Hindawi
Complexity
Volume 2019, Article ID 3821048, 10 pages
https://doi.org/10.1155/2019/3821048

Research Article

Ship Steering Control Based on Quantum Neural Network

Wei Guan, Haotian Zhou, Zuojing Su, Xianku Zhang, and Chao Zhao

Navigation College, Dalian Maritime University, Dalian 116026, China

Correspondence should be addressed to Zuojing Su; suzuojing@163.com

Received 14 August 2019; Accepted 6 November 2019; Published 17 December 2019

Guest Editor: Raúl Baños

During the mission at sea, the ship steering control to yaw motions of the intelligent autonomous surface vessel (IASV) is a very challenging task. In this paper, a quantum neural network (QNN) which takes the advantages of learning capabilities and fast learning rate is proposed to act as the foundation feedback control hierarchy module of the IASV planning and control strategy. The numeric simulations had shown that the QNN steering controller could improve the learning rate performance significantly comparing with the conventional neural networks. Furthermore, the numeric and practical steering control experiment of the IASV BAICHUAN has shown a good control performance similar to the conventional PID steering controller and it confirms the feasibility of the QNN steering controller of IASV planning and control engineering applications in the future.

1. Introduction

In the past decade, the research on intelligent automatic surface ship (IASV) technology in academic and marine industries has continued to grow. These developments have been fuelled by advanced sensing, communication, and computing technology together with the potentially transformative impact on automotive sea transportation and perceived social and economic benefits [1–5]. The ship planning and control strategy for IASV, shown in Figure 1, which based on a module-based hierarchical structure, would be a good navigation strategy. It includes the global routing planning module, behaviour decision-making module, local motion planning module, and feedback control module. These modules are in charge of the different tasks especially the feedback control module is the foundation module as action part of the IASV navigation process. The key function of this module is the IASV steering operation to maintain or change ship course. In this paper, a quantum neural network (QNN) for ship steering control is proposed to address the ship steering control problem based on the IASV planning and control concept.

As a good research foundation of the IASV steering control problem, many effective steering feedback control methods had been surveyed. The ship steering control based on proportional-integral-derivative (PID) strategy is simple and easy to construct. However, the conventional PID occupies the necessary basic controller role in process control, but it is not the trend of controller design due to the lack of learning and adaption capabilities. In addition, the controller parameters are required adjustments in varying conditions, which are time consuming and may not achieve accurate control performance. To solve the issues and obtain better performance, various advanced control strategies have been proposed for the steering control of the ship in recent years, such as adaptive steering control strategy [6–8], steering control strategy based on fuzzy logic algorithm [9, 10], steering controller based on Backstepping controller design method [11–13], and adaptive backstepping method [14, 15]. The robust control schemes such as the sliding mode control method [16, 17] and H_∞ robust control algorithm [18] are also utilized in the ship steering control to achieve better ship course keeping and changing manoeuver.

Since the 1990s, with the introduction of the artificial neural network into the ship steering controller design, experts and scholars had gradually increased ship steering control research on this issue. Witt et al. proposed a PID steering controller to train a neural network, where the output signal of the PID controller acts as the teacher signal and the simulation results showed that the control

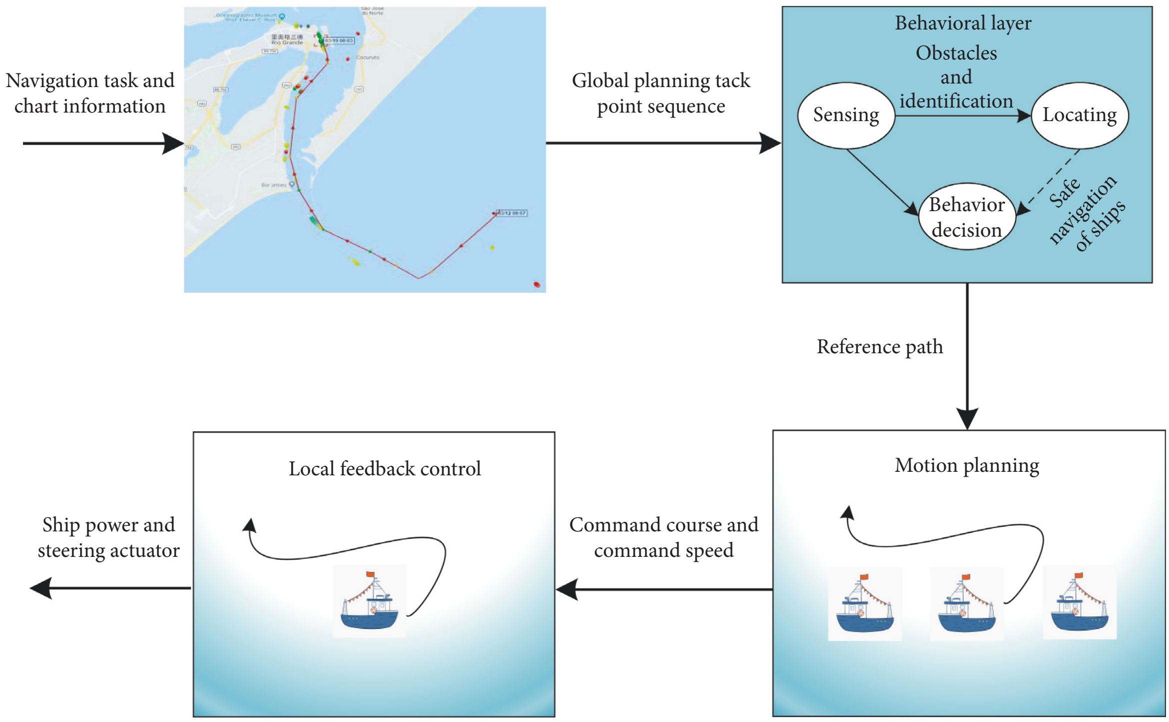

Figure 1: IASV planning and control strategy.

effects of the PID controller and the neural network controller have basically the same course control effect [19]. Hearn et al. proposed an online course control neural network to improve the conventional PID steering control effects, but the slow convergence of the ship steering controller based on the neural network is still a big problem to be solved [20].

In order to overcome the shortcomings of the conventional neural network, a ship steering controller based on the QNN is proposed in this paper under the concept of quantum computing [21–23]. The concept of QNN was first proposed by Toth et al. in 1996 [24]. Then, Matsui et al. used a quantum bit and the quantum revolving door to design a QNN for information processing and expression [25]. Another group of Japanese scholars, Kouda et al., summarized their previous research and summed up an emerging model, which is a quantum neuron model based on general quantum logic gates [26]. In 2018, Jeswal and Chakraverty introduced the latest developments of QNN and discussed the application of QNN [22]. And Xie et al. took the general quantum logic gates as the basis function to design a quantum neural computing network, and the simulation results indicated that the QNN is superior to the classical BP neural network and the radial basis function (RBF) neural network computing model in the financial data analysis [27]. Li and Li also pointed out that QNN based on general quantum gate evolution can improve the convergence performance of the conventional BP neural network [28]. Besides, the QNN has been applied to signature verification [29], audio watermarking [30], cardiovascular diseases risk prediction [31], classification recognition of electronic shock fault [32], and English to Hindi machine translator [33] and other fields.

Motivated by the above observations, a QNN steering controller would be applied to the IASV steering to yaw control. Hence, the remainder of the paper is organized as follows. In Section 2, the mathematical model of the IASV steering to yaw motion is given. Section 3 devotes to a systematic procedure for the QNN steering controller design. In Section 4, the numeric comparison simulations for the QNN steering controller and conventional neural networks steering controller were firstly carried out to demonstrate the faster learning convergence of the proposed QNN steering controller. Then, a numeric and practical experiment on smart IASV BAICHUAN has shown the feasibility of the QNN steering controller in practical engineering practice. Finally, Section 5 gives the conclusions of the paper.

2. IASV Mathematical Model

While a mathematical model of the IASV is fully described by coupled nonlinear differential equations, a simple model with predictive capability is usually preferred for the design of a ship-steering autopilot. A three-degree-of-freedom plane motion including surge, sway, and yaw motion is considered satisfactory. However, roll motion cannot be neglected due to couplings and hence a four-degree-of-freedom plane motion including surge, sway, yaw, and roll motion is used to describe the motion of a ship. Consequently, a fourth-order transfer function relating to the yaw rate to rudder deflection is derived based on the linearized equations of motion. Nevertheless, a fourth-order transfer function is further reduced to a second-order Nomoto model and then to a first-order Nomoto model for ease of controller design. The first-order Nomoto model, from δ to yaw rate r is presented as

$$\frac{r(s)}{\delta(s)} = \frac{K}{(1+Ts)}, \tag{1}$$

where r is the yaw rate, δ is the rudder deflection, T is the time constant of IASV maneuverability, and K is defined as the steering control gain constant of IASV maneuverability. The parameters K and T that describe the ship steering to yaw dynamics can be identified from standard maneuvering tests. Since r is the time derivative of the yaw angle ψ, the transfer function relating to the yaw angle to steering movement can be obtained by adding an integrator (1/s) to the first-order Nomoto model of (1), then we can get

$$\frac{\psi(s)}{\delta(s)} = \frac{K}{s(1+Ts)}, \tag{2}$$

and the corresponding differential equation can be expressed as

$$T\ddot{\psi} + \dot{\psi} = K\delta. \tag{3}$$

The model presented in (3) is modified to include a nonlinear steering condition as discussed in [6], wherein the yaw rate $\dot{\psi}$ term is replaced by a nonlinear function $H(\dot{\psi})$. Then, we can get the following equation:

$$T\ddot{\psi} + H(\dot{\psi}) = K\delta, \tag{4}$$

where

$$H(\dot{\psi}) = \alpha_0 + \alpha_1\dot{\psi} + \alpha_2\dot{\psi}^2 + \alpha_3\dot{\psi}^3. \tag{5}$$

Because of the symmetrical structure of ships, the parameters $a_0 = a_2 = 0$ [34] and α_1 is set as +1 for stable ships and −1 for unstable ones, while the value of α_3, known as the Norbin coefficient [14], can be determined via the ship turning test.

3. QNN Steering Controller Design

In this section, a quantum neural network model was constructed for the ship steering controller design to enhance the convergence performance of the conventional neural network steering controller.

3.1. The Quantum Neuron Model. The structure of the quantum neuron model based on the quantum logic gate is defined as Figure 2, including the input part, phase rotation part, aggregation part, reverse rotation part, and output part. The details of the quantum neural networks working processes are shown as the following steps:

Step 1: let $|x_i\rangle = (\cos t_i, \sin t_i)^T$, and define the qubit phase rotation gate as

$$R(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}. \tag{6}$$

Then, with the aggregation, we can get

$$\sum_{i=1}^{n} R(\theta_i)\left|x_i\right\rangle = [\cos\theta \ \ \sin\theta]^T, \tag{7}$$

where $\theta = \arg\left(\sum_{i=1}^{n} R(\theta_i) \mid x_i\rangle\right) = \arg\tan\left(\sum_{i=1}^{n}\sin(t_i + \theta_i)/\sum_{i=1}^{n}\cos(t_i+\theta_i)\right)$.

Step 2: the result of equation (7) makes the reverse rotation operation by the controlled-NOT gate as follows:

$$U(\gamma) = \begin{pmatrix} \cos\left(f(\gamma)\frac{\pi}{2} - 2\theta_0\right) & -\sin\left(f(\gamma)\frac{\pi}{2} - 2\theta_0\right) \\ \sin\left(f(\gamma)\frac{\pi}{2} - 2\theta_0\right) & \cos\left(f(\gamma)\frac{\pi}{2} - 2\theta_0\right) \end{pmatrix}, \tag{8}$$

where f is the sigmoid function; then, we can get

$$U(\gamma)\sum_{i=1}^{n} R(\theta_i)\left|x_i\right\rangle = \left[\cos\left(\frac{\pi}{2}f(\gamma) - \theta\right) \ \ \sin\left(\frac{\pi}{2}f(\gamma) - \theta\right)\right]^T. \tag{9}$$

Therefore, the relationship between the input and output of the quantum neuron model can be described as

$$y = \sin\left(\frac{\pi}{2}f(\gamma) - \theta\right) = \sin\left(\frac{\pi}{2}f(\gamma) - \arg\left(\sum_{i=1}^{n} R(\theta_i) \mid x_i\rangle\right)\right). \tag{10}$$

3.2. QNN Model. Based on the quantum neuron model, a quantum neural network for the ship steering controller design is constructed as shown in Figure 3. The proposed neural network has three layers including an input layer, hidden layer, and output layer. The concept of QNN is applied to the layer that is between the input layer and the hidden layer; there are n quantum neurons in the input layer, m quantum neurons in the hidden layer, and p conventional neurons in the output layer.

Assuming the input variable is $|x_i\rangle$, the output of the hidden layer is h_j, the output of the QNN is y_k, $R(\theta_{ij})$ is the quantum rotation gate between the input layer and the hidden layer to update the qubits, and w_{jk} is the network weight for the hidden layer and the output layer. Taking the qubit-controlled NOT-gate $U(\gamma_j)$ as the transfer function of the hidden layer, then the output of the QNN can be expressed as

$$\begin{aligned} y_k &= g\left(\sum_{j=1}^{m} w_{jk}h_j\right) = g\left(\sum_{j=1}^{m} w_{jk}\sin\left(\frac{\pi}{2}f(\gamma_j)\right.\right. \\ &\qquad \left.\left. - \arg\left(\sum_{i=1}^{m} R(\theta_{ij}) \mid x_i\rangle\right)\right)\right), \end{aligned} \tag{11}$$

where $i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, m$; and $k = 1, 2, \ldots, p$.

3.3. The Learning Algorithm of QNN. To apply the QNN in practical engineering, the training samples should be transformed into the quantum states. For example, the n-dimensional Euclidean space training sample

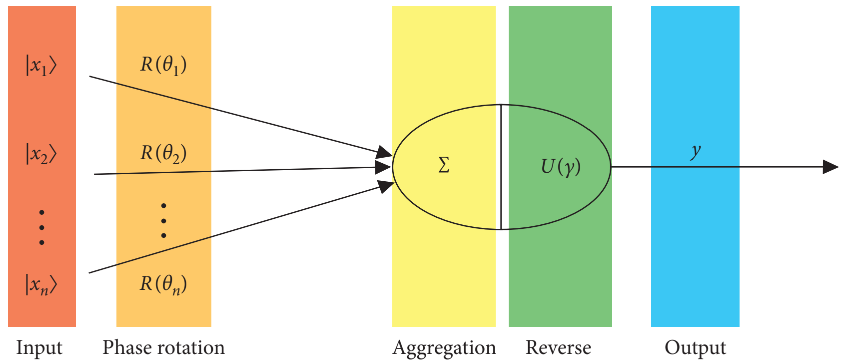

FIGURE 2: The quantum neuron model.

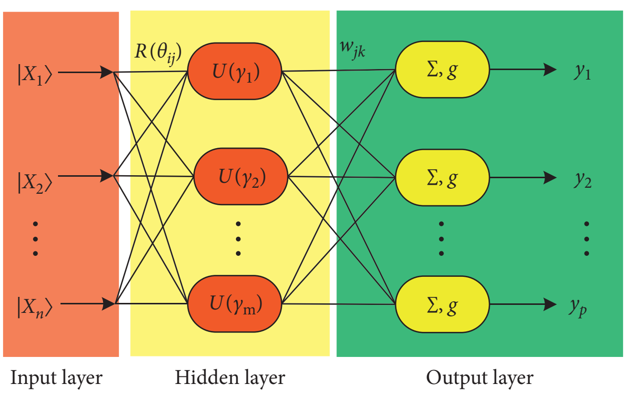

FIGURE 3: The quantum neuron networks model.

$\overline{X} = (\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_n)^T$ can be defined as the corresponding quantum state as

$$|X\rangle = \left(|x_1\rangle, |x_2\rangle, \ldots, |x_n\rangle \right)^T, \tag{12}$$

where

$$\begin{aligned} |x_i\rangle &= \cos\left(\frac{2\pi}{1+\exp(-\overline{x}_i)}\right)|0\rangle + \sin\left(\frac{2\pi}{1+\exp(-\overline{x}_i)}\right)|1\rangle \\ &= \left(\cos\left(\frac{2\pi}{1+\exp(-\overline{x}_i)}\right), \sin\left(\frac{2\pi}{1+\exp(-\overline{x}_i)}\right)\right)^T. \end{aligned} \tag{13}$$

In the three layers of the QNN model as described in Figure 3, there are 3 groups of parameters, namely, phase rotation parameters θ_{ij}, reverse parameters γ_j, and network weights w_{jk} needed to be updated. Firstly, define the error evaluation function as

$$E = \frac{1}{2}\sum_{k=1}^{p}(d_k - y_k)^2, \tag{14}$$

where d_k and y_k are the desired outputs and actual outputs of the normalized quantum neural network, respectively. Let $|x_i\rangle = (\cos\varphi_i, \sin\varphi_i)^T$ and $\beta_j = \arctan(\sum_{i=1}^{n}\sin(\varphi_i + \theta_{ij})/\sum_{i=1}^{n}\cos(\varphi_i + \theta_{ij}))$, then equation (11) could be rewritten as

$$y_k = g\left(\sum_{j=1}^{m} w_{jk}\sin\left(\frac{\pi}{2}f(\gamma_j) - \beta_j\right)\right). \tag{15}$$

Let

$$\begin{cases} S_j = \dfrac{\sum_{i=1}^{n}\sin(\varphi_i + \theta_{ij})}{\sum_{i=1}^{n}\cos(\varphi_i + \theta_{ij})}, \\ S_{j1} = \sum\limits_{i=1}^{n}\cos(\varphi_i + \theta_{ij}), \\ T_j = \dfrac{\left(\cos(\varphi_i + \theta_{ij})S_{j1} + \sin^2(\varphi_i + \theta_{ij})\right)}{S_{j1}^2}. \end{cases} \tag{16}$$

According to the gradient descent method, we can get

$$\begin{cases} \Delta\theta_{ij} = -\dfrac{\partial E}{\partial \theta_{ij}} = -\sum\limits_{k=1}^{p}(d_k - y_k)g' w_{jk}\cos\left(\dfrac{\pi}{2}f(\gamma_j) - \beta_j\right) \\ \quad \cdot \dfrac{T_j}{1 + {S_j}^2}, \\ \Delta\gamma_j = -\dfrac{\partial E}{\partial \gamma_j} = \dfrac{\pi}{2}\sum\limits_{k=1}^{p}(d_k - y_k)g' w_{jk}\cos\left(\dfrac{\pi}{2}f(\gamma_j) - \beta_j\right)f', \\ \Delta w_{jk} = -\dfrac{\partial E}{\partial w_{jk}} = (d_k - y_k)g'\sin\left(\dfrac{\pi}{2}f(\gamma_j) - \beta_j\right). \end{cases} \tag{17}$$

Therefore, the updated rules for phase rotation parameters θ_{ij}, reverse rotation parameters γ_j, and network weights w_{jk} are

$$\begin{cases} \theta_{ij}(t+1) = \theta_{ij}(t) + \eta\Delta\theta_{ij}(t), \\ \gamma_j(t+1) = \gamma_j(t) + \eta\Delta\gamma_j(t), \\ w_{jk}(t+1) = w_{jk}(t) + \eta\Delta w_{jk}(t), \end{cases} \tag{18}$$

where η is the learning rate of the QNN.

3.4. Teacher Controller of QNN. In this paper, the conventional PID steering control controller is acted as the teacher of the QNN controller. The input variable of the PID controller is the heading deviation $\Delta\psi$, and the linear combination of the proportion, integration, and differentiation of the heading deviation $\Delta\psi$ is used as the output value of the PID controller. The command steering angle $\delta(k)$ can be expressed as

$$\delta(k) = k_p \Delta\psi(k) + k_i \sum_{j=0}^{k} \Delta\psi(j) + k_d[\Delta\psi(k) - \Delta\psi(k-1)], \tag{19}$$

where k_p, k_i, and k_d are the controller proportion parameter, integral parameter, and differential parameter, respectively, and k is the sampling time $(k = 0, 1, 2, \ldots)$.

3.5. Design of the QNN Steering Controller. In this section, a three layer 2-5-1 QNN model was constructed. The structure of the QNN steering control system is shown in Figure 4. The two inputs of the QNN steering controller are the heading deviation $\Delta\psi(k)$ and yaw rate $r(k)$, respectively, and the output is the command steering angle $\delta_r^{\mathrm{QNN}}(k)$. The difference between the QNN steering controller outputs and PID course keeping controller outputs is defined as the system error. The mean square of the system error (MSE) is defined as the performance evaluation function of the proposed QNN to evaluate the performance of the QNN learning performance and optimized targets. Generally, its value is set as 0.00001. The activation function of the QNN hidden layer and the output layer is defined as hyperbolic tangent sigmoid function (tan-sigmoid) to accelerate the QNN training and convergence performance in the training process. The gradient descent with a quasi-Newton algorithm [35] is offered to the QNN training, and the momentum parameter of the quasi-Newton algorithm is set as 0.8. The initial values of the QNN weights are randomly generated between the intervals $(-1, 1)$, and the learning rate η of QNN is set as 0.1.

4. Simulations and Analysis

In this section, a series of simulations were used to illustrate the fast convergence characteristics and practical engineering effectiveness of the proposed controller. Especially an IASV BAICHUAN is utilized as a practical experiment for validations of the proposed QNN steering controller. Take $K = 0.6, T = 1.866, a_1 = 1$, and $a_3 = -9.44 \times 10^{-6}$ as the dynamic parameters of the second-order Nomoto ship model equation (4) for IASV BAICHUAN. Set $k_p = 2$, $k_i = 0.00001$, and $k_d = 1.5$ as tuning parameters of the teacher controller equation (19). In the simulations, the initial course of the IASV was set as 000° and the desired course keeping angle was set as 090°. The simulation time was set as 50 s, and the sampling period was set as 0.05 s. From the result of the simulations shown in Figure 5, it can be seen that the PID steering controller, which acted as the teacher of the QNN controller, could track the desired course after 13s, and the result shows that the PID controller is satisfactory to act as a suitable teacher of the QNN steering controller.

To illustrate the practical effectiveness of the proposed QNN steering control system, as shown in Figure 4, during the QNN steering controller training, the values of phase rotation parameters θ_{ij}, reverse rotation parameters γ_j, and network weights w_{jk} would be updated according to equation (18) by using the training data set extracted from PID controller simulation results in Figure 5.

For comparison, a conventional BP neural network steering controller is also trained using the same training data set extracted from the PID control results in Figure 5. To emphasize the advantages of the faster convergence and fewer learning iterations, the QNN steering controller and BP neural network steering controller were trained 8 times, respectively, and then the epochs in each training time are shown in Figure 6. For the BP neural network, the maximum number of training epoch is 9565 (in the 6th training time), the minimum training epoch is 4325 (in the 2nd training time), and the average number of training epochs for the 8 training times is 7022. Although, for the QNN, the maximum number of training epoch is 4625 (in the 2nd training time), the minimum training epoch is 1526 (in the 4th training time), and the average number of sample training epoch of the 8 training times is 3302. Therefore, it can be concluded that the QNN steering controller is improved significantly in the convergence rate compared with the conventional BP neural network steering controller.

To validate the effectiveness of the trained QNN steering controller, an IASV BAICHUAN QNN steering control simulation was carried out. The weights w_{jk} of the QNN controller were extracted from the 2nd training time and selected as the initial weights of the IASV BAICHUAN QNN steering controller. The values are detailed in Table 1. Then, the simulation result are shown in Figure 7.

It can be seen from Figure 7 that the QNN steering controller could track the desired course at about 13 s. The control result is very similar to the PID controller. It can be concluded that the proposed QNN steering controller has a very strong learning ability and could be widely applied to various fields.

To further confirm the proposed QNN steering controller performance, an IASV BAICHUAN course keeping practical engineering experiment was carried out. The experiment environment is shown in Figure 8. The wind direction of the experiment scene was northwest (310°–330°), and the wind velocity varied from 0–0.20 m/s. The temperature was about 8°C. The maximum wave height was about 0–0.05 m. The initial course of the IASV is set at 000°, and the desired course keeping angle is set as 090°. The QNN steering controller's initial parameters are also set as Table 1. Also the PID steering control experiment was carried out for comparison. The parameter of the PID steering controller was also set as $k_p = 2$, $k_i = 0.00001$, and $k_d = 1.5$, as mentioned above. The sampling period of the QNN steering controller and PID steering controller are set as 0.05s, respectively. Finally, the experiment results are shown in Figure 9.

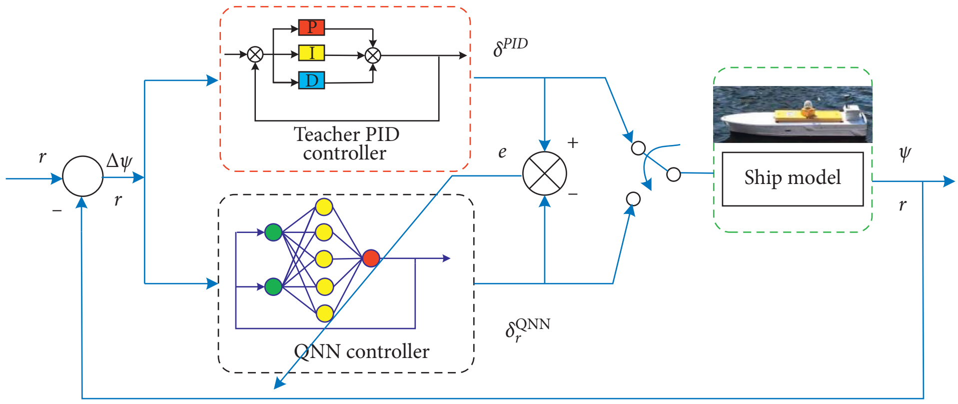

FIGURE 4: IASV QNN-steering autopilot structure.

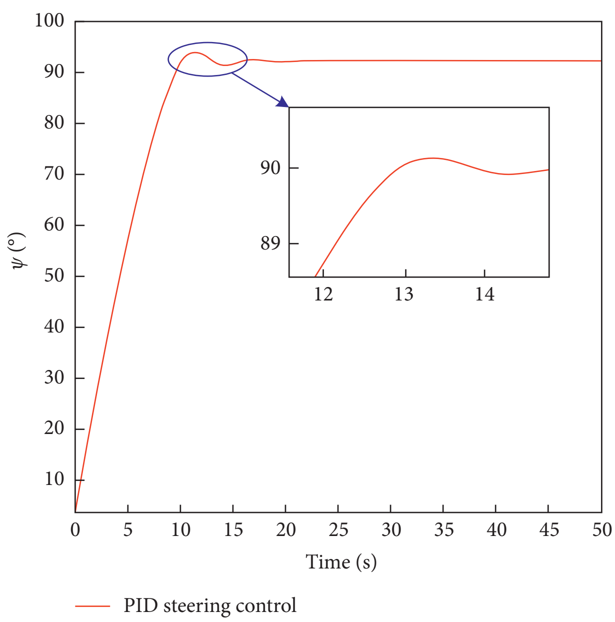

FIGURE 5: The PID steering control results.

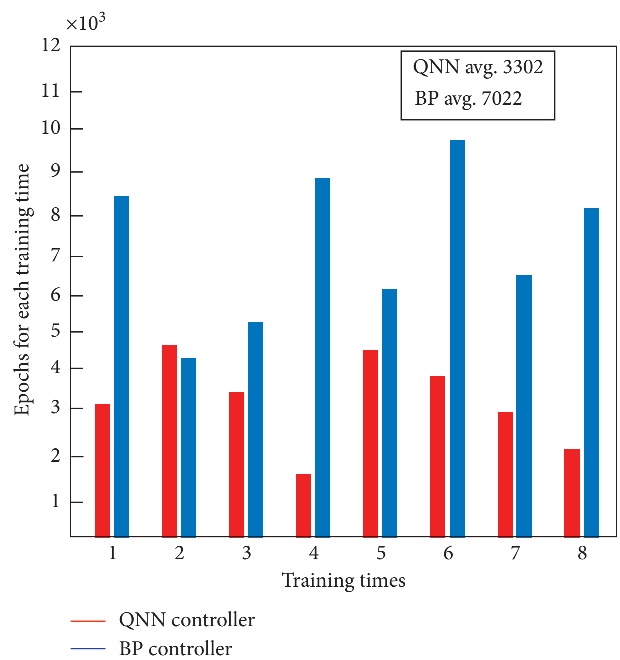

FIGURE 6: The epochs of eight training times.

The left side of Figure 9 shows the course keeping control effect, and the right side shows the output of the IASV BAICHUAN steering control. It can be seen that the rising time of the QNN steering controller is slightly slower than the PID controller, but both controllers can reach the target course rapidly and both of them can achieve a good course keeping control effect. As it can be seen on the right side of Figure 9, the controller output of the two type steering controller is basically the same and the time to reach the static stabilities are also similar, but the response of the QNN steering controller is also slightly slower (about one second) than the PID steering controller.

To further quantify the controller performance, the controller efficiency function (CEF) is defined as

$$\mathrm{CEF} = \frac{1}{n}\left(\sum_{k=1}^{n}(\Delta\psi(k))^2 + \sum_{k=1}^{n}(\Delta\delta(k))^2\right). \tag{20}$$

Then, we can obtain that the CEF of the QNN steering controller is 0.323 and the CEF of the PID steering controller is 0.299. Hence, it is concluded that the control efficiency of the QNN steering controller can get a similar control effect compared with the PID steering controller for the IASV course keeping.

Remark 1. From the numeric simulation and practical engineering experiment, it can be seen that the proposed QNN steering controller has a slightly delayed response compared with the PID steering controller, although the delayed response phenomenon is not obvious in the numeric simulations. The reason of the delayed response phenomenon might be caused by the larger computation burden of the QNN steering controller. This is a potential disadvantage of the QNN. However, due to the features of strong learning ability and fast convergence performance, the proposed QNN steering controller could be used in learning of other advanced controllers, not only restricted in the PID

Table 1: The weights and parameters of the QNN steering controller.

w_{13}	w_{14}	w_{15}	w_{16}	w_{17}	w_{23}	w_{24}	w_{25}	w_{26}	w_{27}	w_{38}	w_{48}	w_{58}	w_{68}	w_{78}	b_1	b_2	b_3	b_4	b_5	b_6
0.0026	−0.165	−0.0012	0.0074	−0.0075	−0.0133	0.1705	−0.0001	0.0230	−0.0233	−0.0190	0.0038	0.0010	0.0271	−0.0275	0.0013	0.0003	−0.0003	0.0026	6.4116	6.4114

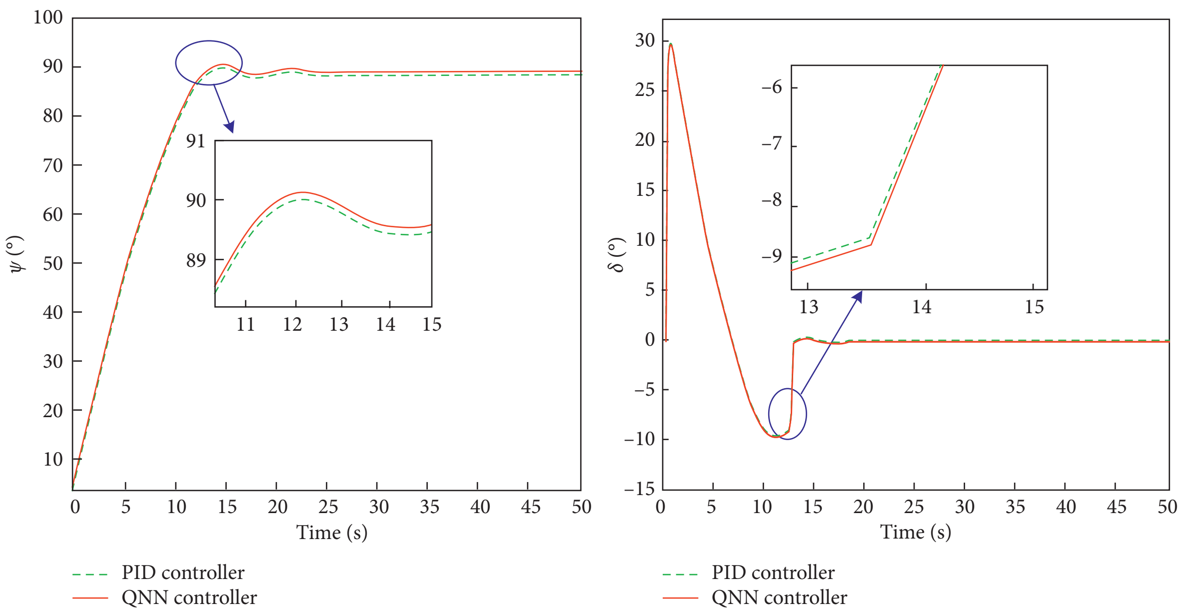

Figure 7: The course keeping simulation results for the PID and QNN steering controller.

Figure 8: The scene of the experiment.

controller. Therefore, the proposed QNN steering controller might be a universal controller design structure and scheme for the future IASV steering feedback control module.

5. Conclusions

In this paper, a QNN steering controller design method based on the planning and control concept is proposed. Through the numeric simulations of the steering controller based on the conventional BP neural network and QNN, it can be inferred that the QNN steering controller has a faster convergence rate than the conventional BP neural network steering controller. Also, the numeric simulation results show that the QNN steering controller has a similar course keeping control performance comparing with the training teacher PID steering controller. Furthermore, the practical QNN steering control experiment on an IASV BAICHUAN has shown that the proposed QNN steering controller is feasible to be equipped to a practical IASV for steering to yaw control in the future IASV planning and control engineering. Especially the strong learning characteristics and efficient convergence performance of the QNN steering controller might be the developing trend of the advanced IASV steering controller. However, the QNN steering controller proposed in this paper might be the first step to apply the advanced AI controller to the IASV. Furthermore, the proposed QNN controller structure could apply to other marine control engineering practices.

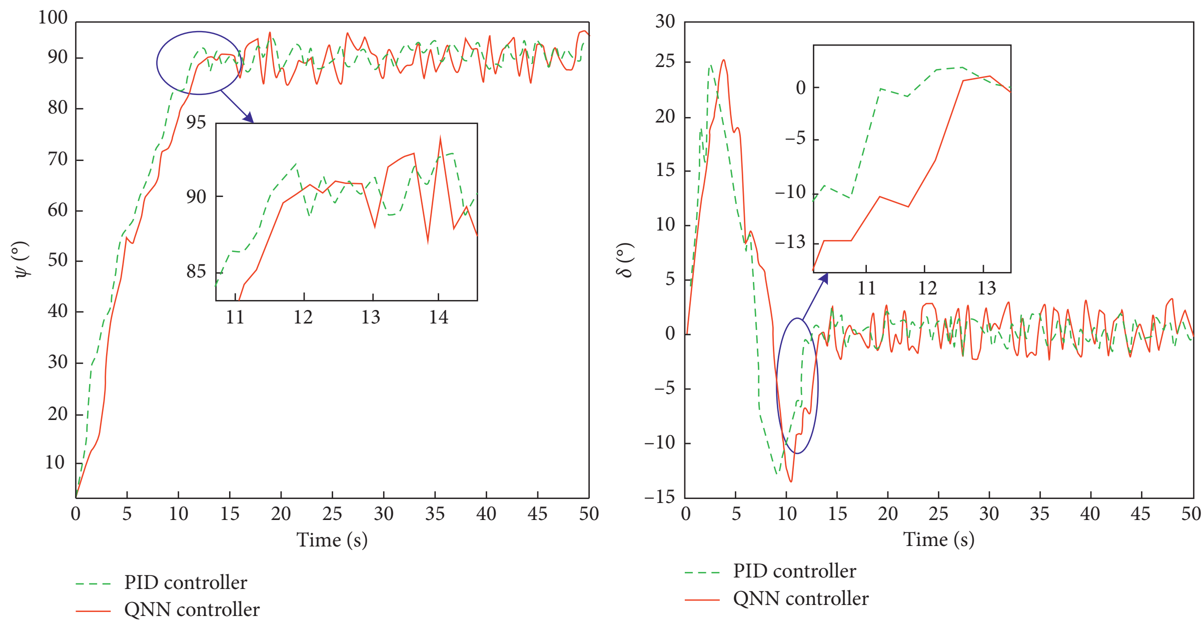

Figure 9: The practical course keeping control results of IASV BAICHUAN.

Data Availability

All data generated or analyzed during this study are included in this article.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Nos. 51409033 and 51679024) and the Fundamental Research Funds for the Central Universities of China (No. 3132019343).

References

[1] J. E. Manley, “Unmanned surface vehicles 15 years of development,” in *Proceedings of the OCEANS*, pp. 1–4, Quebec City, QC, Canada, September 2008.

[2] F. Yunsheng, G. Zenglu, Z. Yongsheng, and W. Guofeng, “Design of information network and control system for USV,” in *Proceedings of the 2015 54th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 1126–1131, Hangzhou, China, July 2015.

[3] N. H. Tran, H. S. Choi, J. Y. Oh, and S.-K. Jeong, “Design and implementation of dynamic positioning control system for USV,” in *AETA 2015. Recent Advances in Electrical Engineering and Related Sciences*, pp. 633–644, Springer, Cham, Switzerland, 2016.

[4] E. I. Sarda, I. R. Bertaska, A. Qu, and K. D. von Ellenrieder, “Development of a USV station-keeping controller,” in *Proceedings of the OCEANS 2015*, pp. 1–10, Genova, Italy, May 2015.

[5] I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks, “Collective memory and spatial sorting in animal groups,” *Journal of Theoretical Biology*, vol. 218, no. 1, pp. 1–11, 2002.

[6] J. Van Amerongen, “Adaptive steering of ships-a model reference approach,” *Automatica*, vol. 20, no. 1, pp. 3–14, 1984.

[7] J. Van Amerongen and A. J. Udink Ten Cate, “Model reference adaptive autopilots for ships,” *Automatica*, vol. 11, no. 5, pp. 441–449, 1975.

[8] C. Y. Tzeng and K. F. Lin, “Adaptive ship steering autopilot design with saturating and slew rate limiting actuator,” *International Journal of Adaptive Control and Signal Processing*, vol. 14, no. 4, pp. 411–426, 2000.

[9] G. Rigatos and S. Tzafestas, “Adaptive fuzzy control for the ship steering problem,” *Mechatronics*, vol. 16, no. 8, pp. 479–489, 2006.

[10] Y. Yang and J. Ren, “Adaptive fuzzy robust tracking controller design via small gain approach and its application,” *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 6, pp. 783–795, 2003.

[11] T. I. Fossen and J. P. Strand, “Tutorial on nonlinear backstepping: applications to ship control,” *Modeling, Identification and Control: A Norwegian Research Bulletin*, vol. 20, no. 2, pp. 83–135, 1999.

[12] K. D. Do, Z. P. Jiang, and J. Pan, “Underactuated ship global tracking under relaxed conditions,” *IEEE Transactions on Automatic Control*, vol. 47, no. 9, pp. 1529–1536, 2002.

[13] S. Das, A. Bhatt, and S. E. Talole, “UDE based backstepping design for ship autopilot,” in *Proceedings of the 2015 International Conference on Industrial Instrumentation and Control (ICIC)*, pp. 417–422, Pune, India, May 2015.

[14] J. Du, C. Guo, S. Yu, and Y. Zhao, “Adaptive autopilot design of time-varying uncertain ships with completely unknown control coefficient,” *IEEE Journal of Oceanic Engineering*, vol. 32, no. 2, pp. 346–352, 2007.

[15] J. Li, T. Li, Z. Fan, R. Bu, Q. Li, and J. Hu, “Robust adaptive backstepping design for course-keeping control of ship with parameter uncertainty and input saturation,” in *Proceedings of the 2011 International Conference of Soft Computing and*

Pattern Recognition (SoCPaR), pp. 63–67, Dalian, China, October 2011.

[16] E. W. McGookin, D. J. Murray-Smith, Y. Li, and T. I. Fossen, "Ship steering control system optimisation using genetic algorithms," *Control Engineering Practice*, vol. 8, no. 4, pp. 429–443, 2000.

[17] L. Yuan and H.-s. Wu, "Terminal sliding mode fuzzy control based on multiple sliding surfaces for nonlinear ship autopilot systems," *Journal of Marine Science and Application*, vol. 9, no. 4, pp. 425–430, 2010.

[18] L. Sheng, Y. Ping, L. Yan-yan, and D. Yan-chun, "Application of H infinite control to ship steering system," *Journal of Marine Science and Application*, vol. 5, no. 1, pp. 6–11, 2006.

[19] N. A. J. Witt, R. Sutton, and K. M. Miller, "A track keeping neural network controller for ship guidance," *IFAC Proceedings Volumes*, vol. 28, no. 2, pp. 385–392, 1995.

[20] G. E. Hearn, Y. Zhang, and P. Sen, "Alternative designs of neural network based autopilots: a comparative study," *IFAC Proceedings Volumes*, vol. 30, no. 22, pp. 83–88, 1997.

[21] R. P. Feynman, "Quantum mechanical computers," *Foundations of Physics*, vol. 16, no. 6, pp. 507–531, 1986.

[22] S. K. Jeswal and S. Chakraverty, "Recent developments and applications in quantum neural network: a review," *Archives of Computational Methods in Engineering*, vol. 26, no. 4, pp. 793–807, 2019.

[23] S. C. Kak, "On quantum neural computing," in *Advances in Imaging and Electron Physics*, vol. 94, pp. 259–313, Elsevier, Amsterdam, Netherlands, 1995.

[24] G. Toth, C. S. Lent, P. D. Tougaw et al., "Quantum cellular neural networks," *Superlattices and Microstructures*, vol. 20, no. 4, pp. 473–478, 1996.

[25] N. Matsui, M. Takai, and H. Nishimura, "A network model based on qubitlike neuron corresponding to quantum circuit," *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, vol. 83, no. 10, pp. 67–73, 2000.

[26] N. Kouda, N. Matsui, H. Nishimura, and F. Peper, "Qubit neural network and its learning efficiency," *Neural Computing and Applications*, vol. 14, no. 2, pp. 114–121, 2005.

[27] G. J. Xie, D. Zhou, and H. Q. Fan, "A neural network model based-on quantum gates cell and its applications," *System Engineering Theory and Practice*, vol. 25, no. 5, pp. 113–117, 2005.

[28] P. C. Li and S. Y. Li, "Learning algorithm and application of quantum BP neural networks based on universal quantum gates," *Journal of Systems Engineering and Electronics*, vol. 19, no. 1, pp. 167–174, 2008.

[29] M. J. Cao, P. C. Li, and H. Xiao, "Application of quantum neural network in PID parameter adjustment," *Computer Engineering*, vol. 37, no. 12, pp. 182–189, 2011.

[30] Y. P. Zhang, L. Chen, and H. A. Hao, "New audio watermarking algorithm based on quantum neural network," *Signal Processing*, vol. 29, no. 6, pp. 684–690, 2013.

[31] N. Renu, S. Sanjai, and G. Achal, "Cardiovascular risk prediction: a comparative study of Framingham and quantum neural network based approach," *Patient Preference and Adherence*, vol. 10, pp. 1259–1270, 2016.

[32] H. Guan, M. Liu, C. Li et al., "Classification recognition model of electric shock fault based on wavelet packet transformation and quantum neural network," *Transactions of the Chinese Society of Agricultural Engineering*, vol. 34, no. 5, pp. 183–190, 2018.

[33] R. Narayan, S. Chakraverty, and V. P. Singh, "Quantum neural network based machine translator for English to Hindi," *Applied Soft Computing*, vol. 38, pp. 1060–1075, 2016.

[34] W. Guan, W. Cao, J. Sun, and Z. Su, "Steering controller design for smart autonomous surface vessel based on CSF L2 gain robust strategy," *IEEE Access*, vol. 7, pp. 109982–109989, 2019.

[35] P. X. Lu, "Research on BP neural network algorithm based on quasi-Newton method," *Applied Mechanics and Materials*, vol. 686, pp. 388–394, 2014.