Supervisory Recurrent Fuzzy Neural Network Guidance Law Design for Autonomous Underwater Vehicle

Yi-Jen Mon and Chih-Min Lin

Abstract

A guidance law, based on supervisory recurrent fuzzy neural network control (SRFNNC), is proposed for the autonomous underwater vehicle (AUV) guidance systems. This SRFNNC system is comprised of a recurrent fuzzy neural network (RFNN) controller and a supervisory controller. The RFNN controller is used to mimic an ideal controller and the supervisory controller is designed to compensate for the approximation error between the RFNN controller and the ideal controller. The proposed design method is applied to investigate the active acoustic homing guidance of an AUV, which is affected by sonar propagation time-delay and measurement noise. A comparison is made for the proposed SRFNNC, a proportional navigation (PN) and a sliding-mode control (SMC) guidance laws. Simulation results show that the proposed SRFNNC guidance law is more robust and can obtain smaller miss distance than the PN and SMC guidance laws.

Keywords: Autonomous underwater vehicle (AUV), Guidance law, Sliding-mode control, fuzzy neural network.

Nomenclature

\[ OXYZ = \text{inertial frame} \]
\[ CX'Y'Z' = \text{local frame} \]
\[ N = \text{navigation ratio} \]
\[ q(t) = \text{line-of-sight (LOS) angle} \]
\[ \hat{q}(t) = \text{estimated LOS angle} \]
\[ \Delta q(t) = \text{deviation of LOS angle} \]
\[ r(t) = \text{measured AUV-to-target range} \]
\[ r_{Y}(t) = \text{component of } r(t) \text{ in } Y'\text{-axis} \]
\[ u = \text{AUV velocity} \]

1. Introduction

An autonomous underwater vehicle (AUV) is a self-propelled guided projectile that operates underwater and is designed to detonate on contact or in proximity to a target. AUV may be guided by wire or passive/active acoustic homing [1, 2]. For the missile guidance systems, because the target measurement is by electric waves, the time-delay problem is negligible. However, in AUV guidance the target measurement is by acoustic homing, so the measurement time-delay from sound velocity in water becomes a significant problem for AUV guidance [3, 4]. Carof has proposed an acoustic positioning and guidance technique based upon differential delay and Doppler tracking system [5]. He used two extended transmitters to obtain the differential delay and used Doppler to estimate the current target position, but the guidance system has not been discussed. Till now, there are few publications for AUV guidance system design. Of these, Hutchins and Roque have proposed a linear-quadratic-regulator (LQR) AUV guidance law [6]. However, in their designs, there is no discussion about the transmission time delays due to acoustic homing. Nakamura and Savant have proposed a nonlinear control [7]. Bessa et al. have proposed an adaptive fuzzy sliding mode control [8].

Recently, neural network based control techniques have presented an alternative design method for the control of dynamic systems [9-10]. The most useful property of neural networks is their ability to approximate linear or nonlinear mapping through learning. Based on this property, neural network based controllers have been

\[ V_{Y}(t) = \text{speed of target in } Y'\text{-axis} \]
\[ V_{I}(t) = \text{speed of AUV in } Y'\text{-axis} \]
\[ w_{x} = \text{angular rates in the local frame corresponding to the roll rate} \]
\[ w_{y} = \text{angular rates in the local frame corresponding to the pitch rate} \]
\[ w_{z} = \text{angular rates in the local frame corresponding to the yaw rate} \]
\[ \phi = \text{rolling angle} \]
\[ \theta = \text{pitch angle} \]
\[ \psi = \text{yaw angle} \]
developed to compensate for the effects of nonlinearities and system uncertainties; thus, improving the stability, convergence and robustness of the control system. Recently, the concept of incorporating fuzzy logic into a neural network has become a popular research topic [11-13]. Fuzzy neural networks have advantages both of fuzzy systems and of neural networks since they combine the capabilities of fuzzy reasoning and neural network on-line learning. However, the neural networks presented in [9-13] are feedforward type, belonging to static mapping networks. Recently, the recurrent neural network has been extensively presented since it has capabilities superior to the feedforward neural network, such as the dynamic response and the information storing ability [14-15]. Since a recurrent neural network has an internal feedback loop, it captures the dynamic response of system with external feedback through delays. Thus, the recurrent neural network is a dynamic mapping network.

This paper presents a supervisory recurrent fuzzy neural network control (SRFNNC) guidance law for an AUV guidance system. This SRFNNC system is comprised of a recurrent fuzzy neural network (RFNN) controller and a supervisory controller. The RFNN controller is used to mimic an ideal controller, and the supervisory controller is designed to compensate for the approximation error between the RFNN controller and the ideal controller. The RFNN is inherently a recurrent multilayered neural network for realizing fuzzy inference using dynamic fuzzy rules. Temporal relations are embedded in the network by adding feedback connections in the second layer of the fuzzy neural network. Moreover, an on-line parameter training methodology, using the gradient descent method and the Lyapunov stability theorem, is proposed to increase the learning ability. In addition, to relax the requirement for the uncertain bound in the supervisory controller, an estimation mechanism is investigated to observe the uncertain bound. Thus the chattering phenomena of the control efforts can be relaxed. Finally, a comparison for the AUV guidance system among PN, SMC and the proposed SRFNNC guidance laws is presented to illustrate the effectiveness of the proposed design method.

2. Formulation of AUV-target Engagement

Figure 1 shows the three-dimensional AUV-target engagement diagram, with two reference frames, of which the $CX'Y'Z'$ frame is the local frame and the $OXYZ$ frame is the inertial frame. During the interception process, the AUV’s roll angle is assumed to be controlled to zero by the autopilot control system, so that the control of elevation loop and azimuth loop for the guidance system can be separately designed. In the following discussion, the azimuth loop is illustrated, and the result is similar for the elevation loop.

Figure 1. Three-dimensional AUV-target engagement diagram.

Figure 2 shows the two-dimensional AUV-target engagement diagram in the final homing phase. The measured line-of-sight (LOS) angle $q(t)$ is assumed to be a small angle in the final homing phase, and it is represented as the azimuth LOS angle; $r(t)$ represents the measured AUV-to-target range; and $r_Y(t)$ represents the component of $r(t)$ in $Y'$-axis.

From the LOS coordinate frame, the AUV-target relative velocity in $Y'$-axis can be represented as

$$\dot{r_Y}(t) = V(t) - V_Y(t), \quad (1)$$

where $V(t)$ and $V_Y(t)$ represent the speed of target and AUV in $Y'$-axis, respectively.

Based on polar geometry, the following equation can be established

$$\frac{r(t)}{r_Y(t)} = \sin(q(t)). \quad (2)$$

where $q(t)$ is the measured LOS angle.

Differentiating (2) with respect to time produces

$$\dot{q}(t) = \frac{\dot{r_Y}(t) - \dot{r}(t)q(t)}{r(t)}. \quad (3)$$

From (1) and (3), it is obtained that

$$\dot{q}(t) = \frac{-\dot{r}(t)}{r(t)}q(t) + \frac{1}{r(t)}(V(t) - V_Y(t)) \quad (4)$$

which is the AUV-target engagement equation in a system without time-delay. However, considering the sonar propagation time-delay problem, the $q(t)$ is not the actual azimuth LOS angle at the sensing time. Thus, a deviation of azimuth LOS angle $\Delta q(t)$ is induced. The actual azimuth LOS angle is defined as

$$\dot{q}(t) = q(t) + \Delta q(t). \quad (5)$$

Considering the time-delay problem, (4) should be re-
placed by
\[ \dot{q}(t) = -\frac{\dot{r}(t)}{r(t)} \dot{q}(t) + \frac{1}{r(t)} \left( V'_i(t) - V'_j(t) \right). \] (6)
that is
\[ \dot{q}(t) = -\frac{\dot{r}(t)}{r(t)} q(t) + \frac{1}{r(t)} \left( V_i(t) - V_j(t) - \Delta q(t) - \frac{\dot{r}(t)}{r(t)} \Delta q(t) \right) \]
\[ = -\frac{\dot{r}(t)}{r(t)} q(t) + \frac{1}{r(t)} \left( V_i(t) - V_j(t) - g(\Delta q(t), \Delta \dot{q}(t)) \right). \] (7)
where \( g(\Delta q(t), \Delta \dot{q}(t)) = \Delta q(t) + \frac{\dot{r}(t)}{r(t)} \Delta q(t) \) denotes the measurement uncertainty coming from time delay.

The control input of (7) is \( V_i(t) \); however, (7) is a time-varying system with unknown \( V_i(t) \), \( \Delta q(t) \) and \( \Delta \dot{q}(t) \). For this time-delay system, a SMC algorithm and a SRFNNC algorithm will be developed to handle the time-varying system in order to cope with the unknown target velocity \( V_i(t) \) and the time-delay angular deviations \( \Delta q(t) \) and \( \Delta \dot{q}(t) \), so that the LOS angle can be controlled to zero.

\[ V_e(q, \dot{q}, \Delta q(t), \Delta \dot{q}(t)) = V_e(q(t)) + V_h(t) \]

where \( V_e(q(t)) \) is an equivalent target velocity, \( V_h(t) \) is a hitting controller designed to disspel the system uncertainties as
\[ V_h(t) = Fr(t) \text{sgn}(s(t)) \]
in which \( F \geq \frac{V_i(t)}{r(t)} - g(\Delta q(t), \Delta \dot{q}(t)) \) is a positive constant and \( \text{sgn}(\cdot) \) is a sign function. From (9), (11) and (12), it is obtained that
\[ \dot{s}(t) = \dot{q}(t) + k_1 \dot{q}(t) + k_2 q(t) \]
\[ = \dot{q}(t) + k_1 \left[ -\frac{\dot{r}(t)}{r(t)} q(t) + \frac{1}{r(t)} \left( V_i(t) - (V_e(t) + V_h(t)) \right) - g(\Delta q(t), \Delta \dot{q}(t)) \right] + k_2 q(t) \]
\[ = k_1 \left[ V'_i(t) - g(\Delta q(t), \Delta \dot{q}(t)) - F \text{sgn}(s(t)) \right]. \] (13)

Choose a Lyapunov candidate as
\[ V_1 = \frac{1}{2} s^2(t). \] (14)

Differentiating (14) with respect to time and using (13), it is obtained that
\[ V'_1 = s(t) s(t) = k_1 s(t) \left[ \frac{V'_i(t)}{r(t)} - g(\Delta q(t), \Delta \dot{q}(t)) - F \text{sgn}(s(t)) \right] \]
\[ \leq k_1 \left[ s(t) \left| \frac{V'_i(t)}{r(t)} - g(\Delta q(t), \Delta \dot{q}(t)) - F \text{sgn}(s(t)) \right| \right] \]
\[ \leq 0. \] (15)

In summary, the sliding-mode control system presented in (10) can guarantee asymptotic stability in the sense of the Lyapunov theorem, even under system uncertainties.

4. Supervisory Recurrent Fuzzy Neural Network Control Guidance Law

In the following, a new guidance law is proposed. If the system uncertainties are well known and measurable, the following ideal controller can be obtained from (7)
\[ V'_i(t) = V'_i(t) - k_1 q(t) + k_2 \int_0^t q(\tau) d\tau \]

Figure 2. Two-dimensional AUV-target engagement LOS coordinate frame diagram.

3. Guidance Law Design

A. Proportional Navigation Guidance Law

For the guidance systems, the most frequently used is the proportional navigation (PN) guidance law. It can be represented as
\[ V_i = N r(t) \dot{q}(t) \] (8)
where \( N \) is the navigation ratio.

B. Sliding Mode Control Guidance Law

The main advantage of SMC is that the system’s uncertainties and external disturbances can be handled under the invariance characteristics of system’s sliding condition. Thus, SMC is used as one of the AUV guidance law. The first step of SMC design is to select a sliding surface that models the desired closed-loop performance in phase-plane space. Then the control law is designed such that the system state trajectories are forced toward the sliding surface and then stay on it. Suppose that an integrated sliding function is defined as
\[ s(t) = \dot{q}(t) + k_1 q + k_2 \int_0^t q(\tau) d\tau. \] (9)
where \( k_1 \) and \( k_2 \) are positive constants. Then the sliding-mode control law can be defined as [16]
\[ V'_i(t) = V'_e(t) + V'_h(t) \] (10)
where \( V'_e(t) \) is an equivalent controller represented as
\[ V'_e(t) = -\dot{r}(t) q(t) + r(t) \left[ \frac{1}{k_1} \dot{q}(t) + \frac{k_2}{k_1} q(t) \right] \] (11)
and \( V'_h(t) \) is a hitting controller designed to disspel the system uncertainties as
\[ V'_h(t) = Fr(t) \text{sgn}(s(t)) \] (12)
in which \( F \geq \frac{V'_i(t)}{r(t)} - g(\Delta q(t), \Delta \dot{q}(t)) \) is a positive constant and \( \text{sgn}(\cdot) \) is a sign function. From (9), (11) and (12), it is obtained that
\[ \dot{s}(t) = \dot{q}(t) + k_1 \dot{q}(t) + k_2 q(t) \]
\[ = \dot{q}(t) + k_1 \left[ -\frac{\dot{r}(t)}{r(t)} q(t) + \frac{1}{r(t)} \left( V_i(t) - (V_e(t) + V_h(t)) \right) - g(\Delta q(t), \Delta \dot{q}(t)) \right] + k_2 q(t) \]
\[ = k_1 \left[ V'_i(t) - g(\Delta q(t), \Delta \dot{q}(t)) - F \text{sgn}(s(t)) \right]. \] (13)

Choose a Lyapunov candidate as
\[ V_1 = \frac{1}{2} s^2(t). \] (14)

Differentiating (14) with respect to time and using (13), it is obtained that
\[ V'_1 = s(t) s(t) = k_1 s(t) \left[ \frac{V'_i(t)}{r(t)} - g(\Delta q(t), \Delta \dot{q}(t)) - F \text{sgn}(s(t)) \right] \]
\[ \leq k_1 \left[ s(t) \left| \frac{V'_i(t)}{r(t)} - g(\Delta q(t), \Delta \dot{q}(t)) - F \text{sgn}(s(t)) \right| \right] \]
\[ \leq 0. \] (15)

In summary, the sliding-mode control system presented in (10) can guarantee asymptotic stability in the sense of the Lyapunov theorem, even under system uncertainties.
Substituting (16) into (7) gives
\[ \ddot{q}(t) + k_1 \dot{q}(t) + k_2 q(t) = 0. \]  
(17)

If \( k_1 \) and \( k_2 \) are chosen to correspond to the coefficients of a Hurwitz polynomial, that is, a polynomial whose roots lie strictly in the open left half of the complex plane, then \( \lim_{t \to \infty} q(t) = 0 \). Since the system parameters may be unknown or perturbed, the ideal controller \( V_{id} \) cannot be implemented. To overcome this, a recurrent fuzzy neural network (RFNN) controller will be designed to approximate this ideal controller. In addition, a supervisory controller is designed to compensate for the approximation error between the RFNN controller and the ideal controller. Thus, the block diagram of the supervisory fuzzy neural network control (SRFNNC) system is shown in Fig. 3, where the inputs of the RFNN controller are \( s(t) \) and \( \dot{s}(t) \). The SRFNNC is assumed to take the following form:

\[ V_r(t) = V_{srn}(t) = V_{rn}(t) + V_{sp}(t) \]  
(18)

where \( V_{rn}(t) \) is the RFNN controller and \( V_{sp}(t) \) is the supervisory controller.

A. Recurrent Fuzzy Neural Network Controller

A four-layer neural network shown in Fig. 4, which is comprised of the input (the \( i \) layer), membership (the \( j \) layer), rule (the \( k \) layer), and output (the \( o \) layer) layers, is used. The recurrent feedback is embedded in the network by adding feedback connections in the second layer of the fuzzy neural network. Since the recurrent neuron has an internal feedback loop, it captures the dynamic mapping network. The signal propagation and the basic function in each layer are introduced as follows:

\[ + r(t)\left( \frac{1}{k_1} \ddot{q}(t) + \frac{k_2}{k_1} q(t) \right) \]  
(16)
this layer is a labeled as $\Sigma$, which computes the overall output as the summation of all incoming signals

$$net^4_o(N) = \sum_k w^4_k x^4_k$$  \hspace{1cm} (25)

$$y^4_o(N) = f^4_o\left(net^4_o(N)\right) = net^4_o(N), \quad o = 1$$  \hspace{1cm} (26)

where the link weight $w^4_k$ is the output action strength of the $o$th output associated with the $k$th rule, $x^4_k$ represents the $k$th input to the node of layer 4, and $y^4_o$ is the output of the recurrent fuzzy neural network controller.

B. On-Line Learning Algorithm

The on-line learning algorithm is a gradient descent algorithm in the space of network parameters and aims to minimize $s(t)\hat{s}(t)$. Therefore, $s(t)\hat{s}(t)$ is selected as the error function. Taking the first derivative of $s(t)$ and using (7), it can be obtained that

$$\dot{s}(t) = \dot{\hat{s}}(t) + k_1 \dot{q}(t) + k_2 q(t)$$

$$= k_1 \left[ -\frac{\dot{r}(t)}{r(t)} q(t) + \frac{1}{r(t)} (V_i(t) - V_r(t)) + V_{sp}(t) \right] - g(\Delta q, \Delta \dot{q}(t)) + A_d(\hat{q}, q)$$  \hspace{1cm} (27)

where $A_d(\hat{q}, q) = \frac{1}{k_1} \hat{q}(t) + k_2 q(t)$. Substituting (18) into (27) and multiplying both sides by $s(t)$, it is obtained that

$$s(t)\dot{s}(t) = s(t)k_1 \left[ -\frac{\dot{r}(t)}{r(t)} q(t) + \frac{1}{r(t)} (V_i(t) - V_r(t)) + V_{sp}(t) \right] - g(\Delta q, \Delta \dot{q}(t)) + A_d(\hat{q}, q) \right].$$  \hspace{1cm} (28)

According to the gradient descent method, the weights in the output layer are updated by the following:

$$\dot{w}^4_k = -\eta_w \frac{\partial s(t)\dot{s}(t)}{\partial w^4_k} = -\eta_w \frac{\partial s(t)\dot{s}(t)}{\partial V_{net}} \frac{\partial V_{net}}{\partial w^4_k}$$

$$= -\eta_w \frac{\partial s(t)\dot{s}(t)}{\partial V_r} \frac{\partial V_r}{\partial w^4_k}$$

$$= \eta_w k_1 \frac{s(t)}{r(t)} x^4_k = \eta_w \frac{s(t)}{r(t)} x^4_k$$  \hspace{1cm} (29)

where $\eta_w$ is the learning rate with a positive constant and $\eta_w' \equiv \eta_w k_1$. Since the weights in the rule layer are unity, only the approximation error term needs to be calculated and propagated by the following:

$$\dot{\delta}^3_k = -\frac{\partial s(t)\dot{s}(t)}{\partial V_{net}} \frac{\partial V_{net}}{\partial \delta^3_k} \frac{s(t)}{r(t)} x^4_k = -\frac{s(t)}{r(t)} x^4_k$$  \hspace{1cm} (30)

The multiplication is done in the membership layer and the error term is computed as follows:

$$\delta^2_j = -\frac{\partial s(t)\dot{s}(t)}{\partial V_{net}} \frac{\partial V_{net}}{\partial \delta^2_j} \frac{s(t)}{r(t)} \frac{\partial V_{net}}{\partial \delta^2_j} \frac{s(t)}{r(t)} \frac{\partial V_{net}}{\partial \delta^2_j}$$

$$= \sum_k \delta^3_k \delta^3_k$$  \hspace{1cm} (31)

The update laws of $m^2_j$, $\sigma^2_{ij}$ and $\theta^2_{ij}$ also can be obtained by the gradient search algorithm, i.e.,

$$\dot{m}^2_j = -\eta_m \frac{\partial s(t)\dot{s}(t)}{\partial m^2_j} = -\eta_m \frac{\partial s(t)\dot{s}(t)}{\partial V_{net}} \frac{\partial V_{net}}{\partial m^2_j}$$

$$= -\eta_m \frac{2(x^2_j + y^2_j(N-1)\theta^2_{ij} - m^2_j)^2}{(\sigma^2_j)^2}$$  \hspace{1cm} (32)

$$\dot{\sigma}^2_j = -\eta_\sigma \frac{\partial s(t)\dot{s}(t)}{\partial \sigma^2_j} = -\eta_\sigma \frac{\partial s(t)\dot{s}(t)}{\partial V_{net}} \frac{\partial V_{net}}{\partial \sigma^2_j}$$

$$= -\eta_\sigma \frac{2(x^2_j + y^2_j(N-1)\theta^2_{ij} - m^2_j)^2}{(\sigma^2_j)^2}$$  \hspace{1cm} (33)

$$\dot{\theta}^2_{ij} = -\eta_\theta \frac{\partial s(t)\dot{s}(t)}{\partial \theta^2_{ij}} = -\eta_\theta \frac{\partial s(t)\dot{s}(t)}{\partial V_{net}} \frac{\partial V_{net}}{\partial \theta^2_{ij}}$$

$$= \eta_\theta \frac{2(x^2_j + y^2_j(N-1)\theta^2_{ij} - m^2_j)^2}{(\sigma^2_j)^2}$$  \hspace{1cm} (34)

where $\eta_m$, $\eta_\sigma$ and $\eta_\theta$ are the learning rates with positive constants.

C. Supervisory Controller

The most useful property of a neural network is its ability to approximate linear or nonlinear mapping through learning. In the following, the layer numbers of the parameters are omitted for convenience. By the universal approximation theorem, there exists an optimal RFNN such that [17]

$$V_{id}(t) = V_{rn}(w^*_k, m^*_j, \sigma^*_j, \theta^*_j, t) + \epsilon(t)$$  \hspace{1cm} (35)

where $w^*_k, m^*_j, \sigma^*_j$ and $\theta^*_j$ are the optimal parameters of $w_k, m_j, \sigma_j$ and $\theta_j$, respectively. The time invariant optimal parameters $w^*_k, m^*_j, \sigma^*_j$ and $\theta^*_j$ are defined as

$$w^*_k = \arg \min_{w_k \in \Omega_k} \sup_{m_j \in \Omega_j, \sigma_j \in \Omega_j, \theta_j \in \Omega_j} |V_{rn} - V_{id}|$$  \hspace{1cm} (36)

$$m^*_j = \arg \min_{m_j \in \Omega_j} \sup_{w_k \in \Omega_k, \sigma_j \in \Omega_j, \theta_j \in \Omega_j} |V_{rn} - V_{id}|$$  \hspace{1cm} (37)
The error bound is assumed to be a constant during the observation; however, it is difficult to measure it in practical applications. Therefore, a bound estimation is developed to observe the bound of the approximation error. Define the estimation error of the bound

\[ \tilde{E}(t) = E - \hat{E}(t) \]

where \( \hat{E}(t) \) is the estimated error bound. The supervisory controller is designed to compensate for the effect of approximation error and is chosen as

\[ V_{sp}(t) = \tilde{E}(t) \text{sgn}(s(t)). \]

By substituting (18) into (7), it is revealed that

\[ \dot{q}(t) = \frac{-f(t)}{r(t)} q(t) + \frac{1}{r(t)} (V_t(t) - (V_{rn}(t) - V_{sp}(t))) \]

\[ - g(\Delta q(t), \Delta \dot{q}(t)). \]

After some straightforward manipulation, the error equation governing the system can be obtained through (9), (16) and (42) as follows:

\[ \dot{s}(t) = \tilde{q}(t) + k_1 \tilde{q}(t) + k_2 q(t) = V_{id}(t) - V_{rn}(t) - V_{sp}(t). \] (43)

The adaptive laws in (29) and (32)-(34) are unable to guarantee that \( w_k \in \Omega_{w_k}, m_{ij} \in \Omega_{m_{ij}}, \sigma_{ij} \in \Omega_{\sigma_{ij}} \) and \( \theta_{ij} \in \Omega_{\theta_{ij}} \). Therefore, these adaptive laws should be modified by using the scalar case of projection algorithm [17], such that the parameters will be constrained inside the compact sets. The modified adaptive laws are given as follows

\[ \sigma_{ij}^* = \arg \min_{\sigma_{ij} \in \Omega_{\sigma_{ij}}} \left\{ \sup_{w_k \in \Omega_{w_k}, m_{ij} \in \Omega_{m_{ij}}, \theta_{ij} \in \Omega_{\theta_{ij}}} \left| V_{rn} - V_{id}(t) \right| \right\} \] (38)

\[ \theta_{ij}^* = \arg \min_{\theta_{ij} \in \Omega_{\theta_{ij}}} \left\{ \sup_{w_k \in \Omega_{w_k}, m_{ij} \in \Omega_{m_{ij}}, \sigma_{ij} \in \Omega_{\sigma_{ij}}} \left| V_{rn} - V_{id}(t) \right| \right\} \] (39)

where \( \Omega_{w_k}, \Omega_{m_{ij}}, \Omega_{\sigma_{ij}} \) and \( \Omega_{\theta_{ij}} \) are compact sets of suitable bounds on \( w_k, m_{ij}, \sigma_{ij} \) and \( \theta_{ij} \), respectively, and they are defined as \( \Omega_{w_k} = \{ w_k | w_k \leq M_{w_k} \} \), \( \Omega_{m_{ij}} = \{ m_{ij} | m_{ij} \leq M_{m_{ij}} \} \), \( \Omega_{\sigma_{ij}} = \{ \sigma_{ij} | \sigma_{ij} \leq M_{\sigma_{ij}} \} \) and \( \Omega_{\theta_{ij}} = \{ \theta_{ij} | \theta_{ij} \leq M_{\theta_{ij}} \} \),

where \( M_{w_k}, M_{m_{ij}}, M_{\sigma_{ij}} \) and \( M_{\theta_{ij}} \) are positive constants and \(|x|\) denotes the absolute value. The \( c(t) \) denotes the approximation error; and it is assumed to be bounded by \( 0 \leq |c(t)| \leq E \) where \( E \) is a positive constant.

The error bound is assumed to be a constant during the observation; however, it is difficult to measure it in practical applications. Therefore, a bound estimation is developed to observe the bound of the approximation error. Define the estimation error of the bound

\[ \tilde{E}(t) = E - \hat{E}(t) \]

(40)

where \( \hat{E}(t) \) is the estimated error bound. The supervisory controller is designed to compensate for the effect of approximation error and is chosen as

\[ V_{sp}(t) = \tilde{E}(t) \text{sgn}(s(t)). \]

By substituting (18) into (7), it is revealed that

\[ \dot{q}(t) = \frac{-f(t)}{r(t)} q(t) + \frac{1}{r(t)} (V_t(t) - (V_{rn}(t) + V_{sp}(t))) \]

\[ - g(\Delta q(t), \Delta \dot{q}(t)). \]

(42)

After some straightforward manipulation, the error equation governing the system can be obtained through (9), (16) and (42) as follows:

\[ \dot{s}(t) = \tilde{q}(t) + k_1 \tilde{q}(t) + k_2 q(t) = V_{id}(t) - V_{rn}(t) - V_{sp}(t). \]

(43)

The adaptive laws in (29) and (32)-(34) are unable to guarantee that \( w_k \in \Omega_{w_k}, m_{ij} \in \Omega_{m_{ij}}, \sigma_{ij} \in \Omega_{\sigma_{ij}} \) and \( \theta_{ij} \in \Omega_{\theta_{ij}} \). Therefore, these adaptive laws should be modified by using the scalar case of projection algorithm [17], such that the parameters will be constrained inside the compact sets. The modified adaptive laws are given as follows

\[ \sigma_{ij}^* = \arg \min_{\sigma_{ij} \in \Omega_{\sigma_{ij}}} \left\{ \sup_{w_k \in \Omega_{w_k}, m_{ij} \in \Omega_{m_{ij}}, \theta_{ij} \in \Omega_{\theta_{ij}}} \left| V_{rn} - V_{id}(t) \right| \right\} \]

(38)

\[ \theta_{ij}^* = \arg \min_{\theta_{ij} \in \Omega_{\theta_{ij}}} \left\{ \sup_{w_k \in \Omega_{w_k}, m_{ij} \in \Omega_{m_{ij}}, \sigma_{ij} \in \Omega_{\sigma_{ij}}} \left| V_{rn} - V_{id}(t) \right| \right\} \]

(39)

Then, the stability of the proposed SRFNNC system is summarized by the following theorem.

**Theorem 1:** Consider the nonlinear system shown in (7) with the control law given in (18). In the control law, \( V_{rn} \) is given by (26), in which the parameters \( w_k, m_{ij}, \sigma_{ij} \) and \( \theta_{ij} \) are adapted by (44)-(47); and the supervisory controller is given as (41). Then, the feedback control system in Fig. 1 guarantees the following properties:

1. \( |w_k| \leq M_{w_k} \), \( |m_{ij}| \leq M_{m_{ij}} \), \( |\sigma_{ij}| \leq M_{\sigma_{ij}} \) and \( |\theta_{ij}| \leq M_{\theta_{ij}} \);

2. \( s(t) \rightarrow 0 \) as \( t \rightarrow \infty \).

**Proof:**

1. To prove \( |w_k| \leq M_{w_k} \), let \( V_w = (1/2)(w_k)^2 \). If the
The first line of (44) is true, then either \( |w_k| < M_{w_k} \) or
\[
\dot{V}_w = \eta_w s(t) \frac{s(t)}{r(t)} w_k \leq 0 \quad \text{for} \quad |w_k| = M_{w_k}, \quad \text{i.e.,} \quad |w_k| \leq M_{w_k}
\]
always be satisfied. If the second line of (44) is true, that is \( |w_k| = M_{w_k} \), then \( \dot{V}_w = 0 \), i.e., \( |w_k| \leq M_{w_k} \). Therefore, \( |w_k| \leq M_{w_k} \) for all \( t > 0 \) is proved. Similarly, \( |m_j| \leq M_{m_j} \), \( |\sigma_j| \leq M_{\sigma_j} \), and \( |\theta_j| \leq M_{\theta_j} \) can be proved.

2) Define a Lyapunov function as
\[
V_2(s(t), \bar{E}(t)) = \frac{1}{2} s^2(t) + \frac{1}{2 \eta_E} \bar{E}^2(t) \tag{48}
\]
where \( \eta_E \) is the learning rate with a positive constant. Differentiating (48) with respect to time and using (18), (35), (41) and (43), yields
\[
\dot{V}_2(s(t), \bar{E}(t)) = s(t) \dot{s}(t) + \bar{E}(t) \ddot{E}(t) / \eta_E
\]
\[
= s(t)(\varepsilon(t) - V_{2p}(t)) + \bar{E}(t) \ddot{E}(t) / \eta_E
\]
\[
= \varepsilon(t)s(t) - \bar{E}(t)s(t) + \bar{E}(t) \ddot{E}(t) / \eta_E \tag{49}
\]
If the adaptive law for the supervisory controller is chosen as
\[
\dot{\bar{E}}(t) = -\bar{E}(t) / \eta_E
\]
then (49) can be rewritten as
\[
\dot{V}_2(s(t), \bar{E}(t)) = \varepsilon(t)s(t) - \bar{E}(t)s(t) - (E - \bar{E}(t)) \|s(t)\|
\]
\[
= \varepsilon(t)s(t) - \bar{E}(t)s(t) - \bar{E}(t) \|s(t)\| < 0.
\]
Since \( \dot{V}_2(s(t), \bar{E}(t)) \) is negative semi-definite, that is \( V_2(s(t), \bar{E}(t)) \leq V(s(0), \bar{E}(0)) \), it implies that \( s(t) \) and \( \bar{E}(t) \) are bounded. Let the function \( \Omega(t) := (E - \varepsilon(t)) \times s(t) \leq (E - \varepsilon(t)) \|s(t)\| \leq -\dot{V}_2(s(t), \bar{E}(t)) \), and integrate \( \Omega(t) \) with respect to time, then it is obtained that
\[
\int_0^t \Omega(\tau) d\tau \leq V_2(s(0), \bar{E}(0)) - V_2(s(t), \bar{E}(t)).
\]
Because \( V_2(s(0), \bar{E}(0)) \) is bounded, and \( V_2(s(t), \bar{E}(t)) \) is nonincreasing and bounded, the following result can be obtained:
\[
\lim_{t \to \infty} \int_0^t \Omega(t) d\tau < \infty.
\]
Also, \( \dot{\Omega}(t) \) is bounded, so by Barbial’s Lemma [18], it can be obtained that \( \lim_{t \to \infty} \Omega(t) = 0 \). That is, \( s(t) \to 0 \) as \( t \to \infty \). As a result, the supervisory recurrent fuzzy neural network control system is asymptotically stable. Moreover, when the state \( q(t) \) is trapped on the sliding-surface, the dynamics of the system are determined by (9) and (17) which is always stable, so the state \( q(t) \) will slide into the origin [19], [20]. This completes the Proof of Theorem 1.

Remark: As shown in (7), the model of AUV guidance system is a time varying and time delay system. Theoretically, a time varying and time delay ideal controller can be designed as in (16) to achieve stable system response. However, this ideal controller cannot be implemented for the unknown uncertainty problem; thus, an adaptive SRFNNC system is used to mimic the unobtainable ideal controller, and a supervisory controller is applied to cope with the approximation error for achieving the system’s stability.

5. Simulation Results

The proposed SRFNNC guidance law is applied to the time-delay AUV-target intercept problem. The AUV is modeled by the dynamic system with six degrees of freedom. Three of them are to specify the position of the center of mass (denoted by \( x, y, z \)), and the other three are the Euler angles (denoted by \( \phi, \theta, \psi \)) to describe the orientation of AUV. If AUV is moving with velocity \( u \) along \( CX' \) axis, then the components of this velocity along \( x, y, \) and \( z \) axes can be given by
\[
\begin{bmatrix}
x
\end{bmatrix} = \begin{bmatrix}
ucos\psi cos\theta \\
usin\psi cos\theta \\
-uisin\theta
\end{bmatrix},
\]
where \( \theta \) is the pitch angle and \( \psi \) is the yaw angle. Then the relation between the time rate of the Euler angles and the angular velocity in the local frame can be written as
\[
\begin{bmatrix}
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi}
\end{bmatrix} = \begin{bmatrix}
1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\
0 & \cos\phi & -\sin\phi \\
0 & \sin\phi\sec\theta & \cos\phi\sec\theta
\end{bmatrix} \begin{bmatrix}
w_x \\
w_y \\
w_z
\end{bmatrix},
\]
where \( w_x, w_y, \) and \( w_z \) are the angular rates in the local frame corresponding to the rolling rate, pitch rate and yaw rate, respectively. The AUV velocity \( u \) is assumed to be a constant and the angular rates \( w_x, w_y \) and \( w_z \) can be controlled by proper orientations of the control planes located at the back of the AUV. For guidance, the control inputs of AUV are pitch rate \( (w_y) \) and yaw rate \( (w_z) \). Based on the polar geometry, the control input \( w_z \) (or \( w_y \)) can be computed as
\[
w_z(t) \quad \text{or} \quad w_y(t) = \frac{V(s) / \Delta}{r(t)}.
\]
The \( w_y \) and \( w_z \) are bounded from \(-0.5\) rad/sec to
0.5 rad/sec and the $u$ is maintained with constant velocity as 18 Knots. Two simulation scenarios are examined. For the first scenario, the speed of a crosswise moving target is set with initial velocity of 10 Knots and with acceleration of 0.05 g after 150 sec. For the second scenario, the speed of diving target is set with initial velocity of 1 Knots and with acceleration of 0.02 g. The sound speed in water is set as 1500m/sec.

Since the sea surface both reflects and scatters sound, it has a profound effect on propagation in most applications of underwater sound with shallow source or receiver, and a criterion for the roughness or smoothness of a surface is given by a sinusoidal function [21]. In addition, the sonar energy which impinges the soft bottom will be attenuated within the sediment, so bottom scatter is also assumed to be a sinusoidal function [22]. In our simulations, the sonar measurement noise of $q(t)$ is assumed to be $n = 3\sin(500t)$ deg, and the sampling time is set as 0.1 sec. A Butterworth low-pass filter with cutoff frequency 0.3 rad/sec is used to filter the noise in the guidance law design.

The simulations are performed for these two scenarios with the initial parameters shown in Table 1. For comparison, the simulation scenarios are performed by a PN guidance law, a SMC guidance law and the proposed SRFNNC guidance law. The PN guidance law is denoted as (8) with the navigation ratio set as 2. The SMC guidance law is obtained as in (10) where $F$ in (12) is set as 0.5. For the SRFNNC guidance law, the parameters are selected as follows: $k_1 = 0.3$, $k_2 = 0.02$, $\eta_w = \eta_m = \eta_p = \eta_0 = 0.9$, and $\eta_E = 1$. For the PN, SMC and SRFNNC guidance laws, the simulation results are shown in Figs. 5 to 7 and Figs. 8 to 10, corresponding to scenarios 1 and 2, respectively. The miss distances are summarized in Table 2. It is shown the SRFNNC guidance law can achieve smaller miss distance than the PN and SMC guidance laws.

![Figure 5(a). The target pursuing trajectory diagram of scenario 1 (PN guidance law).](image)

![Figure 5(b). The control inputs of scenario 1 (PN guidance law).](image)

![Figure 6(a). The target pursuing trajectory diagram of scenario 1 (SMC guidance law).](image)

Table 1. The initial parameters of simulations.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Scenarios</th>
<th>Scenario#1</th>
<th>Scenario#2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(x_0, y_0, z_0)_{AUV} (m)$</td>
<td>(2000,1000,-10)</td>
<td>(2000,1000,-10)</td>
<td></td>
</tr>
<tr>
<td>$(\phi_0, \theta_0, \psi_0)_{AUV} (rad)$</td>
<td>$(0,0,0)$</td>
<td>$(0,0,0)$</td>
<td></td>
</tr>
<tr>
<td>$(\phi_0, \theta_0, \psi_0)_{AUV} (rad/\sec)$</td>
<td>$(0,0,0)$</td>
<td>$(0,0,0)$</td>
<td></td>
</tr>
<tr>
<td>$V_{AUV} (Knots)$</td>
<td>10</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>$a_{AUV} (g)$</td>
<td>0.02</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>$(x_0, y_0, z_0)_{AUV} (m)$</td>
<td>(0,0,-30)</td>
<td>(0,0,-30)</td>
<td></td>
</tr>
<tr>
<td>$(\phi_0, \theta_0, \psi_0)_{ATT} (rad)$</td>
<td>$(0,0,0)$</td>
<td>$(0,0,0)$</td>
<td></td>
</tr>
<tr>
<td>$(\phi_0, \theta_0, \psi_0)_{ATT} (rad/\sec)$</td>
<td>$(0,0,0)$</td>
<td>$(0,0,0)$</td>
<td></td>
</tr>
<tr>
<td>$V_{ATT} (Knots)$</td>
<td>18</td>
<td>18</td>
<td></td>
</tr>
</tbody>
</table>

![Table 2. The miss distance of simulations.](image)
Figure 6(b). The control inputs of scenario 1 (SMC guidance law).

Figure 7(a). The target pursuing trajectory diagram of scenario 1 (RFNNC guidance law).

Figure 7(b). The control inputs of scenario 1 (RFNNC guidance law).

Figure 8(a). The target pursuing trajectory diagram of scenario 2 (PN guidance law).

Figure 8(b). The control inputs of scenario 2 (PN guidance law).

Figure 9(a) The target pursuing trajectory diagram of scenario 2 (SMC guidance law).

Figure 9(b). The control inputs of scenario 2 (SMC guidance law).

Figure 10(a). The target pursuing trajectory diagram of scenario 2 (RFNNC guidance law).
6. Conclusions

In this paper, a supervisory recurrent fuzzy neural network control (SRFNNC) AUV guidance law is designed in the condition of unknown of measurement time delay and disturbance. The proposed SRFNNC combines the advantages of the sliding-mode control with robust characteristics and the neural network with on-line learning ability. A comparison among the PN, SMC and the proposed SRFNNC AUV guidance laws is presented. The simulation results demonstrate that the SRFNNC guidance law can achieve smaller miss distances than the PN and SMC guidance laws. These simulation results also confirm the effectiveness of the proposed SRFNNC guidance law for dealing with the time varying and time delay AUV guidance system. The more complex controller design based on the time varying and time delay AUV guidance system will be the future studies.

Acknowledgment

The authors also want to give their thanks to the Reviewers for their valuable comments. And partially funded by teacher’s research project of Taoyuan Innovation Institute of Technology.

References

Yi-Jen Mon is currently an Associate Professor of Department of Computer Science and Information Engineering, Taoyuan Innovation Institute of Technology, Taiwan. He received the B. S. degree from Department of Electrical Engineering, Chung-Cheng Institute of Technology, Taiwan, in 1984 and received the M. S. and Ph.D. degrees from Department of Electrical Engineering, Yuan-Ze University, Taiwan, in 1997 and 2003, respectively. During 1984-2003, he was with the Chung-Shan Institute of Science and Technology as a researcher in the areas of computer science and control systems design. He has engaged in the educational works since 2003. His research interests include fuzzy logic control, intelligent control, automotive safety systems and embedded system design.

Chih-Min Lin is currently a Chair Professor of Electrical Engineering, Yuan Ze University, Taiwan. He also serves as an Associate Editor of IEEE Transaction on Systems, Man, and Cybernetics-Part B, Cybernetics; Asian Journal of Control; International Journal of Fuzzy Systems and International Journal of Machine Learning and Cybernetics. He is a Board of Governor Member of IEEE Systems, Man, and Cybernetics Society. During 1997–1998, he was the honor research fellow at the University of Auckland, New Zealand. His research interests include fuzzy neural network, cerebellar model articulation controller, and intelligent control systems. He has published 110 journal papers and 148 conference papers. Dr. Lin is an IET Fellow.