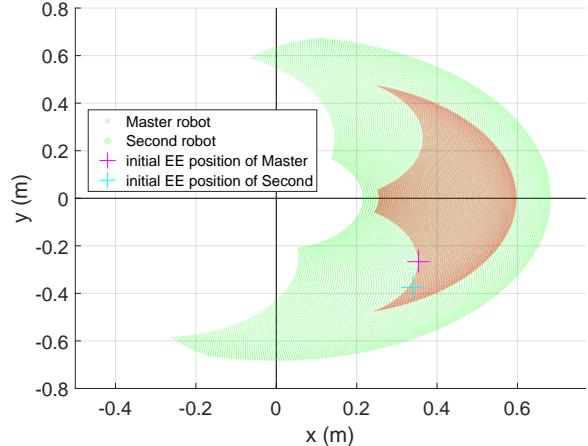


Fig. 3: Workspace of the master robot and the second robot.

TABLE II: Parameterization for experiments with the teleoperation system for robot-assisted rehabilitation.

Parameters	Master Robot	Second Robot	Location
Link length	$[0.254, 0.2667]$	$[0.340, 0.375]$	Eqn.(11,12,13,16)
Spring	$\mathbf{K}_{\text{imp}} = 30\mathbf{I}$	$\mathbf{K}_{\text{imp}} = 20\mathbf{I}$	Eqn.(7)
Damping	$\mathbf{D}_{\text{imp}} = 2\sqrt{30}\mathbf{I}$	$\mathbf{D}_{\text{imp}} = 2\sqrt{20}\mathbf{I}$	Eqn.(7)
Inertia matrix	$\mathbf{M}_{\text{obs}} = \hat{\mathbf{M}}$	$\mathbf{M}_{\text{obs}} = 0.001\mathbf{I}$	Eqn.(10)
Observer gain	$\mathbf{Y}_{\text{obs}} = 1.92\mathbf{I}$	$\mathbf{Y}_{\text{obs}} = 0.048\mathbf{I}$	Eqn.(10)

Note:  $\mathbf{I} \in \mathbb{R}^{2 \times 2}$  denote identity matrix.

As the Rehab robot 1.0 (black) is a planar robot and its links move in the horizontal plane, the gravity-related vector [22] is simply given by

$$\hat{\mathbf{G}} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The identified joint friction model for the Rehab robot 1.0 (black) [22] is given by

$$\hat{\boldsymbol{\tau}}_{\text{fric}} = \begin{bmatrix} \alpha_4 \dot{q}_1 \\ \alpha_5 \dot{q}_2 \end{bmatrix} \quad (19)$$

where  $\alpha_4=0.06510$ ,  $\alpha_5=0.07389$  are experimentally identified.

For the Rehab robot 2.0 (white), the same set of the identified dynamics model presented above is employed. Although the actual identified dynamic parameters of the Rehab robot 2.0 (white) would be quite different from the black one, it is possible to use the same dynamic model since a disturbance observer is employed to compensate for the dynamic uncertainties.

### C. Parameterization

For all experiments in the remaining part of this paper, the parameter values used in the impedance model and NDOB are listed in Table II. Note that the parameter values are tuned by trial and error with the strategy of binary search. A video<sup>1</sup> demonstration of the experiments is available online.

In robot-assisted rehabilitation, a trajectory, either predefined by mathematical equations or customized by the

<sup>1</sup>Online video link: [https://youtu.be/Jokv\\_RPOxEc](https://youtu.be/Jokv_RPOxEc)

therapist, is needed to help patients move their injured limbs around for rehabilitation and recovery [1], [6], [7]. Several pre-defined trajectories are employed for evaluating the teleoperation system performance while trajectory tracking accuracy will be used to evaluate the system performance. Note that in order to present the results more intuitively, the tracking accuracy in experimental results is presented as trajectory plots by overlapping the desired and the actual trajectories into one figure instead of trajectory tracking error plots. A circular and cyclic trajectory can be expressed as a function of time as the following

$$\begin{cases} x_d = R \sin\left(\frac{2\pi}{t_1} t\right) \\ y_d = R \cos\left(\frac{2\pi}{t_1} t\right) \end{cases} \quad (20)$$

where  $R = 0.08$  m is the radius of the circle and  $t_1 = 5$  s is the period for generating a full cycle.

A figure-eight trajectory can be expressed as

$$\begin{cases} x_d = R \sin\left(\frac{2\pi}{t_1} t\right) \cos\left(\frac{2\pi}{t_1} t\right) \\ y_d = R \sin\left(\frac{2\pi}{t_1} t\right) \end{cases} \quad (21)$$

where  $R = 0.1$  m is the amplitude of the figure-eight trajectory,  $t_1 = 5$  s is the period for generating a full cycle.

A tetragon trajectory (also known as beetle curve) can be expressed as a function of time given by

$$\begin{cases} x_d = R \cos^3(t) \\ y_d = R \sin^3(t) \end{cases} \quad (22)$$

where  $R = 0.1$  m is a parameter of the tetragon trajectory.

A pentagram trajectory (also known as hypotrochoid) can be expressed as a function of time given by

$$\begin{cases} x_d = (R - r) \cos(nt) + d \cos\left(\frac{R-r}{r} nt\right) \\ y_d = (R - r) \sin(nt) - d \sin\left(\frac{R-r}{r} nt\right) \end{cases} \quad (23)$$

where  $R = 0.08$ ,  $r = 0.048$ ,  $d = 0.064$ , and  $n = 3$ , are parameters of the pentagram trajectory, and  $R$ ,  $r$ ,  $d$  control the size of the trajectory, and  $n$  controls the period time for a full cycle.

A rose curve trajectory (also known as rhodonea curve) can be expressed as a function of time given by

$$\begin{cases} x_d = R \cos(kt) \cos(t) \\ y_d = R \cos(kt) \sin(t) \end{cases} \quad (24)$$

where  $R = 0.1$ ,  $k = 4$ , are parameters of the rose curve trajectory, and  $R$  controls the size of pattern, and  $k$  controls the number of petals of the pattern.

## IV. EXPERIMENTS AND RESULTS

### A. Exp.1: Individual Robot Performance

In Exp.1, we evaluated the performance of each robot by implementing the impedance controller (3) with and without the nonlinear disturbance observer (NDOB) (10). In this experiment, a circular trajectory (20) is employed.

The experiment results are shown in Fig. 4. In the figure, we can see that the robots cannot perform the trajectory tracking task accurately without NDOB (Fig. 4a, 4c), while accurate tracking performance is obtained (Fig. 4b, 4d) when NDOB is

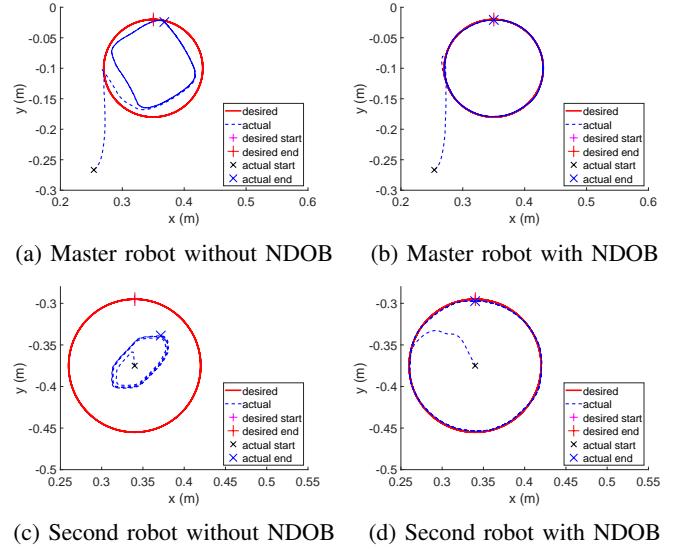


Fig. 4: Exp.1 results of individual performance of the two robots by implementing impedance controller with or without NDOB.

implemented. The reason for failed-tracking (in Fig. 4a, 4c) is due to the inaccurate dynamic model and joint friction model of the robot. This is especially true for the second robot (white) since the identified dynamic model and friction model were specially for the master robot (black). With implementing an NDOB observer, the dynamic uncertainties can be accurately estimated and compensated for in both robots.

Exp.1 indicates that by implementing both the impedance controller and NDOB, the robots can accurately perform trajectory-tracking tasks separately. Even with a borrowed dynamic model, which can be viewed as a roughly estimated dynamic model for the second robot (white), the robot can track accurately by involving a disturbance observer.

### B. Exp.2: Trajectory Tracking

In Exp.2, a set of pre-defined trajectories with patterns varying from simple to complex are employed to evaluate the tracking performance of the teleoperation system, including a circle (20), a figure-eight trajectory (21), a tetragon trajectory (22), a pentagram trajectory (23), and a rose curve trajectory (24). Both the master and the second robots are implemented with an impedance controller and NDOB observer. The master robot is set to track the pre-defined trajectories, while the second robot is set to track the EE position of the master robot. The teleoperation control in Exp.2 is unilateral, *i.e.*, no force feedback on the master robot side, since force feedback is not necessary for the master robot in this experiment.

Fig. 5 shows the tracking performance of the teleoperation system. As can be seen in the figure, the teleoperation system can accurately perform the trajectory tracking tasks from a simple pattern to a complex pattern (Fig. 5a, 5c, 5e, 5g, 5i). Although the tracking accuracy of the complex rose trajectory pattern (Fig. 5i) is not as good as the simple ones (Fig. 5a, 5c, 5e, 5g), it can be further improved by fine-tuning the

impedance gains and observer gains. From the figure, we can also notice that the actual torques for the two robots are not on the same level, and the second robot needs larger torques (maximum around 0.5 Nm) than the master robot (maximum around 0.2 Nm) for tracking the same trajectory. This is reasonable since they are not identical robots, although they have the same form of kinematics. This also reflects that the NDOB can accurately estimate the dynamic uncertainties for the second robot (white) although it employs an identified dynamic mode that is not actually for it.

The results of Exp.2 indicate that the teleoperation system can accurately perform trajectory tracking tasks with varying pre-defined patterns by implementing an impedance controller and NDOB observer. The master robot (black) can accurately follow the pre-defined trajectory while the second robot (white) can accurately follow the master robot.

### C. Exp.3: pHRI

In actual robot-assisted rehabilitation scenarios, pre-defined trajectories might not be always suitable for patients due to their various rehabilitation requirements. Then, the trajectory needs to be customized for individual patients by the therapist which can more effectively help the patient to recover. In Exp.3, physical human-robot interaction (*pHRI*) is involved for the master robot where the therapist moves the master robot EE manually in order to customize trajectories while the second robot follows the trajectory of the master EE.

In this experiment, for the master robot, only an impedance controller is implemented since an NDOB observer will prevent human-robot interaction [9]. The configuration of the second robot remains unchanged, *i.e.*, both impedance controller and NDOB are implemented. The teleoperation control in Exp.3 is bilateral, *i.e.*, the master robot will provide force feedback to its operator when the second robot is in contact with the surrounding environment. Even though, in Exp.3, the operator of the master robot will not receive force feedback since there is no interaction between the second robot and the surrounding environment.

Fig. 6 shows the experimental results when the master robot is operated manually by the operator in the *pHRI* mode while the second robot follows the trajectory of the master robot EE. The results show that the operator of the master robot can customize the trajectory for the master robot, while the second robot can accurately follow the master robot.

### D. Exp.4: Force Feedback

In Exp.3, *pHRI* mode, the operator of the master robot does not receive rendered force feedback due to the fact that the second robot is in free motion and has no interaction with the surrounding environment. In Exp.4, force feedback will be evaluated when the second robot is in contact with external objects. Two scenarios are designed. In scenario 1, the second robot will be in contact with an external stiff wall while the master robot will render force feedback accordingly and provide it to its operator. In scenario 2, the second robot EE is attached to an external object (1130 grams) and drags it around mimicking a rehabilitation scenario of the patient's limb being

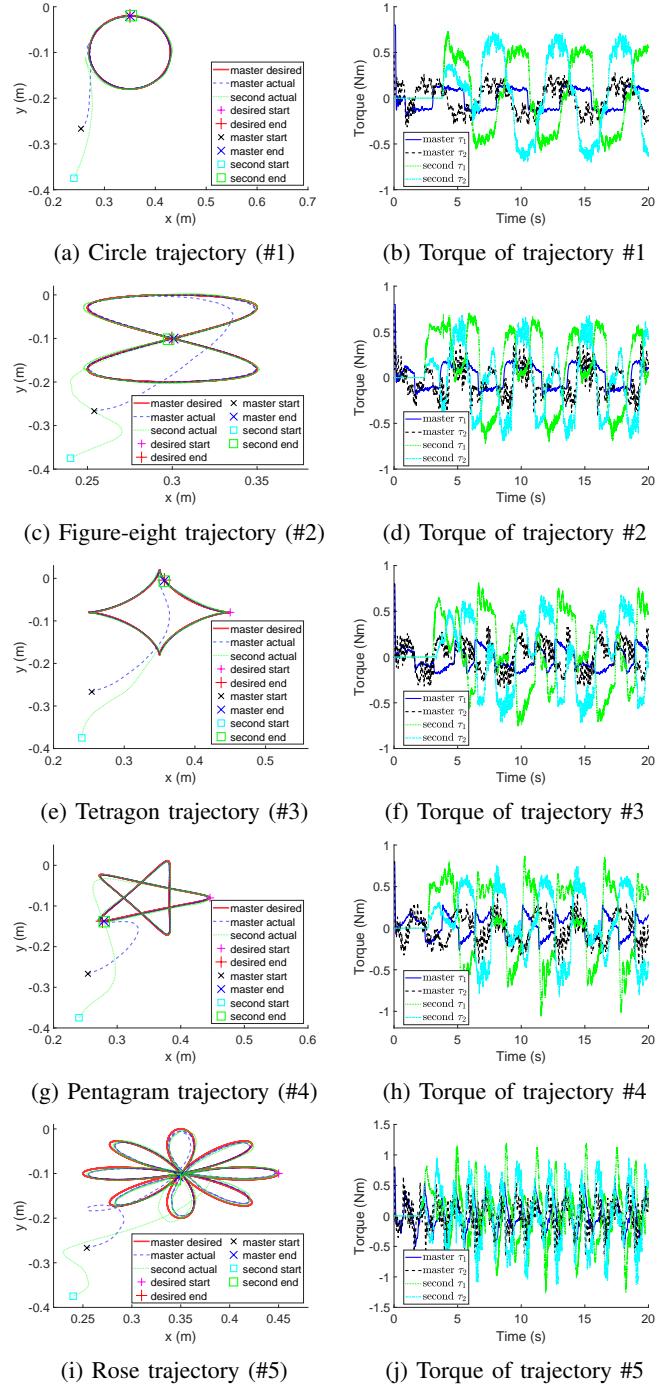
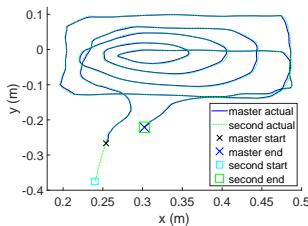
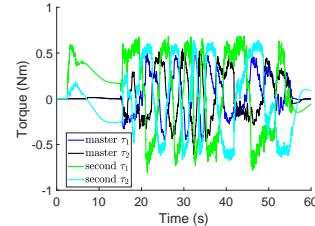


Fig. 5: Exp.2 results of the teleoperation system performance on trajectory tracking tasks. The master robot tracks a pre-defined trajectory while the second robot follows the master robot.



(a) Customized trajectory



(b) Actual torque

Fig. 6: Exp.3 results when the master robot is in pHRI mode. The trajectory of the master robot is defined by the operator while the second robot follows the master robot.

attached to the second robot EE and passively moving around for rehabilitation training.

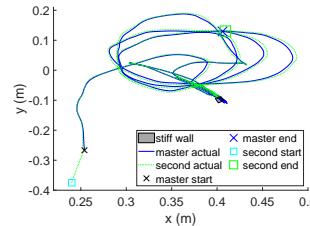
The experimental results of Exp.4 are shown in Fig. 7, where the yellow-colored area represents scenario 1, and the purple-colored area represents scenario 2. In scenario 1 of the yellow-colored area, the operator remotely controls the second robot to probe a stiff wall by operating the master robot while the contact force between the second robot and the stiff wall is rendered by the master robot and delivered to its operator. As can be seen in Fig. 7b, the rendered force feedback is in the range of  $[-7, 7]$  N. In scenario 2 of the purple-colored area, the operator remotely controls the second robot to drag the patient's upper limb (represented by an external object of 1130 g), while the operator receives force feedback which is in the range of  $[-3, 3]$  N. Note that in this work, we only consider scenarios where the patients move their limbs passively, thus we represent the patient's limb by an external object.

The Exp.4 results show that the feedback force can be appropriately rendered according to the position error between the master robot EE and the second robot EE, and then delivered to the operator of the master robot. The feedback force here can be rendered to be stiffer or softer by tuning the corresponding gain.

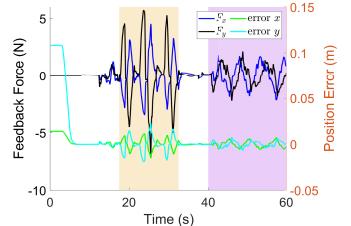
#### E. Exp.5: Record-Replay

In conventional rehabilitation, the therapist often needs to help the patient move the limb repetitively along a customized trajectory, which is time-consuming and strenuous. This can be easily and effectively done by the proposed teleoperation system. The therapist can demonstrate a customized trajectory only once which can be recorded by the robot, then the robot can repeat the trajectory as many times as needed on its own.

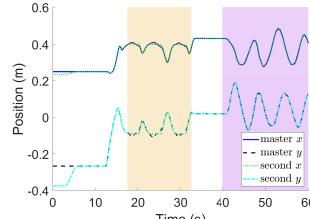
In previous Exp.3, the customized trajectory demonstrated by the operator is recorded. Here in Exp.5, we let the master robot replay the recorded trajectory on its own. The experiment results are shown in Fig. 8. As can be seen in the figure, the teleoperation system can accurately replay the recorded trajectory by comparing it with its original demonstration (Fig. 6). Note that the actual torque (Fig. 8b) is a bit noisy than that in the original demonstration (Fig. 6b). This is reasonable since the recorded velocity and acceleration are discontinuous numerical values. As long as the sampling frequency of the recorded trajectory is sufficiently high, e.g.,



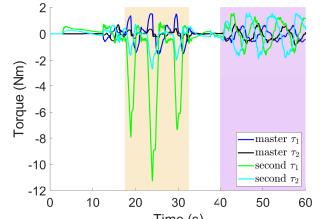
(a) Trajectory



(b) Force Feedback

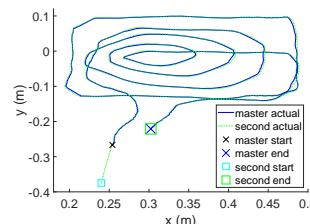


(c) Position

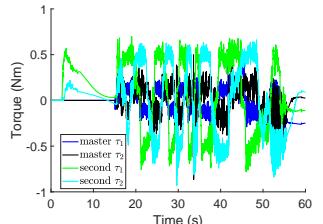


(d) Actual torque

Fig. 7: Exp.4 results of force feedback rendering on the master robot side when the second robot interacts with external objects. The yellow-colored area represents scenario 1, and the purple-colored area represents scenario 2.



(a) Trajectory



(b) Actual torque

Fig. 8: Exp.5 results of replaying the recorded trajectory data from Exp.3. The master robot replays the recorded trajectory while the second robot follows the master robot.

1000 Hz in this experiment, the noise in the actual torque can be largely minimized and has no adverse effect on the system performance.

## V. DISCUSSIONS

In this paper, we propose a teleoperation system for robot-assisted rehabilitation training where patients can passively move their upper limbs with the help of a robot. The control scheme integrated and enabled three main features in one teleoperation system. First, the proposed system allows the therapist to program pre-defined trajectories with various patterns using their mathematical expressions (trajectory-tracking mode), thus helping the patient conduct movement therapy training repetitively. Second, the proposed system allows the therapist to operate the master robot (pHRI mode), then the second robot on the patient side will follow the master robot accurately. In such a way, the therapist can directly help the patient to conduct the movement therapy training remotely. Third, the system can record the customized movement of the

therapist in *pHRI* mode, then replay it repetitively on its own in trajectory-tracking mode to help the patient do the training.

In the proposed teleoperation system, the master robot (on the therapist side) is implemented with both an impedance controller and a disturbance observer in trajectory-tracking mode, while with only an impedance controller in *pHRI* mode. The second robot (on the patient side) is implemented with both an impedance controller and a disturbance observer in both trajectory-tracking mode and *pHRI* mode since it is designed to always follow the master robot. For the master robot in *pHRI* mode, the disturbance observer needs to be turned off since it will prevent human-robot interaction [9].

For the master robot in *pHRI* mode, force feedback is rendered and delivered to the operator (the therapist) when the second robot is in contact with the surrounding environment. This allows the therapist to track the real-time status of the patient via the force feedback. The force feedback is rendered based on the position error between the master robot EE and the second robot EE rather than the actual interaction force. The advantage is that, by appropriate gain-tuning, the rendered force can be tuned to be stiffer or softer, and it is independent of the controller.

In [24], Tavakoli *et al.* summarized typical teleoperation architectures with force feedback including two-channel (2CH) and 4CH control architectures. The 2CH position-error-based (PEB, also called position-position) architecture is a symmetric design where the master robot takes the real-time position of the second robot as its desired position, and vice versa. This 2CH PEB architecture does not need a force sensor while force feedback that is proportional to the position difference will be rendered automatically on both sides due to the symmetric design. However, this design suffers from a distorted perception even when the robot is in free motion due to the non-perfect position tracking (*i.e.*, non-zero position errors), which means that the operator of the master robot will receive proportional force feedback even when the second robot is not in contact with the surrounding environment. Another 2CH architecture is direct force reflection (DFR, also called force-position). This architecture requires a force sensor to measure the interaction force between the second robot and the environment, then only the measured force (not including the position anymore) is sent back to the master robot for generating force feedback. Although the 2CH DFR is proven to be better than the 2CH PEB in terms of force feedback transparency, the operator can still feel the intrinsic inertia of the master robot when the second robot is not in contact with any objects. The 4CH architecture is able to achieve ideal transparency by installing force sensors on both sides to measure the interaction forces. Both the measured force and the real-time position will be sent to the other robot for generating force feedback and position tracking. However, installing force sensors makes the system complex, and usually unnecessary. When measuring the interaction force is not needed on the master (or second) robot side, the corresponding force sensor can be removed, thus a 4CH architecture can be reduced to a 3CH architecture without imposing additional expense on the system transparency. This makes the 3CH architecture extremely attractive. Furthermore, all these 2CH/3CH/4CH architectures are designed for *pHRI*

scenarios and they cannot be directly used for pre-defined trajectory tracking tasks.

In contrast to the 2CH/3CH/4CH architectures introduced in [24], in this work, accurate force feedback (high transparency) is not necessary for rehabilitation. Therefore, force sensors are not needed. Instead, only the position error is used for generating a virtual spring force as the feedback on the master robot side, and by proper gain-tuning, the rendered force can be close to the actual interaction force. The proposed architecture is similar to the 2CH PEB architecture, *i.e.*, both robot real-time positions will be sent to the other robot. But in our designed architecture, the master robot will only use the position of the second robot for force rendering. This also makes the force rendering be independent of the controller design.

One advantage of the proposed architecture is that the teleoperation system can seamlessly switch from trajectory-tracking mode to *pHRI* mode while the architectures in [24] are devotedly focusing on *pHRI* mode. Another advantage is that the proposed control scheme is applicable to all general robotic systems rather than only haptic devices. On the contrary, a potential disadvantage is that the damper term in the controller (6) in *pHRI* mode may not be necessary for a backdrivable haptic device. For a haptic device, with the damper term, the operator will feel an additional damper force generated by this term, while without the damper term, the operator only feels the intrinsic inertia of the robot mechanism. If this is the case, the user can simply remove the damper term when using a haptic device.

A potential limitation of this work is that we are only considering the scenarios where the patients move their upper limbs passively. In future work, we will extend the current work to other scenarios where the patients can move their limbs actively to some extent, and then a robot can help the patients do the rehabilitation training cooperatively.

## VI. CONCLUSIONS

In this paper, a bilateral teleoperation system is constructed for robot-assisted rehabilitation. The teleoperation system is implemented with a combination of impedance controller and disturbance observer where the former can provide compliant robot behavior for ensuring safe robot-human interaction while the latter can compensate for dynamic uncertainties when only a rough dynamic model is available. More importantly, the designed control architecture integrated both trajectory-tracking mode and *pHRI* mode, and allows them to be switched from one mode to another freely. The proposed teleoperation system demonstrated accurate trajectory tracking performance of both the master and second robot, as well as appropriate force feedback on the master robot side. Moreover, the teleoperation system allows the therapist to demonstrate a customized trajectory only once, and then the system, on its own, can accurately replay the recorded trajectory as many times as needed.

The proposed teleoperation system is promising to be used for robot-assisted rehabilitation. It can reduce the workload of the therapist and retain the quality of the work of helping the

patient move along a pre-defined or customized trajectory for rehabilitation training.

#### ACKNOWLEDGMENT

The author would like to thank Dr. Mahdi Tavakoli and Dr. Milad Nazarahari for their invaluable and insightful feedback and suggestions on improving the quality of this work.

#### REFERENCES

- [1] M. Najafi, C. Rossa, K. Adams, and M. Tavakoli, "Using potential field function with a velocity field controller to learn and reproduce the therapist's assistance in robot-assisted rehabilitation," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 3, pp. 1622–1633, 2020.
- [2] R. Ocampo and M. Tavakoli, "Visual-haptic colocation in robotic rehabilitation exercises using a 2D augmented-reality display," in *2019 International Symposium on Medical Robotics (ISMР)*. IEEE, 2019, pp. 1–7.
- [3] N. Hogan, H. I. Krebs, A. Sharon, and J. Charnnarong, "Interactive robotic therapist," Nov. 14 1995, US Patent 5,466,213.
- [4] R. Riener, T. Nef, and G. Colombo, "Robot-aided neurorehabilitation of the upper extremities," *Medical and Biological Engineering and Computing*, vol. 43, pp. 2–10, 2005.
- [5] W. Meng, Q. Liu, Z. Zhou, Q. Ai, B. Sheng, and S. S. Xie, "Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation," *Mechatronics*, vol. 31, pp. 132–145, 2015.
- [6] M. Sharifi, S. Behzadipour, H. Salarieh, and M. Tavakoli, "Assist-as-needed policy for movement therapy using telerobotics-mediated therapist supervision," *Control Engineering Practice*, vol. 101, p. 104481, 2020.
- [7] M. Sharifi, H. Salarieh, S. Behzadipour, and M. Tavakoli, "Patient-robot-therapist collaboration using resistive impedance controlled tele-robotic systems subjected to time delays," *Journal of Mechanisms and Robotics*, vol. 10, no. 6, p. 061003, 2018.
- [8] P. Song, Y. Yu, and X. Zhang, "A tutorial survey and comparison of impedance control on robotic manipulation," *Robotica*, vol. 37, no. 5, pp. 801–836, 2019.
- [9] T. Li, A. Badre, H. D. Taghirad, and M. Tavakoli, "Integrating impedance control and nonlinear disturbance observer for robot-assisted arthroscopic surgery," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 11 172–11 179.
- [10] S. Bruno, S. Lorenzo, V. Luigi, and O. Giuseppe, *Robotics: Modelling, Planning and Control*. Springer, 2010.
- [11] J. Fong, H. Rouhani, and M. Tavakoli, "A therapist-taught robotic system for assistance during gait therapy targeting foot drop," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 407–413, 2019.
- [12] A. Torabi, M. Khadem, K. Zareinia, G. R. Sutherland, and M. Tavakoli, "Application of a redundant haptic interface in enhancing soft-tissue stiffness discrimination," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1037–1044, 2019.
- [13] A. Mohammadi, M. Tavakoli, H. J. Marquez, and F. Hashemzadeh, "Nonlinear disturbance observer design for robotic manipulators," *Control Engineering Practice*, vol. 21, no. 3, pp. 253–267, 2013.
- [14] A. Radke and Z. Gao, "A survey of state and disturbance observers for practitioners," in *2006 American Control Conference*. IEEE, 2006, pp. 5183–5188.
- [15] W.-H. Chen, J. Yang, L. Guo, and S. Li, "Disturbance-observer-based control and related methods — an overview," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 2, pp. 1083–1095, 2016.
- [16] A. Mohammadi, H. J. Marquez, and M. Tavakoli, "Nonlinear disturbance observers: Design and applications to Euler-Lagrange systems," *IEEE Control Systems Magazine*, vol. 37, no. 4, pp. 50–72, 2017.
- [17] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.
- [18] G. Sebastian, Z. Li, V. Crocher, D. Kremers, Y. Tan, and D. Oetomo, "Interaction force estimation using extended state observers: An application to impedance-based assistive and rehabilitation robotics," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1156–1161, 2019.
- [19] J. Hu and R. Xiong, "Contact force estimation for robot manipulator using semiparametric model and disturbance Kalman filter," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3365–3375, 2018.
- [20] S. Liu, L. Wang, and X. V. Wang, "Sensorless force estimation for industrial robots using disturbance observer and neural learning of friction approximation," *Robotics and Computer-Integrated Manufacturing*, vol. 71, p. 102168, 2021.
- [21] M. C. Çavuşoğlu, D. Feygin, and F. Tendick, "A critical study of the mechanical and electrical properties of the Phantom haptic interface and improvements for high performance control," *Presence: Teleoperators & Virtual Environments*, vol. 11, no. 6, pp. 555–568, 2002.
- [22] M. D. Dyck, "Measuring the dynamic impedance of the human arm," Master's thesis, University of Alberta (Canada), 2013.
- [23] A. Torabi, M. Khadem, K. Zareinia, G. R. Sutherland, and M. Tavakoli, "Using a redundant user interface in teleoperated surgical systems for task performance enhancement," *Robotica*, vol. 38, no. 10, pp. 1880–1894, 2020.
- [24] M. Tavakoli, A. Aziminejad, R. V. Patel, and M. Moallem, "High-fidelity bilateral teleoperation systems and the effect of multimodal haptics," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 6, pp. 1512–1528, 2007.