A Neural-Network-Based-Method on Speed Control of Ultrasonic Motors

Y.C. Liang[*, **], X.H. Shi[*], H.P. Lee[**, ***], W.Z. Lin[**], X. Xu[****], S.P. Lim[***] and K.H. Lee[***]

[*] College of Computer Science and Technology, Jilin University, 10 Qian Wei Road, Changchun, 130012, P.R. China, liangyc@ihpc.a-star.edu.sg, shixiaohu@vip.sina.com.cn
[**] Institute of High Performance Computing, 1 Science Park Road, #01-01 The Capricorn, Singapore Science Park II, Singapore 117528, hplee@ihpc.a-star.edu.sg, linwz@ihpc.a-star.edu.sg
[***] Department of Mechanical Engineering, National University of Singapore, 9 Engineering Drive 1, Singapore 119260, mpelimsp@nus.edu.sg, mpeleekh@nus.edu.sg
[****] College of Mathematics, Jilin University, Changchun, 130012, P.R. China, xuxu567@mail.jl.cn

ABSTRACT

An improved Elman network models, output-input feedback (OIF) Elman networks is proposed based on the modified Elman network. A neural-network-based iteration control system (NNICS)) is developed by using the OIF Elman network as a passageway of the error back-propagation. The speed of the ultrasonic motor is identified using the modified Elman network and OIF Elman networks, respectively and some useful comparable results are presented. Numerical results show that the NNICS controller is effective for various kinds of reference speeds of the USM and the proposed scheme is fairly robust against random disturbance to the control variable.

Keywords: Elman neural network, recurrent back propagation, ultrasonic motor, control.

1 INTRODUCTION

The ultrasonic motor (USM) has excellent performances and many useful features, so it has been used in many practical applications and has attracted many researchers in the study on USM modeling [1-2]. The operational characteristics of the USM depend on many factors, therefore it is difficult to construct a precise model of the USM. In recent years, some models of the USM have been proposed, but most models are too complex to apply to practical applications. Therefore, it is of significance in theory and applications to construct practical models and adjust input parameters properly to realize the quick and precise speed control.

Neural networks have good potential for identification and control applications because they can approximate nonlinear input-output mappings of systems. More recently, many papers have been published in this field [3-6]. In this paper, focusing on the USM, an improved Elman neural network identifiers and a newly developed NNICS controller are constructed. Numerical results show that the improved network can approximate the nonlinear input-output mapping of the USM quite well. Good effectiveness of the proposed NNICS controller is obtained for some different kinds of reference speeds.

2 IMPROVED ELMAN NEURAL NETWORKS

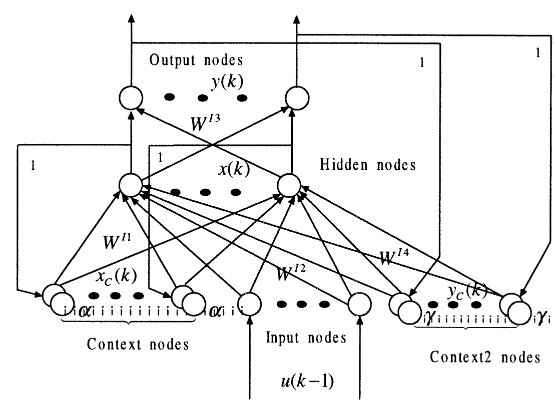

Figure 1: Architecture of the OIF Elman network.

Based on the Elman and modified Elman networks [7-9], one improved Elman network, named as output-input feedback Elman (OIF Elman) network is presented in this paper, which add the feedback of the output nodes to the first layer. Fig 1 shows its structure. There are r nodes in the input layer, n nodes in the hidden and context layers, and m nodes in the output and the second context layers, respectively. The input u is an r dimensional vector, the output x of the hidden layer and the output x_C of the context nodes n dimensional vectors, respectively, the output y of the output layer and output y_C m dimensional vector, and the weights W^{I1}, W^{I2}, W^{I3} and W^{I4} the $n\times n$, $n\times r$, $m\times n$, and $n\times r$ dimensional matrices, respectively.

The mathematical model of OIF Elman neural network is

$$x(k)=f(W^{I1}x_C(k)+W^{I2}u(k-1)+W^{I4}y_C(k)) \quad (1)$$

$$x_C(k)=\alpha x_C(k-1)+x(k-1); \quad (2)$$

$$y_C(k)=\gamma y_C(k-1)+y(k-1); \quad (3)$$

$$y(k)=W^{I3}x(k) \quad (4)$$

Let the kth desired output of the system be $y_d(k)$. Define the error as $E(k)=\frac{1}{2}(y_d(k)-y(k))^T(y_d(k)-y(k))$. Differentiating E with respect to the weights of the networks, according to the gradient descent method, we obtain the learning algorithms of the OIF Elman network. The equations for modifying W^{I1}, W^{I2}, W^{I3} and W^{I4} are as follows

$$\Delta w_{ij}^{I3}=\eta_3\delta_i^0 x_j(k)\quad(i=1,2,\cdots,m;\ j=1,2,\cdots,n) \tag{5}$$

$$\Delta w_{jq}^{I2}=\eta_2\delta_j^h u_q(k-1)\quad(j=1,2,\cdots,n;\ q=1,2,\cdots,r) \tag{6}$$

$$\Delta w_{jl}^{I1}=\eta_1\sum_{i=1}^{m}(\delta_i^0 w_{ij}^{I3})\frac{\partial x_j(k)}{\partial w_{jl}^{I1}}\quad(j=1,2,\cdots,n;\ l=1,2,\cdots,n) \tag{7}$$

$$\Delta w_{js}^{I4}=\eta_4\sum_{i=1}^{m}(\delta_i^0 w_{ij}^{I3})\frac{\partial x_j(k)}{\partial w_{js}^{I4}}\quad(j=1,2,\cdots,n;\ s=1,2,\cdots,m) \tag{8}$$

$$\delta_i^0=(y_{d,i}(k)-y_i(k)) \tag{9}$$

$$\delta_j^h=\sum_{i=1}^{m}(\delta_i^0 w_{ij}^{I3})f_j'(\cdot) \tag{10}$$

$$\frac{\partial x_j(k)}{\partial w_{jl}^{I1}}=f_j'(\cdot)x_l(k-1)+\alpha\frac{\partial x_j(k-1)}{\partial w_{jl}^{I1}} \tag{11}$$

$$\frac{\partial x_j(k)}{\partial w_{js}^{I4}}=f'_j(\cdot)y_s(k-1)+\gamma\frac{\partial x_j(k-1)}{\partial w_{js}^{I4}} \tag{12}$$

where η_1, η_2, η_3 and η_4 are the learning steps of W^{I1}, W^{I2}, W^{I3} and W^{I4}, respectively.

3 SPEED CONTROL OF USM BASED ON NNICS

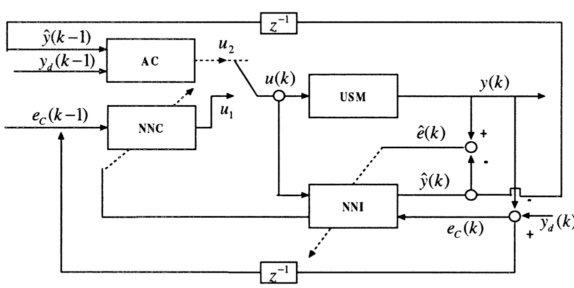

Figure 2: The Block of NNICS.

In this section an artificial neural network and iteration based control system (NNICS) is proposed for controlling the speed of the USM. The block diagram of the speed control is shown in Fig. (2). In the figure, NNI (Neural Network Identifier) represents the neural network identifier. The OIF Elman network proposed in the last section is employed as the NNI whose function is to provide a back propagation passage for system errors. The input of the NNI is the output $u(k)$ of the NNC. The output of the NNI is obtained through on-line learning described in the last section and is denoted as $\hat{y}(k)$. The training error is $\hat{e}(k)=y(k)-\hat{y}(k)$. NNC is the neural network controller, which performs the initial control. When the control error is small than a threshold defined in the IC, an iterative controller will replace the NNC to perform the control.

3.1 Neural Network Controller (NNC)

The NNC has one input node, three hidden nodes and one output node. The input of the NNC is the system error in the last time, which is denoted as $e_C(k-1)$. The output of the NNC is taken as the control parameter $u(k)$ of the USM. Because that the derivative of the error of the last time with respect to the weights is employed during the learning, the model possesses recurrent back propagation characteristics.

The outputs of the hidden layer and the output layer are $v_i(k)=f(e(k-1)w_i^{C1}(k))$, $u(k)=\sum_{i=1}^{3}v_i(k)w_i^{C2}(k)$ respectively. Where w^{C1} and w^{C2} are the weights of the first and second layers, v and u the outputs of the hidden layer and the output layer, and $f(\cdot)$ the sigmoidal activation function, respectively. The input and output of the USM are $u(k)$ and $y_C(k)$, respectively. Let the desired output of the USM be $y_d(k)$ and the difference between the actual output and the desired output be $e_C(k)=y_d(k)-y_C(k)$. Define the error of the system as $E=\frac{1}{2}e_C{}^2(k)=\frac{1}{2}(y_d(k)-y_C(k))^2$. Differentiating E with respect to the weights of the networks, according to the gradient descent method, we obtain the learning algorithm of NNC as:

$$w^{Cd}(k+1)=w^{Cd}(k)+\Delta w^{Cd}(k)\quad(d=1,2) \tag{13}$$

$$\Delta w^{Cd}(k)=-\eta^{Cd}\frac{\partial E}{\partial w^{Cd}}\quad(d=1,2) \tag{14}$$

$$\frac{\partial E}{\partial w^{Cd}}\approx -e_C(k)\frac{\partial\hat{y}(k)}{\partial u}\frac{\partial u(k)}{\partial w^{Cd}}\quad(d=1,2) \tag{15}$$

$$\frac{\partial u(k)}{\partial w_i^{C2}}\approx v_i(k)-\sum_{j=1}^{3}v_j(k)(1-v_j(k))w_j^{C1}\frac{\partial\hat{y}(k-1)}{\partial u}\frac{\partial u(k-1)}{\partial w_i^{C2}}w_j^{C2} \tag{16}$$

$$\frac{\partial u(k)}{\partial w_i^{C1}}=e_C(k)v_i(k)(1-v_i(k))w_i^{C2}-\sum_{j=1}^{3}v_j(k)(1-v_j(k))w_j^{C2}\frac{\partial\hat{y}(k-1)}{\partial u}\frac{\partial u(k-1)}{\partial w_i^{C1}}w_j^{C1}, \tag{17}$$

$\frac{\partial y_C(k)}{\partial u}$ could be obtained by the following Eqs. (18-20):

$$\frac{\partial\hat{y}(k)}{\partial u}=\sum_{i=1}^{n}\left(x_i(1-x_i)w_i^{I3}w_i^{I2}+\frac{\partial y_C(k)}{\partial u}w_i^{I4}\right), \tag{18}$$

$$\frac{\partial y_C(k)}{\partial u}=\gamma\frac{\partial y_C(k-1)}{\partial u}+\frac{\partial\hat{y}(k-1)}{\partial u}, \tag{19}$$

$$\frac{\partial y_C(0)}{\partial u}=0;\ \frac{\partial\hat{y}(0)}{\partial u}=0 \tag{20}$$

where η^{C1} and η^{C2} are the learning steps of w^{C1} and w^{C2}, respectively.

3.2 Iterative Controller (IC)

In order to accelerate the convergent speed of the controller, we propose that an iterative controller (IC) is used to replace the NNC described in Section 3.1. When $e(k) < e_p$, we consider that the control parameters have nearly approached the expected ones. Then we use the IC to accelerate the convergent speed. Assume that the NNI could approach the input and output relationship of the USM $y(k) = f\left(y(k-1), y(k-2), ..., y(k-n), u(k-1), u(k-2), ..., u(k-m)\right)$ in a given precision, that is, $\forall \varepsilon > 0$, $\exists$W, such that the NNI satisfies $\|y(k) - \hat{y}(k)\| < \varepsilon$. Define $g(u) = y_d(k) - \hat{y}(k)$. If $g(u)$ is small enough, then $y_d(k)$-$y(k)$ would be also small .

Expand $g(u)$ in Taylor's series and take the linear form. Let it equal to zero, that is, $g(u) = g(u_0) + g'(u_0)(u - u_0) = 0$, then we have

$$u = u_0 - \frac{g(u_0)}{g'(u_0)} = u_0 + \frac{y_d(k) - \hat{y}(k)}{\partial\hat{y}(k)/\partial u} \tag{21}$$

Eq. (21) is the formula of Newton's algorithm to solve the linear equation. To guarantee the convergence, we introduce the learning rate λ into the above Eq. as follows:

$$\begin{cases} u_2(k, i+1) = u_2(k, i) + \lambda \dfrac{y_d(k) - \hat{y}(k)}{\partial\hat{y}(k)/\partial u} \\ u_2(k, 0) = u_2(k-1, M) \\ u_2(k_0, 0) = u_1(k_0) \end{cases} \tag{22}$$

The above iterations are performed between two consecutive samplings. M is the number of the iterations, $\lambda > 0$ is the learning rate, k_0 is the moment of the control error reached e_p, $\frac{\partial\hat{y}(k)}{\partial u}$ could be obtained by Eqs. (18-20).

4 NUMERICAL RESULTS

Numerical simulations are performed using the proposed method for the identification and speed control of

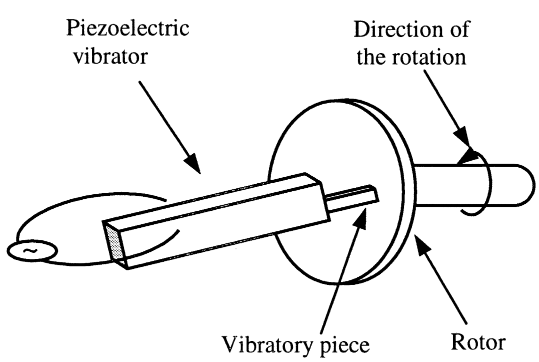

Figure 3:Schematic diagram of the USM.

frequency 27.8 *kHZ*, amplitude of driving voltage 300*V*, permissible output moment $2.5kg \cdot cm$, rotation speed 3.8 *m/s*. Both gain factors α and γ in the NNI are taken as 0.4.

4.1 Simulation on USM Using Modified and Improved Elman Networks

The USM model is simulated by using the modified Elman network and the OIF Elman network. Figs. 4 and 5 show the simulated speed curves for samples of 1000 times/s and 300 times/s using the USM and using two neural network models, respectively, a random disturbance is added at the second 0.4.

Examining the simulated results we could observe that the modified Elman model is superior to the improved one when the sampling frequency is higher, whereas the improved model are superior to the modified Elman model when the sampling frequency is lower. For example, the simulation results of the modified Elman network are superior to those of the improved Elman network for samples of 1000 times/s, on the other hand, the modified Elman model is superior to the improved model when the samplings are more than 1000 times/s. The simulation results are basically the same when the samplings are around 600 times/s. These observed results enable us to determine which neural network models should be used according to the situation of samplings in the testing ground.

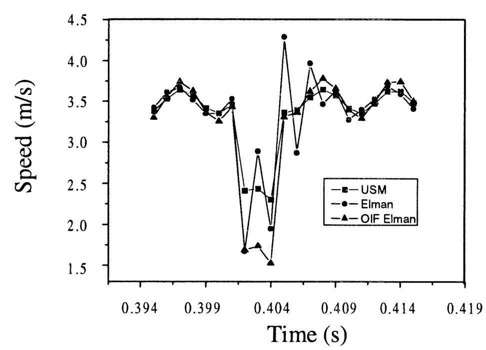

Figure 4: Speed curves for samples of 1000 times/s.

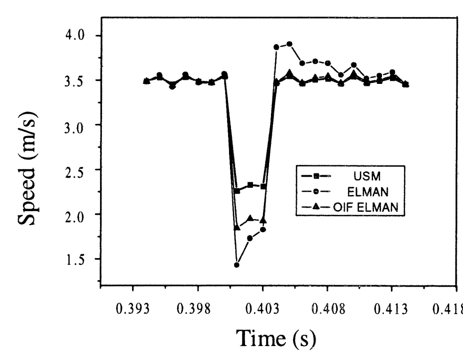

Figure 5: Speed curves for samples of 300 times/s.

4.2 Speed Control of USM Using NNICS Network

Fig. 6 shows the USM speed control curves using the NNICS method and the method in [6] when the control speed is taken as 3.6 m/s and the external moment is taken as 1 $N \cdot cm$. From the figure it can be seen that the fluctuation is large when the method in [6] is employed, the average fluctuation reaches 1.6% when the existing method is employed, whereas, it is only 0.26% when the method proposed in this paper is used. The fluctuation is defined as $\zeta = (V_{\max} - V_{\min}) / V_{aver} \times 100\%$. Where $V_{\max}, V_{\min}$ and V_{aver} represent the maximum, minimum and average values of the speeds, respectively. Figure 7 shows the speed control curves that the reference speed varies as sinusoidal type. From the figure it can be seen that the proposed method possesses good control precision.

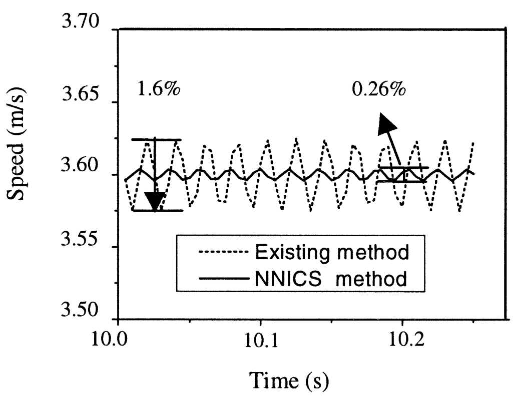

Figure 6:Comparison of speed control curves using different schemes.

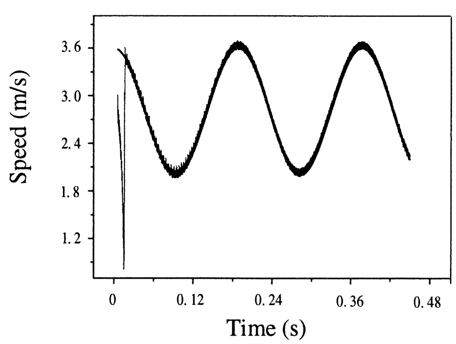

Figure 7: Speed control for sinusoidal type.

5 CONCLUSIONS

An improved neural network model adding recurrent output is presented based on the modified Elman network. Good results are obtained in numerical experiments by using the modified and improved models to simulate the USM. Simulated results show that the modified Elman model is superior to the improved model when the sampling frequency is higher, whereas the improved model are superior to the modified Elman model when the sampling frequency is lower. The observed results enable us to determine the suitable selection on neural network models according to the samplings.

The OIF Elman network is employed to be the passageway of the error back propagation. A neural-network-based recurrent back propagation controller is developed and the control on the USM is performed. Good control effect can be obtained using the improved proposed NNICS controller for different kinds of speeds of constant, and step types. Comparisons with the existing method show that the precision of control can be increased using the proposed method.

REFERENCES

[1] Senjyu T, Yokoda S and Uezato K. A study on high-efficiency drive of ultrasonic motors Electric Power Components and Systems, 2001. 29: 179-189.

[2] Uehino K. Piezoelectric motors: Overview. Smart Materials and Structures , 1998. 7: 273-285.

[3] Senjyu T, Miyazato H, Yokoda S and Uezato K. Speed control of ultrasonic motors using neural network, IEEE Transactions on Power Electronics, 1998. 13(3): 381-387.

[4] Senjyu T, Yokoda S and Uezato K. Speed control of ultrasonic motors using fuzzy neural network, Journal of Intelligent Fuzzy System, 2000. 8 (2): 135-146.

[5] Lin FJ, Wai RJ and Hong CM. Recurrent neural network control for LCC-resonant ultrasonic motor drive, IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, 2000. 47 (3): 737-749.

[6] Shi XH, Liang YC, Lee HP, Lin WZ, Xu X and Lim SP. Improved Elman networks and applications for controlling ultrasonic motors, submitted to Applied Artificial Intelligence.

[7] Elman JL. Finding structure in time, Cognitive Science, 1990. 14: 179-211.

[8] Pham DT and Liu X. Dynamic system modeling using partially recurrent neural networks Journal of Systems Engineering, 1992. 2: 90-97.

[9] Pham DT and Karaboga D. Training Elman and Jordan networks for system identification using genetic algorithms. Artificial Intelligence in Engineering, 1999. 13: 107–117.

[10] Xu X, Liang YC, Lee HP, Lin WZ, Lim SP, Lee KH and Shi XH. Mechanical modeling of a longitudinal oscillation ultrasonic motor and temperature effect analysis. Submitted to Smart Materials and Structures

[11] Sashida T. Trial construction and operation of an ultrasonic vibration driven motor. Applied Physics, 1982. 51(6): 713-720.