An Adaptive Controller with An Orthogonal Neural Network and A Third Order Sliding Mode Observer for Robot Manipulators

Quang Dan Le and Hee-Jun Kang*
Graduate school of Electrical Engineering University of Ulsan, 680-749, Ulsan, South Korea
Email: ledantm@gmail.com, hjkang@ulsan.ac.kr

Tien Dung Le
The University of Danang – University of Science & Technology, Danang, Vietnam
Email: ltdung@dut.udn.vn

***Abstract*—In this paper, An adaptive controller with an orthogonal neural network (ONN) and a third order sliding mode(TOSM) observer for robot manipulators is proposed. Firstly, the TOSM observer is designed to observe joint velocities. Then, the ONN is designed to compensate robot dynamic uncertainties on line inside a computed torque control structure. Therefore, the proposed controller allows only position measurements due to the TOSM observer and achieve highly accurate trajectory tracking performance due to the ONN's uncertainty compensation. Finally, computer simulation for a 2-DOF manipulator is performed to show verify the effectiveness of the proposed controller.**

***Index Terms*—orthogonal neural network, third order sliding mode observers, on line dynamic compensation**

I. INTRODUCTION

Now a day, robots have been widely applied in industry and daily life. The robot manipulator control is always a challenge in control fields because of the high nonlinearity and the dynamic uncertainties. Various control schemes have been developed over the past few decades, including PID controller [1], Sliding mode controller [2], adaptive controller [3], Fuzzy [4] and neural network controller[5]. But, the most of these are based on the supposition that position and velocity are available measurements. The robot usually uses only position encoder to measure joint position. The numerical differentiation of the joint position is often used to obtain the joint velocities so that the unwanted high velocities is caused over fast and unsmooth trajectories. The tachometers could be used for measuring the joint velocity. But, the use of the tachometeris limited due to noises in signal response and more expensiveness. In order to solve this problem, the velocity observer design has been studied with various techniques such as linear observers [6], traditional sliding mode observer[7-8], neural network observer and fuzzy observer. In this paper, a TOSM observer is used to reduce the estimation error with finite time convergence in our control scheme.

Adaptive methods based on intelligent techniques using neural network for robot manipulator have been proposed by many researchers [9-10]. Most researches are based on feed forward neural network which has some drawbacks such local minimum, slow convergence and the difficulty of choosing learning rate and initial weight values. The problem of selecting the initial weights and slow convergence has been improved, seen in [11]. To overcome most of above problems, Yen and Chen [12] proposed an orthogonal neural network and orthogonal active function. The ONNs are also used in anti-lock braking system [13] and robot control [14]. In this paper, the ONN will be used to estimate the dynamic uncertainties and added to the proposed control structure.

In this paper, an adaptive controller with an ONN and a TOSM observer for robot manipulators is proposed. The rest of this paper is arranged as follow. Problem formulation is given based on the robot dynamic models and conventional computed torque controller in section 2. Section 3 presents third order sliding mode observers to estimate joint velocities. Section 4 shows the structure of an orthogonal neural network. In section 5, the proposed control structure is presented. Computer simulations for a 2-DOF manipulator is shown in section 6 and conclusions are given in section 7.

II. STATEMENT OF THE PROBLEM

The dynamic models of a robot manipulator is given as

$$M(q)\ddot{q}+V(q,\dot{q})+G(\mathrm{q})+F_f+\tau_c=\tau \qquad (1)$$

where $\ddot{q},\dot{q},q\in\Re^n$ are the vector of joint accelerations, velocity and position, respectively. $M(q)\in\Re^{n\times n}$ is inertia matrix, $V(q,\dot{q})\in\Re^n$ represents the centripetal and Coriolis matrix, $\mathrm{G}(q)\in\Re^n$ represent the gravitation

Manuscript received September 21, 2017; revised December 18, 2017.

doi: 10.18178/ijmerr.7.2.169-173

torques, F_f is friction term, τ_c is uncertainties in robot dynamic and τ is torque provided at joint.

The equation (1) can be rewritten as

$$\ddot{q} = M^{-1}(q)\{\tau - H(q,\dot{q})\} + M^{-1}(q)\{-F_f - \tau_c\} \quad (2)$$

where $H(q,\dot{q}) = V(q,\dot{q}) + G(q)$. By define $x_1 = q \in \Re^n$ and $x_2 = \dot{q} \in \Re^n$. Eq. (2) can be described in state space form as

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(x_1, x_2, u) + \phi(x_1, x_2, t) \end{cases} \quad (3)$$

where $u = \tau$, $f(x_1, x_2, u) = M^{-1}(q)\{\tau - H(q,\dot{q})\}$ and $\phi(x_1, x_2, t) = M^{-1}(q)\{-F_f - \tau_c\}$. Here, ϕ represents the dynamic uncertainties.

The computed torque control law is given as

$$\tau = M(q)[\ddot{q}_d + K_v(\dot{q}_d - q) + K_e(q_d - q)] + H(q,\dot{q}) \quad (4)$$

where q_d, $\dot{q}_d$ and $\ddot{q}_d$ are the desired position, velocity and acceleration, respectively. K_v and K_e are positive gain matrix. The conventional computed torque controller (CTC) has drawbacks in real application such as requirement of an exact model of the robot dynamics, which might be impossible and low robustness to structured and unstructured uncertainties.

The paper thus aims to design an adaptive controller with an orthogonal neural network so that it could compensates the dynamic uncertainties.

III. Third Order Sliding Mode Observer for Joint Veolocity

In this section, the TOSM observer is presented. The TOSM observer is expressed in form of state observer as

$$\begin{cases} \dot{x}_1 = x_2 + k_1 \left| x_1 - x_1 \right|^{2/3} sign(x_1 - x_1) \\ \dot{x}_2 = f(x_1, x_2, u) + k_2 \left| \dot{x}_1 - x_2 \right|^{1/2} sign(\dot{x}_1 - x_2) + \hat{z} \\ \dot{\hat{z}} = k_3 sign(\dot{x}_1 - x_2) \end{cases} \quad (5)$$

where k_i is the sliding mode gain to be designed and x_1 and x_2 are estimates of x_1 and x_2, respectively.

From Eqs. (3) and (5), the dynamics of state estimation error can be written as

$$\begin{cases} \dot{x}_1 = x_2 + k_1 \left| x_1 - x_1 \right|^{2/3} sign(x_1 - x_1) \\ \dot{x}_2 = d(x_1, x_2, x_2, u) + \phi(x_1, x_2, t) - k_2 \left| \dot{\hat{x}}_1 - \hat{x}_2 \right|^{1/2} sign(\dot{\hat{x}}_1 - \hat{x}_2) - \hat{z} \\ \dot{\hat{z}} = k_3 sign(\dot{x}_1 - x_2) \end{cases} \quad (6)$$

Where $x = x - x$ is the state estimate error. The differentiator converges to zero as

$$\phi(x_1, x_2, t) - k_2 \left| \dot{x}_1 - x_2 \right|^{1/2} sign(\dot{x}_1 - x_2) - \hat{z} = 0 \quad (7)$$

In Eq. (7), after the estimate state (x_1, x_2) converges to the true state (x_1, x_2) the Eq. (7) becomes as

$$\hat{z} = \phi(x_1, x_2, t) \quad (8)$$

From Eqs. (5) and (8), the uncertainties can be directly identified by the equivalent output injection $\hat{z}$.

IV. The Orthogonal Neural Network

In this section, the orthogonal neural network(ONN) is explained. It is based on feed forward neural network(FNN) and polynomial functions as active functions. According to the theory of orthogonal function, an arbitrary function $f(x), f:[a,b] \to \Re$ will have an orthogonal polynomial as

$$F_n(x) = w_1\lambda_1(x) + w_2\lambda_2(x) + ... + w_n\lambda_n(x) \quad (9)$$

Such that

$$\lim_{n\to\infty} \int_a^b (f(x) - F_n(x))^2 dx = 0 \quad (10)$$

where

$$\int_a^b \lambda_i(x)\lambda_j(x)dx = \begin{cases} 0 & i \neq j \\ A_i & i = j \end{cases} \quad (11)$$

$$w_i = \int_a^b f(x)\lambda_i(x)dx / A_i \quad i = 1, 2, ..., n \quad (12)$$

which $\{\lambda_1(x), \lambda_2(x), ...\}$ is an orthogonal set. The orthogonal functions such as Fourier series, Bessel function, Legendre, and Chebyshev polynomial.

In case that there is a function with m variables, an orthogonal function set is denoted as $\{\Psi_1(x), \Psi_2(x), ...\}$. Then, each orthogonal function is defined as

$$\Psi_i(X) = \lambda_{1i}(x_1)\lambda_{2i}(x_2)...\lambda_{mi}(x_m) \quad (13)$$

where $X = [x_1, x_2, ..., x_n]^T$ is input vector.

ONN based on feedforward network with one hidden layer is shown in Figure 1 by Tseng and Chen. Fig. 1 shows a) multi input-one output and b) multi input-multi output. The weight connections between input layer and hidden layer are 1 and the bias is 0. The $\lambda_i \ (i = 0,1,..,n)$ is an orthogonal function. The weights between them are w_{ij} (with i is number orthogonal function and j is number output). The output of ONN can be shown as

$$y = F(X) = \sum_{i=1}^{n} w_i(t)\Psi_i(X) = W^T(t)\Psi \qquad (14)$$

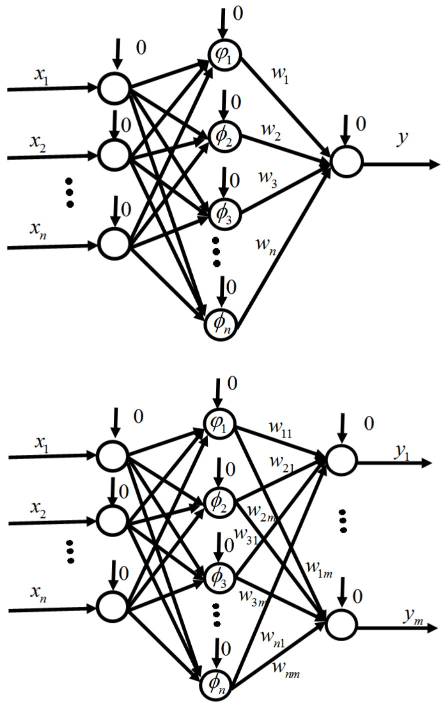


Figure 1.The orthogonal neural network. a) Single output b) Multi output

V. THE PROPOSED CONTROLLER

In this section, an adaptive control with an ONN and a TOSM observer is expressed as

$$\tau = M(q)[\ddot{q}_d + K_v(\dot{q}_d - q) + K_e(q_d - q)] + H(q,\dot{q}) + f_{ONN} \quad (15)$$

where f_{ONN} is output of ONN. The ONN is based on feed forward network and has three layers.

1) The input layer: The input vector of the ONN is denoted as

$$X = [x_1, x_2, x_3, x_4]^T = [e_1, \dot{e}_1, e_2, \dot{e}_2]^T \qquad (16)$$

where $e_i = q_d - q, \dot{e}_i = \dot{q}_d - \dot{q}, i = 1,2$ .In the ONN, the input must be inside [-1,1].The input inside [a, b] can be transformed as

$$T_i = \frac{2}{b-a}x_i - \frac{b-1}{b-a}, \quad T_i \in [-1,1] \qquad (17)$$

2) The hidden layer: The hidden layer using Chebyshev polynomial of the first kind for orthogonal function is expressed as

$$\lambda_i(x) = \frac{(-2)^i i!}{(2i)!}\sqrt{1-x^2}\frac{d^i}{dx^i}(1-x^2)^{i-1/2} \quad i = 0,1,2,...,n \quad (18)$$

Eq.(18) can be rewritten as

$$\begin{aligned}
&\lambda_0(x) = 1\\
&\lambda_1(x) = x\\
&\lambda_2(x) = 2x^2 - 1\\
&\lambda_3(x) = 4x^3 - 3x\\
&\lambda_4(x) = 8x^4 - 8x^2 + 1\\
&\lambda_5(x) = 16x^5 - 20x^3 + 5x\\
&\quad\vdots\\
&\lambda_{n+1}(x) = 2x\lambda_n(x) - \lambda_{n-1}(x)
\end{aligned} \qquad (19)$$

In this paper, n is chosen as 10.

3) The output layer: The weight connection matrix between the hidden layer and output layer is shown as

$$W = [w_1, w_2, ..., w_n]^T \qquad (20)$$

The output of ONN is simply expressed as

$$y = W^T\Psi \qquad (21)$$

where $\Psi = \{\Psi_1(X), \Psi_2(X), ..., \Psi_n(X)\}$ is shown in Eq. (13) .

The separate training of the ONN for each joint could be allowed. It means that the columns of weight matrix in Eq. (20) can be separately adjusted like column by column. And also, the weights of the ONN are adjusted with the gradient descent method. The weights update law of the ONN is given as

$$\dot{W} = \eta e \Psi \qquad (22)$$

where η and e are the learning rate and the learning error, respectively.

The learning error is defined with the sliding function as

$$s = \dot{e} + \Gamma e \qquad (23)$$

where Γ is positive matrix.

The block diagram of the proposed adaptive controller with an ONN and a TOSM observer is shown in Fig. 2.

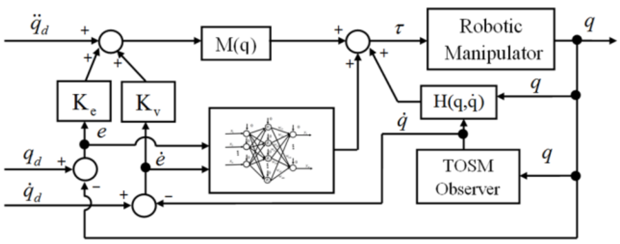


Figure 2. The block diagram of the proposed controller

VI. The Result Simulation

To show the effectiveness of the proposed controller, computer simulation for a 2DOF manipulator was carried out.

The kinematic and dynamic parameters of the manipulator shown in Fig. 3 are given in Table I.

TABLE. I. The Parameters of a Robot Manipulator.

Links	Parameter of each links	
	Length(m)	***Weight(Kg)***
Link 1	$l_1 = 1$	$m_1 = 1$
Link 2	$l_2 = 1$	$m_2 = 1$

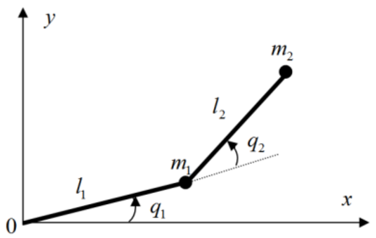


Figure 3. The 2-DOF robot manipulator .

The dynamic models of the manipulator in Eq. (1) is given as

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

$$V(q,\dot{q}) = \begin{bmatrix} -m_2 l_1 l_2 (2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2) s_2 \\ m_2 l_1 l_2 \dot{q}_1^2 s_2 \end{bmatrix}$$

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

F_f and τ_c are the friction matrix and load disturbance matrix, respectively. For this simulation, they are assumed to be as

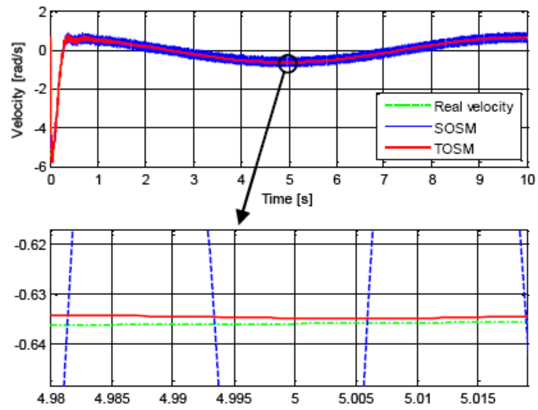
$$F_f = \begin{bmatrix} 1.2 q_1^2 \\ 3.2 q_2^2 \end{bmatrix}$$

and

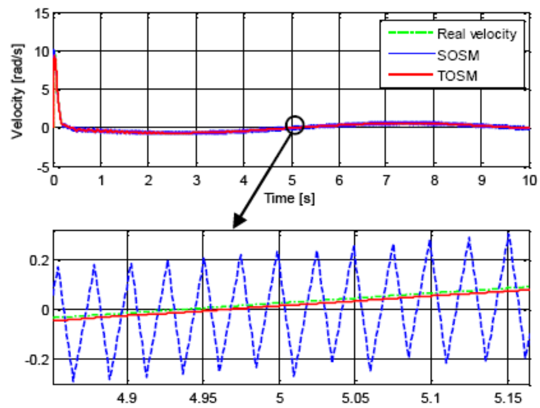
$$\tau_c = \begin{bmatrix} 0.25 \sin(2\dot{q}_1) \\ 3 \sin(2\dot{q}_2) \end{bmatrix}$$

where $c_i = \cos(q_i), s_i = \sin(q_i), i = 1, 2$, $s_{12} = \sin(q_1 + q_2)$, $c_{12} = \cos(q_1 + q_2)$ $g = 9.81(m/s^2)$, m_i is weight of link- i, l_i is length of link- i.

The desired inputs are set to be $q_1 = \sin(2t/\pi) - 1$ and $q_2 = \sin(t/\pi + \pi/2)$. The control gain parameters are set as $K_e = 20 I_{2\times2}$, $K_v = 100 I_{2\times2}$, where $I_{2\times2}$ is identity matrix with dimension 2x2. The learning rate of ONN is set as $\eta = 0.5$. The third order sliding mode gains are set as $K_1 = 90, K_2 = 20, K_3 = 50$. To show the effectiveness of the TOSM observer and the ONN compensation, the velocity estimation results of the TOSM is compared with the second order sliding mode (SOSM) observer and the joint position tracking errors from the proposed control is compared with those from the conventional computed torque control(CTC). Those results are given in Figs. 4-5.

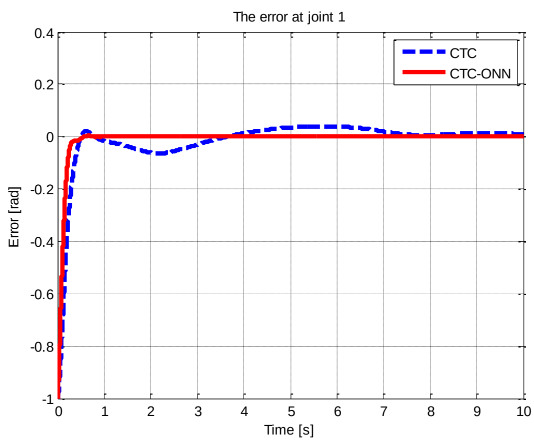


a) Velocity at joint 1

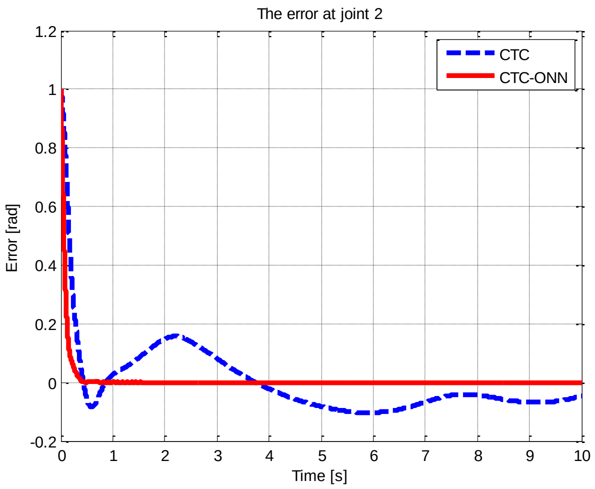


b) Velocity at joint 2

Figure 4.The estimate velocity at each joint.



a) Error at joint 1



b) Error at joint 2

Figure 5.The error at each joint.

VII. The Conclusion

This paper, a adaptive controller using combination of orthogonal neural network and third order sliding mode observers are presented. The result of tracking trajectory had been shown that the propose controller achieve higher accuracy than conventional CTC also the third order sliding mode observers reduce chatting than SOSM observers .

In this paper, an adaptive control with an orthogonal neural network (ONN) and a third order sliding mode (TOSM) observer for robot manipulators is proposed. From computer simulation for the 2 DOF manipulator, the TOSM observer is superior to the SOSM observer in estimating joint velocities and reducing chattering. And the proposed control is very effective in reducing joint position tracking errors compared with the conventional computed torque control.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2016R1D1A3B03930496)

References

[1] G. L. Luo and N. Saridis, “L-Q design of PID controllers for robot arms,” *IEEE Journal of Robotics and Automation*, vol. RA-1, no. 3, pp. 152-159, September 1985.

[2] F. Harashima, J. Xu, and H. Hashimoto, “Tracking control of robot manipulator using sliding mode,” *IEEE Transaction on Power Electronics*, vol. PE-2, no. 2, pp. 169-176, April 1987.

[3] K. Y. Lim and M. Eslami, “Adaptive controller design for robot manipulator systems using Lyapunov direct method,” *IEEE Transaction on Automatic Control*, vol. AC-30, no. 12, pp 1229-1233, December 1985.

[4] S. Y. Yi and M. J. Chung, “A robbust fuzzy logic controller for robot manipulators with uncertainties,” *IEEE Transaction on Systems, man, and Cybernetics*, vol. 27, no. 4, pp 706-713, August 1997.

[5] F. L. Lewis, “ Neural network control of robot manipulators,” *IEEE Expert*, vol. 11, No. 3, pp 64-75 , Jun 1996 .

[6] M. A. Arteaga and R. Kelly, “Robot control without velocity measurements: new theory and experimantal results,” *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp 297-308, April 2004.

[7] M. Van, H. Kang and Y. Suh, “Second order sliding mode-based output Feedback tracking control for uncertain robot manipulator,” *International Journal of Advanced Robotic Systems*, vol. 10, January 2013.

[8] C. C. De Wit and J. J. E. Slotine, “Sliding observers for robot manipulator,” *Automatica*, vol. 27, no. 5, pp. 859-864, 1991.

[9] Q. D. Le, H. J. Kang, T. D. Le, Adaptive extended computed torque control of 3 DOF planar parallel manipulators using neural network and error compensator. In: Huang, D.-S., Han, K., Hussain, A. (eds.) ICIC 2016. LNCS, vol. 9773, pp. 437–448. Springer, Cham (2016).

[10] Y. Gao, M. J. Er, S. Yang, “Adaptive control of robot manipulators using fuzzy neural networks,” *IEEE Transactions on Industrial Electronics*, vol. 48, pp. 1274-1278, no. 6, Dec 2001.

[11] Y. F. Yam, T. W. S Chow, C. T Leung, “A new method in determining the initial weights of feed forward neural networks,” *Neuralcomputing*, vol. 16, pp. 23-32, Junly 1997.

[12] S. S. Yang and C. S. Tseng, “An orthogonal neural network for function approximation,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 26, no. 5, pp 779-785, Oct 1996.

[13] S. L. Peric, D. S. Antic, M. B. Milovanovic, D. B. Mitic, M. T. Milojkovic, S. S. Nikolic, “Quasi-sliding mode control with orhtogonal endocrine neural network-based estimator applied in anti-lock braking system,” *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 2 pp. 754-764, April 2016.

[14] P. Wang, “Control of robot manipulators based on Legendre orthogonal neural network,” *Applied Mechanics and Materials*, vol. 427-429, pp. 1089-1092, Sept 2013.