Bilateral Teleoperation with Computed Torque Control for RRR Type Robot Manipulator

Ammar URGAN[1], **and Janset DAŞDEMİR**[1]

[1] Yıldız Teknik Üniversitesi, Kontrol ve Otomasyon Mühendisliği

Abstract: This paper presents a control mechanism designed for an RRR-type robot manipulator, Geomagic Touch Phantom Omni^{TM}, to track a desired reference signal. Utilizing the properties of the haptic device, a force controller is designed to allow the operator to feel the hard contact state. Since external torque or force measurements are not directly accessible, a method similar to the position-position method is used. The study used an API-based library running in the MATLAB/Simulink^{TM} environment instead of a data transfer unit. Stability analysis shows that the system error dynamics are globally asymptotically stable and all signals within the closed-loop system are bounded. The applicability and performance of the proposed method are validated by experimental results. We propose a bilateral teleoperation system based on the Computed Torque Control (CTC), a nonlinear control approach, method that ensures safe interaction with the surroundings without assuming any prior knowledge of the surroundings.
Keywords: Robot manipulators, Haptic teleoperation, Computed torque control.

RRR Tipi Robot Manipulatörü İçin Hesaplanmış Tork Kontrolü Yöntemiyle Bilateral Teleoperasyon

Özet: Bu çalışma, RRR tipi bir robot manipülatörü olan Geomagic Touch Phantom Omni^{TM} için istenen bir referans sinyalini takip etmek üzere tasarlanmış bir kontrol mekanizması sunmaktadır. Haptik cihazın özelliklerinden yararlanılarak, operatörün sert temas durumunu hissetmesini sağlamak için bir kuvvet kontrolörü tasarlanmıştır. Harici tork veya kuvvet ölçümlerine doğrudan erişilemediği için pozisyon-pozisyon yöntemine benzer bir yöntem kullanılmıştır. Çalışma boyunca veri aktarım ünitesi yerine MATLAB/Simulink^{TM} ortamında çalışan API tabanlı bir kütüphane kullanılmıştır. Kararlılık analizi, sistem hata dinamiklerinin küresel asimptotik kararlı olduğunu ve kapalı döngü sistemindeki tüm sinyallerin sınırlı olduğunu göstermektedir. Önerilen yöntemin uygulanabilirliği ve performansı deneysel sonuçlarla desteklenmiştir. Hesaplanmış Tork Kontrolü (HTK), doğrusal olmayan bir kontrol yaklaşımı, yöntemine dayanan ve çevre hakkında herhangi bir ön bilgi varsaymaksızın çevre ile güvenli etkileşim sağlayan iki taraflı bir teleoperasyon sistemi önerilmektedir.
Anahtar Kelimeler: Robot manipulatörleri, Haptik teleoperasyon, Hesaplanmış tork kontrolü.

RESEARCH PAPER

Corresponding Author: Ammar URGAN, ammar@yildiz.edu.tr

Reference: A. Urgan, and J. Daşdemir, (2024), ITU Computer Science, AI and Robotics, 1, (1) 26-33

Submission Date: 24/01/2024
Acceptance Date: 06/05/2024
Online Publishing: 20/07/2024

1 Introduction

Robotic manipulators, commonly used in industrial applications, pose challenges due to non-linear components and dynamically changing behaviors ([1]). Achieving desired trajectories in three dimensions is a primary goal for industrial robots. To design fast and precise motion control mechanisms, both dynamic and kinematic models of robots are essential ([1], [2]). The kinematic model of a robot manipulator plays a key role, as it determines the joint movements necessary for a given end-effector trajectory. In parallel, dynamic models are critical, as they are used to calculate the control inputs for joint actuators, ensuring precise execution of the desired robot movements ([3]).

A fully known model is either provided by the manufacturer or obtained by user-applied system identification methods. When a full model is available, feedback linearization methods can be applied. The computed torque control (CTC) method is recognized as an effective control strategy for robotic manipulators, representing a specialized implementation of feedback linearization to manage the inherent non-linearities in such systems. This approach significantly improves the precision and adaptability of manipulators by effectively linearizing their complex dynamics. This allows individual control of each joint movement using known linear control strategies ([2]). Simulation studies have demonstrated the success of the computed torque control method in trajectory tracking problems for 2-axis ([4]) and 4-axis ([5]) robots. Research conducted using the Phantom Omni^{TM} haptic device has demonstrated the effectiveness of both inverse dynamic control and computed torque control methodologies ([6], [7]).

Teleoperation systems have been utilized for years in transporting hazardous materials ([8]), (remote) surgical procedures ([9]–[11]), underwater vehicles ([12]), space robots ([13]), and mobile robots ([14]). These systems can be classified differently depending on the direction of information flow. If the information is transmitted from the master robot to the slave robot without any feedback to the master, the system can be considered unilateral (Figure 1). If the master robot can sense the environmental conditions around the slave robot, communication is bilateral (Figure 2). The communication between the master robot and the slave robot can impact system behavior and must be considered ([15]). In IP communication, if the integrity of transmitted information needs to be maintained, TCP should be used. However, if transmission delays negatively impact system performance and need to be as low as possible, UDP should be used ([16]). In cases where force/torque sensors are not present, a position-position (PP) control scheme is commonly used. In the presence of force/torque sensors, a force-position (FP) control scheme becomes applicable. Lawrence proposed a 4-channel (4C) architecture using force and velocity changes between the master and slave robots for bilateral teleoperation ([17]). FP and 4C use both force and position sensors. In teleoperation with easily movable master robots and industrially sized rigid slave robots, PP control demonstrates stability. However, unlike FP and 4C control schemes, PP does not have the capability to provide easy maneuverability and effective force feedback ([18]).

Real-time systems inherently encounter communication delays, impacting controller performance ([19]). To mitigate this, controllers must be designed to function effectively under such conditions. The wave-based teleoperation method is proposed for systems with time delays ([20]), while Passive Model Predictive Controllers have been suggested for constant time delays ([21]). Although scattering-based schemes offer delay-independent stability, this often comes at the cost of reduced tracking performance. Passive output synchronization demonstrates robustness to variable time delays, but position tracking is not guaranteed. Adaptive controllers have been shown to be effective in scenarios with constant time delays, maintaining robust tracking accuracy. However, their performance in terms of tracking precision tends to be compromised in environments characterized by variable time delays ([22]).

This study explores the implementation of bilateral teleoperation utilizing the Computed Torque Control method, adapted for a robotic manipulator devoid of torque sensors. Consequently, a position-position control scheme is deployed. The primary objective is to create a teleoperation experience where the operator perceives no force under normal conditions but can discern the torque from any encountered obstacle. To mitigate communication delays, the system is designed to transmit error values of position, velocity, and acceleration to the master side, instead of their absolute measures. This approach demonstrates the potential to reduce the impacts of communication latencies, thereby enhancing system performance without necessitating complex controller architectures.

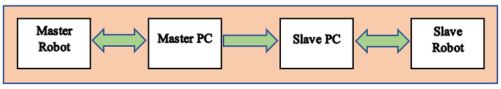

Fig. 1 Communication of unilateral teleoperation

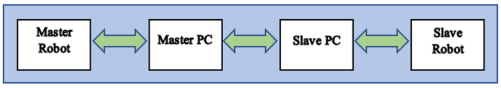

Fig. 2 Communication of bilateral teleoperation

2 Dynamic model of the Robot Manipulator

The joint frames and model parameters for the haptic device are illustrated in Figure 3. The dynamic model of the

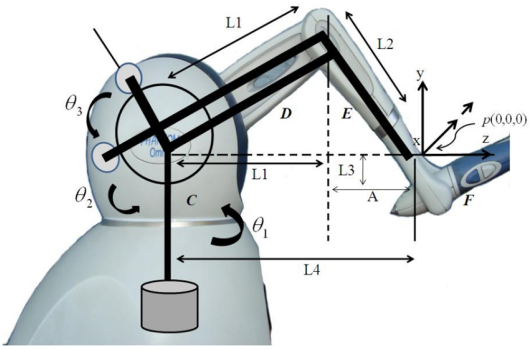

Fig. 3 The kinematics of the Phantom Omni^{TM} haptic device ([23])

Phantom Omni^{TM} haptic device was formulated utilizing the Euler-Lagrange method ([24]).

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + N(\theta) = \tau \tag{1}$$

Here, $M = M^T \in R^{3x3}$ represents the inertia matrix, $C \in R^{3x3}$ the Coriolis and centripetal forces matrix, $N \in R^{3x1}$ the gravity matrix, $\theta = [\theta_1, \theta_2, \theta_3]^T \in R^{3x1}$ the joint positions, and $\tau = [\tau_1, \tau_2, \tau_3]^T \in R^{3x1}$ the torques acting on the joints.

The dynamic model given in equation (1) satisfies the following properties.

Property 1 ([25]) The symmetric, positive definite inertia matrix $M(\theta)$ satisfies the inequality given in the form

$$\mu_1\|\alpha\|^2 \leq \alpha^T M(\theta)\alpha \leq \mu_2\|\alpha\|^2, \qquad \forall\alpha \in R^3 \tag{2}$$

Here, μ_1 and μ_2 are positive scalars and $\|.\|$ is the standard Euclidian norm.

Property 2 ([25]) There exists a skew-symmetric relationship between the derivative of the inertia and centripetal-Coriolis matrices as follows

$$\alpha^T\left(\frac{1}{2}\dot{M}(\theta) - C\left(\theta,\dot{\theta}\right)\right)\alpha = 0, \qquad \forall\alpha \in R^3 \tag{3}$$

Property 3 ([25]) Gravity Forces matrix N,

$$\|N(\theta)\| \leq \mu_3 \tag{4}$$

can be constrained by the inequality above, where μ_3 is a scalar.

The inertia matrix M for the dynamic model given in Equation (1) can be expressed as

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,1} & m_{3,2} & m_{3,3} \end{bmatrix} \tag{5}$$

where the elements are:

$$m_{1,1} = k_1 + k_2 cos(2\theta_2) + k_3 cos(2\theta_3) + k_4 cos(\theta_2) sin(\theta_3)$$

$$m_{1,2} = k_5 sin(\theta_2)$$

$$m_{1,3} = 0$$

$$m_{2,1} = k_5 sin(\theta_2)$$

$$m_{2,2} = k_6$$

$$m_{2,3} = -0.5 k_4 sin(\theta_2 - \theta_3)$$

$$m_{3,1} = 0$$

$$m_{3,2} = -0.5 k_4 sin(\theta_2 - \theta_3)$$

$$m_{3,3} = k_7$$

The Coriolis/Centrifugal Forces Matrix C can be expressed as

$$C = \begin{bmatrix} c_{1,1} & c_{1,2} & c_{1,3} \\ c_{2,1} & c_{2,2} & c_{2,3} \\ c_{3,1} & c_{3,2} & c_{3,3} \end{bmatrix} \tag{6}$$

where the elements are:

$$c_{1,1} = -k_2\dot{\theta}_2 \sin(2\theta_2) - k_3\dot{\theta}_3 \sin(2\theta_3)$$

$$-0.5k_4\dot{\theta}_2 \sin(\theta_2)\sin(\theta_3) + 0.5k_4\dot{\theta}_3 \cos(\theta_2)\cos(\theta_3)$$

$$c_{1,2} = -k_2\dot{\theta}_1 \sin(2\theta_2) - 0.5k_4\dot{\theta}_1 \sin(\theta_2)\sin(\theta_3)$$

$$+k_5\dot{\theta}_2 \cos(\theta_2)$$

$$c_{1,3} = -k_3\dot{\theta}_1 \sin(2\theta_3) - 0.5k_4\dot{\theta}_1 \cos(\theta_2)\cos(\theta_3)$$

$$c_{2,1} = k_2\dot{\theta}_1 \sin(2\theta_2) + 0.5k_4\dot{\theta}_1 \sin(\theta_2)\sin(\theta_3)$$

$$c_{2,2} = 0$$

$$c_{2,3} = 0.5k_4\dot{\theta}_3 \cos(\theta_2 - \theta_3)$$

$$c_{3,1} = k_3\dot{\theta}_1 \sin(2\theta_3) + 0.5k_4\dot{\theta}_1 \cos(\theta_2)\cos(\theta_3)$$

$$c_{3,2} = -0.5k_4\dot{\theta}_2 \cos(\theta_2 - \theta_3)$$

$$c_{3,3} = 0$$

The gravitational Force matrix can be expressed as

$$N = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix} \tag{7}$$

where the elements are:

$$n_1 = 0$$

$$n_2 = k_8 \cos(\theta_2) + k_{10}(\theta_2 - 0.5\pi)$$

$$n_3 = k_9 \sin(\theta_3)$$

The values of the variables used in the equations are as follows ([24]):

$$k_1 = 1.798x10^{-3} \qquad k_2 = 0.864x10^{-3}$$

$$k_3 = 0.486x10^{-3} \qquad k_4 = 2.766x10^{-3}$$

$$k_5 = 0.308x10^{-3} \qquad k_6 = 2.526x10^{-3}$$

$$k_7 = 0.652x10^{-3} \qquad k_8 = 164.158x10^{-3}$$

$$k_9 = 94.050x10^{-3} \qquad k_{10} = 117.294x10^{-3}$$

3 Position Controller Design

The dynamic equation of the system can be rewritten as given in Equation (8).

$$\tau = \hat{M}(\theta)\ddot{\theta} + \hat{C}(\theta,\dot{\theta})\dot{\theta} + \hat{N}(\theta) \tag{8}$$

Here, $\hat{M}$, $\hat{C}$ ve $\hat{N}$ are estimation matrices, $\theta, \dot{\theta}$, $\ddot{\theta}$ are position, velocity and acceleration respectively. The error and error dynamics for use in the control system are defined as follows.

$$e = \theta_d - \theta \tag{9}$$

$$\dot{e} = \dot{\theta}_d - \dot{\theta} \tag{10}$$

$$\ddot{e} = \ddot{\theta}_d - \ddot{\theta} \tag{11}$$

When Equation (1) is rearranged in terms of $\ddot{\theta}$, the following equation is obtained:

$$\ddot{\theta} = M^{-1}(\theta)\left(\tau - C(\theta,\dot{\theta})\dot{\theta} - N(\theta)\right) \tag{12}$$

Accordingly, the control input signal, τ, is designed as in Equation (13):

$$\tau = \hat{M}(\theta)u + \hat{C}(\theta,\dot{\theta})\dot{\theta} + \hat{N}(\theta) \tag{13}$$

With substituting equation (13) into equation (12), equation (14) is obtained.

$$\begin{aligned}\ddot{\theta} =& M^{-1}(\theta)\left(\hat{M}(\theta)u + \hat{C}(\theta,\dot{\theta})\dot{\theta} + \hat{N}(\theta)\right) \\ & - M^{-1}(\theta)\left(C(\theta,\dot{\theta})\dot{\theta} + N(\theta)\right)\end{aligned} \tag{14}$$

When the assumption is made that the estimation matrices are equal to the actual matrices, equation (14) transforms into equation (15).

$$\ddot{\theta} = u \tag{15}$$

When the control signal u is selected as in Equation (16),

$$u = \ddot{\theta}_d + K_d\dot{e} + K_p e \tag{16}$$

and using equations (11), (15), and (16), the error dynamics is obtained as follows:

$$\ddot{\theta}_d - \ddot{\theta} + K_d\dot{e} + K_p e = \ddot{e} + K_d\dot{e} + K_p e = 0 \tag{17}$$

As known from linear system theory, when the condition $(K_p > 0,\ K_d > 0)$ is satisfied, the error dynamics are asymptotically stable, ensuring that the error signal will converge to zero. The boundedness of e and $\dot{e}$ restricts $\ddot{e}$ according to equation (17). Since the reference trajectory signal and its derivatives are bounded, the real trajectory and its derivatives are also bounded according to equations (9), (10), and (11). All signals on the right-hand side of equation (13) are bounded, which in turn limits the control signal. The control scheme is illustrated in Figure 4.

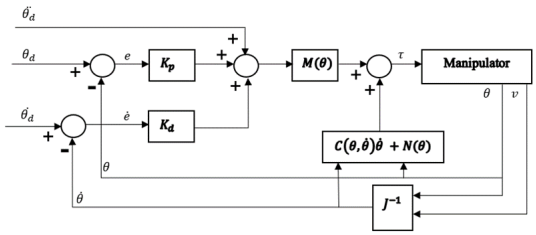

Fig. 4 The Position Controller Scheme

4 Force Controller Design

The error and its first two derivatives for both the master and slave robots have been redefined for use during bilateral teleoperation.

$$e_m = \theta_s - \theta_m \tag{18}$$

$$\dot{e}_m = \dot{\theta}_s - \dot{\theta}_m \tag{19}$$

$$\ddot{e}_m = \ddot{\theta}_s - \ddot{\theta}_m \tag{20}$$

$$e_s = \theta_m - \theta_s \tag{21}$$

$$\dot{e}_s = \dot{\theta}_m - \dot{\theta}_s \tag{22}$$

$$\ddot{e}_s = \ddot{\theta}_m - \ddot{\theta}_s \tag{23}$$

In these equations, the subscript "m" represents the master robot, and the subscript "s" represents the slave robot. To minimize the impact of potential time delays during communication, the master robot sends position, velocity, and acceleration information to the slave robot, while the slave robot sends position, velocity, and acceleration errors to the master robot. The control signal for the slave robot, denoted by the subscripts, can be rearranged as follows:

$$\tau_s = \hat{M}(\theta_s)(\ddot{\theta}_m + K_d\dot{e}_s + K_p e_s) + \hat{C}(\theta_s,\dot{\theta}_s)\dot{\theta}_s + \hat{N}(\theta_s) \tag{24}$$

When designing the control signal for the master robot, a threshold value ε for the error signal has been introduced and the determination of this value is explained in the subsequent section. According to this value, the applied torque signal is defined as follows:

$$\tau_h = \begin{cases} 0, & |e_m| < \varepsilon \\ \hat{M}(\theta_m)(\ddot{\theta}_s + K_d\dot{e}_m + K_p e_m) + & \\ \hat{C}(\theta_m, \dot{\theta}_m)\dot{\theta}_m + \hat{N}(\theta_m), & |e_m| \geq \varepsilon \end{cases} \quad (25)$$

These torque signals carry information about the dynamics of the robot and the obstacle. A set of equations must be written to describe how the information about the obstacle is transferred from the slave robot to the master robot. First, the torque equation from the system dynamics must be written for both the slave (τ_{sd}) and the master (τ_{md}) robot.

$$\tau_s = \hat{M}(\theta_s)(\ddot{\theta}_m + K_d\dot{e}_s + K_p e_s) + \hat{C}(\theta_s, \dot{\theta}_s)\dot{\theta}_s + \hat{N}(\theta_s) \quad (26)$$

$$\tau_{sd} = \hat{M}(\theta_s)(\ddot{\theta}_s) + \hat{C}(\theta_s, \dot{\theta}_s)\dot{\theta}_s + \hat{N}(\theta_s) \quad (27)$$

$$\tau_{so} = \hat{M}(\theta_s)(\ddot{e}_s + K_d\dot{e}_s + K_p e_s) \quad (28)$$

$$\tau_s = \tau_{sd} + \tau_{so} \quad (29)$$

where τ_{so} is the torque caused by the obstacle at the end of the effector or any joint.

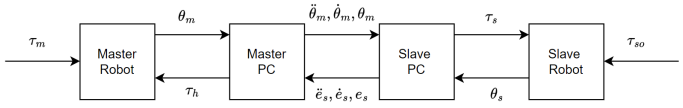

Fig. 5 Information flow of the teleoperation system

$$\tau_h = \hat{M}(\theta_m)(\ddot{\theta}_s + K_d\dot{e}_m + K_p e_m) + \hat{C}(\theta_m, \dot{\theta}_m)\dot{\theta}_m + \hat{N}(\theta_m) \quad (30)$$

$$\tau_{md} = \hat{M}(\theta_m)(\ddot{\theta}_m) + \hat{C}(\theta_m, \dot{\theta}_m)\dot{\theta}_m + \hat{N}(\theta_m) \quad (31)$$

$$\tau_h = \tau_{md} + \hat{M}(\theta_m)(\ddot{e}_m + K_d\dot{e}_m + K_p e_m) \quad (32)$$

$$\tau_h = \tau_{md} - \hat{M}(\theta_m)(\ddot{e}_s + K_d\dot{e}_s + K_p e_s) \quad (33)$$

$$\tau_h = \tau_{md} - \frac{\hat{M}(\theta_m)}{\hat{M}(\theta_s)}\tau_{so} \quad (34)$$

$$\tau_h = \tau_{md} + \tau_m \quad (35)$$

where τ_m is the torque to inform the operator about the obstacle. The relation between τ_{so} and τ_m can be derived from equation (34) and equation (35).

5 Results

In the experimental setup, a master-slave system comprising two robots and two computers was established. The communication between these computers was facilitated using the UDP (User Datagram Protocol) communication protocol, adhering to a bilateral structure specifically designed for this study. The master robot's role encompassed generating the reference trajectory and delivering force feedback to the operator during the experiment. Correspondingly, the master computer was tasked with transmitting this reference trajectory to the slave computer and managing the control input to the master robot, based on feedback received from the slave computer. The slave computer, on the other hand, received the trajectory, performed the necessary calculations, and then controlled the slave robot. It also relayed important information about the slave robot back to the master computer.

The communication speed between computer and robot pairs was chosen as 1600 Hz. Communication was facilitated using the PhanTorque library in the MATLAB/Simulink^{TM} platform ([26]). This platform allows us to send torque commands to the manipulator and receive joint angular position, Cartesian position, and Cartesian velocity. Joint angular velocity, Cartesian velocity, and the Jacobian matrix are calculated by using Simulink. This platform, based on S-functions, is perceived not as hardware but as a simulation model. In systems operating as simulations, there is no fixed time between two steps, but the time required for the necessary calculations. Similarly, the completion of the simulation is related not to the simulation time but to the completion of the calculations. However, in Hardware-in-the-Loop (HIL) systems, time must flow at the same speed and in equal steps for both simulation and hardware in the loop. To address this issue, the Real-Time Desktop application in Simulink^{TM} was employed.

The experimental study demonstrated the success of the proposed control method. Graphs in Figure 6, Figure 7, and Figure 8 show the robot positions for each joint (top), the error signals (middle), and the torque signals applied to the joints (bottom), respectively. The error values for each joint are analyzed separately to determine if they surpass a predetermined threshold. As a result, when an obstacle is encountered, the torque signals generated in response are specific to the affected joints, rather than influencing all joints. This selective response ensures that only the relevant joints react to the obstacle. To demonstrate how the system reacts to an obstacle, the slave robot was made to touch a nearby sponge surface in a perpendicular and parallel manner. This physical constraint allowed the response of the system to be observed and analyzed in the presence of an obstacle.

In the development of the force controller, selecting an optimal threshold value is critical for balance. A higher threshold value facilitates smoother maneuvering but may delay obstacle detection. In contrast, a lower threshold value enhances the speed of obstacle detection yet can hinder maneuverability in the absence of obstacles. Thus,

a judicious choice was made in alignment with the efficacy of the position controller. Given that the position controller effectively limited errors to approximately [2, 6, 6] degrees, the threshold was consequently set to around [3, 7, 7] degrees. As illustrated in Figure 6, the slave robot's first

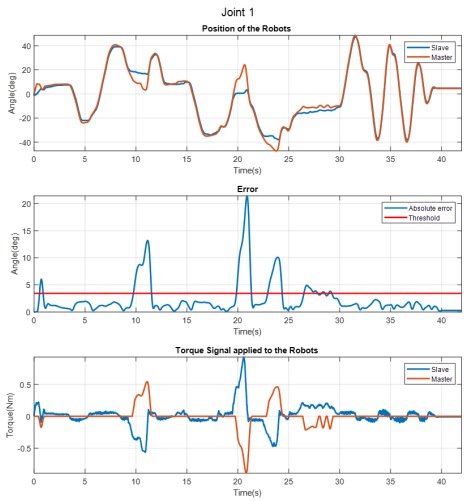

Fig. 6 Results of bilateral teleoperation for the first joint

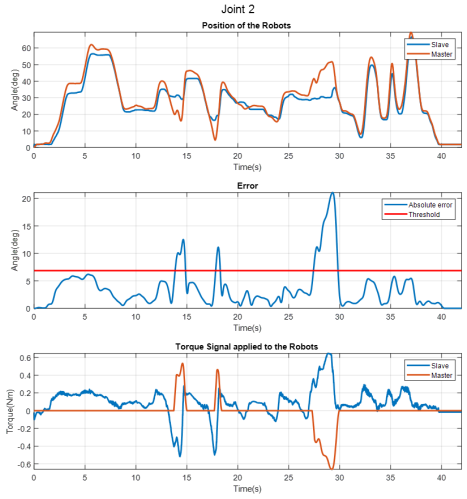

Fig. 7 Results of bilateral teleoperation for the second joint

joint can track the master robot's movements successfully during obstacle-free intervals (approximately between 0-9, 12-18, and 30-45 seconds). However, during periods when an obstacle is encountered (around 9-12 and 18-30 seconds), the slave robot experiences difficulty in following the master robot. This difficulty leads to an increase in the control input to the slave robot. Simultaneously, control input is

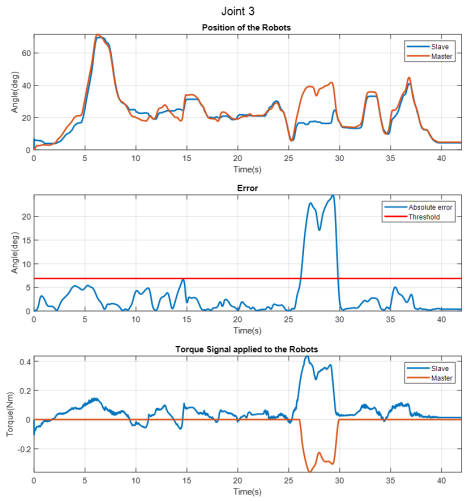

Fig. 8 Results of bilateral teleoperation for the third joint under

also applied to the master robot, facilitating force feedback to the operator, thereby signaling the presence of a potential obstacle. For the second joint, as depicted in Figure 7, encounters with obstacles occurred approximately between 14-15, 17-18, and 27-30 seconds. Regarding the third joint, obstacle encounters are observed between 25-30 seconds, as shown in Figure 8. In both the second and third joints, the controller exhibits similar responses to those observed in the first joint.

5.1 Effect of communication delay

The results obtained in the previous section show that the communication delays that are already present in the system do not degrade the controller's performance. Then, we tested our method with a round-trip communication delay of 500 ms on both sides while sending the signal between the computers, i.e. adding 1 s delay to the system in total, are analyzed. The results are shown in Figure 9, 10 and 11. Since the system is recorded from a single location during operation, the positions of the robots appear to be without delay. However, when looking at the torque graph, it is obvious that there is a delay in the system. When the slave robot did not encounter any obstacle, it provided acceptable results that demonstrated the effectiveness of the method. No torque was applied to the master robot during the time no obstacle was encountered. When the slave robot encounters an obstacle, the torque applied to the relevant joint of the slave robot increases. However, the torque exerted on the relevant joint of the master robot is applied with a subsequent delay due to the additional latency, and this time lag between the torque signals is evident in the graph (Figures 9, 10, and 11). The time duration, t_e, it takes for the error to decrease below the threshold value

after passing the threshold value can be considered as a performance criterion. Although the time of encountering and being exposed to the obstacle is random, the operator has tried to move away from the obstacle. It can be seen from the graphs that the added extra delay has no obvious effect on the t_e time.

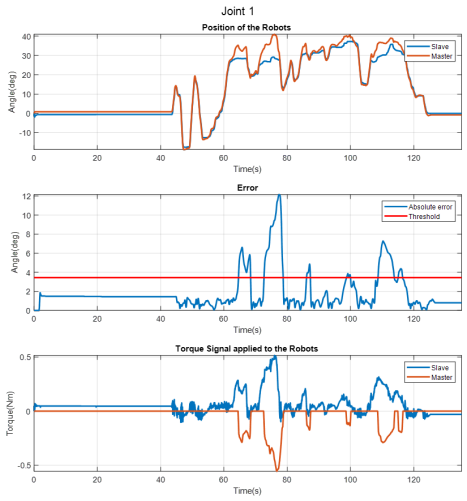

Fig. 9 Results of bilateral teleoperation under extra communication delay for the first joint

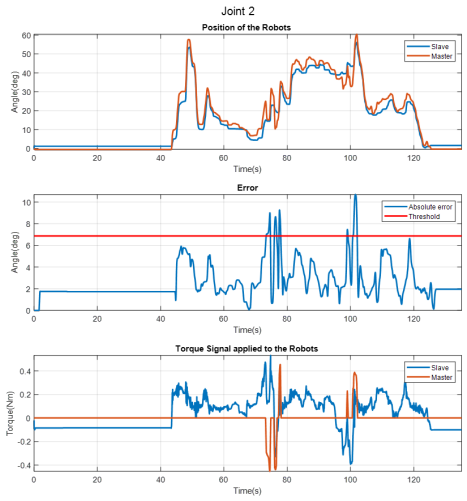

Fig. 10 Results of bilateral teleoperation under extra communication delay for the second joint

6 Conclusion

In this study, a haptic teleoperation system was designed for an RRR-type robot manipulator using the computed

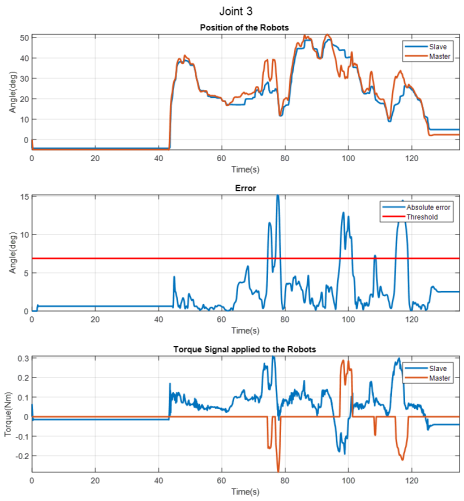

Fig. 11 Results of bilateral teleoperation under extra communication delay for the third joint

torque control method. According to the stability analysis, it was shown that the error dynamics are asymptotically stable, and the error signal converges to zero. When examining the experimental results, it can be observed that force feedback is not applied to the operator when the tracking task is successfully performed, and force feedback is applied to the operator when an obstacle is encountered. Extra communication delay between computers did not affect the controller's performance.

Since a data acquisition card was not used during the study, delays due to processor-related issues on the computer can occur in this model, which operates directly on the computer's processor. Additionally, because the communication proceeds through API-based requests and responses, this can also lead to delays. To make real-time communication more reliable, serial communication should be used instead of API-based libraries by finding and using parameter register addresses. Furthermore, the use of a data acquisition card or the assignment of specific processor cores for this process could enhance real-time communication to a more reliable level.

In future studies, the use of industrial robots and adaptive controller design against parametric uncertainties will be considered.

References

[1] M. Vukobratovic and V. Potkonjak, *Dynamics of Manipulation Robots: Theory and Application*. Springer-Verlag, 1982.

[2] K. S. Fu and R. C. Gonzalez, *Robotics: Control, Sensing, Vision, and Intelligence*. McGraw-Hill, 1987.

[3] L. Sciavicco and B. Siciliano, *Modelling and Control of Robot Manipulators*. Springer Science & Business Media, 2001.

[4] A. N. Sharkawy and P. Koustoumpardis, "Dynamics and computed-torque control of a 2-dof manipulator: Mathematical analysis," *International Journal of Advanced Science and Technology*, vol. 28, no. 12, pp. 201–212, 2019.

[5] T. Q. Nguyen, V. T. Phan, D. T. Vo, *et al.*, "Kinematics, dynamics and control design for a 4-dof robotic manipulator," in *2021 International Conference on System Science and Engineering (ICSSE)*, 2021, pp. 138–144.

[6] T. Sansanayuth, I. Nilkhamhang, and K. Tungpimolrat, "Teleoperation with inverse dynamics control for phantom omni haptic device," in *2012 Proceedings of SICE Annual Conference (SICE)*, 2012, pp. 2121–2126.

[7] A. Urgan and J. Daşdemir, "Hesaplanmış tork kontrolü yöntemi ile rrr tipi robot manipülatörünün yörünge kontrolü," in *Elektrik-Elektronik ve Biyomedikal Mühendisliği Konferansı (ELECO 2022)*, Nov. 2022.

[8] W. Wei and Y. Kui, "Teleoperated manipulator for leak detection of sealed radioactive sources," in *IEEE International Conference on Robotics and Automation (ICRA'04)*, vol. 2, 2004, pp. 1682–1687.

[9] J. W. Hills and J. F. Jensen, "Telepresence technology in medicine: Principles and applications," *Proceedings of the IEEE*, vol. 86, no. 3, pp. 569–580, 1998.

[10] R. Taylor and D. Stoianovici, "Medical robotics in computer-integrated surgery," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 5, pp. 765–781, 2003.

[11] M. Tavakoli, R. V. Patel, and M. Moallem, "Haptic interaction in robot-assisted endoscopic surgery: A sensorized end-effector," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 1, no. 2, pp. 53–63, 2005.

[12] J. Funda and R. P. Paul, "A symbolic teleoperator interface for time-delayed underwater robot manipulation," in *OCEANS 91 Proceedings*, 1991, pp. 1526–1533.

[13] W. K. Yoon, T. Goshozono, H. Kawabe, *et al.*, "Model-based space robot teleoperation of ets-vii manipulator," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 602–612, 2004.

[14] D. Lee, O. Martinez-Palafox, and M. W. Spong, "Bilateral teleoperation of a wheeled mobile robot over delayed communication network," in *Proceedings 2006 IEEE International Conference on Robotics and Automation (ICRA 2006)*, 2006, pp. 3298–3303.

[15] S. Avila-Becerril, G. Espinosa-Pérez, E. Panteley, and R. Ortega, "Consensus control of flexible-joint robots," *International Journal of Control*, vol. 88, no. 6, pp. 1201–1208, 2015.

[16] E. N. Ortega and L. B. Villaluenga, "Haptic guidance with force feedback to assist teleoperation systems via high-speed networks," in *37th International Symposium on Robotics (ISR)*, VDI Verlag, 2006.

[17] D. A. Lawrence, "Stability and transparency in bilateral teleoperation," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 3, pp. 624–637, 1993.

[18] I. Aliaga, A. Rubio, and E. Sanchez, "Experimental quantitative comparison of different control architectures for master-slave teleoperation," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 1, pp. 2–11, 2004.

[19] K. Tindell, A. Burns, and A. Wellings, "Analysis of hard real-time communications," *Real-Time Systems*, vol. 9, pp. 147–171, 1995.

[20] G. Niemeyer and J.-J. E. Slotine, "Telemanipulation with time delays," *The International Journal of Robotics Research*, vol. 23, no. 9, pp. 873–890, 2004.

[21] N. Piccinelli and R. Muradore, "A bilateral teleoperation with interaction force constraint in unknown environment using non linear model predictive control," *European Journal of Control*, vol. 62, pp. 185–191, 2021.

[22] E. Nuño, L. Basañez, and R. Ortega, "Passivity-based control for bilateral teleoperation: A tutorial," *Automatica*, vol. 47, no. 3, pp. 485–495, 2011.

[23] A. J. Silva, O. A. D. Ramirez, V. P. Vega, and J. P. O. Oliver, "Phantom omni haptic device: Kinematic and manipulability," pp. 193–198, 2009.

[24] A. Nygaard, "High-level control system for remote controlled surgical robots," M.S. thesis, Norwegian University of Science and Technology, 2008.

[25] F. L. Lewis, D. M. Dawson, and C. T. Abdallah, *Robot Manipulator Control: Theory and Practice*. CRC Press, 2003.

[26] C. I. Aldana, E. Nuño, L. Basañez, and E. Romero, "Operational space consensus of multiple heterogeneous robots without velocity measurements," *Journal of the Franklin Institute*, vol. 351, no. 3, pp. 1517–1539, 2014.