ADAPTIVE CONTROL OF AUTONOMOUS UNDERWATER VEHICLE BASED ON FUZZY NEURAL NETWORK

Received 20th April 2009; accepted 15th September 2009.

Zheng Qin, Jason Gu

Abstract:

This paper presents an adaptive control method based on fuzzy neural network for Autonomous Underwater Vehicle (AUV). The Fuzzy Neural Network (FNN) could build the inverse model of AUV through on-line learning algorithm, which is free of fuzzy neural network structure knowledge and prior fuzzy inference rules. The adaptive controller for AUV based on FNN is proposed, and then the stability of the resulting AUV closed-loop control system is analyzed by Lyapunov stability theory. The validity of the proposed control method has been verified through computer simulation experiments.

Keywords: *autonomous underwater vehicle, fuzzy neural network, adaptive control, stability.*

1. Introduction

An autonomous underwater vehicle may be defined as a vehicle with a sensorial system and an actuator system, working in an uncertain and unstructured underwater environment (unknown and under severe disturbance), managed by control system and able to undertake a user specified mission (such as ocean mapping, surveying or monitoring). Because of the highly nonlinear dynamics of an autonomous underwater vehicle (AUV) and the difficulty in modeling the environment and its interaction with the AUV, controlling an AUV in an underwater environment presents many theoretical and engineering challenges. Various advanced control systems have been proposed such as sliding-mode, robust and nonlinear control strategies have been applied with some success [1]–[5]. Adaptive control as well as fuzzy-logic control and neural-network-based control have also been proposed [6]–[13].

One of the earlier researches in AUV control was developed by Yoerger and Slotine [5]. They proposed a sliding-mode controller and investigated the effects of uncertainty of hydrodynamic coefficients and negligence of cross-coupling terms. The result showed that the adaptive sliding-mode controller outperformed a conventional linear controller for a wide range of velocity. Fossen and Blanke [3] derived an output feedback controller using nonlinear control theory and feedback from the axial water velocity. They proved that a nonlinear observer combined with an output feedback integral controller provides exponential stability. Healey and Lienard [4] designed a sliding-mode controller for a six-degrees-of-freedom (DOF) AUV control. They decomposed the system into noninteracting subsystems and grouped certain key functions for the separate functions of steering, diving and speed control. Choi and Yuh [6] developed an adaptive controller based on bound estimation and implemented it for AUV control. Ishii *et al.* [7] proposed a neural-network-based controller associated with an adaptation method named "Imaginary Training" for heading-keeping control of an AUV called "Twin-Burger."

In view of the versatilities of neural networks and fuzzy logic, a fuzzy neural network can be expected to exhibit many advantages. The combination of fuzzy inferences and neural networks has been researched extensively recently. Fuzzy Neural systems are multilayered connected networks that realize the elements and functions of traditional fuzzy logic control, decision systems. A trained fuzzy neural system is isomorphic to a fuzzy system extracted and interpreted from the network. The system can automatically and simultaneously identify fuzzy logic rules and adapt its membership function. By utilizing the learning capability of neural network, the systems can construct input-output mapping for many applications.

Jang [14] proposed architecture of fuzzy neural network called Adaptive-Network-Based Fuzzy Inference System (ANFIS), which can be taken as a basis for constructing a set of fuzzy rules, and generate the stipulated input-output pairs. Pattern learning or on-line learning is adopted to update the network. Cho and Wang [15] used the number of input variables and number of rules to determine the structure of their neural network. Then the network is trained by using back-propagation algorithm. Although the above methods can construct the input-output mapping for many applications. However no efficient process for reducing the complexity of a fuzzy neural network has been suggested. To increase the efficiency and reduce the complexity of the network, many self-constructing fuzzy neural networks have been proposed in the literature. These systems are inherently modified Takagi-Sugeno-Kang type fuzzy rule based models processing neural network's learning ability. Juang and Lin [16] proposed such a system. There were no rules in that SONFIN initially. The inference rules were created and adapted as on-line learning process *via* simultaneous structure and parameter identification. Based on that generalized fuzzy neural networks, Juang and Lin [17] also designed an adaptive controller for the robotic manipulators. The simulation results showed that the error convergence rate with the Adaptive Fuzzy Neural Control (AFNC) was fast, the flexibility adaptation and tracking performance of this adaptive control system was verified theoretically. The asymptotic stability of the MIMO control system was also established and analyzed using Lyapunov approach [18].

Wang and Lee [19] develop their previous feed forward neuro-fuzzy systems to recurrent neuro-fuzzy systems with better control performance and learning convergence. They use recurrent neuro-fuzzy network to model the inverse dynamics of an AUV and the feedback-error-learning method to on-line fine-tune the parameters of the recurrent Neuro-fuzzy controllers.

This paper extends the previous fuzzy-neural control approaches and presents an adaptive controller based on FNN to learn the inverse model of AUV. The research aims to develop an AUV motion controller, which is free of the pre-designed structure of Neural Network and is free of the pre-training process. The inverse dynamic model of AUV could be obtained through on-line adaptive learning, resulting in improved flexibility and robustness of AUV control system.

The rest of the paper is organized as follows. Section II briefly describes the structure and learning algorithm of FNN. The dynamic model for AUV is proposed in section III. In section IV, the adaptive controller based on FNN is designed and its stability is proved in section V. In section VI computer simulations of the proposed control scheme and PD controller are conducted and their performances are compared to validate the effectiveness of the proposed approach. Finally, section VII concludes the paper.

2. The Architecture and Learning Algorithm of FNN

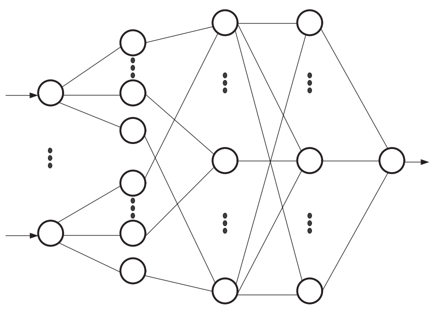


Fig. 1 Architecture of FNN.

The FNN is based on extended RBF neural network. The architecture is shown in Fig. 1, which includes 5 layers, where

Layer 1: Each node in layer 1 represents an input linguistic variable.

Layer 2: Each node in layer 2 represents a membership function (MF), which is in the form of Gaussian functions:

$$\mu_{ij} = \exp[-\frac{(x_i - c_{ij})^2}{\sigma_j^2}],\ i=1,2,\cdots r;\ j=1,2,\cdots u \qquad (1)$$

Where μ_{ij} is jth membership function of x_i; c_{ij} is the center of the jth Gaussian membership function of x_i; σ_j is the width of the jth Gaussian membership function of x_i; r is member of input variables; u is the member of membership functions.

Layer 3: Each node in layer 3 represents a possible IF-part for fuzzy rules. For the jth rule R_j, its output is:

$$\phi_j = \exp[-\frac{\sum_{i=1}^{r}(x_i - c_{ij})^2}{\sigma_j^2}] = \exp[-\frac{\|X - C_j\|^2}{\sigma_j^2}] \qquad (2)$$

$$j=1,2,\cdots u$$

Where $X=(x_1, x_2, \cdots x_r)$ and C_j is the center of jth RBF unit.

Layer 4: We refer to these nodes as N (normalized) nodes. Obviously, the number of N nodes equals to that of R nodes. The output of the N_j node is:

$$\psi_j = \frac{\phi_j}{\sum_{k=1}^{u}\phi_k} = \frac{\exp[-\frac{\|X - C_j\|^2}{\sigma_j^2}]}{\sum_{k=1}^{u}\exp[-\frac{\|X - C_k\|^2}{\sigma_k^2}]},\ j=1,2,\cdots u \qquad (3)$$

Layer 5: Each node in this layer represents an output variable as the summation of incoming signals

$$y(X) = \sum_{k=1}^{u} w_{2k}\psi_k = \frac{\sum_{k=1}^{u} w_{2k}\exp[-\frac{\|X - C_j\|^2}{\sigma_j^2}]}{\sum_{k=1}^{u}\exp[-\frac{\|X - C_k\|^2}{\sigma_k^2}]} \qquad (4)$$

Where y is the value of an output variable and w_{2k} is the weight of each rule.

(4) can be also written as

$$Y=W^T\Psi \qquad (5)$$

An on-line GD-FNN learning algorithm can be applied for the above network, which is described in detail in [21] and [22]. The learning algorithm includes structure identification and parameters estimation, which are performed automatically and simultaneously. Where the structure identification is to determine the number of membership functions and gain fuzzy rules. Parameters estimation include modification the parameters in IF parts and THEN parts of the fuzzy rules. There are two criteria on the new fuzzy rules generation, that are the system error must be larger than threshold K_e, and the minimum Mahalanobis distance must be larger than the threshold K_{md}. When a new rule is generated, the parameters of GD-FNN would be updated. The current training sample inputting to the network are used as the center of a Gussian membership function of a new rule. The initial weights of GD-FNN can be determined by orthogonal least squares algorithm. The test results with an articulated two-link manipulator show that the learning algorithm is superior in terms of learning efficiency and performance.

3. Dynamic model for AUV

In order to research on the controller, we are considering the 6-DOF dynamic model of AUV described in [23]. In general, the nonlinear dynamic equations of motion of a six- DOF AUV expressed in the body-fixed coordinate frame can be written as:

$$M\dot{v}+C(v)v+D(v)v+g(\eta)+\tau_d=\tau \qquad (6)$$

$$\dot{\eta}=J(\eta)v \qquad (7)$$

Where M is the inertial matrix including both rigid-body mass and added mass; $C(v)$ is the coriolis and centripetal matrix including rigid-body mass and added mass; $D(v)$ is the total hydrodynamic matrix that includes radiation-induced potential damping, linear skin friction, wave drift damping and damping due to vortex shedding; $g(\eta)$ contains the restoring terms formed by the vehicle's buoyancy and gravitational terms; τ_d represents disturbances (e.g., wave and current) from environmental forces and moments acting on the vehicle. τ includes the control forces and moments; $J(\eta)$ is a velocity transformation matrix (a Jacobian matrix) that transforms the vehicle-fixed velocities to those of the earth-fixed reference frame; Translational and rotational movements in the global reference frame are represented by η that includes earth-fixed position and Euler angles; v consists of six velocity components of motion (surge, sway, heave, roll, pitch, and yaw) in the vehicle coordinate system. In the body fixed frame, the dynamic model has the following characteristics:

$$M=M^T>0, M=0;\ C(v)=-C^T(v);\ D(v)=0.$$

The dynamic model can be written in the global coordinate frame as follows:

$$M_\eta(\eta)\dot{\eta}+D_\eta(v,\eta)\dot{\eta}+g(\eta)+\tau_{d\eta}=J^{-T}(\eta)\tau \qquad (8)$$

Where $M_\eta(\eta)=J^{-T}(\eta)MJ^{-1}(\eta)$;

$C_\eta(v,\eta)=J^{-T}(\eta)[C(v)-MJ^{-1}(\eta)\dot{J}(\eta)]J^{-1}(\eta)$;

$D_\eta(v,\eta)=J^{-T}(\eta)D(v)J^{-1}$;

$g_\eta(\eta)=J^{-T}(\eta)g(\eta)$; $\tau_{d\eta}=J^{-T}(\eta)\tau_d$;

$$v=J^{-1}(\eta)\dot{\eta} \qquad (9)$$

$$\dot{v}=J^{-1}(\eta)[\ddot{\eta}-\dot{J}(\eta)J^{-1}(\eta)\dot{\eta}] \qquad (10)$$

In the global coordinate frame, the dynamic model has the following characteristics:

$$M_\eta(\eta)=M_\eta^T(\eta)>0,\ \dot{M}_\eta(\eta)\neq 0;$$

$$C_\eta(v,\eta)\neq -C_\eta^T(v,\eta);\ D_\eta(v,\eta)>0.$$

4. The adaptive controller based on fuzzy neural network

Suppose that the expected motion state of AUV is limitary, that is:

$$\left\|\begin{bmatrix}\eta_d^T & \dot{\eta}_d^T & \ddot{\eta}_d^T\end{bmatrix}^T\right\|\leq\eta_B \qquad (11)$$

Where, η_d is the expected position vector in the global coordinate frame; $\dot{\eta}_d$ is the expected velocity vector in the global coordinate frame; $\ddot{\eta}_d$ is the expected acceleration vector in the global coordinate frame; η_d is a positive constant.

The filtering tracking error of AUV can be defined as:

$$s=\dot{\tilde{\eta}}+\lambda\tilde{\eta} \qquad (12)$$

Where $\tilde{\eta}=\eta_d-\eta$, $\dot{\tilde{\eta}}=\dot{\eta}_d-\dot{\eta}$, and λ is a positive constant.

Therefore (12) can be described as the following equations

$$s=\dot{\eta}_r-\dot{\eta} \qquad (13)$$

$$\dot{\eta}_r=\eta_d+\lambda\tilde{\eta} \qquad (14)$$

Where, $\dot{\eta}_r$ is the virtual reference track in the global coordinate frame.

The virtual reference track meets with the following equations:

$$\dot{\eta}_r=J(\eta)v_r \qquad (15)$$

$$\dot{v}_r=J^{-1}(\eta)[\ddot{\eta}_r-\dot{J}(\eta)J^{-1}(\eta)\dot{\eta}_r] \qquad (16)$$

Taking derivatives to both sides of (13) and substituting the dynamic model of AUV, the dynamic equation of system error of AUV can be organized as following:

$$\begin{aligned}M_\eta\dot{s}&=M_\eta\ddot{\eta}_r-M_\eta\ddot{\eta}=\\&-(D_\eta+C_\eta)s+J^{-T}[M\dot{v}_r\\&+C(v)v_r+D(v)v_r+g(\eta)+\tau_d-\tau]\end{aligned} \qquad (17)$$

To obtain (17), the following equation has been applied:

$$M_\eta\ddot{\eta}_r+C_\eta\dot{\eta}_r+D_\eta\dot{\eta}_r+g(\eta)$$

$$=J^{-T}(M\dot{v}_r+C(v)v_r+D_\eta v_r+g) \qquad (18)$$

Define that

$$f(\dot{v}_r,v_r,v,\eta)=M\dot{v}_r+C(v)v_r+D(v)v_r+g(\eta) \qquad (19)$$

The FNN and its learning algorithm are adopted to approach , $f(\dot{v}_r,v_r,v,\eta)$ yielding

$$f(\dot{v}_r,v_r,v,\eta)=W^T\psi(\dot{v}_r,v_r,v,\eta)+\varepsilon \qquad (20)$$

Where, ε is the approaching error, and $\|\varepsilon\|\leq\varepsilon_N$, ε_N, W is positive constant; is the vector of weights of the ideal FNN after learning.

The control instance is designed as:

$$\tau=\hat{f}(\dot{v}_r,v_r,v,\eta)+J^TK_ds+\alpha \qquad (21)$$

$$\dot{\hat{W}}=g(\psi,s,\hat{W}) \qquad (22)$$

$$\hat{f}(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) = \hat{W}^{\mathrm{T}} \psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) \quad (23)$$

Substituting (19), (21) and (23) into (17), gives the dynamic equations of error of the closed-loop system:

$$M_\eta \dot{\boldsymbol{s}} = -(D_\eta + K_d)\boldsymbol{s} - C_\eta \boldsymbol{s} + J^{-\mathrm{T}}[\tilde{f}(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) + \varepsilon + \tau_d - \alpha] \quad (25)$$

$$\tilde{f}(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) = W^{\mathrm{T}}\psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) - \hat{f}(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) = \tilde{W}^{\mathrm{T}}\psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) \quad (26)$$

$$\tilde{W} = W - \hat{W} \quad (27)$$

Where, $\tilde{f}(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta)$ is the estimated error;

$\hat{f}(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta)$ is the estimation of $f(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta)$;

$\hat{W}$ is the estimation of matrix W; K_d is the gain matrix, which meets with $K_d = K_d^{\mathrm{T}} > 0$; α is the robust controller which is used to increase system robustness to approaching error of the neural network and the environment disturbances. From (21) we can see that the adaptive controller is the integration of fuzzy neural network controller $\hat{f}(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta)$, PD controller $J^{\mathrm{T}}K_d s$ and robust controller α.

5. Stability analysis

In order to guarantee the stability of the control system, the Fuzzy Neural Network must be convergent. In other words, the parameters of fuzzy neural network must be bounded. From (5), we know that if the weights of the network W are bounded, then the network must be bounded. Define the constraints set Γ for resulting weight matrix W:

$$\Gamma = \{\|w_k\| \le \|w_k(0)\|, k = 1, \ldots, N_0\} \quad (28)$$

Also define

$$B = J^{-1}\boldsymbol{s} = [b_1, b_2, \cdots, b_{N_0}]^{\mathrm{T}} \quad (29)$$

The adaptive law of the weights of network can be described as:

$$\dot{w}_k = \begin{cases} k\psi b_k; \\ \text{if } (\|w_k\| = \|w_k(0)\| \text{ or} \\ (\|w_k\| = \|w_k(0)\| \text{ and } w_k^{\mathrm{T}}\psi b_k \le 0) \\ k(I - \dfrac{w_k w_k^{\mathrm{T}}}{\|w_k\|^2})\psi b_k; \\ \text{if } (\|w_k\| = \|w_k(0)\|) \text{ and } w_k^{\mathrm{T}}\psi b_k > 0 \end{cases} \quad (30)$$

Where, k is a positive constant.

Theorem 1. If the initial values of the weights satisfy $w_k(0) \in \Gamma$, and the adaptive law (30) is adopted, then the weights satisfy $w_k(t) \in \Gamma$, $\forall t > 0$.

Proof: We consider the following Lyapunov function candidate:

$$V_b = \frac{1}{2} w_k^{\mathrm{T}} w_k \quad (31)$$

Taking the derivative of the Lyapunov function with respect to time, we have

$$\dot{V}_b = \frac{1}{2} w_k^{\mathrm{T}} \dot{w}_k \quad (32)$$

When $\|w_k\| = \|w_k(0)\|$ and $w_k^{\mathrm{T}}\psi b_k \le 0$, $\dot{V}_b \le 0$ thus it can be guaranteed that $\|w_k\| \le \|w_k(0)\|$ when $(\|w_k\| = \|w_k(0)\|)$ and $w_k^{\mathrm{T}}\psi b_k > 0$, $\dot{V}_b = 0$. Thus $\|w_k\| < \|w_k(0)\|$ is also guaranteed. W is bounded by constraint set Γ for all $t \ge 0$.

Theorem 2. Suppose that the expected trajectory of AUV is bounded; the unknown external force disturbance τ_d and the approaching error of neural network ε are both zero. If the input to the controller is:

$$\tau = \hat{f}(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) + J^{\mathrm{T}} K_d \boldsymbol{s} \quad (33)$$

And the adaptive law of neural network is given by (30), and then the tracking error $\boldsymbol{s}(t)$ will approach to zero.

Proof: Now that the assumption of the conditions are satisfied and the control law (33) is adopted, thus the error dynamic equation of closed loop system will be:

$$M_\eta \dot{\boldsymbol{s}} = -(D_\eta + K_d)\boldsymbol{s} - C_\eta \boldsymbol{s} + J^{-\mathrm{T}}\tilde{W}^{\mathrm{T}}\psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) \quad (34)$$

We consider the following Lyapunov function

$$V(t) = \frac{1}{2}\boldsymbol{s}^{\mathrm{T}} M_\eta \boldsymbol{s} + \frac{1}{2} k^{-1} \mathrm{tr}\{\tilde{W}^{\mathrm{T}} \tilde{W}\} \quad (35)$$

Taking the derivative to the above Lyapunov function gives:

$$\dot{V}(t) = \frac{1}{2}(\dot{\boldsymbol{s}}^{\mathrm{T}} M_\eta \boldsymbol{s} + \boldsymbol{s}^{\mathrm{T}} M_\eta \dot{\boldsymbol{s}} + \boldsymbol{s}^{\mathrm{T}} \dot{M}_\eta \boldsymbol{s}) + k^{-1}\mathrm{tr}\{\tilde{W}^{\mathrm{T}} \dot{\tilde{W}}\} \quad (36)$$

Substituting $\boldsymbol{s}^{\mathrm{T}}(\dot{M}_\eta - 2C_\eta)\boldsymbol{s} = 0$ and $\dot{\boldsymbol{s}}^{\mathrm{T}} M_\eta \boldsymbol{s} = \boldsymbol{s}^{\mathrm{T}} M_\eta \dot{\boldsymbol{s}}$ into (36) yields:

$$\begin{aligned} \dot{V}(t) &= \boldsymbol{s}^{\mathrm{T}}(M_\eta \dot{\boldsymbol{s}} + C_\eta \boldsymbol{s}) + k^{-1}\mathrm{tr}\{\tilde{W}^{\mathrm{T}} \dot{\tilde{W}}\} \\ &= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} + (J^{-1}\boldsymbol{s})^{\mathrm{T}} \tilde{W}^{\mathrm{T}}\psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta) \\ &\quad + k^{-1}\mathrm{tr}\{\tilde{W}^{\mathrm{T}} \dot{\tilde{W}}\} \\ &\le -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} \\ &\quad + \mathrm{tr}\{\tilde{W}^{\mathrm{T}}[\psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta)(J^{-1}\boldsymbol{s})^{\mathrm{T}} + k^{-1}\dot{\tilde{W}}]\} \end{aligned} \quad (37)$$

Under the condition 1 of (30), (37) becomes

$$\dot{V}(t) \le -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} + \mathrm{tr}\{\tilde{W}^{\mathrm{T}}[\psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta)(J^{-1}\boldsymbol{s})^{\mathrm{T}}$$
$$+ k^{-1}\dot{\tilde{W}}]\} = -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s}$$
$$+ \mathrm{tr}\{\tilde{W}^{\mathrm{T}}[(\psi\, b_1, \psi\, b_2, \cdots, \psi\, b_{N_0})$$
$$- k^{-1}(k\psi b_1, k\psi b_2, \cdots, k\psi b_{N_0})]\}$$
$$= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} \le 0 \tag{38}$$

Under the condition 2 of (30), (37) becomes

$$\dot{V}(t) \le -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s}$$
$$+ \mathrm{tr}\{\tilde{W}^{\mathrm{T}}[\psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta)(J^{-1}\boldsymbol{s})^{\mathrm{T}} + k^{-1}\dot{\tilde{W}}]\}$$
$$= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} + \mathrm{tr}\{\tilde{W}^{\mathrm{T}}[(\psi\, b_1, \cdots, \psi\, b_{N_0})$$
$$- k^{-1}(k\psi b_1, k\psi b_2, \cdots, k\psi b_{N_0})$$
$$+ (\frac{w_1 w_1^{\mathrm{T}}}{\|w_1\|}\psi\, b_1, \mathrm{L}\ , \frac{w_{N_0} w_{N_0}^{\mathrm{T}}}{\|w_{N_0}\|}\psi\, b_{N_0})]\}$$
$$= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s}$$
$$+ \mathrm{tr}\{\tilde{W}^{\mathrm{T}}[\frac{w_1 w_1^{\mathrm{T}}}{\|w_1\|^2}\psi\, b_1, \mathrm{L}\ , \frac{w_{N_0} w_{N_0}^{\mathrm{T}}}{\|w_{N_0}\|^2}\psi\, b_{N_0}]\}$$
$$= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} + \sum_{k=1}^{N_0}[(w_k^* - w_k)\frac{w_k w_k^{\mathrm{T}}}{\|w_k\|^2}\psi\, b_k]$$
$$= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} + \sum_{k=1}^{N_0}(\frac{(w_k^{*\mathrm{T}})\, w_k}{\|w_k\|^2} - 1)w_k^{\mathrm{T}}\psi\, b_k$$
$$= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} \le 0 \tag{39}$$

Theorem 3. Suppose that the expected trajectory of AUV is bounded; the unknown external force disturbance τ_d and the approaching error of neural network ε are both zero. If the input to the controller is (21), the adaptive law of neural network is (30), and robust controller is defined as:

$$\alpha = (\varepsilon_N + d_B)\frac{J^{-1}\boldsymbol{s}}{\|J^{-1}\boldsymbol{s}\|} \tag{40}$$

Then the tracking error $\boldsymbol{s}(t)$ will approach to zero.
Proof: We consider the Lyapunov function (35), thus

$$\dot{V}(t) = \boldsymbol{s}^{\mathrm{T}}(M_\eta \dot{\boldsymbol{s}} + C_\eta \boldsymbol{s}) + \mathrm{tr}\{\tilde{W}^{\mathrm{T}}\Gamma^{-1}\dot{\tilde{W}}\}$$
$$= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} + (J^{-1}\boldsymbol{s})^{\mathrm{T}}[\tilde{W}^{\mathrm{T}}\psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta)$$
$$+ \varepsilon + \tau_d - \alpha] + \mathrm{tr}\{\tilde{W}^{\mathrm{T}}\Gamma^{-1}\dot{\tilde{W}}\}$$
$$\le -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s}$$
$$+ \mathrm{tr}\{\tilde{W}^{\mathrm{T}}[\psi(\dot{\boldsymbol{v}}_r, \boldsymbol{v}_r, \boldsymbol{v}, \eta)(J^{-1}\boldsymbol{s})^{\mathrm{T}} + \Gamma^{-1}\dot{\tilde{W}}]\}$$
$$+ (J^{-1}\boldsymbol{s}^{\mathrm{T}})[\varepsilon + \tau_d - \alpha]$$
$$\le -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} + (J^{-1}\boldsymbol{s})^{\mathrm{T}}[\varepsilon + \tau_d - \alpha]$$
$$= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} + (J^{-1}\boldsymbol{s})^{\mathrm{T}}[\varepsilon + \tau_d]$$
$$- (J^{-1}\boldsymbol{s})^{\mathrm{T}}(\varepsilon_N + d_B)\frac{J^{-1}\boldsymbol{s}}{\|J^{-1}\boldsymbol{s}\|}$$
$$= -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} + (J^{-1}\boldsymbol{s})^{\mathrm{T}}[\varepsilon + \tau_d]$$
$$- \|J^{-1}\boldsymbol{s}\|(\varepsilon_N + d_B) \le -\boldsymbol{s}^{\mathrm{T}}(D_\eta + K_d)\boldsymbol{s} \le 0 \tag{41}$$

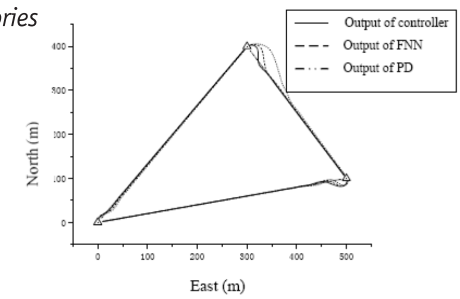
6. Simulation results

The simulation could be done by substituting some parameters into the 6 DOF dynamic AUV model (8)-(10). The goal of the simulation is to verify the validity of the proposed adaptive control based on fuzzy neural network by having AUV track a series of predetermined trajectories under the disturbance of the ocean current and comparing the proposed control with other control methods.

The current velocity function is defined as $0.5 + 0.1\sin(0.01t)$ (knot); the current direction function is defined as $50 + 10\sin(0.01t)$ (degree). The key parameters in the controller are assumed as following: The width of Gaussian function is determined as 0.1; k = 1000; k_e = 100; k_{md} = 0.083; λ = diag (0.1 0.1 0.1 0.1 0.1 0.1); k_d = diag (2×10^4 2×10^4 2×10^4 3×10^5 3×10^5 3×10^5).

In the simulation experiments, AUV is required to track a user-planned trajectory at constant speed 4kn. After finishing the simulation experiment, there are inference rules that have been generated. The system takes about 300s-400s to learn the new rules. From the errors analysis in table 1, Fig. 2(b), Fig. 3(b), Fig. 4(b) and Fig. 5(d), we can see at the beginning of the simulation, the results of PD and the adaptive controller („output of controller" in the following plots) are similar. While after the learning process being accomplished, the output of the adaptive controller and the output of fuzzy neural network are very close, and are superior to PD controller output. From the effects of trajectory tracking in Fig. 2(a), Fig. 3(a), Fig. 4(a) and Fig. 5(a)-(c), we can conclude that our proposed adaptive controller is advanced in its self constructing and self learning. The controller is also robust to the disturbance of complex ocean current.

(a) Trajectories



(b) Tracking Errors

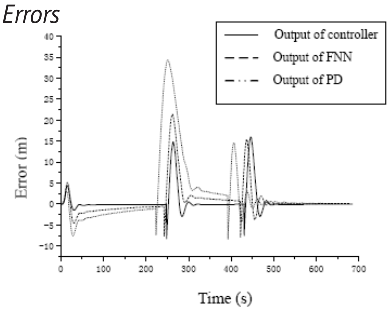
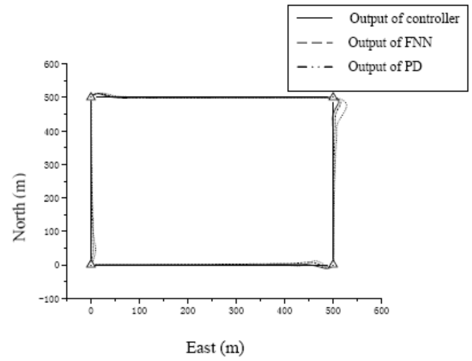


Fig. 2. Tracking a triangle trajectory.

(a) Trajectories



(b) Tracking Errors

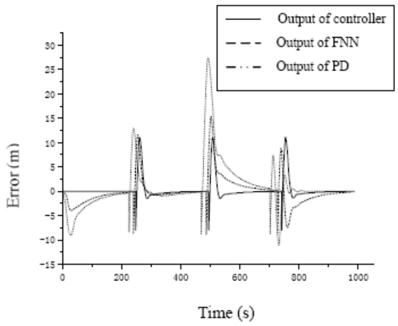
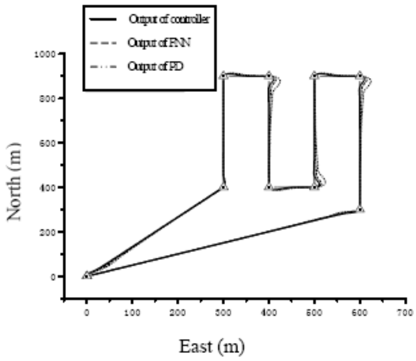


Fig. 3. Tracking a rectangular trajectory.

(a) Trajectories



(b) Tracking Errors

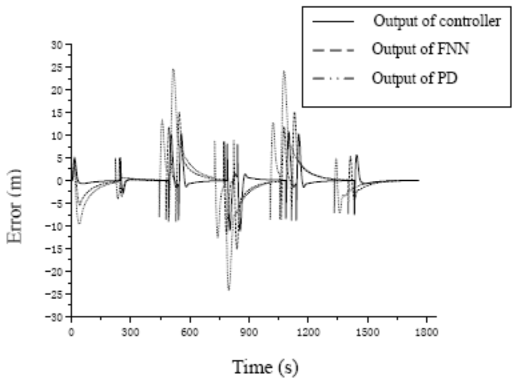
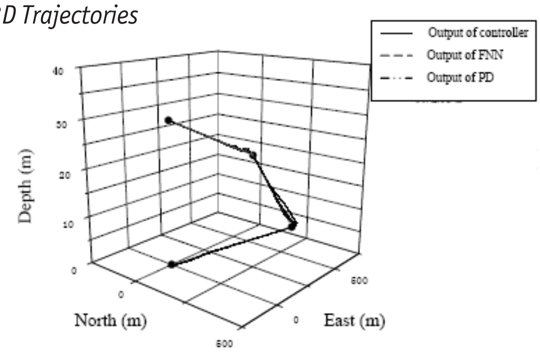
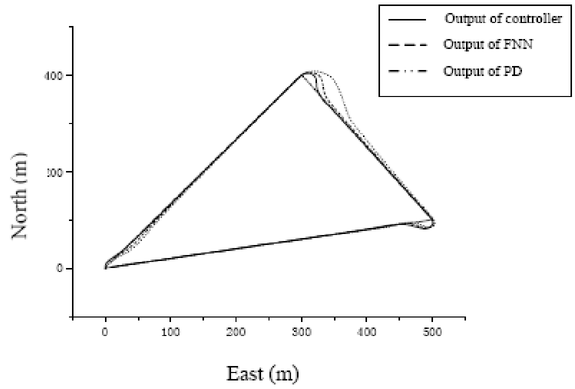


Fig. 4. Tracking a comb-shape trajectory.

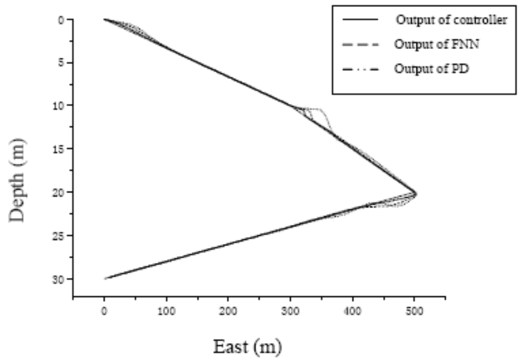
(a) 3D Trajectories



(b) Trajectories projection on Northeast plane



(c) Trajectories projection on Depth-east plane



(d) Tracking Errors

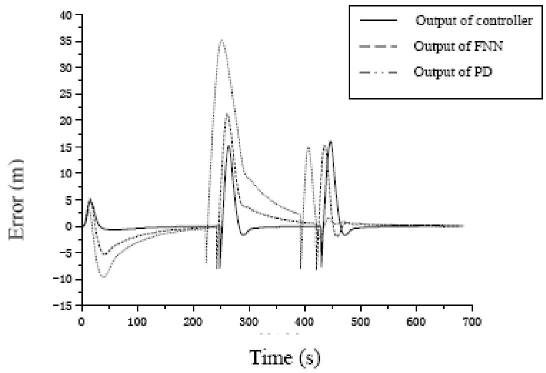


Fig. 5. 3D trajectory tracking.

Table 1. Performance comparison.

Simulation	*Root mean square error (m)*		*Maximum error (m)*		*Minimum error (m)*	
	PD	*Adaptive*	*PD*	*Adaptive*	*PD*	*Adaptive*
Triangle trajectory	*16.11*	*5.44*	*37.13*	*16.26*	*0.55*	*0.06*
rectangular trajectory	*10.47*	*2.26*	*27.61*	*13.75*	*0.51*	*0.02*
Comb-shape trajectory	*6.25*	*1.31*	*25.45*	*10.56*	*0.42*	*0.03*
3D trajectory	*17.63*	*4.87*	*36.32*	*15.71*	*0.53*	*0.05*

7. Conclusion

An adaptive controller based on Fuzzy Neural Network is proposed and its stability for the closed loop control system is proved theoretically. The stability and robustness of the controller are also verified by simulation experiments in which AUV is required to track trajectories with current disturbance. Through the theoretical analysis and simulation verification, we can conclude:

The on-line learning algorithm of the fuzzy neural network is advanced in its self-constructing and self-learning properties, so that the complex unknown model can be approached precisely based on this kind-learning algorithm.

The fuzzy neural network is also able to learn the disturbance to AUV under complex underwater environment, so that the disturbing effects to AUV motion control would decrease efficiently.

The adaptive controller based on fuzzy neural network can be applied in trajectory tracking control for AUV. The system tracking error decreases as the on-line learning processes.

AUTHORS
Zheng Qin* - Department of Electrical and Computer Engineering, Dalhousie University, Halifax, B3J 2X4, Canada. E-mail: Zheng.Qin@dal.ca.
Jason Gu - Department of Electrical and Computer Engineering, Dalhousie University, Halifax, B3J 2X4, Canada. E-mail: Jason.Gu@dal.ca.
* Corresponding author

References

[1] Song X., Ye J., Wu L., „Integral sliding mode controller based on fuzzy logic for the heading control of the submersible vehicle", *ICICIC '06. First International Conference on Innovative Computing, Information and Control*, 2006, pp. 183-186.
[2] Tangirala S., Dzielski J., „A variable buoyancy control system for a large AUV", *IEEE Journal of Oceanic Engineering*, vol. 32, no. 4, 2007, pp. 762-771.
[3] FossenT.I., Blanke M., „Nonlinear output feedback control of AUV propellers using feedback form estimated axial flow velocity", *IEEE Journal of Oceanic Engineering*, vol. 25, no. 4, 2000, pp. 241-255.
[4] HealeyA.J., Lienard D., „Multivariable sliding mode control for autonomous diving and steering of unmanned AUVs", *IEEE Journal of Oceanic Engineering*, vol. 18, no. 7, 1993, pp. 327-339.
[5] Yoerger D.N., Slotine J.E., „Robust trajectory control of AUVs", *IEEE Journal of Oceanic Engineering*, vol. 10, no.9, 1985, pp. 462-470.
[6] Choi S.K., Yuh J., „Experimental study on a learning control system with bound estimation for AUVs", *Underwater Robots*, Norwell, MA: Kluwer, 1996, pp. 113-120.
[7] Ishii K., Fujii T., Ura T., „An on-line adaptation method in a neural network-based control system for AUVs", *IEEE Journal of Oceanic Engineering*, vol. 20, no. 7, 1995, pp. 221-228.
[8] Jordan M.A., Bustamante J.L., „Guidance of AUVs with cable tug perturbations under fixed and adaptive control systems", *IEEE Journal of Oceanic Engineering*, vol. 33, no. 4, 2008, pp. 579-598.
[9] Cavalletti M., Ippoliti G., Longhi S., „Switching adaptive control for a remotely operated vehicle", *American Control Conference*, 2007, pp. 2799-2806.
[10] Kim H., Shin Y., „Design of adaptive fuzzy sliding mode controller using FBFE for UFV depth control", *SICE-ICASE*, 2006, pp. 3100-3103.
[11] Kim H., Shin Y., „Design of expanded adaptive fuzzy sliding mode controller using FBFE for UFV depth control", *International Conference on Automation and Systems*, 2007, pp. 2654-2657.
[12] Moattari M., Khayatian A., „A neural network controller for diving of a variable mass autonomous underwater vehicle", *2008 SICE Annual Conference*, 2008, pp. 1339-1344.
[13] Xu J., Zhang M., „Neural network modeling and generalized predictive control for an autonomous underwater vehicle", *6th IEEE International Conference on Industrial Informatics*, 2008 pp. 487-491.
[14] Jang J.S.R., „ANFIS: adaptive-network-based fuzzy inference system". *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, no. 3, 1993, pp. 665-684.
[15] Cho K.B., Wang B.H., „Radial basis function based adaptive fuzzy systems and their applications to system identification and prediction", *Fuzzy Sets Systems*, vol. 83, no. 3, 1996, pp. 325-339.
[16] Juang C.F., Lin C.T., „An online self-constructing neural fuzzy inference network and its applications", *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 1, 1998, pp. 12-32.
[17] Juang C.F., Lin C.T., „A recurrent self- organizing neural fuzzy inference network", *IEEE Transactions on Neural Networks*, vol. 10, no. 4, 1999, pp. 828-845.
[18] Er M.J., Gao Y., „Robust adaptive control of robot manipulators using generalized fuzzy neural networks", *IEEE Transactions on Industrial Electronics*, vol. 48, no. 6, 2001, pp. 1274-1278.

[19] Wang J.S., Lee C.S.G., „Self-adaptive recurrent neuro-fuzzy control of an autonomous underwater vehicle”, *IEEE Transactions on Robotics and Automation*, vol. 11, no. 4, 2003, pp. 462-477.

[20] Wang J.S., Lee C.S.G., „Structure and learning in self-adaptive neural fuzzy inference systems”, *International Journal of Fuzzy Systems*, vol. 2, no. 1 2000, pp. 12-22.

[21] Er M.J., Gao Y., „Robust adaptive control of robot manipulators using generalized fuzzy neural networks”, *IEEE Transactions on Industrial Electronics*, vol. 50, no. 3, 2003, pp. 620-628.

[22] Gao Y., Er M.J., „Online adaptive fuzzy neural identification and control of a class of MIMO nonlinear systems”, *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 4, 2003, pp. 462-477.

[23] Yuh J., „Underwater Robotic Vehicles: Design and Control?”, *USA: TSI Press Series Networks*, 1995, pp. 15-40.