International Journal of Control, Automation and Systems 16(5) (2018) 2405-2420
http://dx.doi.org/10.1007/s12555-017-0739-1
ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

Controller Design Based on Wavelet Neural Adaptive Proportional Plus Conventional Integral-derivative for Bilateral Teleoperation Systems with Time-varying Parameters

Soheil Ganjefar, Mohammad Afshar, Mohammad Hadi Sarajchi, and Zhufeng Shao*

Abstract: In this study, a new controller method based on *wavelet neural adaptive proportional plus conventional integral-derivative* (WNAP+ID) controller through adaptive learning rates (ALRs) for the Internet-based bilateral teleoperation system is developed. The PID controller design suffers from dealing with a plant with an intricate dynamic model. To make an adaptive essence for PID controller, this study uses a trained offline self-recurrent wavelet neural network as a processing unit (SRWNN-PU) in parallel with conventional PID controller. The SRWNN-PU parameters are updated online using an SRWNN-identifier (SRWNNI) in order to reduce the controller error in real-time function. Using feedback linearization method and a PID controller, the presented control method reduced the tracking error in the subsystems of the teleoperation system, i.e., master and slave which are stabilized, respectively. Additionally, time-varying delay in teleoperation systems is considered as noise making the master signals be modulated because wavelt neural networks have a high susceptibility to remove the noise, thus the WNAP+ID controller is able to eliminate the noise effect. In this paper, we concentrated on the efficiency and stability of the teleoperation system with time-varying parameters through simulation outcomes. Moreover, the results of the WNNs are compared with those of multi-layer perceptron neural networks (MLPNNs).

Keywords: Adaptive PID control, bilateral teleoperation systems, time-varying delay, wavelet neural network.

1. INTRODUCTION

Nowadays, there is no doubt that teleoperation systems play a key role in humans modern lifestyle. Teleoperation systems not only are utilized in perilous sites or to shift nuclear wastes but also they are broadly employed in different tasks from underwater to space actions [1]. Moreover, controlling something remotely provides the human operator with this chance to work in environments where were not accessible or feasible earlier leading to a noticeable effect on reducing the expenses; therefore, it is obvious that working on the teleoperation systems and improving their performance how much is beneficial and essential for the human beings in order to experience a better and easier lifestyle [2].

In this regard, Zhao*et al*. addressed an extended state observer and sliding mode controller for the nonlinear bilateral teleoperation system to handle lumped system uncertainties [3]. Ollin *et al*. employed predictive control based on the coupling matrix for the master-slave teleoperation system to improve stability and transparency [4]. Lee *et al*. presented a newly developed egocentric method for bilateral teleoperation system to facilitate robot motion despite human cognitive and operational constraints [5]. Lu *et al*. proposed a novel approach with multiple adaptive dominance factors to enhance the transparency of the teleoperation system and reduce the effect of time-varying delay in the communication channel of the teleoperation system [6]. Mellah *et al*. addressed two adaptive neural fuzzy controllers for a nonlinear bilateral teleoperation system; the first one predicts communication delay and the second one controls the slave manipulator and improves system performance [7]. A small-gain framework was developed by Polushin *et al*. [8] in order to improve the stability of a nonlinear bilateral teleoperation system, in which a new neuro-fuzzy controller was addressed. A small-gain structure based on the energy-bounding approach was developed by Uddin *et al*. [9] to

Manuscript received November 26, 2017; revised March 7, 2018; accepted March 26, 2018. Recommended by Associate Editor Huanqing Wang under the direction of Editor Hamid Reza Karimi. This work was supported by the National Natural Science Foundation of China (No. 51575292) and National Science and Technology Major Project of the Ministry of Science and Technology of China (No. 2018ZX04000020).

Soheil Ganjefar and Mohammad Afshar are with the Department of Electrical Engineering, Bu-Ali Sina University, Chahar Bagh Shahid Mostafa Ahmadi Roshan, Hamedan, Iran (e-mails: s_ganjefar@basu.ac.ir, afshar.basu@yahoo.com). Mohammad Hadi Sarajchi and Zhufeng Shao are with the State Key Laboratory of Tribology & Institute of Manufacturing Engineering, and Beijing Key Lab of Precision/Ultra-precision Manufacturing Equipment and Control, Tsinghua University, 30 Shuangqing Rd, Haidian Qu, Beijing Shi, China (e-mails: sarajchi@tsinghua.edu.cn, shaozf@tsinghua.edu.cn).
* Corresponding author.

©ICROS, KIEE and Springer 2018

force-reflecting teleoperation systems. Bekiaris-Liberis and Krstic developed a robust delay predictor to remove the negative effect of time-delay in Internet-based connection. The simulation and experimental results proved the efficiency of the proposed approach in time-constant delay and time-varying delay [10].

Since teleoperation system suffers from uncertainty kinematically and dynamically, Liu *et al*. proposed a new nonlinear adaptive controllers for which no thorough knowledge should be gained in terms of master-slave kinematics as well as operator-master-slave-environment dynamics [11]. In Li and Su [12], an adaptive optimal control for teleoperation systems was examined in case of time-delays. Similarly, Hue *et al*. developed two optimal controllers for a nonlinear bilateral teleoperation system [13]. In the former case , there were differences between some parameters, including the position error, the master and the slave, and velocity error, and the acceleration error. In the latter case, along with unavailable acceleration signal, there were new synchronization variables in a way that master-slave tracking error converged exponentially to zero. Jian-Ning Li and Lin-Sheng Li presented an adaptive neuro-fuzzy inference system (ANFIS) for teleoperation system. The simulation results proved that it could improve the stability and performance of the system and reduce the tracking error of the master-slave manipulators [14]. Zhai and Xia addressed the adaptive sliding mode controller for teleoperation systems. As sliding mode controller is designed according to the Lyapunov theory; as a result, not only it guarantees the stability of the system but also reduces master-slave trajectory tracking error to a minimum value. [15]. Ganjefar *et al*. proposed an adaptive developed PID controller for a nonlinear bilateral teleoperation system. In their study, they used a neural network predictor to reduce the effect of time-varying delay. Simulation results obviously indicate the superiority of the proposed approach in comparison with the other methods [16].

As the neural network has many benefits, such as learning ability and adaptability, it is adept at working out the target threat evaluation contrasted with the other traditional approaches. However, no one can deny the superiority of the wavelet neural network (WNN) in comparison with other neural networks, for instance, the WNN parameters (weights and hidden nodes) are faster defined rather than the Recursive neural network (RNN); WNN needs fewer learning iteration than convolutional neural network (CNN). In addition, although WNN needs fewer nodes, it enjoys a high-speed convergence. The other way round, the teleoperation system strongly suffers from time-varying delay and WNN with adaptive learning rate can compensate this problem appropriately rather other traditional controllers. Also, some scholars attempted to examine applications of WNN by an approximation of nonlinearity in the control systems [17–19]. WNN is completely centralized, unlike the sigmoidal functions employed in multilayer perceptron (MLP) network. Such functions determine more effectiveness for WNN learning capability as compared with neural network based on MLP network. Thus, the WNN-based controller can provide the controller designer with more options and facilities rather than the traditional NN-based controller [17] causing many scholars to put greater emphasis on WNNs for control and identification [20].

This study aimed to introduce a novel controller relying on the control characteristics and noise immunity properties for WNNs. The adaptive designed wavelet PID controller for the slave is not affected by the noise of time-varying delay. Consequently, system stability is preserved and an increase in the convergence of master and slave occurs. The learning rates, for the training of neural network controller, are derived from a Lyapunov function guaranteeing transparency between states of master and the slave manipulators. To demonstrate the capability of the proposed controller, system responses for parameter uncertainty of slave subsystem are simulated and results firmly confirm the superiority of the developed WNN-based controller. This paper is developed in the following sections. Section 2 mainly concentrates on teleoperation systems. In Section 3, the proposed self-recurrent wavelet neural network (SRWNN) is explained. Section 4 presents (WNAP+ID) controller. In Section 5, simulation results validate the performance of the proposed controller in comparison with other controllers. The final section i.e., Section 6 schemes conclusion.

2. TELEOPERATION SYSTEMS

Over the recent years, a large number of scholars have deployed bilateral teleoperation systems in various situations. Today, the master-slave teleoperation systems are used in different areas of concern, such as space and submarine explorations, military and public services [21].

A teleoperation system as an electro-mechanical system contains a master and a slave, interconnected by a communication channel. The human operator is able to generate control signals for the slave through interaction with the master device. These command signals represent position and velocity. Fig. 1 presents the standard l framework of teleoperation system. The framework consists of five parts: human operator, master, control and communication, slave, and environment. A human operator commands via master manipulator by applying a force F_m to drive it with $X_m = [x_m \quad \dot{x}_m]$, that is forwarded to the slave side via the communication block. The slave manipulator is moved by a local control T_s on the slave side. If the slave interacts a far environment and/or some external force, the remote force F_s that shifts slave manipulator with $X_s = [x_s \quad \dot{x}_s]$ that is transmitted back to the master manipulator through the communication block. Control

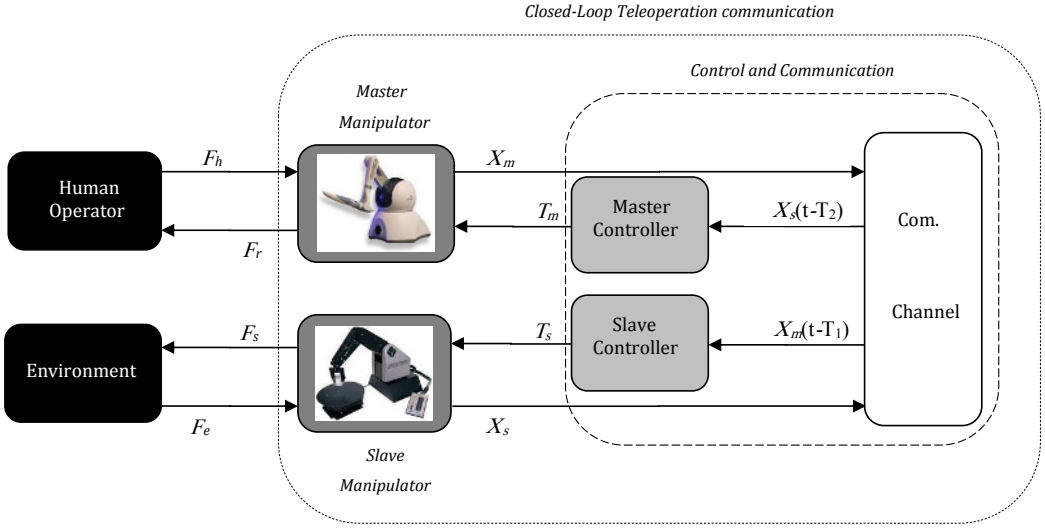

Fig. 1. Structure of teleoperation systems.

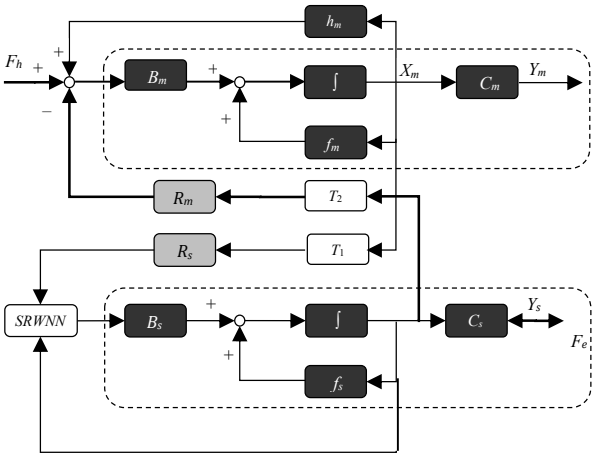

Fig. 2. Position-position structure of 1-DOF nonlinear teleoperation system.

signal T_m or reflected force F_r is received at the master side that operator senses it. The human operator handles the local master manipulator to remotely move the slave one to perform a given act. THe system must be entirely "transparent"; thus, the operator can make a sense as if he was able to control the far environment directly [22, 23].

Fig. 2 shows a position-position framework of 1-DOF nonlinear bilateral teleoperation system:

F_h and F_e : operator and environment force, respectively,

u_m and u_s : master and slave control input, respectively,

X_m and X_s : master and slave states, respectively,

Y_m and Y_s : output for the master and slave, respectively,

h_m : master linearization matrix,

R_m : slave-master interaction and allows calculation of the force reflection to the master,

R_s : master-slave interaction.

The master and slave subsystems are then introduced:

$$\begin{cases} \dot{X}_m(t) = f_m(X_m) + B_m u_m, \ f_m(X_m) = A_m X_m + \Delta_m(X_m), \\ Y_m(t) = C_m X_m(t), \end{cases} \tag{1}$$

$$\begin{cases} \dot{X}_s(t) = f_s(X_s) + B_s u_s, \ \ f_s(X_s) = A_s X_s + \Delta_s(X_s), \\ Y_s(t) = C_s X_s(t). \end{cases} \tag{2}$$

where A_m, B_m, Δ_m, C_m, A_s, B_s, Δ_s, and C_s are general matrices or functions which can be defined as follows: The blocks of the time-delay represent time-delay in the communication channel.

2.1. Teleoperation modeling

The general form of motion equation for a couple of n-DOF nonlinear robot manipulators in the lack of friction or other disturbances is expressed as [24]:

$$M(p)\ddot{p} + C(p,\dot{p}) + G(p) = T, \tag{3}$$

where $M(p)$ and $C(p,\dot{p}) \in R^{n\times n}$ are positive definite inertia matrices and the Coriolis/Centripetal vector, respectively. Moreover, $G(p)$ denotes the gravity vector, and T stands as the torque vector. In this study, the degree of freedom (DOF) for the master and slave manipulators is considered one. The dynamic model of a 1-DOF manipulator in Fig. 3 is given as:

$$J\ddot{\theta}(t) + b\dot{\theta}(t) + mgl\sin\theta(t) = u(t), \tag{4}$$

where J is the inertia; m and l are the mass and length of the manipulator, respectively; g is the gravity acceleration; $\theta(t)$ is the angle of the rotation; $u(t)$ is the applied control

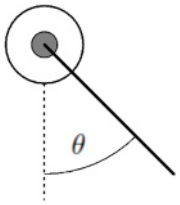

Fig. 3. Diagram of 1-DOF manipulator.

signal; and b is the viscous friction coefficient. The proof is given in [25].

Based on Fig. 2, the following equation is obtained by using $B_m h_m(X_m)$:

$$B_m h_m(X_m) = B_m K_m X_m(t) - \Delta_m(X_m), \tag{5}$$

where the distinct nonlinear term in the master subsystem is merged. ($\Delta_m(X_m)$ is a function of $\sin\theta(t)$, and K_m is a state feedback vector in the master subsystem); therefore, if the position and velocity states are considered ($x_1(t) = \theta(t)$), and ($x_2(t) = \dot{\theta}(t)$), master and slave manipulators can be symbolized in state-space description as:

$$\begin{cases} \begin{bmatrix} \dot{x}_{m1}(t) \\ \dot{x}_{m2}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{b_m}{J_m} \end{bmatrix} \begin{bmatrix} x_{m1}(t) \\ x_{m2}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J_m} \end{bmatrix} u_m(t), \\ y_m(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_{m1}(t) \\ x_{m2}(t) \end{bmatrix}, \end{cases} \tag{6}$$

$$\begin{cases} \begin{bmatrix} \dot{x}_{s1}(t) \\ \dot{x}_{s2}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{b_s}{J_s} \end{bmatrix} \begin{bmatrix} x_{s1}(t) \\ x_{s2}(t) \end{bmatrix} \\ \qquad\qquad + \begin{bmatrix} 0 \\ -\frac{m_s g l_s}{J_s} \sin(x_{s2}) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J_s} \end{bmatrix} u_s(t), \\ y_s(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_{s1}(t) \\ x_{s2}(t) \end{bmatrix}. \end{cases} \tag{7}$$

2.2. Environment

Environmental force produced by the interacting slave with remote environment plays a key role in teleoperation systems. Thus, a mathematical model should be introduced to calculate this reaction force. The Kelvin's simplified model for the environment is employed in this study [26]. This reaction force, which performs opposite the slave according to the PD control low, is given as:

$$f_s(t) = k_e \theta_s(t) + b_e \dot{\theta}_s(t), \tag{8}$$

where k_e represents stiffness, and b_e shows viscous friction. The R_m vector in the slave-master force feedback is described as:

$$R_m = \begin{bmatrix} r_{m1} & r_{m2} \end{bmatrix} = \begin{bmatrix} k_f k_e & k_f b_e \end{bmatrix}, \tag{9}$$

where k_f is the gain of the force feedback.

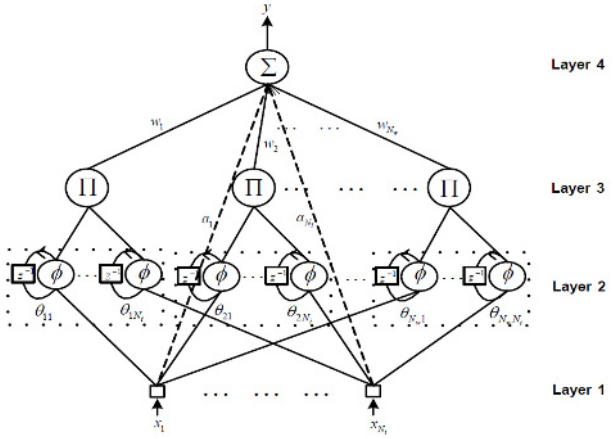

Fig. 4. The SRWNN structure.

3. SRWNN

SRWNN farmework demonstrated in Fig. 4 has four layers: N_i inputs, $N_i \times N_w$ mother wavelets, and one output as well as a product layer [27]. There are a self-feedback loop and a mother wavelet for each node in the mother wavelet layer (layer 2). The mother wavelet function is selected as the first derivative of a Gaussian function $\phi(x) = x\exp(-0.5x^2)$. For each node, ϕ_{ij} is determined as the following:

$$\phi_{ij}(z_{ij}) = \phi\left(\frac{u_{ij} - m_{ij}}{d_{ij}}\right), \quad z_{ij} = \frac{u_{ij} - m_{ij}}{d_{ij}}, \tag{10}$$

where d_{jk} and m_{jk} are the dilation and translation wavelet factors, respectively. In this paper, the jth input term of the ith wavelet is represented by ij index; however, in the nth sample time, the inputs of this layer are computed as:

$$u_{ij}(n) = x_j(n) + \phi_{ij}(n-1)\,\theta_{ij}, \tag{11}$$

where θ_{ij} denotes the weight of the self-feedback loop weight. The pase information of neural networks can be reserved by the memory option ($\phi_{ij}(n-1)$). Consequently, SRWNNs are used more than WNNs in complex systems. Here, θ_{ij} shows the information storage rate highlighting the main distinction between SRWNNs and WNN. The third layer nodes are given by:

$$\psi_i(x) = \prod_{j=1}^{N_i} \phi(z_{ij}) = \prod_{j=1}^{N_i} \left[-z_{ij} \exp\left(-\frac{1}{2}(z_{ij})^2\right)\right]. \tag{12}$$

The node in the fourth layer or the SRWNN output is calculated for each parameters and self-recurrent wavelet in the following:

$$y(n) = \sum_{i=1}^{N_w} w_i \psi_i(x). \tag{13}$$

3.1. Training of the SRWNN

In an SRWNN, the following quadratic cost function is minimized:

$$J(n) = \frac{1}{2}[y_d(n) - \hat{y}(n)]^2 = \frac{1}{2}e^2(n), \tag{14}$$

where $\hat{y}(n)$ is the actual output and $y_d(n)$ is the desired ouput of SRWNN for the nth sample time. The GD method adjust the SRWNN weights; therefore, after a given number of training iterations, the error is minimized. The following equation determines the GD method:

$$W^i(n+1) = W^i(n) + \eta^i\left(-\frac{\partial J(n)}{\partial W^i(n)}\right), \tag{15}$$

where $W = [t_{ij}\ \ d_{ij}\ \ \theta_{ij}\ \ w_{ij}]^T$ and $\eta = [\eta^t\ \ \eta^d\ \ \eta^\theta\ \ \eta^w]^T$; η^i and W^i adress the learning rate matrix and an arbitary weighting vector, respectively in the SRWNN. In the other word, W^i can be t_{ij} or d_{ij} or θ_{ij} or w_{ij} and η^i can be η^t or η^d or η^θ or η^w. The partial derivative of the cost function with respect to W^i is:

$$\frac{\partial J(n)}{\partial W^i i(n)} = -e(n)\frac{\partial \hat{y}(n)}{\partial W^i(n)}. \tag{16}$$

The weighting vector is updated by applying the chain rule recursively as the following:

$$\frac{\partial \hat{y}(n)}{\partial t_{ij}(n)} = -w_i\psi_i\left(\frac{-1}{d_{ij}}\right)\left(\frac{1}{z_{ij}-z_{ij}}\right), \tag{17}$$

$$\frac{\partial \hat{y}(n)}{\partial d_{ij}(n)} = z_{ij}\frac{\partial \hat{y}(n)}{\partial t_{ij}(n)}, \tag{18}$$

$$\frac{\partial \hat{y}(n)}{\partial \theta_{ij}(n)} = -\phi_{ij}(n-1)\frac{\partial \hat{y}(n)}{\partial t_{ij}(n)}, \tag{19}$$

$$\frac{\partial \hat{y}(n)}{\partial w_i(n)} = \psi_i(x). \tag{20}$$

4. WNAP+ID CONTROLLER DESIGNING

4.1. WNAP+ID architecture

The overall architecture of (WNAP+ID) cotroller is shown in Fig. 5(a). There are two SRWNN structures, i.e., identifier and controller. Fig. 5(b) and (c) represent the structures of the "Identifier" and "PU", respectively.

The system dynamics and sensitivity are approximated by the identifier. The error of the reference model output and the actual system output, $e(n) = y_d(n) - y(n)$, is minimized by the control signal. Therefore, the control signal can be represented as:

$$u(n) = K_p.PE(n) + K_i\frac{e(n)}{S} + (K_d e(n))S, \tag{21}$$

where $PE(n)$ stands the processed error (PE) signal in $n$the sample and the propotional, integral and derivative

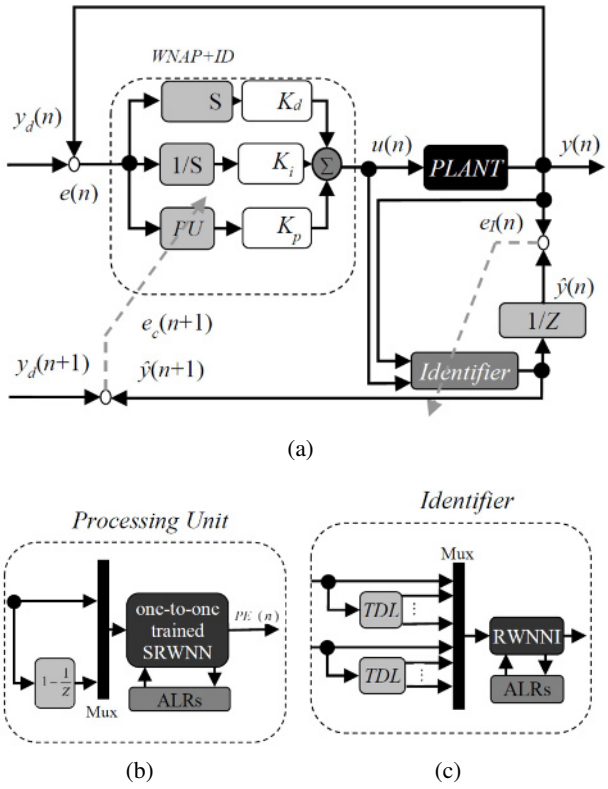

Fig. 5. Structure of the proposed controller.

gains as well as K_d, respectively. The inputs and output of SRWNN-PU are demonstrated by:

$$x_c = [e(n),\ (e(n) - e(n-1))], \tag{22}$$

$$PE(n) = f_c(x_c), \tag{23}$$

where the input ector of the SRWNN-PU is represented by x_c. Minimizing the PE signal results in optimizing the contol signal, $u(n)$, in turn, leading to deriving a cost function:

$$J_c(n) = \frac{1}{2}[e_c^2(n) + \beta u^2(n)], \tag{24}$$

where $e_c(n) = y_d(n) - y(n)$, and $y_d(n)$ are the desired output and β is a weighting factor used to modify the performance index for system responses due to creating a favorable compromise between the used force and the control performance. Using (15), the derivative of the cost function, $J_c(n)$, with respect to weighting parameters W_C^i of the SRWNN-PU is:

$$\frac{\partial J_c(n)}{\partial W_C^i(n)} = K_p\left[-e_c(n)\frac{\partial y(n)}{\partial u(n)} + \beta u(n)\right]\frac{\partial PE(n)}{\partial W_C^i(n)}, \tag{25}$$

then

$$W_C^i(n+1) = W_C^i(n) + eta_C^i\frac{\partial J_C(n)}{\partial W_C^i(n)}, \tag{26}$$

where W_C^i and η_C^i represent the weight and learning rates in the SRWNN-PU, respectively.

In (25), $\partial PE(n)/\partial W_C^i(n)$ is computed by (17)-(20), and $\partial y(n)/\partial u(n)$ which denotes the system sensitivity, but it is not possible to derive it from the output. As a result, the system sensitivity must be derived by the identifier output, moreover, a real-time approximation SRWNNI is used to calculating the sensitivity. The SRWNNI output can be addressed as:

$$x_I = [(u(n), u(n-1), ..., u(n-p)), \\ (\omega(n), \omega(n-1), ..., \omega(n-p))], \tag{27}$$

$$\hat{y}(n+1) = f_I(x_I), \tag{28}$$

where x_I is the input vector of the identifier, $= omega$ is the control signal error, and $\hat{y}(n+1)$ is the predicted output of the system. Using (17)-(20), let us minimize the identification cost function of online or offline training procedure:

$$J_I(n) = \frac{1}{2}[y(n) - \hat{y}(n)]^2 = \frac{1}{2}e_I^2(n), \tag{29}$$

then

$$W_I^i(n+1) = W_I^i(n) + \eta_I^i \left(e_I(n) \frac{\partial \hat{y}(n)}{\partial W_I^i(n)} \right), \tag{30}$$

where W_I^i and η_I^i are the weighting and learning rates in the SRWNNI, respectively. The SRWNNI has a significant effect on improving the achievement of the SRWNN-PU, so it must be trained using the offline procedure at the beginning and finally, it has to be trained in online mode operation and be sited in the system configuration. Now the system sensitivity can be calculated using the plant model by the chain rule as follows:

$$\frac{\partial y(n+1)}{\partial u(n)} \approx \frac{\partial \hat{y}(n+1)}{\partial x_I} \frac{\partial x_I}{\partial u(n)}. \tag{31}$$

From (27), can derive $\partial x_I/\partial u(n)$ as:

$$\frac{\partial x_I}{\partial u(n)} = [(1 \;\; f_1(z), ..., f_p(z)) \;\; (0 \;\; 0, ..., 0)]^T, \tag{32}$$

where $f_i(z) = z^{-i}$ and also:

$$\frac{\partial \hat{y}(n+1)}{\partial x_{I,j}} = \sum_{i=1}^{N_{I,w}} w_{I,i} \psi_{I,I} \left(\frac{1}{d_{ij}} \right) \left(\frac{1}{z_{ij} - z_{ij}} \right), \tag{33}$$

where $N_{I,w}$ represents the node number in the product layer of SRWNNI.

4.2. Convergence analysis via ALRs for identifier

The optimal learning rate is designed using the convergence of the SRWNN. As the performance of the neuro-identifier is determined by the learning rate and the neuro-controller is trained through the GD method, we should find the optimal learning rate. The discrete-type Lyapunov function is represented as:

$$V_I(n) = J_I(n) = \frac{1}{2}{e_I}^2(n). \tag{34}$$

The Lyapunov function difference is derived by:

$$\Delta V_I(n) = V_I(n+1) - V_I(n) = \frac{1}{2}\left[e_I^2(n+1) - e_I^2(n)\right]. \tag{35}$$

The change in the error is illustrated by:

$$\Delta e_I(n) = e_I(n+1) - e_I(n) \approx \left[\frac{\partial e_I(n)}{\partial W_I^i(n)} \right]^T \Delta W_I^i(n). \tag{36}$$

From (29) and (30):

$$\Delta W_I^i(n) = \eta_I^i e_I(n) \frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)}. \tag{37}$$

The error difference can be rewritten as:

$$\Delta e_I(n) = e_I(n+1) - e_I(n) \\ \approx \left[\frac{\partial e_I(n)}{\partial W_I^i(n)} \right]^T \eta_I e_I(n) \frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)}. \tag{38}$$

Then, a convergence theorem is represented.

Theorem 1: Let η_I^i be defined as the learning rate for the SRWNNI weights as well as $g_{I,\max}^i$ be determined as $g_{I,\max}^i = \max_n \|g_I^i(n)\|$ where $g_I^i(n) = \partial \hat{y}(n)/\partial W_I^i$ and $\|\cdot\|$ represents the Euclidean norm in R^n, then the convergence occurs if adopted η_I^i an satisfy:

$$0 < \eta_I^i < \frac{2}{\left(g_{I,\max}^i\right)}. \tag{39}$$

From (34)-(38), $\Delta V_I(n)$ is defined as:

$$\Delta V_I(n) = -\lambda_I e_I^2(n), \tag{40}$$

where

$$\lambda_I = 0.5\eta_I^i\} g_I^i(n)\}^2 \left(2 - \eta_I^i \|g_I^i(n)\|^2\right). \tag{41}$$

Proof: See Appendix A. □

If $\lambda_I > 0$ is satisfied; hence, $\Delta V_I < 0$ which guarantees the asymptotic convergence of the SRWNNI.

Let $\eta_a = \eta_I^i (g_{I,\max}^i)^2$ then:

$$\lambda_I \geq 0.5\eta_I^i\} g_I^i(n)\}^2 (2 - \eta_a) > 0. \tag{42}$$

From (41), we obtain $0 < \eta_a < 2$ following (39) fulfills the proof.

Remark 1: The maximum learning rate guaranteeing convergence is:

$$\eta_I^i = \left(\frac{1}{g_{I,\max}^i} \right), \tag{43}$$

where this presents the half of the upper boundary condition in Theorem 1 [28]. The quantity of $g_{I,\max}^{i}$ for each learning rate η_I^w, η_I^t, η_I^d, η_I^θ in SRWNN structure can be assessed. Then, the following Theorem comes up.

Theorem 2: Let be the learning rates for the SRWNNI, respectively. The learning rates of the SRWNN identifier for the maximum convergence are as follows:

$$\eta_I^{w,\max} = \frac{1}{N_{I,w}}, \tag{44}$$

$$\eta_I^{t,Max} = \frac{1}{N_{I,w}N_{I,i}}\left[\frac{|d_{I,\min}|}{2\exp(-0.5)|w_{I,\max}|}\right]^2, \tag{45}$$

$$\eta_I^{d,Max} = \frac{1}{N_{I,w}N_{I,i}}\left[\frac{|d_{I,\min}|}{2\exp(0.5)|w_{I,\max}|}\right]^2, \tag{46}$$

$$\eta_I^{\theta,Max} = \frac{1}{N_{I,w}N_{I,i}}\left[\frac{|d_{I,\min}|}{2\exp(-0.5)|w_{I,\max}|}\right]^2, \tag{47}$$

where $N_{I,i}$ and $N_{I,w}$ are the number of inputs and nodes in the product layer of SRWNNI, respectively.

Proof: See Appendix B. □

4.3. Convergence analysis via ALRs for controller

In this section, the convergence analysis of the SRWNN-PU is determined. A discrete-type Lyapunov function is represented as:

$$V_c(n) = \frac{1}{2}\left[e_{c,1}^2(n) + e_{c,2}^2(n)\right], \tag{48}$$

where $e_{c,1}(n) = (y_d(n)y(n))$ and $e_{c,2}(n) = \beta^{0.5}u(n)$ are the errors in the closed-loop control system. During the training process, the deference of the Lyapunov function is derived by:

$$\begin{aligned}\Delta V_c(n) =& V_c(n+1) - V_c(n)\\ =& \frac{1}{2}\sigma_{j=1}^2 e_{c,j}^2(n+1) - e_{c,j}^2(n),\end{aligned} \tag{49}$$

and difference of the errors is represented by:

$$\begin{aligned}e_{c,j}(n+1) &= e_{c,j}(n) + \Delta e_{c,j}(n)\\ &\approx e_{c,j}(n) + \left[\frac{\partial e_{c,j}(n)}{\partial W_c^i}\right]^{\mathrm{T}} \Delta W_c^i,\end{aligned} \tag{50}$$

where

$$\frac{\partial e_{c,1}(n)}{\partial W_c^i} = -K_P S\frac{\partial PE(n)}{\partial W_c^i}, \tag{51}$$

$$\frac{\partial e_{c,2}(n)}{\partial W_c^i} = -K_P\sqrt{\beta}\frac{\partial PE(n)}{\partial W_c^i}. \tag{52}$$

From (25) and (26):

$$\begin{aligned}\Delta W_c^i =& \eta_c^i\left(-\frac{\partial J_c(n)}{\partial W_c^i}\right)\\ =& \eta_c^i\left[K_P S e_{c,1}(n) - K_P\sqrt{\beta}e_{c,2}(n)\right]\frac{\partial PE(n)}{\partial W_c^i},\end{aligned} \tag{53}$$

where $S \approx \partial\hat{y}(n+1)/\partial u$ and hence the convergence theorem as follows:

Theorem 3: Let $g_{c,\max}^i$ as the learning rate for the SRWNN-PU weights and is determined as $g_{c,\max}^i = \max_n \|g_c^i(n)\|$, where $g_c^i\ \partial PE(n)/\partial W)c^i$ and $\|\cdot\|$ represents the Euclidean norm in R^n, then the convergence occurs if adoted η_I^i can satisfy:

$$0 < \eta_c^i < \frac{2}{K_P^2(S^2+\beta)(g_{c,\max}^i)^2}. \tag{54}$$

Proof: From (48) to (54), $\Delta V_c(n)$ can be represented as:

$$\begin{aligned}\Delta V_c(n) &= \sum_{j=1}^{2}\Delta e_{c,j}(n)\left(e_{c,j}(n) + \frac{1}{2}\Delta e_{c,j}(n)\right)\\ &= -\left(K_P S e_{c,1}(n) - K_P\sqrt{\beta}e_{c,2}(n)\right)^2\lambda_c,\end{aligned} \tag{55}$$

where

$$\lambda_c = \eta_c^i\|g_c^i(n)\|^2\left(1 - \frac{1}{2}\eta_c^i\|g_c^i(n)\|^2K_P^2\left(S^2+\beta\right)\right). \tag{56}$$

If $\lambda_I > 0$ is satisfied; hence, $\Delta V_I < 0$ which guarantees the asymptotic convergence of the SRWNNI.

Let $\eta_a = \eta_c^i(g_{c,\max}^i)^2$, then:

$$\begin{aligned}\lambda_c =& 0.5\eta_c^i\|g_c^i(n)\|^2\left(2 - \eta_a\frac{\|g_c^i(n)\|^2}{(g_{c,\max}^i)^2}K + P^2(S^2+\beta)\right)\\ \le& 0.5\eta_c^i\|g_c^i(n)\|^2(2 - \eta_a K_P^2(S^2+\beta)) > 0.\end{aligned} \tag{57}$$

$0 < \eta_a < 2/K_P^2(S^2+\beta)$ is derived from (57) and (54) and this totalizes the proof. □

Corollary 1: From Remark 1, the optimal learning rates of SRWNN-PU for the asymptotic convergence are as follows:

$$\eta_c^i = \frac{1}{K_P^2(S^2+\beta)(g_{c,\max}^i)^2}. \tag{58}$$

This shows the half of the upper boundary condition in Theorem 3. The quantity of $g_{c,\max}^i$ for each learning rate in SRWNN structure can be assessed. Consequently, the following Theorem is derived:

Theorem 4: Let η_c^w, η_c^t, η_c^d, η_c^θ be the learning rates of the SRWNN-PU, the optimal learning rates guaranteeing the convergence of the SRWNN-PUare as follows:

$$\eta_c^{w,\max} = \frac{1}{N_{c,w}K_P^2(S^2+\beta)}, \tag{59}$$

$$\eta_c^{t,Max} = \frac{1}{N_{c,w}N_{c,i}K_P^2(S^2+\beta)}\left[\frac{|d_{c,\min}|}{2\exp(-0.5)|w_{c,\max}|}\right]^2, \tag{60}$$

$$\eta_c^{d,Max} = \frac{1}{N_{c,w}N_{c,i}K_P^2(S^2+\beta)}\left[\frac{|d_{c,\min}|}{2\exp(0.5)|w_{c,\max}|}\right]^2, \tag{61}$$

$$\eta_c^{\theta,Max} = \frac{1}{N_{c,w}N_{c,i}K_P^2(S^2+\beta)}\left[\frac{|d_{c,\min}|}{2\exp(-0.5)|w_{c,\max}|}\right]^2, \tag{62}$$

where $N_{c,i}$ and $N_{c,w}$ are the number of inputs and nodes in the product layer of SRWNN-PU. Prepared by the identifier, the system sensitivity, $S \approx \partial\hat{y}(n_1)/\partial u$ must be replaced by Smax represented in the following:

$$S_{\max} = \sqrt{N_{I,w}}|w_I|_{\max}\frac{2\exp(-0.5)}{|d_I|_{\min}}. \tag{63}$$

Proof: See Appendix C. □

5. SIMULATION

The master-slave parameters are represented as:

$$J_m = 1.5\,\text{kg.m}^2, \qquad J_s = 2\ \text{kg.m}^2,$$
$$b_m = 11\,\frac{\text{Nm}}{\text{rad/s}}, \qquad b_s = 15\,\frac{\text{Nm}}{\text{rad/s}},$$
$$l_m = 0.4\,\text{m}, \qquad l_s = 2\ \text{m},$$
$$m_m = 0.3\,\text{kg}, \qquad m_m = 1.5\ \text{kg}.$$

The state space representation of the master and slave is as follow:

$$A_m = \begin{bmatrix} 0 & 1 \\ 0 & \frac{-11}{1.5} \end{bmatrix},\ B_m = \begin{bmatrix} 0 \\ \frac{1}{1.5} \end{bmatrix},\ C_m = \begin{bmatrix} 1 & 0 \end{bmatrix}$$
$$A_s = \begin{bmatrix} 0 & 1 \\ 0 & \frac{-15}{2} \end{bmatrix},\ B_s = \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix},\ C_s = \begin{bmatrix} 1 & 0 \end{bmatrix},$$
$$k_m = [k_{m1} \quad k_{m2}] = [-150 \quad -67],$$
$$R_m = [k_f k_e \quad k_f b_e] = [10 \quad 0.1],$$
$$R_s = [r_{s1} \quad r_{s2}] = [12.5 \quad 0.125],$$

where R_m represents the interaction between the slave and remote environment depending on stiffness(k_e)and viscous friction (b_e) parameters, respectively. Therefore,

$$b_e = 1\,\frac{\text{Nm}}{\text{rad/s}},\ k_e = 100\,\frac{\text{Nm}}{\text{rad}},\ k_f = 0.1.$$

In this paper, the teleoperation system is simulated by using (WNAP+ID) and MLPNN controllers. A thirdorder identifier is used in this study, in this regard; p and q values in (27) are considered as 2. The SRWNNI has 6, 48, 8 and 1 neurons in the input, mother wavelet, product, and output layers, respectively. Also, the SRWNN-PU has 2, 12, 6 and 1 neurons at the corresponding layers, in the other word, the SRWNN-PU enjoys a simple framework with only two inputs.

Fig. 6 falls into four parts related to simulation results of constant time-delay, 500 msec. Part (a) contains stepvarying input with different amplitudes. When we apply this input to system and output response puts in a strong performance and does good tracking, controller design proves acceptable. Part (b) consists of master and slave position for varying-step input relating to part (a) in constant and adaptive learning rates. In order to show the effect of adaptive learning rate in training of SRWNN controller, the responses of teleoperation system are shown for both constant and adaptive learning rates, in which the weights using fewer iterations of the online simulation are updated. Controller based on ALR not only has more acceptable overshoot relative to that of CLR but also converges to the desired value of master for each step of input. Some of these adaptive learning rates are shown in Fig. 7. SRWNNs result in fast convergence using fewer neurons as compared with MLPNNs; therefore in order to prove this purpose, we used a MLPNN to control the teleoperation system. In this paper, the SRWNN controller used for controlling has just 12 mother wavelets ($N_i \times N_w = 12$), but MLPNN controller has one hidden layer and there are 50 neurons in the hidden layer. Responses of teleoperation system for varying-step input relating to part (a) in 500 msec time-delay are shown in part (c). The other controller designed by Azorin *et al.* [29] is represented in this Figure. Also, we call it previous controller (Pre. Co.). The previous controller was selected for comparison because it has the same bilateral teleoperation structure as mentioned in Section 2. When responses of WNAP+ID and Pre. Co. are compared together, it is clear that slave of SRWNN converges more rapidly to the desired value of master. In part (d), control signals of WNAP+ID, MLPNN and Pre. Co. for varying-step input relating to part (a) for 500 msec time-delay are shown. It is clearly evident that each of the three signals has nearly the same amplitude but as can be seen in part (c), WNAP+ID slave converges rapidly and this is of considerable significance for the proposed controller. On the other hands, all of them reach zero for each pulse of input and therefore slave reaches master position because the controller is designed for convergence of master and slave.

Adaptive learning rates for SRWNN controller (part (b) of Fig. 6) are explained in Fig. 7. As seen in part (a), this rate is primary constant for many input pulses, then after a time, it will switch to another value. Learning rate will stay at this value with a new input pulse; it will switch to another value. In part (b), a signal presents two types of learning rate with the same performance. This signal is varied exactly at the time when the signal of part (a) changes into another value. As seen in this part, the signal of part (b) converges to the constant value for several input pulses too, which its amplitude is much less than that of part (a). This reflects the minimal impact of this signal relative to the signal of part (a). Part (c) indicates the third

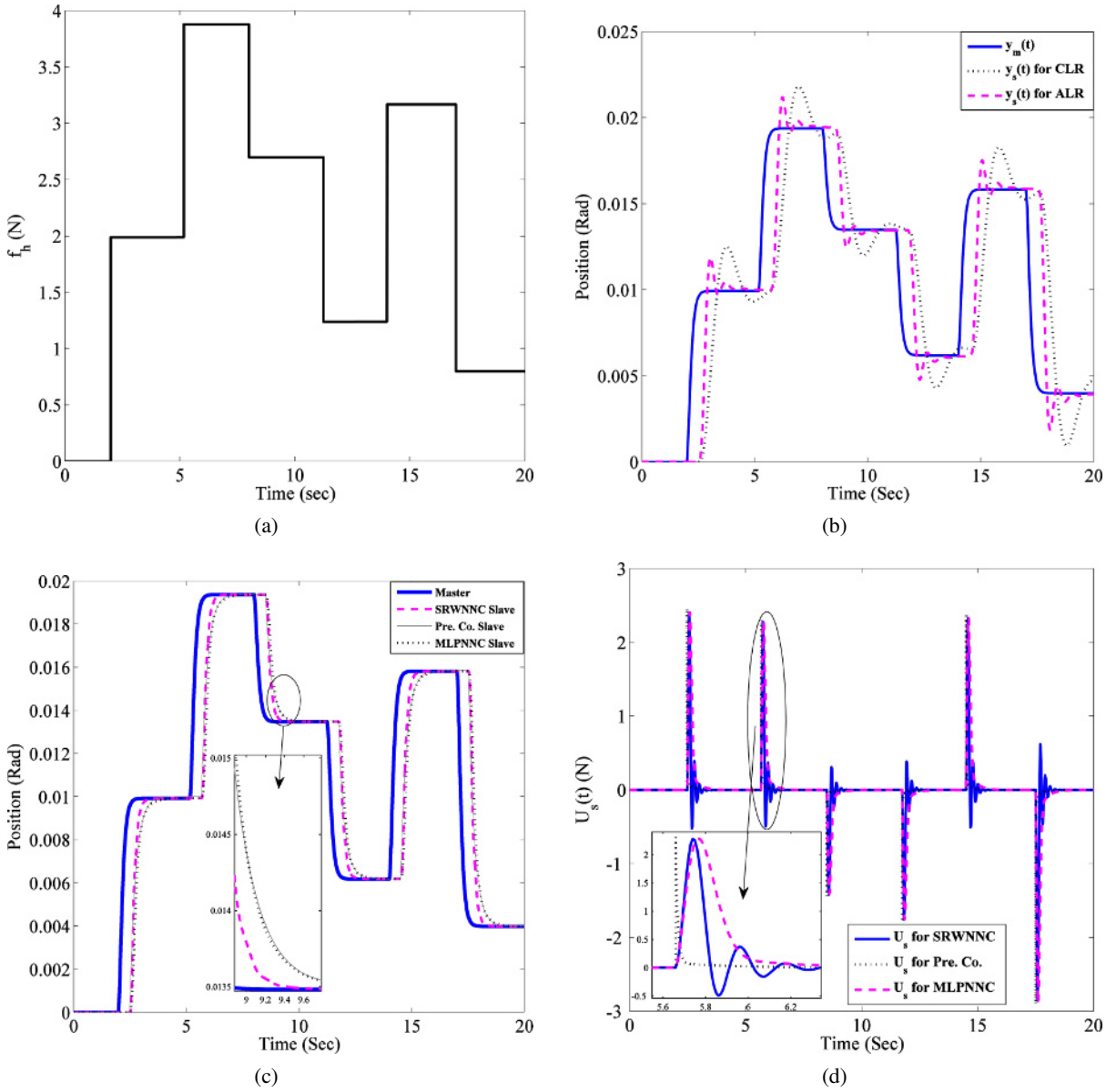

Fig. 6. Results for 500 msec time-delay. (a) Step-varying input signal. (b) Response for constant learning rate (CLR) and adaptive learning rate (ALR) for less iteration to updating of in online simulation. (c) Comparison among responses of three different controller methods for 4000 times updating of weights in online simulation. (d) Comparison among control signals of three different controller methods.

type of learning rate with a structure like part (b). Difference between signals of parts (b) and (c) lies in the signal amplitude. As shown in this figure, the amplitude of the signal in part (c) is around 10 percent of it in part (b). In other words, this signal demonstrates minimal impact as compared with part (b).

There are four parts in Fig. 8, indicating simulation results for time-varying delay of part (a). The Internet is utilized in teleoperation systems as a communication network. As its time-delay types are variable, we examine controller by using time-varying delay as shown in part (a). In part (b), noise filtering in the proposed controller is considered. As mentioned above, time-varying delay causes noise in teleoperation systems, which makes master signals modulated and applied to the slave subsystem as the input of WNAP+ID controller. In this part, two signals are shown: an inappropriate signal with inappropriate fluctuation and the solid line showing the input of WNAP+ID controller and a signal indicating slave subsystem response. The input signal is disturbed but output signal is exactly accepted, with no undesirable effect of time-varying delay. This shows that the proposed controller is robust in cases of noise and disturbance from the environment. Part (c) demonstrates master and slave posi-

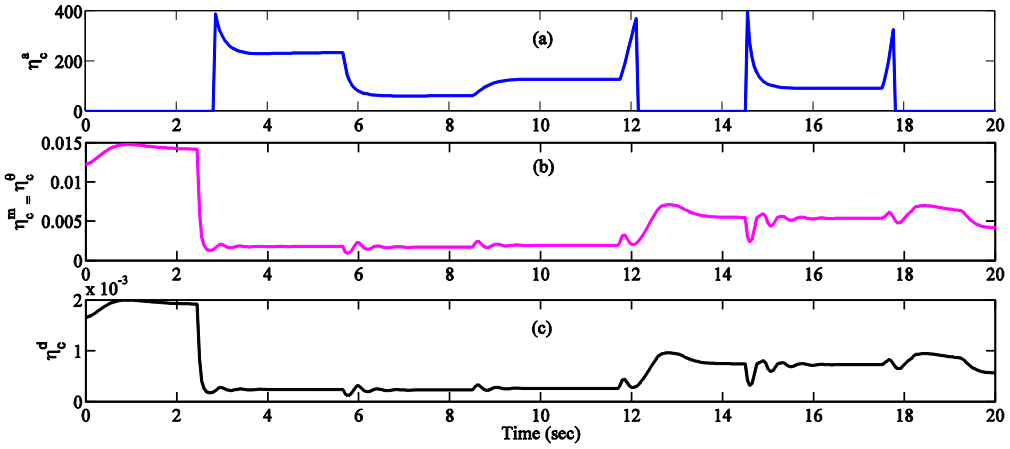

Fig. 7. Adaptive learning rates for 500 msec time-delay.

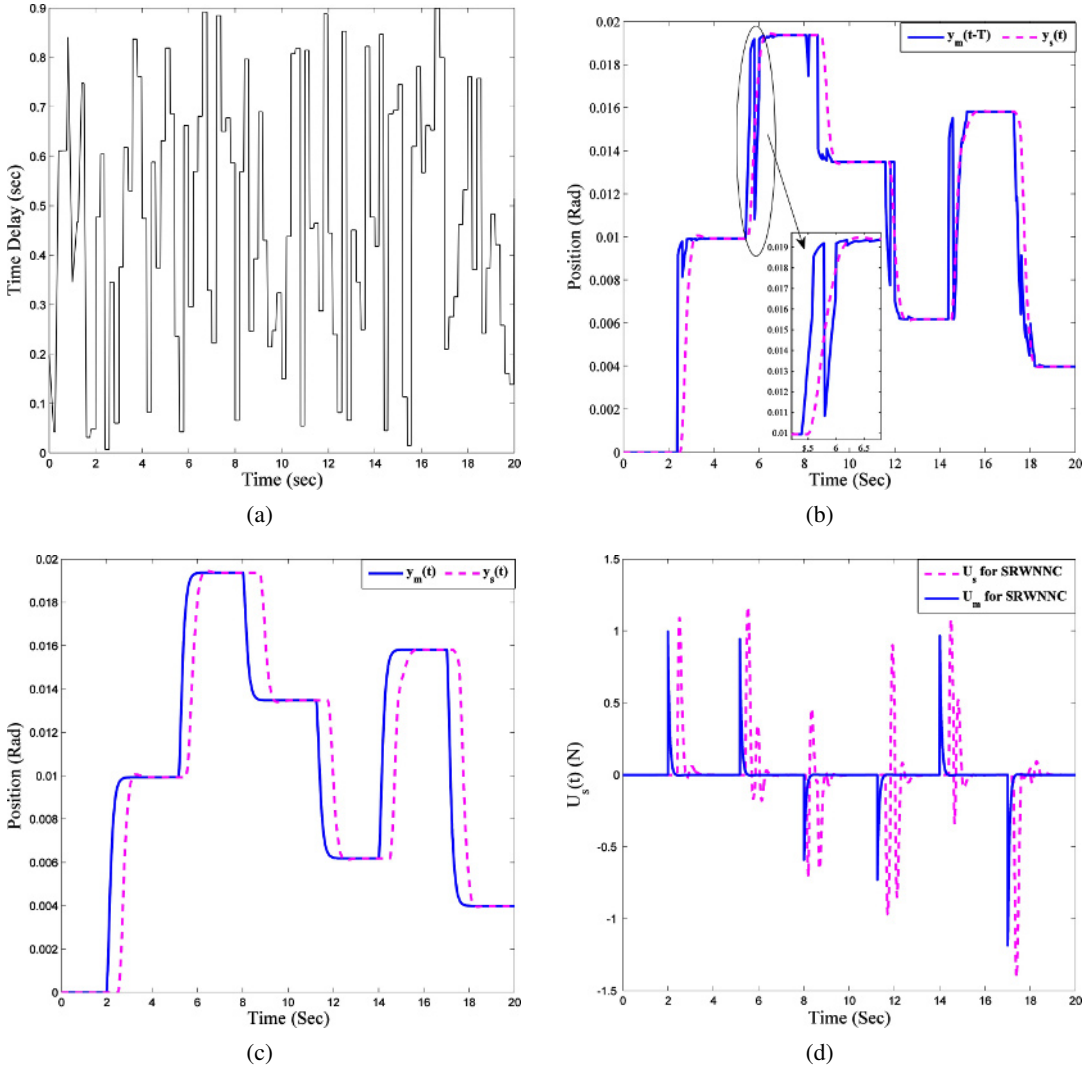

Fig. 8. Adaptive results for time-varying delay. (a) Time-varying delay signal. (b) Input and output of the slave subsystem. (c) Positions of master and slave. (d) Control signals of master and slave.

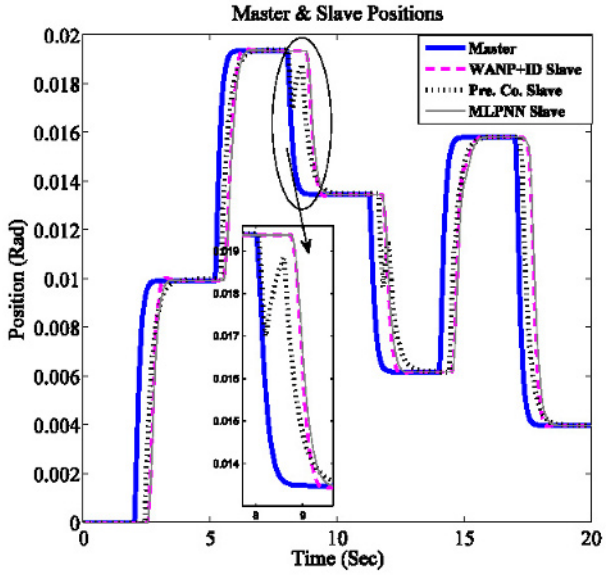

Fig. 9. Master and slave positions of SRWNN and MLPNN and previous controllers for time-varying delay.

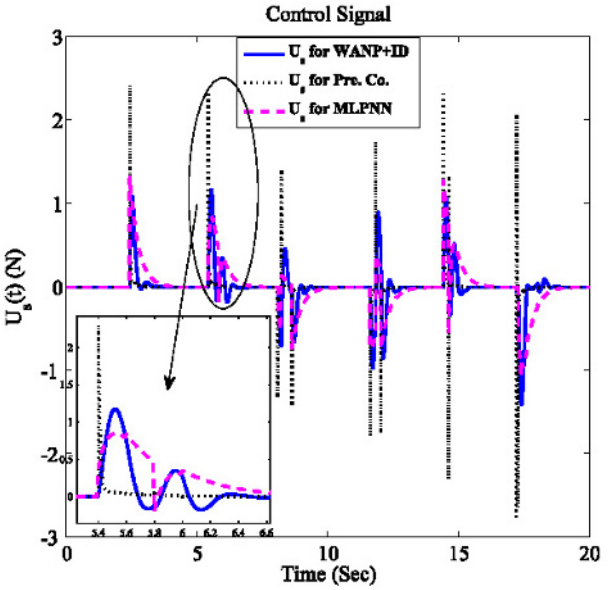

Fig. 10. Slave control signals of SRWNN and MLPNN and previous controllers for time-varying delay.

tion obtained by applying step-varying input for timevarying delay. It is clear that slave converges rapidly to master for different steps of input. Hence, SRWNN controller was designed well; on the other hand, no timevarying delay exerts an impact on the system response. This again is an indication of robustness in case of timevarying delay. Part (d) contains the master and slave control signal for time-varying delay, both of which have limited amplitude without any undesirable fluctuation. In a sense, they are accepted and can be implemented.

Fig. 9 shows comparison between master and slave positions for different types of controller. In this part, there are four signals: master, slave of WNAP+ID, slave of MLPNN and slave of Pre. Co. for step-varying input and time-varying delay. This figure shows absolute superiority of proposed controller in rapid convergence, robust on adverse effect of time-varying delay and high performance. It is worth noting that neurons of SRWNN are much less in number as compared with MLPNN. Inappropriate signal of Pre. Co. is very harmful because it leads to unwanted movements of robot arm at slave. In Fig. 10, slave control signals of SRWNN, MLPNN, and Pre. Co. controllers are explained. Despite slave control signal of Pre. Co. has a higher amplitude as compared with other control signals; all of these signals can be applied. In the zoomed part, the control signal of WNAP+ID obtains flexibility. All in all, WNAP+ID demonstrates more effectiveness than the other method, because of its more appropriate position as well as its more desirable control signal.

Fig. 11 falls into four parts related to simulation results of time-varying parameters demonstrated in parts (a) and (b) of Fig. 8. Part (a) contains time-varying input with different amplitudes as well as fixed time-step equal to 2.5 sec. It is of great importance to have been considered that time-step for all of parts in this table are fixed and equal to 2.5 sec. As shown in part (b), time-varying mass is strongly considered as the most important parameter of the slave subsystem dynamic in all of complex bilateral teleoperation systems. With the slave manipulator moving any object, its dynamic changes due to the object mass added to that of manipulator. In fact, net mass of slave manipulator is 1.5 kg. in this study and after moving any object, manipulator mass switches to different value. Responses of teleoperation system for time-varying input and delay as well as mass are described in part (c) and compared with each other. This part clearly shows the proposed controller based on WNAP+ID providing more appropriate output in comparison with Pre. Co. approach. Since the proposed controller is adaptive and tunes its parameters online; hence, WNAP+ID-controlled system will be robust on time-varying parameters. On the other hand, a clear-cut superiority of the presented approach is described in part (d). Slave control signal of WNAP+ID converge at zero much faster than that of Pre. Co. method. In the other word, taking advantage of less effort, the proposed controller rapidly and in a strong manner converge at desirable value defined by master in many a step of input.

In order to compare the simulation results in different conditions, we employed the following equation:

$$ESS = \int e^2 dt,$$

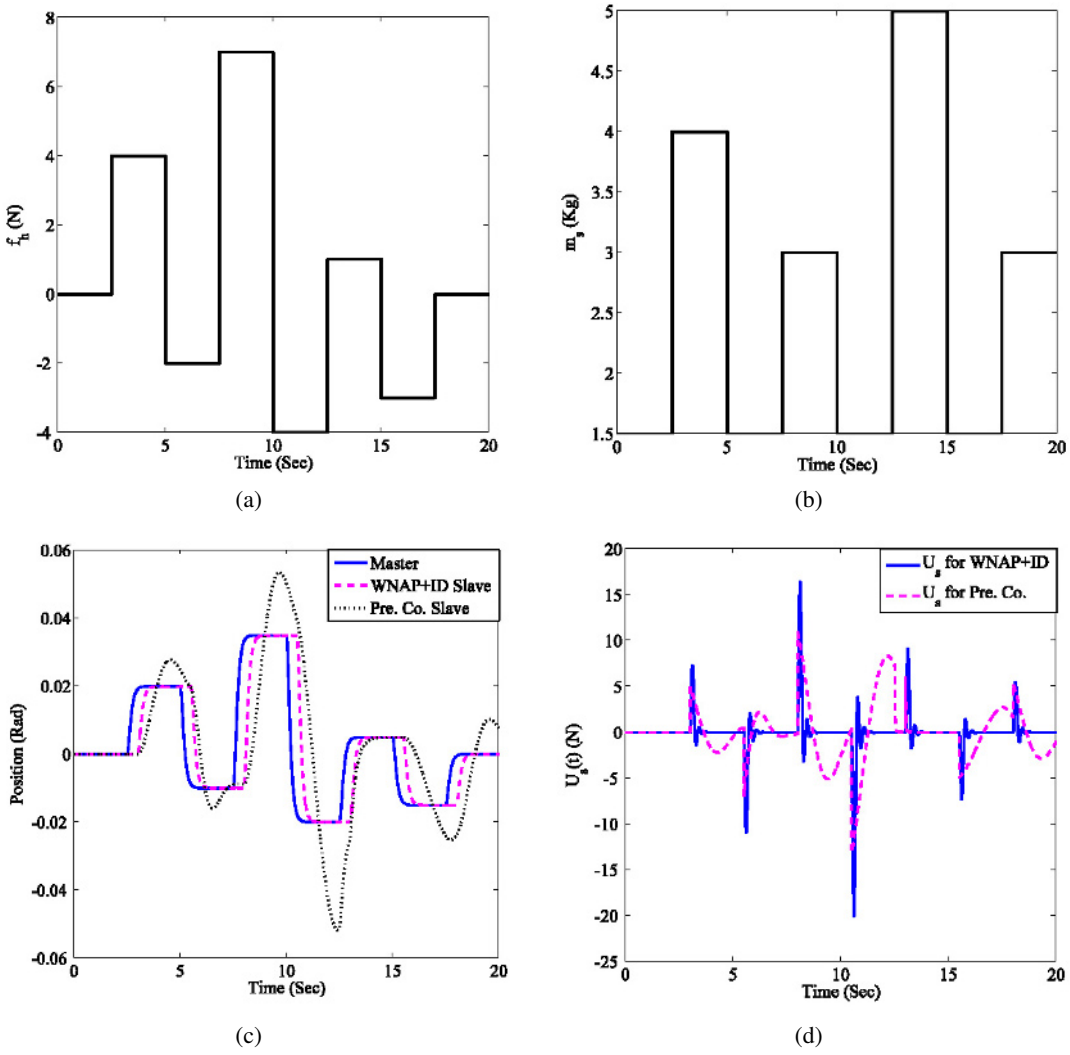

Fig. 11. Results for time-varying mass of slave. (a) Time-varying input signal. (b) Time-varying mass of the slave subsystem. (c) Positions of master and slave. (d) Slave control signals.

where

$$e(t) = x_{m1}(t) - x_{s1}(t).$$

$x_{m1}(t)$ and $x_{s1}(t)$ show the master and slave position, respectively. Moreover, there is desirable transparency for a correct rendition of the environment to the operator. Transparency takes the second most significant position in the design of teleoperation systems. Difference between the master and slave positions is considered as the transparency and is fulfilled as follows:

$$\int e^2 dt < \beta.$$

β is a small positive value. Simulation results for the different types of time-delay are presented in Table 1. Comparison of results in Table 2 reveals that the WNAP+ID controller using ALRs has higher performance than the MLPNN and WNAP+ID based on CRL.

Table 1. ESS for different types of time-delay. (All the values are according to the Rad.)

Delay	WNAP+ID	MLPNN	Pre. Co.
500 msec	0.0183	0.0212	0.0201
Time-Varying	0.0105	0.0208	0.0121

Table 2. Control performance comparison of the WNAP+ID and the MLPNN.

Number of	WNAP+ID (Adaptive)	WNAP+ID (Static)	MLPNN
Product nodes (N_w)	6	6	-
Inputs (N_i)	2	2	2
Training parameters	42	42	201
Sampling rate	0.05	0.01	0.01
Learning rate	Adaptive	0.01	0.1
Iteration	4,00	2,000	2,000

6. CONCLUSION

This study aimed at proposing a new controller based on Wavelet Neural Adaptive Proportional plus conventional Integral-Derivative (WNAP+ID). In this approach, the self-recurrent wavelet neural networks via adaptive learning rates for the nonlinear bilateral teleoperation systems were utilized. Because the WNAP+ID have recurrent and dynamic structures, they are appropriate for control and identification of systems. Adaptive learning rates for training WNAP+ID controller and identifier, derived from a proper Lyapunov function, cause faster convergence and provide the acceptable tracking between the master and slave. Despite the time-varying delay of the communication channel in the bilateral teleoperation systems, the system performance is difficult to stabilize and improve. Hence, a large number of studies have examined controller design based on the adaptive controller. This type of controller ensures the stability, transparency, and high performance of the system. Additionally, it takes both the communication latency and the slave-environment interaction into account. Because the time-varying delay in the bilateral teleoperation systems makes an inappropriate signal modulated with the master signal, a WNN controller is utilized to eliminate this noise, time-varying delay, and to improve the distortion of the delayed signal, using SRWNN via adaptive learning rates. In addition to stabilization of the closed-loop system, SRWNN controller eliminates this distortion. The findings demonstrated that not only do responses to the proposed controller rapidly converge into desired values, but fewer neurons are also utilized in comparison with MLPNNs to control and identify systems. The simulation results validate the proposed control approach and its excellent performance in motion tracking. This new controller puts the system in a great performance on constant and time-varying delay. It is of great importance to say that mass of slave is also considered as a variable parameter like the Internet delay because the slave can move different materials with various mass affecting slave manipulator mass.

APPENDIX A

From (34)-(38), $V_I(n) > 0$ is defined as:

$$\begin{aligned}
\Delta V_I(n) =& \frac{1}{2}\left(e_I^2(n+1) - e_I^2(n)\right) \\
=& \Delta e_I(n)\left[e_I(n) + \frac{1}{2}\Delta e_I(n)\right] \\
=& \left[\frac{\partial e_I(n)}{\partial W_I^i(n)}\right]^T \eta_I^i e_I(n) \frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)} \\
& \times \left(e_I(n) + \frac{1}{2}\left[\frac{\partial e_I(n)}{\partial W_I^i(n)}\right]^T \eta_I^i e_I(n) \frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)}\right) \\
=& -\left[\frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)}\right]^T \eta_I^i e_I(n) \frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)} \\
& \times \left(e_I(n) - \frac{1}{2}\left[\frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)}\right]^T \eta_I^i e_I(n) \frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)}\right) \\
=& -e_I^2(n) \sum_{i=1}^{4} \left[\eta_I^i \left(\frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)}\right)^2 \right. \\
& \left. \times \left\{1 - \frac{1}{2}\eta_I^i \left(\frac{\partial \hat{y}_I(n)}{\partial W_I^i(n)}\right)^2\right\}\right] \\
=& -e_I^2(n)\left(0.5\eta_I^i \|g_I^i\|^2 \left(2 - \eta_I^i \|g_I^i\|^2\right)\right) \\
=& -e_I^2(n)\lambda.
\end{aligned}$$

APPENDIX B

First, Proof of (44):

$$\begin{aligned}
&\frac{\partial \hat{y}(n)}{\partial w_i(n)} = \psi(x), \\
&g_I^i(n) = \frac{\partial \hat{y}(n)}{\partial w_i(n)} \\
&\quad = \left[\frac{\partial \hat{y}(n)}{\partial w_1(n)} \;\; \frac{\partial haty(n)}{\partial w_2(n)} \;\; \cdots \;\; \frac{\partial \hat{y}(n)}{\partial w_{N_i}(n)}\right] \\
&\quad = [\psi_{I,1}(x) \;\; \psi_{I,2}(x) \;\; \cdots \;\; \psi_{I,N_i}(x)] \\
&\quad = \Psi.
\end{aligned}$$

where Ψ is the input vector of the SRWNN based identifier and $w_i = [w_1, w_2, ..., w_{N_i}]^T$. Then we have $\|g_I^1(n)\| \leq (N_i)^{0.5}|\psi_{I,\max}|$. Therefore, from Theorem 1, we find that $0 < \eta_I^w < 2/(g_{I,\max}^1(n))^2$. Thus, from the Remark 1, we have:

$$\eta_I^{w,\max} = \frac{1}{N_{I,w}}. \tag{B.1}$$

Next, Proof of (45):

Lemma 1: Let $f(t) = t\exp(-t^2)$. Then $|f(t)| < 1$, $f \in R$.

Lemma 2: Let $g(t) = t^2\exp(-t^2)$. Tehn $g(t)| < 1$, $g \in R$.

The learning rate η_I^t of the translation weight t_I:

$$\begin{aligned}
g_I^2(n) =& \frac{\partial \hat{y}}{\partial t_I} \\
=& \sum_{j=1}^{N_w} w_{I,j}\left(\frac{\partial \Phi_j}{\partial t_I}\right) \\
=& \sum_{j=1}^{N_w} w_{I,j}\left\{\sum_{k=1}^{N_i} \frac{\prod_{k=1}^{N_i}\varphi(z_{jk})}{\varphi(z_{jk})}\left(\frac{\partial \varphi(z_{jk})}{\partial z_{jk}}\frac{\partial z_{jk}}{\partial t_I}\right)\right\} \\
<& \sum_{j=1}^{N_w} w_{I,j}\left\{\sum_{k=1}^{N_i} \max\left(\frac{\partial \varphi(z_{jk})}{\partial z_{jk}}\frac{\partial z_{jk}}{\partial t_I}\right)\right\} \\
<& \sum_{j=1}^{N_w} w_{I,j}\left\{\sum_{k=1}^{N_i} \max\left(2e^{-0.5}\left(-\frac{1}{d_I}\right)\right)\right\}.
\end{aligned}$$

According to Lemma 2,

$$\left|\left(\frac{1}{2}z_{jk}^2-\frac{1}{2}\right)\exp\left\{-\left(\frac{1}{2}z_{jk}^2-\frac{1}{2}\right)\right\}\right|<1.$$

Therefore, we have:

$$\begin{aligned}\|g_I^2(n)\| &< \sum_{j=1}^{N_w} w_{I,j}\sqrt{N_i}\left(\frac{-2\exp(-0.5)}{d_{I,\min}}\right)\\ &< \sqrt{N_w}\sqrt{N_i}|w_{I,\max}|\left|\frac{2\exp(-0.5)}{d_{I,\min}}\right|.\end{aligned}$$

Thus,

$$\eta_I^{t,\max}=\frac{1}{N_{I,w}N_{I,j}}\left[\frac{|d_{I,\min}|}{2\exp(-0.5)|w_{I,\max}|}\right]^2.$$

Next, Proof of (46):

$$\begin{aligned}g_I^3(n)&=\frac{\partial\hat{y}}{\partial d_I}\\ &=\sum_{j=1}^{N_w} w_{I,j}\left(\frac{\partial\Phi_j}{\partial d_I}\right)\\ &=\sum_{j=1}^{N_w} w_{I,j}\left\{\sum_{k=1}^{N_i}\frac{\prod_{k=1}^{N_i}\varphi(z_{jk})}{\varphi(z_{jk})}\left(\frac{\partial\varphi(z_{jk})}{\partial z_{jk}}\frac{\partial z_{jk}}{\partial d_I}\right)\right\}\\ &<\sum_{j=1}^{N_w} w_{I,j}\left\{\sum_{k=1}^{N_i}\max\left(\frac{\partial\varphi(z_{jk})}{\partial z_{jk}}\frac{\partial z_{jk}}{\partial d_I}\right)\right\}\\ &<\sum_{j=1}^{N_w} w_{I,j}\left\{\sum_{k=1}^{N_i}\max\left(2e^{-0.5}\left(-\frac{1}{d_I}\right)\right)\right\}.\end{aligned}$$

According to Lemma 2,

$$\left|\left(\frac{1}{2}z_{jk}^2-\frac{1}{2}\right)\exp\left\{-\left(\frac{1}{2}z_{jk}^2-\frac{1}{2}\right)\right\}\right|<1.$$

Therefore, we have:

$$\begin{aligned}\|g_I^3(n)\| &< \sum_{j=1}^{N_w} w_{I,j}\sqrt{N_i}\left(\frac{2\exp(0.5)}{d_{I,\min}}\right)\\ &< \sqrt{N_w}\sqrt{N_i}|w_{I,\max}|\left|\frac{2\exp(0.5)}{d_{I,\min}}\right|.\end{aligned}$$

Thus, according to Theorem 1:

$$0<\eta_I^d<\frac{2}{(g_{I,\max}^3)^2}=\frac{2}{N_wN_i}\left[\frac{1}{|w_{I,\max}|\left(\frac{2\exp(0.5)}{|d_{i,\min}|}\right)}\right]^2.$$

Based on Remark 1:

$$\eta_I^{d,\max}=\frac{1}{N_{I,w}N_{I,j}}\left[\frac{|d_{I,\min}|}{2\exp(-0.5)|w_{I,\max}|}\right]^2.$$

Next, Proof of (47):

$$\begin{aligned}g_I^4(n)&=\frac{\partial\hat{y}}{\partial\theta_I}\\ &=\sum_{j=1}^{N_w} w_{I,j}\left(\frac{\partial\Phi_j}{\partial\theta_I}\right)\\ &=\sum_{j=1}^{N_w} w_{I,j}\left\{\sum_{k=1}^{N_i}\frac{\prod_{k=1}^{N_i}\varphi(z_{jk})}{\varphi(z_{jk})}\left(\frac{\partial\varphi(z_{jk})}{\partial z_{jk}}\frac{\partial z_{jk}}{\partial\theta_I}\right)\right\}\\ &<\sum_{j=1}^{N_w} w_{I,j}\left\{\sum_{k=1}^{N_i}\max\left(\frac{\partial\varphi(z_{jk})}{\partial z_{jk}}\frac{\partial z_{jk}}{\partial\theta_I}\right)\right\}\\ &<\sum_{j=1}^{N_w} w_{I,j}\left\{\sum_{k=1}^{N_i}\max\left(2e^{-0.5}\left(-\frac{1}{d_I}\right)\right)\right\}.\end{aligned}$$

According to Lemma 2,

$$\left|\left(\frac{1}{2}z_{jk}^2-\frac{1}{2}\right)\exp\left\{-\left(\frac{1}{2}z_{jk}^2-\frac{1}{2}\right)\right\}\right|<1.$$

Therefore, we have:

$$\begin{aligned}\|g_I^4(n)\| &< \sum_{j=1}^{N_w} w_{I,j}\sqrt{N_i}\left(\frac{2\exp(-0.5)}{d_{I,\min}}\right)\\ &< \sqrt{N_w}\sqrt{N_i}|w_{I,\max}|\left|\frac{2\exp(-0.5)}{d_{I,\min}}\right|.\end{aligned}$$

Thus, according to Theorem 1:

$$0<\eta_I^d<\frac{2}{(g_{I,\max}^4)^2}=\frac{2}{N_wN_i}\left[\frac{1}{|w_{I,\max}|\left(\frac{2\exp(-0.5)}{|d_{i,\min}|}\right)}\right]^2.$$

Based on Remark 1:

$$\eta_I^{\theta,\max}=\frac{1}{N_{I,w}N_{I,j}}\left[\frac{|d_{I,\min}|}{2\exp(-0.5)|w_{I,\max}|}\right]^2.$$

APPENDIX C

$$\begin{aligned}S_{\max}=\frac{\partial y(n+1)}{\partial u}&\approx\frac{\partial\hat{y}(n+1)}{\partial u}=\sum_{i=1}^{N_{I,w}} w_{I,i}\left(\frac{\partial\psi_i}{\partial u}\right)\\ &=\sum_{i=1}^{N_{I,w}} w_{I,i}\sum_{j=1}^{N_{I,i}}\frac{\prod_{j=1}^{N_{I,j}}\varphi(z_{ij})}{\varphi(z_{ij})}\left(\frac{\partial\varphi(z_{ij})}{\partial z_{ij}}\frac{\partial z_{ij}}{\partial x_{I,j}}\frac{\partial x_{I,j}}{\partial u}\right).\end{aligned}$$

In order to simplify computing; $\partial x_I/\partial u=[\{1,0,0,...,0\}\{0,0,0,,0\}]$, therefore:

$$<\sum_{i=1}^{N_{I,w}} w_{I,i}\left\{\max\left(\frac{\partial\varphi(z_{ij})}{\partial z_{ij}}\frac{\partial z_{ij}}{\partial x_{I,j}}\right)\right\}. \tag{C.1}$$

In Section 3, the author assumed the $\varphi(x)=x\exp(-0.5x^2)$; thereby:

$$\frac{d\varphi(x)}{dx}=(1-x^2)e^{-\frac{1}{2}x^2}.$$

The maximum value of the above equation is 1. Thus:

$$\frac{d\varphi(x)}{dx} < 2e^{-\frac{1}{2}} \approx 1.21. \tag{C.2}$$

Moreover, according to (10) and (11):

$$\frac{dz_{ij}}{dx_{I,j}} = \frac{1}{d_I}. \tag{C.3}$$

Based on (C.1), (C.2), and (C.3), we have:

$$< \sum_{i=1}^{N_{I,w}} w_{I,i} \left\{ \max \left(\frac{\partial \varphi(z_{ij})}{\partial z_{ij}} \frac{\partial z_{ij}}{\partial x_{I,j}} \right) \right\}$$
$$< \sum_{i=1}^{N_{I,w}} w_{I,i} \left\{ \max \left(\frac{2\exp(-0.5)}{d_I} \right) \right\}.$$

And according to the CauchySchwarz inequality, we have:

$$< \sqrt{N_{I,w}} |w_I|_{\max} \left(\frac{2\exp(-0.5)}{|d_I|_{\min}} \right).$$

Proof pf Appendix C.
Then:

$$< \sum_{i=1}^{N_{I,w}} w_{I,i} \left\{ \max \left(\frac{\partial \varphi(z_{ij})}{\partial z_{ij}} \frac{\partial z_{ij}}{\partial x_{I,j}} \right) \right\}$$
$$< \sum_{i=1}^{N_{I,w}} w_{I,i} \left\{ \max \left(\frac{2\exp(-0.5)}{d_I} \right) \right\}$$
$$< \sqrt{N_{I,w}} |w_I|_{\max} \left(\frac{2\exp(-0.5)}{|d_I|_{\min}} \right).$$

Thus:

$$S_{\max} = \sqrt{N_{I,w}} |w_I|_{\max} \left(\frac{2\exp(-0.5)}{|d_I|_{\max}} \right).$$

APPENDIX D

According to the CauchySchwarz inequality:

$$\left(\sum_{i=1}^{N} ab \right)^2 \leq \left(\sum_{i=1}^{N} a^2 \right) \left(\sum_{i1}^{N} b^2 \right),$$

we consider $a = 1$, $b = X$:

$$\left(\sum_{i=1}^{N} X \right)^2 = \left(\sum_{i=1}^{N} 1 \times X \right)^2$$
$$\leq \left(\sum_{i=1}^{N} 1^2 \right) \left(\sum_{i=1}^{N} X^2 \right) = N \sum_{i=1}^{N} X^2,$$
$$\left(\sum_{i=1}^{N} X \right)^2 \leq N \sum_{i=1}^{N} X^2,$$
$$\left(\sum_{i=1}^{N} X \right) < \sqrt{N} \sum_{i=1}^{N} |X|_{\max}.$$

REFERENCES

[1] R. J. Anderson and M. W. Spong, "Bilateral Control of Operators with Time Delay," *IEEE Transactions on Automation Control*, vol. 34, no. 5, pp. 494-501, 1989.

[2] T. B. Sheridan, "Telerobotics," *Automatica*, vol. 25, no. 4, pp. 487-507, 1989.

[3] L. Zhao, H. Zhang, Y. Yang, and H. Yang, "Integral sliding mode control of a bilateral teleoperation system based on extended state observers," *International Journal of Control, Automation and Systems*, vol. 15, no. 5, pp. 2118-2125, 2017.

[4] O. Peñaloza-Mejía, L. A. Márquez-Martínez, J. Alvarez-Gallegos, and J. Alvarez, "Master-slave teleoperation of underactuated mechanical systems with communication delays," *International Journal of Control, Automation and Systems*, vol. 15, no. 2, pp. 827-836, 2017.

[5] H. G. Lee, H. J. Hyung, and D. W. Lee, "Egocentric teleoperation approach," *International Journal of Control, Automation and Systems*, vol. 15, no. X, pp. 2744-2753, 2017.

[6] Z. Lu, P. Huang, P. Dai, Z. Liu, and Z. Meng, "Enhanced transparency dual-user shared control teleoperation architecture with multiple adaptive dominance factors," *International Journal of Control, Automation and Systems*, vol. 15, no. 5, pp. 2301-2312, 2017.

[7] R. Mellah, S. Guermah, and R. Toumi, "Adaptive control of bilateral teleoperation system with compensatory neural-fuzzy controllers," *International Journal of Control, Automation and Systems*, vol. 15, no. 4, pp. 1949-1959, 2017.

[8] I. G. Polushin, S. Dashkovskiy, A. Takhmar, and R. V. Patel, "A small gain framework for networked cooperative force-reflecting teleoperation," *Automatica*, vol. 49, no. 2, p. 338348, February 2013.

[9] R. Uddin, S. Park, S. Park, and J. Ryu, "Projected predictive Energy-Bounding Approach for multiple degree-of-freedom haptic teleoperation," *International Journal of Control, Automation and Systems*, vol. 14, no. 6, pp. 1561-1571, 2016.

[10] N. Bekiaris-Liberis and M. Krstic, "Robustness of nonlinear predictor feedback laws to time- and state-dependent delay perturbations," *Automatica*, vol. 49, no. 6, pp. 1576-1590, 2013.

[11] X. Liu, R. Tao, and M. Tavakoli, "Adaptive control of uncertain nonlinear teleoperation systems," *Mechatronics*, vol. 24, no. 1, pp. 66-78, 2014.

[12] Z. Li and C. Y. Su, "Neural-adaptive control of single-master-multiple-slaves teleoperation for coordinated multiple mobile manipulators with time-varying communication delays and input uncertainties," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 9, pp. 1576-1590, 2013.

[13] 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, no. 3, pp. 204-212, 2013.

[14] J. N. Li and L. S. Li, "Reliable control for bilateral teleoperation systems with actuator faults using fuzzy disturbance observer," *IET Control Theory & Applications*, vol. 11, no. 3, pp. 446-455, 2017.

[15] D. H. Zhai and Y. Xia, "Adaptive control for teleoperation system with varying time delays and input saturation constraints," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 6921-6929, 2016.

[16] S. Ganjefar, S. Rezaei, and F. Hashemzadeh, "Position and force tracking in nonlinear teleoperation systems with sandwich linearity in actuators and time-varying delay," *Mechanical Systems and Signal Processing*, vol. 86, Part A, pp. 308-324, 2017.

[17] C. K. Lin, "Adaptive tracking controller design for robotic systems using Gaussian wavelet networks," *IEE proc. Control Theory App*, vol. 149, no. 4, pp. 316-322, 2002.

[18] C. D. Sousa, E. M. Hemerly, and R. K. H. Galvao, "Adaptive control for mobile robot using wavelet networks," *IEEE Trans. System Man Cyber*, vol. 32, no. 4, pp. 493-504, 2002.

[19] C. F. Hsu, C. M. Lin, and T. T. Lee, "Wavelet adaptive backstepping control for a class of nonlinear systems," *IEEE Trans. Neural Network*, vol. 17, no. 5, pp. 1175-1183, 2006.

[20] R. M. Sanner and J. J. E. Slotine, "Structureally dynamic wavelet networks for adaptive control of robotic systems," *Int. J. Control*, vol. 70, no. 3, pp. 405-421, 1998.

[21] T. B. Sheridan, *Telerobotics, Automation, and Human Supervisory Control*, the MIT Press, Cambridge, Massachusetts, 1992.

[22] K. Taylor and J. Trevelyan, "Australia's telerobot on the World Wide Web," *Proc. of 26th International Symposium on Industrial Robots*, Singapore, pp. 39-44, 1995.

[23] K. Goldberg, S. Gentner, C. Sutter, and J. Wiegley, "The mercury project: a feasibility study for Internet robots," *IEEE Robotics & Automation Magazine*, vol. 7, no. 1, pp. 35-40, 2000.

[24] D. Lee and M. W. Spong, "Passive bilateral teleoperation with constant time-delay," *IEEE Transactions On Robotics*, vol. 22, no. 2, pp. 269-281, 2006.

[25] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, Pearson Prentice Hall, London, 1989.

[26] F. Janabi-Sharifi, "Modelling, simulation and identification of robotic manipulators interacting with environments," *Journal of Intelligent and Robotic systems*, vol. 13, no. 1, pp. 1-44, 1995.

[27] S. J. Yoo, J. B. Park, and Y. H. Choi, "Indirect adaptive control of nonlinear dynamic systems using self-recurrent wavelet neural networks via adaptive learning rates," *Information Sciences*, vol. 177, no. 15, p. 30743098, August 2007.

[28] S. J. Yoo, J. B. Parkn and Y. H. Choi, "Stable predictive control of chaotic systems using self recurrent wavelet neural network," *International Journal of Control, Automation, and Systems*, vol. 3, no. 1, p. 4355, 2005.

[29] J. M. Azorin, O. Reinoso, J. M. Sabater, and C. Perez, "A new control method of teleoperation with time delay," *Proc. of The International Conference on Advanced Robotics Coimbra*, Portugal, pp. 100-105, 2003.

Soheil Ganjefar received his B.Sc. degree from the Ferdoowsi University, Mashhad, Iran, in 1994, and the M.Sc. and Ph.D. degrees from the Tarbiat Modares University, Tehran, Iran, in 1997 and 2003, respectively, all in electrical engineering. He is currently a Professor in the Department of Electrical Engineering, Bu-Ali Sina University, Hamedan, Iran. His main research interests include Teleoperation systems control, neural network, and Renewable Energy.

Mohammad Afshar received his M.Sc. degree in control engineering from the Bu-Ali Sina University of Hamedan, Iran, in 2013. His current research interests include neural network, intelligent systems control, robotic systems, and teleoperation systems.

Mohammad Hadi Sarajchi received his B.Sc. degree in electrical engineering from Razi University in 2010, and the M.S. degree in electrical engineering from Bu-Ali Sina University in Iran in 2013. In 2017, he joined the Department of Mechanical Engineering, Tsinghua University, in China as a Post Master Researcher. His current research interests include teleoperation system, artificial intelligence, cable-driven parallel robot (CDPR), and drone.

Zhufeng Shao is an associate professor in the Department of Mechanical Engineering, Tsinghua University. He received his Ph.D. degree in Mechanical Engineering from Tsinghua University in 2011. He joined Tsinghua University in the same year where he is teaching mechanical design and control of parallel manipulator. His research interests include cable-driven robot, motion control and optimal design.