Neural Network Control of Teleoperation Systems with Delay and Uncertainties based on Multilayer Perceptron Estimations

Parham M. Kebria, Abbas Khosravi, and Saeid Nahavandi
Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University
Waurn Ponds, VIC 3216, Australia
{parham.kebria,abbas.khosravi,saeid.nahavandi}@deakin.edu.au

***Abstract*—This paper investigates a novel synchronisation strategy for controlling Internet-based teleoperation systems. These kinds of systems considerably suffer from network-induced latencies. Random time-varying delays resulted by the Internet deteriorate the stability and performance of teleoperation processes. Moreover, uncertain dynamic elements, including human operators and partially known remote environments introduce further difficulties to the control design of such systems. Utilising the learning capabilities of artificial neural networks, this paper develops an adaptive algorithm to deal with time-delays and uncertainties negatively affecting an Internet-based teleoperation process. The stable convergence of the proposed control algorithm is proved by Lyapunov-Krasovskii stability criteria. Moreover, the robust performance of the controller is also verified via experimental evaluations.**

***Index Terms*—Neural network, multilayer perceptron, teleoperation systems, time-delays, uncertainties**

I. INTRODUCTION

Teleoperation technology has recently enabled the human being to do difficult or impossible tasks that are extremely delicate, dangerous or spatially inaccessible [1]–[3]. Generally, teleoperation systems consist of a master device receiving the desired task commands directly from the human operator and a slave system. The slave system is also called the teleoperator and its duty is to perform the desired task in the remote environment (Figure 1). Uncertain dynamics and partially known elements, including the human operator and the remote workspace, severely make the control design for these systems a very problematic challenge. On the other hand, the master and slave systems transfer task commands and sensory information through a communication network. Stochastic latencies induced by the network further threatens the stability of the teleoperation process. Several research studies and researchers have been investigating solutions for the teleoperation control problem.

Control approaches based on computational intelligence techniques have shown much effective performance compared to conventional controllers [4]. For designing an effective controller for teleoperation systems, it is required to derive a reliable and efficiently accurate model of all elements involved in the teleoperation process. However, derivation of such a model for the human operator and the remote environment, for example, is a troublesome task by itself [5]–[12]. Learning and approximation capabilities of artificial neural networks (NN) provided researchers to develop control algorithms to deal with uncertainties and nonlinearities in complex applications, such as teleoperation systems [13]–[17]. Researchers in [18] proposed an adaptive NN control for teleoperation systems, however, under constant time-delay. The adaptive NN control developed in [12] enhanced the position tracking performance of an uncertain teleoperation, however, under constant time-delays. Targeting model uncertainties in master and slave subsystems, researchers have developed NN-based adaptive control algorithms for teleoperation applications [7], [18]–[20]. The solution proposed in [16] improved the performance of the teleoperation process under uncertainties and disturbances. [21] has proposed a nonlinear mapping algorithm for teleoperation, however, precise model of the systems is required.

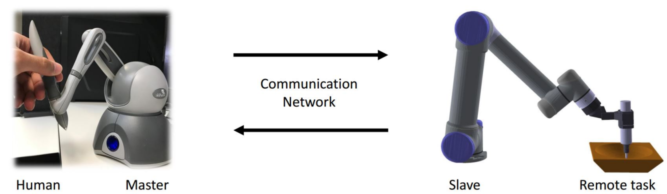

Fig. 1. General schematic of a teleoperation system.

Furthermore, uncertainties and time-delays are not the only major concerns. Also, transparency and haptic are other main performance objectives in teleoperation applications [4], [22]. The efficiency of haptic requires an effective force-feedback, which is considerably vulnerable to time-delays. Employing neural-networks for pattern extraction through the sensed force signals, and also, estimating the future trends of the network delay improved the performance of teleoperation systems in terms of transparency [7], [23], [24]. However, most of these studies in the literature have considered teleoperation systems with fully-known dynamics and kinematics. Kinematic dissimilarity between the master and slave systems is also another challenging topic in teleoperation applications [25], [26]. Therefore, investigation for an effective control strategy for guaranteeing the robust stability and performance

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

of teleoperation systems is still a question to be answered.

Taking advantage of learning potentials of NN, this paper proposes an adaptive technique to deal with instability and performance degradation due to uncertainties and time-varying delays on Internet-based teleoperation systems. To achieve this goal, the main contributions of this study are listed as:

- A multilayer perceptron (MLP) methodology is employed for learning and estimation purposes. Although most of the existing NN-based control strategies in the literature of teleoperation systems have used radial basis function (RBF) NN structures [7], superiority of the MLP networks in estimation tasks has been proven in [27], [28]. Hence, in this paper, MLP networks are considered for estimation purposes for dealing with delay and uncertainties in teleoperation processes.
- In addition to learning capacities of MLP networks, an adaptive online learning algorithm is also proposed to improve the real-time performance of the system. Moreover, global convergence and stability of the online learning algorithm is mathematically proved by Lyapunov-Krasovskii theorem.
- Employing MLP algorithm, the proposed adaptive NN strategy provides either master and slave systems with a smooth and possibly the most accurate estimation of their corresponding reference signals.
- Evaluation of the proposed adaptive NN controller is experimentally carried out on a real-world teleoperation setup established over commercial Internet network.

The paper is organised as follows. Section II describes the teleoperation problem considered in this study. Derivation of the proposed adaptive NN control algorithm is detailed in Section III, followed by the experimental evaluation in Section IV. Finally, conclusion remarks and future directions of the study are discussed in Section V.

II. Problem Description

This section studies the dynamical expressions and time-delay considerations in an Internet-based teleoperation system schemed in Figure 1.

A. Teleoperation Dynamics

Generally, dynamic equations of the master and slave subsystems are expressed as:

$$\begin{aligned} M_m(q_m)\ddot{q}_m + C_m(q_m,\dot{q}_m)\dot{q}_m + g_m(q_m) &= \\ \tau_m + J_m^T(q_m) f_h + \mu_m & \\ M_s(q_s)\ddot{q}_s + C_s(q_s,\dot{q}_s)\dot{q}_s + g_s(q_s) &= \\ \tau_s - J_s^T(q_s) f_e + \mu_s & \end{aligned} \tag{1}$$

in which, q_i, $\dot{q}_i$, and $\ddot{q}_i$ ($i \in \{m$: master, s: slave$\}$) are angular position, velocity and acceleration of the joints, $M_i(q_i)$ is the inertia matrix, $C_i(q_i, \dot{q}_i)$ is the Coriolis and centripetal matrix, $g_i(q_i)$ is the gravity vector, μ_i is uncertain torques including friction, unmodelled dynamics, and external disturbances. Notably, this torque vector depends on systems' states q_i, $\dot{q}_i$, $\ddot{q}_i$, and time t, i.e. $\mu_i(q_i, \dot{q}_i, \ddot{q}_i, t)$. τ_i is the input control torque with unknown external disturbances τ_i^d, respectively. Every term in (1) is prone to uncertainties. Derivation of (1) involves enormous mathematical and computational effort that depending on the master and slave systems' dynamics might be practically impossible in some cases. In this regard, reference [29] has proposed a symbolic modelling toolbox for robotic manipulators. f_h and f_e are the Cartesian forces exerted by the human operator and remote environment, J_i^T the Jacobian matrix of the manipulators. The external Cartesian forces are modelled by non-homogeneous mass-spring-damper dynamics [4], [9]:

$$\begin{cases} f_h = f_{h0} - M_h\ddot{\mathbf{x}}_m - B_h\dot{\mathbf{x}}_m - K_h\mathbf{x}_m \\ f_e = f_{e0} + M_e\ddot{\mathbf{x}}_s + B_e\dot{\mathbf{x}}_s + K_e\mathbf{x}_s \end{cases} \tag{2}$$

where, M_j, B_j, and K_j $(j \in \{h, e\})$ are the mass, damping, and spring coefficients, with the non-homogeneous term f_{j0}. All these parameters are (partially) unknown and introduce severe uncertainties to the teleoperation performance. $\mathbf{x}_i$, $\dot{\mathbf{x}}_i$, and $\ddot{\mathbf{x}}_i$ $(i \in \{m, s\})$ are respectively Cartesian position, velocity, and acceleration of the master and slave systems. These variables are in relation with joint variables based on the corresponding forward kinematics and Jacobian:

$$\begin{cases} \mathbf{x}_i = \mathbf{p}_i(q_i) \\ \dot{\mathbf{x}}_i = J_i(q_i)\dot{q}_i \\ \ddot{\mathbf{x}}_i = \dot{J}_i(q_i)\dot{q}_i + J_i(q_i)\ddot{q}_i \end{cases} \tag{3}$$

Jacobian expression in (3) is another source of uncertainty that requires a great deal of attention in terms of singularity and feasibility of the commanded teleoperation task [25]. Equation (1) is the expression of dynamics of the teleoperation system in joint (q_i) space. However, the desired teleoperation tasks are generally defined and commanded in the Cartesian workspace of the master and slave robots [25]. Hence, the Cartesian equivalent of (1) is expressed as:

$$\mathbf{M}_i\ddot{q}_i + \mathbf{C}_i(q_i,\dot{q}_i)\dot{q}_i + \mathbf{g}_i(q_i) = \tau_i + \mathbf{f}_j + \mu_i \tag{4}$$

in which

$$\mathbf{M}_i = M_i + J_i^T M_j J_i \quad , \qquad \mathbf{C}_i = C_i + J_i^T M_j \dot{J}_i + J_i^T B_j J_i$$

$$\mathbf{g}_i = g_i + J_i^T K_j \mathbf{p}_i \quad , \qquad \mathbf{f}_j = J_i^T f_{j0}$$

$(i, j) \in \{(m, h), (s, e)\}$. It should be noted that the uncertainty vectors μ_i are upper-bounded because of mechanical limitations of the systems. In other words, q_i, $\dot{q}_i$, and $\ddot{q}_i$ are physically limited and cannot be unbounded, and therefore, $\|\mu_i\| \leq \Gamma_i$.

B. Transmission Delay in Teleoperation

In a teleoperation system, the Reference signals are transmitted trough the communication channels and are affected by the delays:

$$\begin{cases} \mathbf{x}_s^*(t) = \mathbf{x}_m^d(t) = \mathbf{x}_m(t - T_f(t)) \\ f_h^*(t) = f_e^d(t) = f_e(t - T_b(t)) \end{cases} \tag{5}$$

with $T_f(t)$ and $T_b(t)$ as forward and backward time-varying delays, respectively. In this paper, we employ a neural network-based prediction method for smoothly approximating the delayed reference signals.

III. Adaptive Neural Control

This section describes the derivation of the proposed adaptive NN control algorithm for uncertain Internet-based teleoperation systems. Left-hand side of (4) has a property called linearity in parameters [7], [20]; i.e.:

$$\mathbf{M}_i\ddot{q}_i + \mathbf{C}_i(q_i,\dot{q}_i)\dot{q}_i + \mathbf{g}_i(q_i) = \Psi_i(q_i,\dot{q}_i,\ddot{q}_i)\vec{\theta}_i \quad (6)$$

with the parameters vector $\vec{\theta}_i$ and the regressor matrix Ψ_i. This property is very useful in adaptive control design practices for these systems [7], [18]–[20]. Combining (4) and (6) we have:

$$\Psi_i(q_i,\dot{q}_i,\ddot{q}_i)\vec{\theta}_i = \tau_i + \mathbf{f}_j + \mu_i \quad (7)$$

Assuming that the desired reference signals result in the desired regressor Ψ_i^* and parameters $\vec{\theta}_i^*$, and subsequently, the desired control torque τ_i^*,

$$\Psi_i^*(q_i,\dot{q}_i,\ddot{q}_i)\vec{\theta}_i^* = \tau_i^*$$

Utilising NN proficiency, we develop an adaptive control algorithm to estimate the desired torques τ_i^* by $\hat{\tau}_i$ in the presence of unknown external elements $\mathbf{f}_j$ and μ_i. In other words, the neural adaptive strategy generates $\hat{\tau}_i$ to effectively approximate τ_i^* that guarantees the stable performance of the teleoperation system. Accordingly, $\hat{\tau}_i$ will also compensate for the time-delays and uncertainties taking place on the teleoperation process.

A. Multilayer Perceptron Neural Networks

Among the NN architectures, multilayer perceptron (MLP) is one of the most commonly used NN techniques [30]. Here, we employ MLP for the estimation purpose in this study. In MPL networks, nodes (neurons) are hierarchically structured in several fully connected layers. The first and the last layers are respectively called the input and output layers. The layers sit in between are called hidden layers. Hidden layers carry out the main computational tasks to approximate the desired output [31]. Fig 2 schemes the MLP structure with the corresponding layers and functions used in this study.

In our application, output of the MLP is derived as:

$$\xi_i = W_1^i e_i + \beta_1^i \quad , \quad \delta_l^i = \frac{1}{1+e^{-\xi_i}} \quad , \quad \hat{\gamma}_i = W_2^i\delta_l^i + \beta_2^i \quad (8)$$

in which, $e_s = x_s^* - \hat{x}_s$ for the slave robot, and $e_m = f_h^* - \hat{f}_h$ for the master device, are the training errors being fed as the network input. ξ_i is the output of nodes (neurons) in the first layer. $W_1^i = [w_1^i]_{m\times n}$ and $W_2^i = [w_2^i]_{o\times m}$ are the weight matrices and β_1^i and β_2^i are the biases for each layer. m and n are the number of nodes (neurons) in the antecedent and consequent layers, respectively. l is the number of hidden layers with the corresponding activation (sigmoid) function δ_l^i.

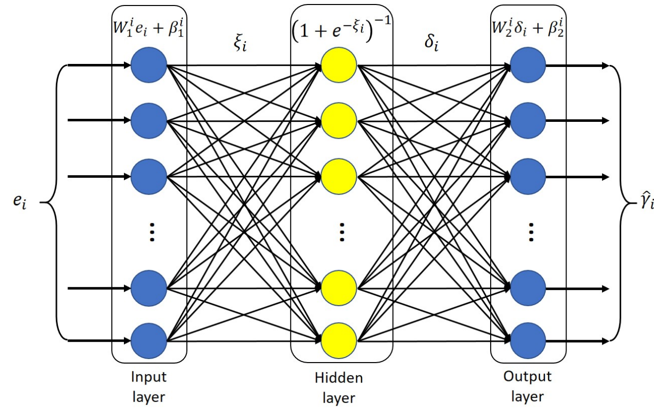

Fig. 2. Schematic of the MLP neural network used in this study.

The cost function $E_i = \frac{1}{2}e_i^T e_i$ is considered for the gradient-based updating rules as follow ($p = 1\ldots n, q = 1\ldots m$):

$$\begin{aligned} w_{1p}^i(t+1) &= \alpha_i w_{1p}^i(t) + \lambda_i(t)\frac{\partial E_i}{\partial w_{1p}^i} \\ w_{2q}^i(t+1) &= \alpha_i w_{2q}^i(t) + \lambda_i(t)\frac{\partial E_i}{\partial w_{2q}^i} \end{aligned} \quad (9)$$

with the adaptive learning rate $\lambda_i(t)$ and a sufficiently small positive constant α_i. Derivation of the gradient terms and the adaptation law of the learning rate in (9) is later detailed in this section. Calculating the one-step derivative of (9) gives:

$$\begin{aligned} \dot{w}_{1p}^i &= (\alpha_i - 1)w_{1p}^i + \lambda_i\frac{\partial E_i}{\partial w_{1p}^i} \\ \dot{w}_{2q}^i &= (\alpha_i - 1)w_{2q}^i + \lambda_i\frac{\partial E_i}{\partial w_{2q}^i} \end{aligned} \quad (10)$$

Then, we will compute the partial derivative terms in (10) in order to stably minimise the cost functions (errors). To avoid getting stuck in local minima, we apply Lyapunov-Krasovskii theorem to derive the adaptive learning laws. For this purpose, we propose the Lyapunov-Krasovskii functional below based on the cost function E_i:

$$\begin{aligned} V_i =& E_i + \int_{t-T_{f,b}(t)}^{t} E_i d\sigma + \int_{t-T_{f,b}(t)}^{t} \dot{E}_i d\sigma \\ &+ \int_{-T_{max}}^{0}\int_{t+\delta}^{t} E_i d\delta d\sigma + \frac{1}{2}\tilde{\lambda}_i^2 \end{aligned} \quad (11)$$

in which, T_{max} is the maximum delay happening through the communication network, and $\tilde{\lambda}_i = \lambda_i - \lambda_i^*$ is the adaptation error of the learning rate λ_i to reach its desired value λ_i^*. To obtain the globally-optimal update laws (10) we calculate the time-derivative of the Lyapunov-Krasovskii function V_i to examine its negative definiteness. Differentiation of V_i gives:

$$\begin{aligned} \dot{V}_i =& \dot{E}_i + E_i^2 - (1-\dot{T}_{f,b})E_i(t-T_{f,b})^2 + \dot{E}_i^2 \\ &- (1-\dot{T}_{f,b})\dot{E}_i(t-T_{b,f})^2 + T_{max}E_i^2 \\ &- \int_{t+\delta}^{t} E_i^2 d\delta + \dot{\tilde{\lambda}}_i\tilde{\lambda}_i \end{aligned} \quad (12)$$

Knowing that $T_{max} < \infty$ and $\dot{T}_{f,b} \leq 1$ [9], and the fact that reference signals are physically limited due to mechanical constraints of the robotic systems $\max(\|x_s^*\|, \|f_h^*\|, \|\dot{x}_s^*\|, \|\dot{f}_h^*\|) \leq \Delta < \infty$;

$$\dot{V}_i \leq \dot{E}_i + (1 + T_{max})E_i^2 + \dot{E}_i^2 + \dot{\tilde{\lambda}}_i \tilde{\lambda}_i \tag{13}$$

Considering $\lambda_i^* > (1 + T_{max})$, and proposing the adaptive learning rate as:

$$\dot{\lambda}_i = \dot{\tilde{\lambda}}_i = E_i^2 = \frac{1}{4}\|e_i\|^4 \tag{14}$$

(13) becomes:

$$\dot{V}_i < \dot{E}_i + \dot{E}_i^2 + \lambda_i E_i^2 \tag{15}$$

Applying chain rule and (10) results in:

$$\dot{V}_i < \sum_{p=1}^{n} \frac{\partial E_i}{\partial w_{1p}^i} \dot{w}_{1p}^i + \sum_{q=1}^{m} \frac{\partial E_i}{\partial w_{2q}^i} \dot{w}_{2q}^i + \dot{E}_i^2 + \lambda_i E_i^2 \tag{16}$$

substituting (10) into (16):

$$\begin{aligned} \dot{V}_i <& \sum_{p=1}^{n} \frac{\partial E_i}{\partial w_{1p}^i} \Big((\alpha_i - 1) w_{1p}^i + \lambda_i \frac{\partial E_i}{\partial w_{1p}^i} \Big) \\ &+ \sum_{q=1}^{m} \frac{\partial E_i}{\partial w_{2q}^i} \Big((\alpha_i - 1) w_{2q}^i + \lambda_i \frac{\partial E_i}{\partial w_{2q}^i} \Big) \\ &+ \dot{E}_i^2 + \lambda_i E_i^2 \end{aligned} \tag{17}$$

Hereafter, we only present the procedure for one weight w_{1p}^i as the process is exactly identical for all other weights. Therefore, simplifying (17) results in:

$$\begin{aligned} \dot{V}_i <& \frac{\partial E_i}{\partial w_{1p}^i} \Big((\alpha_i - 1) w_{1p}^i + \lambda_i \frac{\partial E_i}{\partial w_{1p}^i} \Big) \\ &+ \frac{1}{n+m} \Big(\dot{E}_i^2 + \lambda_i E_i^2 \Big) \end{aligned} \tag{18}$$

which can be rewritten in a quadratic form:

$$\begin{aligned} \dot{V}_i <& \underbrace{\lambda_i}_{a_{1p}^i} \Big(\frac{\partial E_i}{\partial w_{1p}^i} \Big)^2 + \underbrace{(\alpha_i - 1) w_{1p}^i}_{b_{1p}^i} \frac{\partial E_i}{\partial w_{1p}^i} \\ &+ \underbrace{\frac{1}{n+m} \Big(\dot{E}_i^2 + \lambda_i E_i^2 \Big)}_{c_{1p}^i} \end{aligned} \tag{19}$$

and then solving for $\frac{\partial E_i}{\partial w_{1p}^i}$ requires:

$$\begin{aligned} &(b_{1p}^i)^2 - 4 a_{1p}^i c_{1p}^i = \\ &\quad ((\alpha_i - 1) w_{1p}^i)^2 - \frac{4\lambda_i}{n+m} \Big(\dot{E}_i^2 + \lambda_i E_i^2 \Big) = 0 \end{aligned} \tag{20}$$

which has solutions naming $\omega_{1p,1}^i$ and $\omega_{1p,2}^i$ for w_{1p}^i. Then, the solutions of (19) for $\frac{\partial E_i}{\partial w_{1p}^i}$ will be:

$$\frac{(1 - \alpha_i)\omega_{1p,1}^i}{2\lambda_i} \quad , \quad \frac{(1 - \alpha_i)\omega_{1p,2}^i}{2\lambda_i} \tag{21}$$

Remark: Maximum solutions in (21) makes $\dot{V}_i$ much negative, and accordingly, guarantees the stable performance of the teleoperation system under the controller (8) with the adaptive learning rate (14) and weights' update laws (9). It also demonstrates the global convergence of the learning algorithms (9) and (10) that will not get stuck in local optima. It should be mentioned that the same proof procedure applies for all other weights in (8). The next section studies the effectiveness of the proposed neural adaptive control methodology on a real-world teleoperation system.

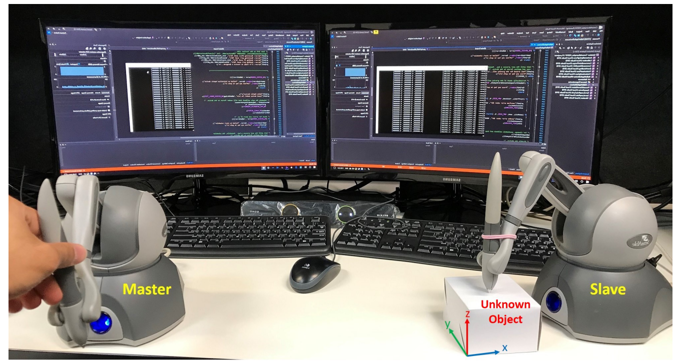

Fig. 3. The bilateral teleoperation setup considered in this study for experimental evaluations. The setup consists of two haptic devices as the master and slave subsystems. Sampling rate for data collection and control implementation is 1,000 samples per sec (1kHz).

IV. Experimental Results and Discussions

This section presents the experimental evaluation carried out on a real-world teleoperation setup under the proposed adaptive neural controllers (8). For this purpose, we have considered two Omni haptic devices as the master and slave systems. The considered teleoperation setup is shown in Fig 3. Each device has three degrees of freedom (DoF) operating in a Cartesian workspace with six DoF. The communication medium between the master and slave systems of the experimental teleoperation is established through a local area network (LAN). Based on previous experienced latency measurements in [8], [9], [11], forward and backward delays implemented between the two sides of the teleoperation setup are shown in Fig 4. At the slave side, we have placed an object with unknown dynamics (M_e, B_e and K_e). The same condition applies at the master side where the human operator is unknown (M_h, B_h and K_h). The initial values of these parameters for training and experimental purposes are considered as: $M_h = I_{3\times3} = B_h = K_h$, $M_e = 0.1 \times I_{3\times3} = B_e = K_e$. f_{e0} and f_{e0} are both considered $[0 \;\; 0 \;\; 0]^T$. It should be noted that the training parameters for the neural network are considered as 6 nodes in the input layer ($n = 6$), 18 nodes for the hidden layer ($m = 18$) and 3 nodes for the output layer ($o = 3$).

The evaluation results are accordingly shown in Figures 5 to 10. It should be mentioned that as shown in Figure 1, the human operator manipulates the unknown object through the slave device interacting with the remote environment. In this scenario, the contact forces should be as if the human operator interacts with the object directly. In other words,

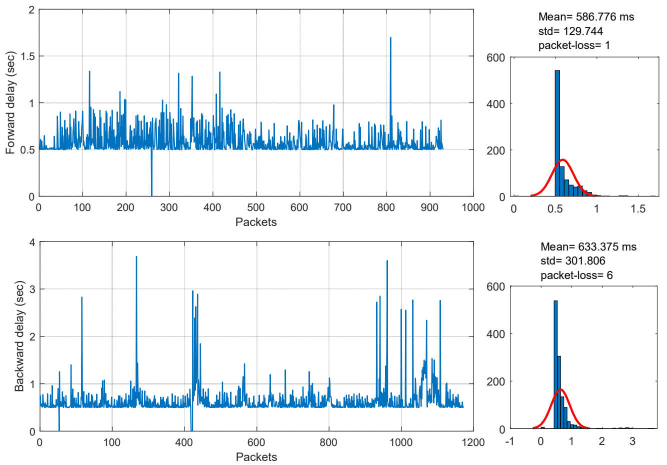

Fig. 4. The implemented large time-varying delay functions in the experimental teleoperation setup. The mean and standard deviation of the forward and backward delays, respectively, are: $\bar{T}_f = 586.776$ (ms), $\text{std}_f = 129.744$, $\bar{T}_b = 633.375$ (ms), and $\text{std}_b = 301.806$. The zero-valued samples are the lost packets during the transmission. Notably, in the forward channel one packet is lost, wile that number for the backward channel is 6 packets.

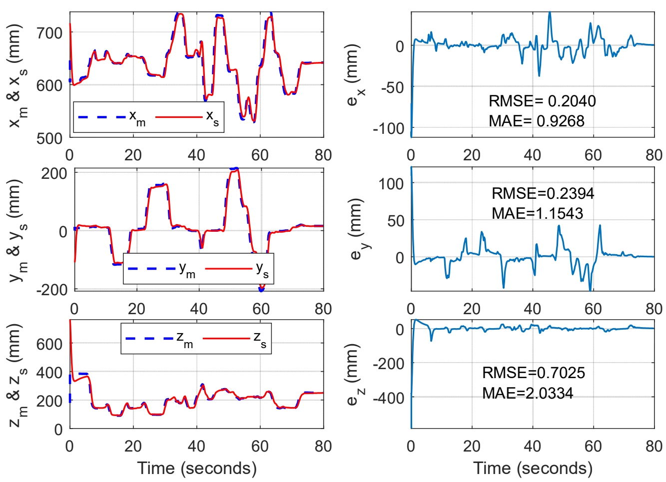

Fig. 5. Position tracking of the slave system.

forces imposed between the slave system and object are desired to be the same as those exposed between the master device and the human hand. Figure 5 illustrates the position tracking and errors of the slave robot under the developed controller. Velocity tracking of the slave robot is also shown in Figure 6. That figure shows the largest amount of errors due to the fluctuating behaviour of the motion of the human hand. Although, the controller effectively commands the robot to follow the same motion. Moreover, Figure 7 shows joints angles of the both robots to have a better understanding of the systems' dynamic behaviour. Demonstrated in Figure 8, stable force reflection of the teleoperation setup is significant due to the effective estimations provided by the proposed adaptive MLP. Eventually, convergence of the adaptive and training parameters of the controller is shown in Figure 10.

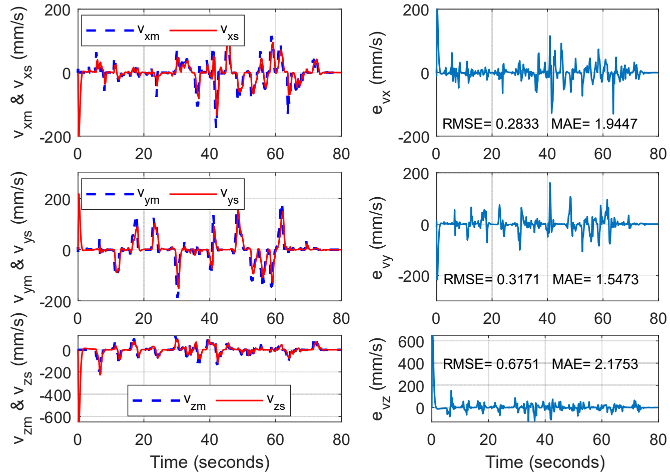

Fig. 6. Velocities of the master and slave systems.

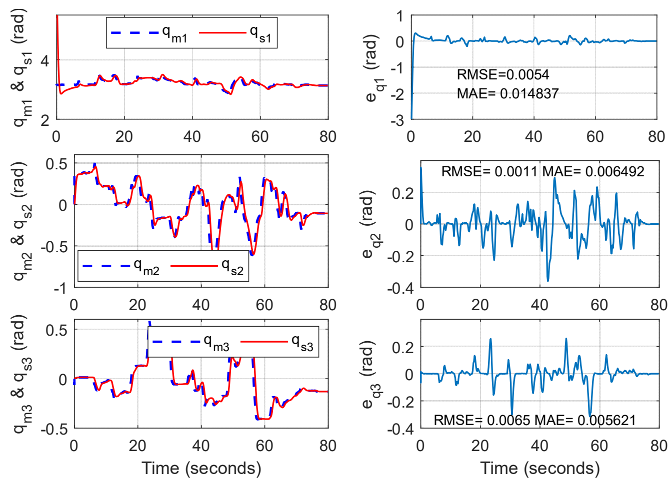

Fig. 7. Joint angles of the master and slave.

V. Conclusions

This study concerned the control design for teleoperation systems suffering from uncertainties in communication and modelling. There have been a large number of conventional control strategies for such systems, however, they mostly fail to deal with unknown situations occurring in real-world practices. On the other side, artificial neural networks have shown a promising performance in uncertain conditions and estimating complex, nonlinear models. Hence, the current article enjoyed the approximation capabilities of MLP networks and proposed a stable control methodology for uncertain teleoperation systems under randomly time-varying delays. The convergence and stability of the proposed method have been proven via Lyapunov-Krasovskii theorem. Moreover, practical effectiveness of the proposed MLP-based control technique was verified through a real-time Internet-based communication between two haptic devices. This research opens new doors to future investigations in further empowering the teleoperation applications by employing more advanced computational in-

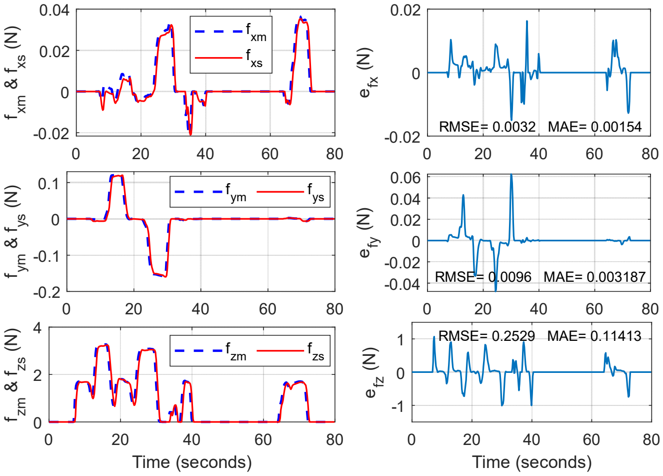

Fig. 8. Force reflection of the master device.

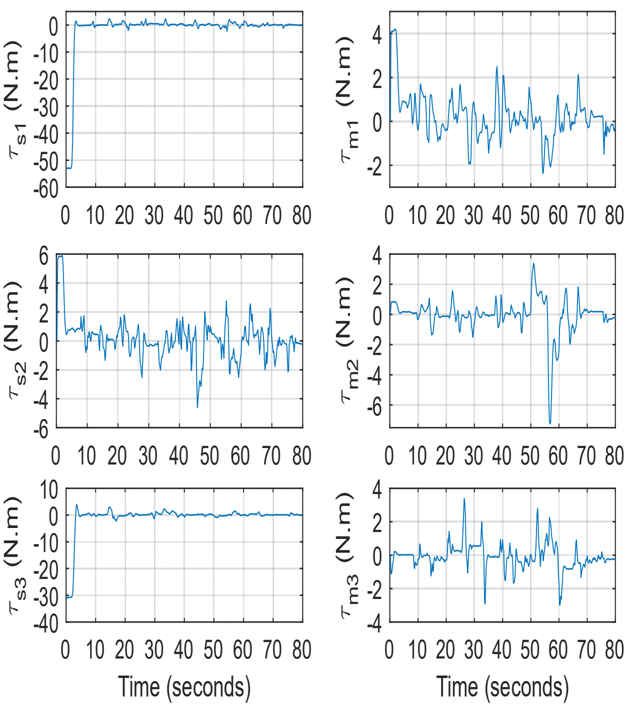

Fig. 9. Torque signals generated by the controllers.

telligence approaches.

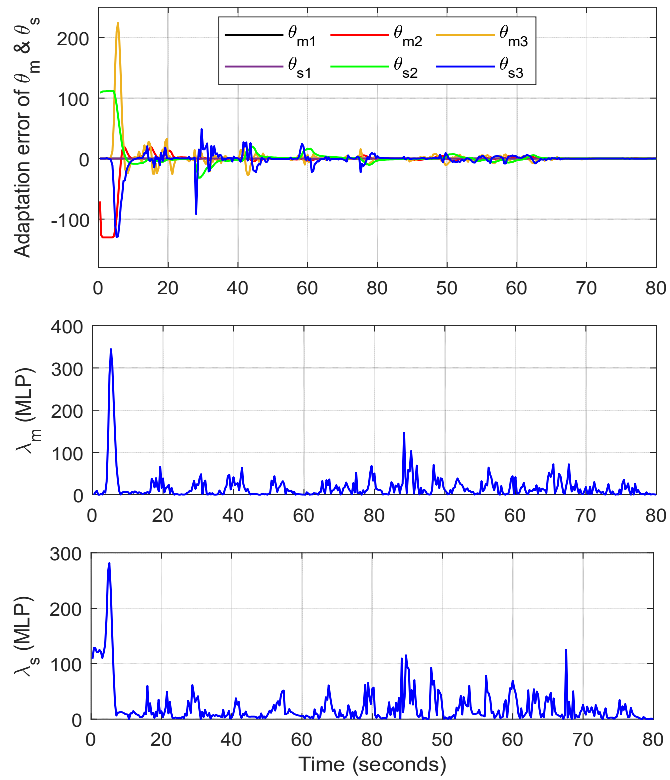

Fig. 10. Adaptive control parameters and learning rates of the two systems.

References

[1] Y. Gao and S. Chien, "Review on space robotics: Toward top-level science through space exploration," *Science Robotics*, vol. 2, no. 7, p. eaan5074, 2017.

[2] M. M. Dalvand, S. Nahavandi, M. Fielding, J. Mullins, Z. Najdovski, and R. D. Howe, "Modular instrument for a haptically-enabled robotic surgical system (herosurg)," *IEEE Access*, vol. 6, pp. 31 974–31 982, 2018.

[3] Q. Miao, A. McDaid, M. Zhang, P. Kebria, and H. Li, "A three-stage trajectory generation method for robot-assisted bilateral upper limb training with subject-specific adaptation," *Robotics and Autonomous Systems*, vol. 105, pp. 38–46, 2018.

[4] P. M. Kebria, H. Abdi, M. M. Dalvand, A. Khosravi, and S. Nahavandi, "Control methods for internet-based teleoperation systems: A review," *IEEE Transactions on Human-Machine Systems*, vol. 49, no. 1, pp. 32–46, 2019.

[5] P. M. Kebria and *et al.*, "Deep imitation learning: The impact of depth on policy performance," in *International Conference on Neural Information Processing*. Springer, 2018, pp. 172–181.

[6] ——, "Evaluating architecture impacts on deep imitation learning performance for autonomous driving," in *ICIT 2019: Proceedings of the 2019 20th IEEE International Conference on Industrial Technology*, 2019, pp. 865–870.

[7] ——, "Neural network adaptive control of teleoperation systems with uncertainties and time-varying delay," in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2018, pp. 252–257.

[8] ——, "Robust adaptive control of internet-based bilateral teleoperation systems with time-varying delay and model uncertainties," in *2019 IEEE International Conference on Industrial Technology*. IEEE, 2019, pp. 187–192.

[9] ——, "Robust adaptive control scheme for teleoperation systems with delay and uncertainties," *IEEE transactions on cybernetics (Early Access)*, pp. 1–11, 2019.

[10] P. M. Kebria, A. Khosravi, S. Nahavandi, D. Wu, and F. Bello, "Adaptive type-2 fuzzy neural-network control for teleoperation systems with delay and uncertainties (*Early Access*)," *IEEE Transactions on Fuzzy Systems*, 2019.

[11] P. M. Kebria and *et al.*, "Robust adaptive synchronisation of a single-master multi-slave teleoperation system over delayed communication," in *2019 IEEE International Conference on Industrial Technology*. IEEE, 2019, pp. 193–198.

[12] Y. Ji, D. Liu, and Y. Guo, "Adaptive neural network based position

tracking control for dual-master/single-slave teleoperation system under communication constant time delays," *ISA transactions*, 2019.

[13] A. Khosravi, S. Nahavandi, and D. Creighton, "A neural network-garch-based method for construction of prediction intervals," *Electric Power Systems Research*, vol. 96, pp. 185–193, 2013.

[14] S. Refaat, J. M. Herve, S. Nahavandi, and H. Trinh, "Asymmetrical three-dofs rotational-translational parallel-kinematics mechanisms based on lie group theory," *European journal of mechanics-A/Solids*, vol. 25, no. 3, pp. 550–558, 2006.

[15] H. Wang, P. X. Liu, and S. Liu, "Adaptive neural synchronization control for bilateral teleoperation systems with time delay and backlash-like hysteresis," *IEEE transactions on cybernetics*, vol. 47, no. 10, pp. 3018–3026, 2017.

[16] Y. Yang, C. Ge, H. Wang, X. Li, and C. Hua, "Adaptive neural network based prescribed performance control for teleoperation system under input saturation," *Journal of the Franklin Institute*, vol. 352, no. 5, pp. 1850–1866, 2015.

[17] P. M. Kebria, A. Khosravi, S. M. Salaken, and S. Nahavandi, "Deep imitation learning for autonomous vehicles based on convolutional neural networks," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 82–95, January 2020.

[18] C.-C. Hua, Y. Yang, and X. Guan, "Neural network-based adaptive position tracking control for bilateral teleoperation under constant time delay," *Neurocomputing*, vol. 113, pp. 204–212, 2013.

[19] L. Cheng, Z.-G. Hou, and M. Tan, "Adaptive neural network tracking control of manipulators using quaternion feedback," in *2008 IEEE International Conference on Robotics and Automation*. IEEE, 2008, pp. 3371–3376.

[20] ——, "Adaptive neural network tracking control for manipulators with uncertain kinematics, dynamics and actuator model," *Automatica*, vol. 45, no. 10, pp. 2312–2318, 2009.

[21] P. Malysz and S. Sirouspour, "Nonlinear and filtered force/position mappings in bilateral teleoperation with application to enhanced stiffness discrimination," *IEEE Transactions on Robotics*, vol. 25, no. 5, pp. 1134–1149, 2009.

[22] Z. Najdovski and S. Nahavandi, "Extending haptic device capability for 3d virtual grasping," in *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*. Springer, 2008, pp. 494–503.

[23] C. Yang, X. Wang, Z. Li, Y. Li, and C.-Y. Su, "Teleoperation control based on combination of wave variable and neural networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 8, pp. 2125–2136, 2016.

[24] T. Zhang, Z. McCarthy, O. Jowl, D. Lee, X. Chen, K. Goldberg, and P. Abbeel, "Deep imitation learning for complex manipulation tasks from virtual reality teleoperation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–8.

[25] P. M. Kebria and *et al.*, "Experimental comparison study on joint and cartesian space control schemes for a teleoperation system under time-varying delay," in *2019 IEEE International Conference on Industrial Technology*, 2019, pp. 108–113.

[26] P. M. Kebria, S. Al-Wais, H. Abdi, and S. Nahavandi, "Kinematic and dynamic modelling of ur5 manipulator," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2016, pp. 4229–4234.

[27] M. Kashaninejad, A. Dehghani, and M. Kashiri, "Modeling of wheat soaking using two artificial neural networks (mlp and rbf)," *Journal of food engineering*, vol. 91, no. 4, pp. 602–607, 2009.

[28] M. Kumar and N. Yadav, "Multilayer perceptrons and radial basis function neural network methods for the solution of differential equations: a survey," *Computers & Mathematics with Applications*, vol. 62, no. 10, pp. 3796–3811, 2011.

[29] P. M. Kebria, H. Abdi, and S. Nahavandi, "Development and evaluation of a symbolic modelling tool for serial manipulators with any number of degrees of freedom," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2016, pp. 4223–4228.

[30] D. E. Rumelhart, J. L. McClelland, P. R. Group *et al.*, "Parallel distributed processing: Explanations in the microstructure of cognition. vol. 1," pp. 318–362, 1986.

[31] D. Svozil, V. Kvasnicka, and J. Pospichal, "Introduction to multi-layer feed-forward neural networks," *Chemometrics and intelligent laboratory systems*, vol. 39, no. 1, pp. 43–62, 1997.