A Neural Network-based Self-tuning PID Controller of an Autonomous Underwater Vehicle

Enzeng Dong\textsuperscript{1,2}, Shuxiang Guo\textsuperscript{1,2}, Xichuan Lin\textsuperscript{2}, Xiaoqiong Li\textsuperscript{3} and Yunliang Wang\textsuperscript{1}

\textsuperscript{1} School of Electrical Engineering, Tianjin University of Technology, Xiqing District, 300384, Tianjin, China
\textsuperscript{2} Faculty of Engineering, Kagawa University, Takamatsu, 761-0396, Kagawa, Japan
\textsuperscript{3} School of Life Science, Beijing Institute of Technology, Haidian District, 100081, Beijing, China

\{dong, guo and lixq\}@eng.kagawa-u.ac.jp, wangyl@tjut.edu.cn

\textbf{Abstract} – Taking into account the complex interferences in underwater environment, this paper presents a neural network-based self-tuning PID controller for a spherical AUV. The control system consists of neural network identifier and neural network controller, and the weights of neural networks are trained by using Davidon least square method. The proposed controller is characterized with a strong anti-interference ability and a fast convergence rate. For its simple structure, the controller can be easily realized in hardware. The linear velocity of the spherical AUV can be controlled to precisely track any desired trajectory in vehicle-fixed coordinate system. The effectiveness of the controller is verified by simulation results.

\textbf{Index Terms} - spherical AUV; neural network; Davidon least square method; PID controller

\section{I. INTRODUCTION}

Research on autonomous underwater robots (AUVs) is attracting increased attention around the world. Different kinds of AUVs were developed and used successfully in various applications, such as oceanographic surveys, bathymetric measurements, underwater maintenance activities. Most of these underwater vehicles are torpedo-like with a streamline body [1-3], and some of them are underwater biomimetic micro-robots [4-7]. Recently, Lin et al designed a kind of AUV with a spherical body [8-9].

AUVs are currently being utilized for scientific, commercial and military underwater applications. These vehicles require autonomous guidance and control systems in order to perform underwater tasks. The studies on automatic control of these vehicles are still major active areas of research and development [10-12]. The idea of having a self-driven device performing various tasks underwater without human intervention is appealing from both an industrial and a theoretical viewpoint.

The PID control is a common control strategy which is easily realized in hardware [13-14]. For example, a regular PID was adopted in tracking control of an AUV named Mako [15]. Considering the complex ocean current activities and noise interference in underwater environment, this paper presents a self-tuning PID controller for an autonomous underwater vehicle with a spherical body. The proposed controller consists of neural network identifier and neural network controller; the weights of neural networks are trained by using Davidon least square method. A three-layer feed-forward neural network identifier is applied to model the vehicle’s dynamics and external interference; a self-tuning PID controller is realized by using two-layer neural network. The proposed controller is characterized with a strong anti-interference ability and a fast convergence speed. For its simple structure, the controller can be easily realized physically. The linear velocity of the spherical AUV can be controlled to accurately track any desired target in vehicle-fixed coordinate system. The effectiveness of the proposed controller is verified by simulation results.

\section{II. THE DYNAMICS MODEL OF SPHERICAL AUV}

\subsection{A. Vehicle Kinematics}

In this section, we introduce a coordinate system fixed on the surface of earth as the earth-fixed coordinate system \( E : \{E - x, y, z\} \). The earth-fixed coordinate system is used to express the position and orientation of the spherical AUV. Correspondingly, we introduce a vehicle-fixed coordinate system \( S : \{S - x_r, y_r, z_r\} \) which moves along with the spherical AUV. The vehicle-fixed coordinate system is applied to express the linear and angular velocity of the spherical AUV. The position and attitude of the AUV in earth-fixed coordinate can be expressed by vectors \( \eta_1 = (x, y, z) \) and \( \eta_2 = (\varphi, \theta, \psi) \) respectively, then define \( \eta = (\eta_1, \eta_2) \). At the same time, the linear velocity and angular velocity of the AUV in vehicle-fixed coordinate system can be expressed as \( v_1 = (v_x, v_y, v_z) \) and \( v_2 = (\omega_x, \omega_y, \omega_z) \) respectively, then we define \( v = (v_1, v_2) \). The notation of the position, orientation, velocity and force of the 6 DOF of an underwater vehicle is deeply analyzed by Fossen [16]. The transformation of the linear velocity can be described as

\[ \eta_1 = J_1(\eta_2)v_1 \] (1)

where \( J_1(\eta_2) \) is the transformation matrix which related to the function of Euler angles [13].

\[ J_1(\eta_2) = \begin{bmatrix} \cos \psi \cos \theta & \sin \psi \cos \theta \cos \phi - \sin \phi \sin \psi & \sin \psi \cos \theta \sin \phi + \cos \phi \sin \psi \\ \sin \psi \cos \theta & \cos \psi \cos \theta \cos \phi + \sin \phi \sin \psi & \cos \psi \cos \theta \sin \phi - \sin \phi \sin \psi \\ -\sin \theta & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix} \] (2)

where \( s() = \sin() \), \( c() = \cos() \).

And the transformation of the angular velocity can be described as

\[ \eta_2 = J_1(\eta_2)v_2 \] (3)

where \( J_1(\eta_2) \) can be expressed as

\[ J_1(\eta_2) = \begin{bmatrix} c\psi s\theta & c\psi s\theta \sin \phi + c\phi \sin \psi & c\psi s\theta \cos \phi - c\psi \cos \phi \\ s\psi c\theta & s\psi s\theta \cos \phi - c\phi \sin \psi & s\psi s\theta \sin \phi + c\phi \sin \psi \\ -s\theta & c\theta \sin \phi & c\theta \cos \phi \end{bmatrix} \]
\[
J_2(\eta_2) = \begin{bmatrix}
1 & \sin \varphi \tan \theta & \cos \varphi \tan \theta \\
0 & \cos \varphi & -\sin \varphi \\
0 & \sin \varphi / \cos \theta & \cos \varphi / \cos \theta
\end{bmatrix}
\] (4)

So, the whole transformation matrix which transforms velocity from vehicle-fixed coordinate to earth-fixed coordinate can be described as
\[
\begin{bmatrix}
\dot{\eta}_1 \\
\dot{\eta}_2
\end{bmatrix} = \begin{bmatrix}
J_1(\eta_1) & 0 \\
0 & J_2(\eta_2)
\end{bmatrix} \begin{bmatrix}
v_1 \\
v_2
\end{bmatrix}
\] (5)

B. Vehicle Dynamics Model

Firstly, for the purpose of simplifying, we assume that the underwater environment is waveless. Then, the dynamics equation of motion for spherical underwater vehicle can be expressed in vehicle-fixed coordinate system as
\[
M\ddot{\mathbf{v}} + C(\mathbf{v})\dot{\mathbf{v}} + D(\mathbf{v})\mathbf{v} + G(\eta) = \boldsymbol{\tau}
\] (6)

where \(\tau \in \mathbb{R}^6\) is the vector of control inputs, \(\dot{\mathbf{v}}\) is the time derivative of velocity, \(G(\eta) \in \mathbb{R}^6\) is vector of restoring forces and moments, \(D(\mathbf{v}) \in \mathbb{R}^6\) is damping matrix, \(M \in \mathbb{R}^{6\times6}\) is inertia mass including added mass and \(C(\mathbf{v}) \in \mathbb{R}^6\) is coriolis and centripetal matrix.

The conceptual design and experimental prototype of the spherical AUV developed by Lin et al [17-18] are shown in Fig.1. There are three water-jet propellers, used as propulsion system in the spherical body of the vehicle, and three water-jet propellers are equipped in accordance with angle of 120° degree. The structural design of the robot is symmetric about the Z-axis. Its total diameter is 40 cm, and its weight is 6.3kg. This spherical AUV is not bottom-heavy, the weight of the inside components is distributed. The Z-axis position of the waterproof box could be adjusted by four long screws. Hence, the center of mass of the robot is adjustable.

\[
\begin{bmatrix}
23.05 & 0 & 0 & \dot{u}_x \\
0 & 23.05 & 0 & \dot{v}_x \\
0 & 0 & 0.5145 & \dot{w}_x \\
0.02 & 0.3 & 0 & \dot{v}_y \\
0 & 0.02 & 0.3 & \dot{w}_y \\
0 & 0 & 3.8 \times 10^{-4} & \dot{w}_z
\end{bmatrix}
\] (7)

where
\[
\tau = [(F_1 + F_2) \cos \theta, (F_1 - F_2) \cos \theta, 0.2 \cdot (F_1 - F_2) \cos \theta]^T.
\]

In this paper, the linear velocity in X-axis direction of the spherical AUV is controlled to track the desired signal. The structure of the spherical AUV from top view is shown in Fig.2.

III. THE NEURAL NETWORK-BASED SELF-TUNING PID CONTROLLER

The structure of whole closed loop is shown in Fig.3, the whole system consists of two neural networks, the system is identified by using a three layer feed-forward neural network, and the PID controller is realized by a two layer neural network. The self-tuning mechanism is as follows: the model of the spherical AUV is online identified by using the neural network identifier; the weights of neural network PID are update by using Davidon least squares method, and the parameters \(k_p\), \(k_i\) and \(k_d\) can be adjusted in real time.

A. Neural Network identifier of The Spherical AUV

![Fig. 3. The block diagram of closed-loop system.](image-url)
Taking into account the coupling properties of the spherical AUV, the velocity \( u_v \) is identified and predicted by using a three-layer feed-forward neural network. The structure of the identifier is shown in Fig. 4. We take state variables \( u_v(t), v_v(t), r_r(t) \) and the controller \( u(t) \) as the input signals of the neural network identifier.

![Fig. 4. The structure of feed-forward neural network NNI](image)

The one step prediction of the linear velocity in X-axis direction is described by

\[
\hat{u}_v(t+1) = f_{NN}(u_v(t), v_v(t), r_r(t), u(t), W(t))
\]

where \( u_v(t) \) and \( v_v(t) \) are linear velocities in X-axis direction and Y-axis direction; \( r_r(t) \) is the angular velocity which rotates around Z-axis; \( u(t) \) is the control action at the sampling time \( t \); and \( W(t) \) is the whole weights matrix of the neural network NNI. The input layer of the neural network NNI consists of four nodes, the specific forms are given by

\[
X = [u_v(t), v_v(t), r_r(t), u(t)]^T
\]

We define the number of nodes in the hidden layer as \( m_i \), so the following equations can be obtained.

\[
net_i^j(t) = \sum_{i=1}^{m_i} W_i^j(t)X_i(t), \quad i = 1, \ldots, m_i
\]

\[
net_j^i(t) = f(net_i^j(t)), \quad i = 1, \ldots, m_i
\]

where \( W^j \) is the weights matrix which is between the input layer and the hidden layer; and the activation function of the hidden layer node is Sigmoid function.

There is a single node in output layer, which can be described as

\[
\hat{u}_v(t+1) = \sum_{j=1}^{m_h} W_j^m(t)net_j^i(t)
\]

where \( W^m \) represents the weights matrix between hidden layer and output layer.

The cost function of the neural network identifier is defined as

\[
E(t) = \frac{1}{2}(u_v(t+1) - \hat{u}_v(t+1))^2
\]

To improve the convergence rate, the connection weights in the neural network are trained by using Davidon least square method. The corresponding weight increments can be recursively calculated by following equations.

\[
W(t+1) = W(t) - \frac{H(t)\nabla\phi(t)\phi(t)}{\eta + \nabla^2\phi(t)H(t)\nabla\phi(t)}
\]

\[
H(t+1) = \frac{1}{\eta}[H(t) + \nabla^2\phi(t)H(t)\nabla\phi(t)]
\]

where \( W(t) \) represent the weights vector which consists of all the weights in the neural network; \( \phi(t) = \hat{u}_v(t) - u_v(t) \); \( \nabla\phi(t) = \frac{\partial\phi(t)}{\partial W(t)} \) is the gradient vector according to \( W(t) \); and \( \eta \in (0, 1] \) is the forgetting factor.

For the weights matrix \( W^j(t) \), the gradient vector can be calculate by

\[
\frac{\partial\phi(t)}{\partial W^j(t)} = net_j^i(t)
\]

All elements in \( \nabla\phi(t) \) can be obtained by using equation (16) and equation (17).

**B. The Neural Network PID Controller**

To realize the self-turning function of the proposed PID controller, the parameters \( k_p, k_i, k_d \) are represented by the weights of a two-layer neural network, and the output of the neural network controller is the control increment. The structure of the two-layer neural network is shown in Fig. 5.

![Fig. 5. The self-turning PID controller neural network.](image)

The input layer of the neural network controller is given by

\[
X^c = [e_1(t), e_2(t), e_3(t)]^T
\]

where \( e_1(t) = u_v^d(t) - u_v(t), \quad e_2(t) = \Delta e_1(t) = e_1(t) - e_1(t-1), \quad e_3(t) = \Delta e_2(t) = e_2(t) - e_2(t-1) \) and \( u_v^d(t) \) is the desired signal.

Obviously, the input signals \( e_1(t), e_2(t) \) and \( e_3(t) \) are system tracking error, one-order differential form of error and second order differential form of error, the values of these
signals can be obtained by error sampling and preprocessing. According to the specific form of the incremental PID controller, the output of the neural network controller is described as

$$\Delta u(t) = k_p e_1(t) + k_i e_1(t) + k_d e_1(t)$$  \hspace{1cm} (19)$$

So, the PID control strategy is obtained by

$$u(t) = u(t-1) + \Delta u(t)$$ \hspace{1cm} (20)$$

Taking into account the precise fitting capability of the three-layer neural network NNI, the output of NNI can converge to the actual output of the spherical AUV. The cost function of the neural network PID controller is defined as

$$E^2(t) = \frac{1}{2} (\hat{u}_1(t+1) - u^d(t+1))^2$$ \hspace{1cm} (21)$$

The parameters $k_p(t), k_i(t)$ and $k_d(t)$ are represented by the weights of the neural network controller, and they are trained by Davidon least square method. The weight increments can be calculate by

$$P(t+1) = \lambda P(t) - \lambda \nabla e^T(t) P(t) \nabla e(t)$$ \hspace{1cm} (23)$$

where $K(t) = [k_p(t), k_i(t), k_d(t)]^T$, $e(t) = \hat{u}_1(t+1) - u^d(t+1)$, $\nabla e(t)$ is the gradient vector according to $\phi(t)$, and $\lambda \in (0,1]$ is the forgetting factor.

$$\frac{\partial e(t)}{\partial k_i} = \frac{\partial \hat{u}_1(t+1)}{\partial u(t)} \frac{\partial u(t)}{\partial k_i}$$ \hspace{1cm} (24)$$

$$\frac{\partial \hat{u}_1(t+1)}{\partial u(t)} = \sum_{j=1}^{m} W_{o}^j f'(\text{net}_j^i(t))W_{i,j}^i$$ \hspace{1cm} (25)$$

C. The whole process of the proposed algorithm

In this section, the process of the whole algorithm is summarized as follows:

1. Setting up the initial weight values of the neural network identifier, the neural network controller, covariance matrix $H(0), P(0)$, and forgetting factor $\eta, \lambda$.
2. Sampling the values of $u_i(t), v_i(t), r_i(t)$, and calculating $e_i(t), e_c(t), e_n(t)$.
3. Calculating the output $\Delta u(t)$ of the neural network controller and the PID control strategy $u(t) = u(t-1) + \Delta u(t)$.
4. Updating the weight values in the neural network identifier NNI and the neural network controller NNPID.
5. If the expected step is reached, then stop the process of self-tuning PID; else, turn to step (3).

IV. NUMERICAL SIMULATIONS

In this section, we take the spherical AUV as the study object.

The linear velocity in X-axis direction is controlled to track the sine wave and square wave. The effectiveness of proposed control method is verified by simulation results.

A. The Simulation of Tracking to Sine Wave

As shown in Fig.6, the velocity in X-axis direction is controlled to track a desired sine signal. Comparing to the tracking effectiveness of regular PID in Fig.9, the proposed self-tuning PID controller has a better tracking precision. Fig.7 shows that the PID parameters $k_p, k_i$ and $k_d$ can automatically evolve with the variation of the desired signal, and the corresponding PID controller is shown in Fig. 8.
Fig. 9. The simulation result of regular PID controller with $k_p = 0.9$, $k_i = 0.2$, $k_d = 0.3$.

B. The Simulation of Tracking to Square wave

Fig. 10. The simulation result of the velocity in X-axis direction tracking to sine wave signal.

Fig. 11. The evolutions of the parameters of PID controller.

Fig. 12. The corresponding PID controller.

C. The Analysis of Tracking Effectiveness With Noise Interference

Fig. 13. The simulation result of the velocity in X-axis direction with white noise interference.

Fig. 14. The evolutions of the parameters of PID controller.

Fig. 15. The corresponding PID controller.

There exits all kinds of interferences in underwater environment, such as current activities, measurement noise.
etc. Consider the impact of these disturbances, we add the white noise in the dynamics model of the spherical AUV, and the variance of white noise is 0.05. Fig.13 shows that the high precision tracking of the velocity of the AUV with the noise interference. The complex evolutions of the parameters of PID controller are shown in Fig.14, and Fig.15 shows the corresponding PID controller.

V. CONCLUSIONS

Considering the complex ocean current activities and noise interferences in underwater environment, this paper has proposed a self-tuning PID controller for a spherical AUV. The whole closed loop system consists of two neural networks; a three-layer feed-forward neural network is used to indentify the linear velocity of the spherical AUV, and a two-layer neural network is applied to adjust the parameters of the PID controller online. All the weights of two neural networks are trained by using Davidon least square method, which has a faster convergence rate. The numerical analysis shows that the linear velocity in X-axis direction of the spherical AUV can be controlled to precisely track different signals. These simulation results have been showed demonstrating the effectiveness of the proposed control strategy. The proposed controller is characterized with a strong anti-interference ability and a fast convergence rate. For its simple structure, the controller can be easily realized in hardware.

ACKNOWLEDGMENT

This work was supported by Guolab, Faculty of Engineering, Kagawa University.

REFERENCES
