ICIC Express Letters
Volume **10**, Number **6**, June **2016**

 ISSN 1881-803X
pp. **1489–1495**

SHIP COURSE CONTROL BASED ON REINFORCEMENT LEARNING

Haiqing Shen[1], Chen Guo[1], Tieshan Li[2] and Ronghui Li[2]

[1]Information Science and Technology College
[2]Navigation College
Dalian Maritime University
No. 1, Linghai Road, Dalian 116026, P. R. China
flydm@aliyun.com; {dmuguoc; tieshanli}@126.com; lironghui@163.com

Received November 2015; accepted February 2016

Abstract. *In this paper, a control policy based on reinforcement learning algorithm is proposed for ship course due to the ship motion characteristics, which are big inertia, nonlinearity and uncertainty. The framework of actor-critic is adopted in this algorithm and two separated back propagation neural networks are used to implement actor and critic. Without the mathematical model of the object and prior knowledge, the algorithm can effectively control complex nonlinear system by trial-and-error learning. The simulation results demonstrate that the autopilot has satisfied control performance, which is robust to the external disturbances and nonlinearity of the ship motion.*
Keywords: Ship course, Reinforcement learning, Actor-critic, Neural network

1. **Introduction.** Ship motion control design has been a challenge which is characterized by big inertial, nonlinear and time-varying uncertainties [1]. In the past, several design methods have been proposed for autopilots such as classical control, adaptive control, proportional-integrate derivative control (PID), and other modern control techniques [2-4]. The autopilot based on PID has been designed and equipped on board since 1950s. However, owing to the ship's complicated characteristics, there are some difficulties in controlling the ship course perfectly by PID method. The majority of modern control techniques have several limitations. The first one is that these methods have high requirements for ship mathematical model, while ship motion with complicated characteristics is difficult to establish accurate mathematical model. The second one is the limitation of "explosion of complexity" in these control algorithms, which may lead to a more complicated controller and a larger control input [5].

Reinforcement learning (RL) has been widely applied to human-level control, robot control and so on [6,7]. Without the mathematical object model and the prior knowledge, RL algorithm utilizes reinforcement signals provided by environment to evaluate the current action and its effect in the future. Owing to the few information provided by the external environment, RL must rely on its own experience through a lot of trial and error to improve strategies to adapt to the environment and get good control quality [8]. The actor-critic RL algorithm used in this paper is evolved from the adaptive heuristic critic (AHC) algorithm, which has been detailedly introduced by Sutton and Barto [8]. Compared with other algorithms, this algorithm needs minimal amount of calculation to choose action and it is very useful in dealing with the no-Markov case [8]. However, RL is rarely used for ship motion control up to the present. Therefore, in this paper, a control policy based on actor-critic RL algorithm with two separated back propagation (BP) neural networks for ship course control is designed. To verify the performance of the proposed method, the external disturbances are considered, and a traditional PID course autopilot is also designed to compare with the proposed RL method. Then, the compared

simulations, between the proposed RL algorithm and traditional PID with disturbances or without disturbances, are performed and discussed.

The paper is organized as follows. A mathematical model of maneuvering motion for ship is introduced briefly in Section 2. In Section 3, the framework of actor-critic network adopted in RL algorithm is presented in detail. In Section 4, simulation results are presented to confirm the effectiveness of the proposed method. Finally, Section 5 concludes the paper.

2. **Ship Mathematical Model.** The motion of ship is very complex, which is with six degrees of freedom. We need to do something to simplify its model in response to the majority of ship motion and control issues. For the issue of ship course control, we could ignore the heaving, pitching and rolling, so the ship mathematical model is simplified to three degrees of freedom just considering the surging, swaying and yawing in the horizontal motion. Definition of motions in the horizontal plane diagram is shown in Figure 1. The course angle ψ can be changed only by controlling the rudder angle δ. The mathematical model related to the rudder angle and the heading rate of ship can be described in the following form [9]

$$\dot{r} + \frac{1}{T}r = \frac{K}{T}\delta \tag{1}$$

wherein r is the yaw rate, K is gain, and T is time constant.

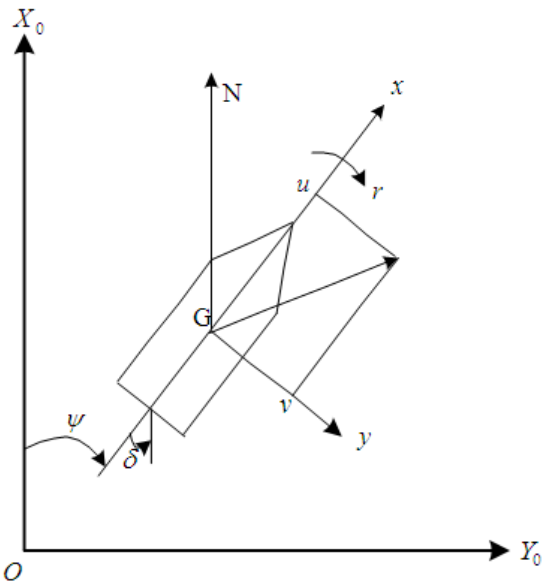

FIGURE 1. Ship's motion in the horizontal plane

In this paper, the characteristic of rudder is also considered, which can be expressed as [9]

$$T_E\dot{\delta} = \delta_E - \delta \tag{2}$$

wherein δ_E is command rudder angle, T_E is steer gear time constant; wherein T_E is 2.5s, $\left|\dot{\delta}\right| \leq 3^\circ/\text{s}$, $|\delta_E| \leq 35^\circ$ is the rudder angle limit and rudder rate limit.

When the ship is navigating on the sea, it will be influenced by the disturbances. We could suppose the disturbances including wave, current and wind which could be treated as an equivalent rudder angle. The disturbances can be described as follows [10]

$$\omega(t) = 2 + 0.5\sin(0.0523t) + rand(-0.5, 0.5) \tag{3}$$

wherein $rand(-0.5, 0.5)$ is a random function, generating random number between -0.5 and 0.5.

Finally, the ship mathematical model is composed of Equation (1), Equation (2) and Equation (3), which will be used in the simulation studies later.

3. Control System Structure.

3.1. Actor-critic control system.

Actor-critic methods are Temporal Differences (TD) methods [11] that have separated memory structure to explicitly represent the policy independent of the value function. The structure of actor-critic is shown in Figure 2 [8]. The policy structure used to select actions is known as the actor. The estimated value function is known as the critic, which criticizes the actions made by the actor. Learning is always on-policy: the critic must learn about and criticize whatever policy is currently being followed by the actor. The critique takes the form of TD error. This scalar signal is the sole output of the critic and it is also used to drive all learning in both actor and critic. Then, BP neural network, which has good ability of approximation, is applied to implementing actor and critic.

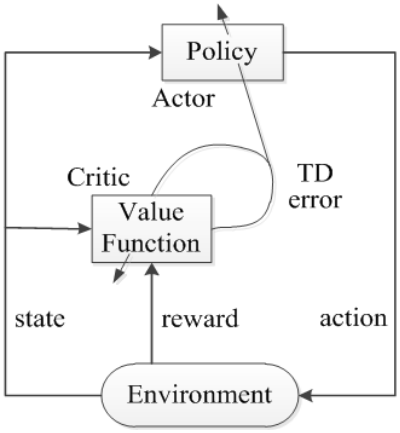

FIGURE 2. The structure of actor-critic

3.2. Critic network.

The critic network criticizes the actions according to external reward and system's states signal. The critic network seeks to maximize the expected discounted payoff, and its output is defined as some specific functions of the discounted reward sequence [8]:

$$V(t) = r_e(t+1) + \gamma r_e(t+2) + \gamma^2 r_e(t+3) + \cdots = \sum_{k=0}^{\infty} \gamma^k r_e(t+k+1) \tag{4}$$

wherein r_e is the reward, relying on the system's states and selected action and r_e would be -1 for each failure and at all other times; γ is the discount rate, $0 \leq \gamma \leq 1$, and the discount rate determines the present value of future rewards: a reward received k time steps in the future is worth only γ^{k-1} times what it would be worth if it were received immediately.

Hence from (4):

$$V(t-1) = r_e(t) + \gamma r_e(t+1) + \gamma^2 r_e(t+2) + \cdots = \sum_{k=0}^{\infty} \gamma^k r_e(t+k) \tag{5}$$

Therefore, from (4) and (5), the TD error for state-value prediction is [8]

$$\delta(t) = r_e(t) + \gamma V(t) - V(t-1) \tag{6}$$

When the TD error tends to zero, critic network will approximate Equation (4), and also remember the function prediction value $V(t-1)$. Then, $V(t+1)$ could be calculated

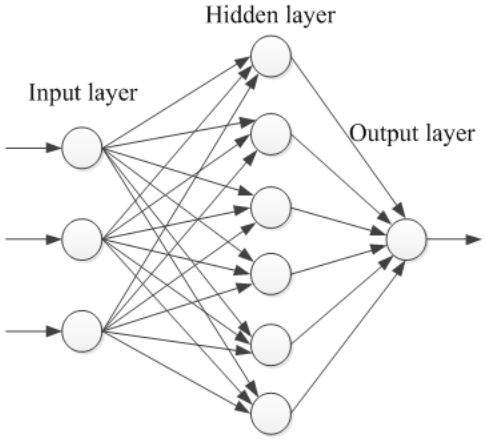

FIGURE 3. Architecture of critic network

without additional prediction model. After accomplishing learning, the prediction value of failure signal should be between -1 and 0 [12].

In critic network, BP neural network with three layers is used for training value function. The structure is shown in Figure 3. Let the number of input layer nodes be IN, and the output layer has only one node, the number of hidden layer nodes is HN, with the following activation function:

$$f(x) = \left(1 - e^{-x}\right) / \left(1 + e^{-x}\right) \tag{7}$$

Then, at time step t, the input of hidden layer nodes is defined by

$$u_{cj}(t) = \sum_{i=1}^{IN} w_{cij}(t) * X_{ci}(t) \tag{8}$$

wherein j denotes the jth hidden node. w_{cij} denotes the weights between the jth hidden node and input nodes. X_{ci} denotes the ith input node.

Hence, the output of hidden layer nodes can be expressed as

$$h_{cj}(t) = \left(1 - e^{-u_{cj}(t)}\right) / \left(1 + e^{-u_{cj}(t)}\right) \tag{9}$$

Finally, we can obtain the output of output node

$$V(t) = \sum_{j=1}^{HN} w_{cj}(t) * h_{cj}(t) \tag{10}$$

wherein w_{cj} denotes the weights between the jth hidden node and the output node.

The energy function of neural network is defined as Equation (11) [13]. The weight of neural network is regulated by gradient descent method.

$$E_c = \frac{1}{2}\left[\delta(t)\right]^2 = \frac{1}{2}\left[r_e(t) + \gamma V(t) - V(t-1)\right]^2 \tag{11}$$

3.3. **Actor network.** The goal of actor network is to maximize the future reward. Both critic network and actor network learn online in order to find an optimal policy. The structure of actor network is similar to critic network, shown in Figure 3. We also let the number of input layer nodes be IN and the output layer has only one node, the number of hidden layer nodes is HN, with the activation function Equation (7). At time step t, the input of hidden layer nodes is defined by

$$u_{aj}(t) = \sum_{i=1}^{IN} w_{aij}(t) * X_{ai}(t) \tag{12}$$

wherein j denotes the jth hidden node. w_{aij} denotes the weights between the jth hidden node and input nodes. X_{ai} denotes the ith input node.

Hence, the output of hidden layer node can be expressed as

$$h_{aj}(t) = \left(1 - e^{-u_{aj}(t)}\right) / \left(1 + e^{-u_{aj}(t)}\right) \tag{13}$$

Finally, we can obtain the output of output node

$$A(t) = \sum_{j=1}^{HN} w_{aj}(t) * h_{aj}(t) \tag{14}$$

wherein w_{aj} denotes the weights between the jth hidden node and output node.

The energy function of neural network is defined as Equation (15) [13]. The weight of neural network is regulated by gradient descent method.

$$E_a = \frac{1}{2}\left[V(t)\right]^2 \tag{15}$$

4. **Simulation Studies.** In this section, the simulation studies are presented to demonstrate the effectiveness of the proposed actor-critic RL algorithm for ship course control. In RL, the purpose of the agent is formalized in terms of a special reward signal passing from the environment to the agent. Meanwhile, the use of a reward signal formalizing the idea of a goal is one of its most distinctive features [8]. The purpose of ship course control is to apply rudder to ship tracking the order course ψ_d. The reinforcement signal $r_e(t)$ is defined as

$$r_e(t) = \begin{cases} 0, & |e_\psi(t)| \le \varepsilon \text{ or } |e_\psi(t)| \le |e_\psi(t-1)| \\ -1, & \text{otherwise} \end{cases} \tag{16}$$

wherein ε is the tolerance error band, e_ψ is the error between real course ψ and order course ψ_d, $e_\psi = \psi - \psi_d$. The definition of reinforcement signal can both reflect the quality of the current action and compare the current action with neighboring action.

In these simulations, the considered vessel is a training ship called "Yulong", which belongs to Dalian Maritime University. The detailed parameters can be obtained from [9], and by calculation we obtain $K = 0.478$, $T = 216$. The structure of actor network consists of a layer of two input nodes, a layer of six hidden nodes, and a final output node; the inputs to the network are the error of course angle e_ψ and the heading rate r, and the output is the command rudder angle δ_E. The structure of critic network is similar to actor network except that the number of input nodes is three, while the inputs are e_ψ, r and δ_E.

Firstly, we train the actor-critic network without external disturbances. We initialize $\gamma = 0.9$, $\varepsilon = 1.5°$, $\psi_d = 0°$ and ψ is a random value between 0° and 90°. In each episode, a failure is considered to occur if the trials are more than 1000 steps, keeping ψ_d are more than 10000 steps, it is considered a success. Episodes will terminate when either failure or success; if success, only ψ is reset to random value, otherwise the system returns to initial state, and then this episode restarts. After training steps, the obtained control network is used for simulating the two ship course control experiments with either including or excluding disturbances, which are described in Equation (3). We also design a traditional PID course autopilot for the "Yulong", which is used to compare with the proposed RL algorithm. The PID autopilot is also used for simulating another two experiments as same as the RL algorithm, and the parameters of PID are kept unchanged. Then, initialize the system states $\psi = 0°$, $r = 0°/\text{s}$, $\delta = 0°$ and set a square wave with 500s period and 30° amplitude as the order course signal. The four compared experimental results are shown in Figure 4 and Figure 5. Figure 4 shows that the PID autopilot can accurately track order course signal without disturbances, but it has an apparent deviation with disturbances. However, from Figure 4, we can see that the RL autopilot, which is almost not influenced

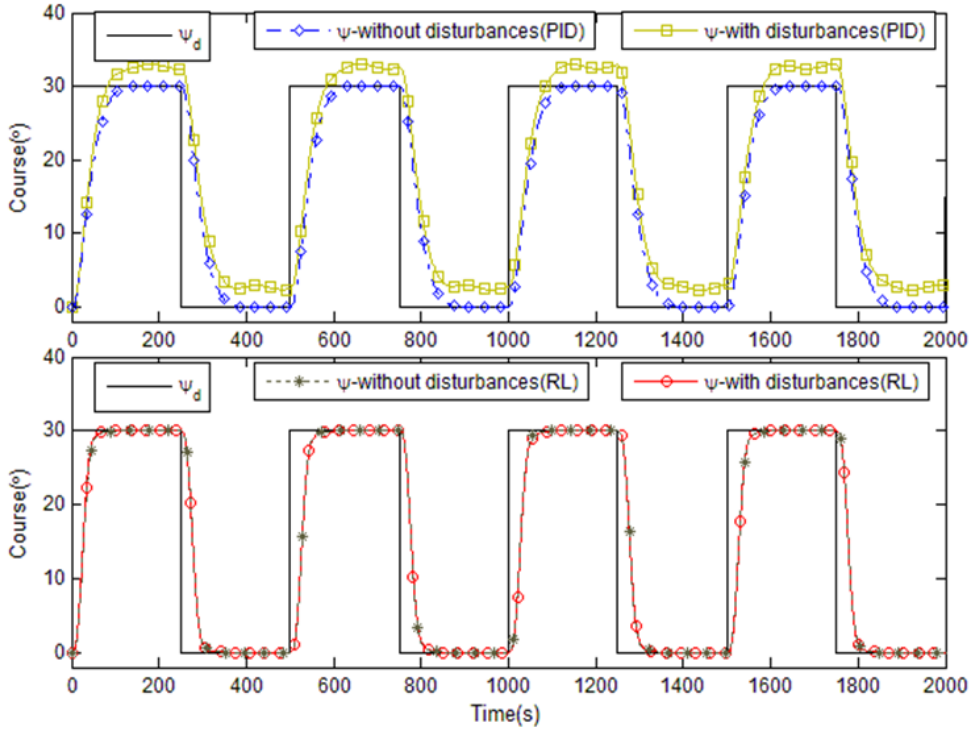

FIGURE 4. Real course and order course in RL autopilot and PID autopilot

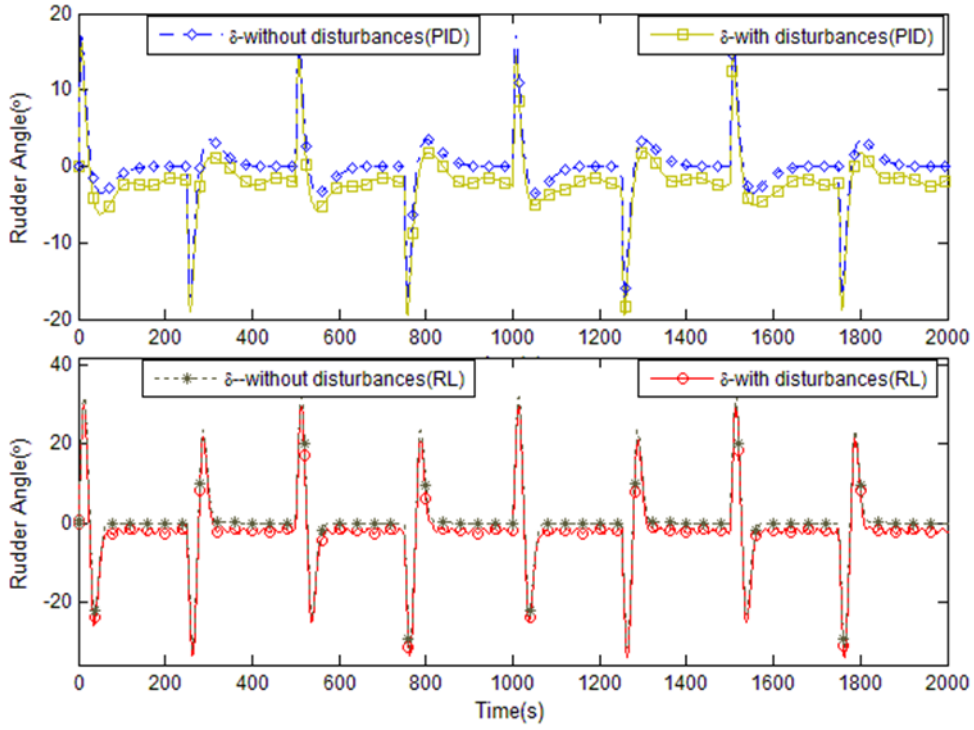

FIGURE 5. Real rudder angle of RL autopilot and PID autopilot

by disturbances, can accurately track and keep order course signal without overshoot, and the responses of ship course are smoother and quicker than the PID autopilot. From Figure 5, we can see that the responses of rudder are satisfied without excess actions in the RL autopilot which are of great benefit to reduce energy consumption. Figure 5 also shows that the RL autopilot overcomes the disturbances by changing small rudder angle periodically, but the PID autopilot does not make effective steering to overcome the

disturbances. The simulation results apparently indicate that the RL autopilot obtained a satisfied control performance.

5. **Conclusion.** In summary, the proposed RL method can be used for complex systems, such as ship course control system which is extremely difficult to establish the accurate model and has strong disturbances. By trial-and-error learning, the method can gradually adapt to the environment and the weights of BP network are updated; thus, we can obtain the optimal control policy. In the simulations of ship course control, we trained the actor-critic network firstly and then the obtained actor-critic network is used for simulating the two compared experiments with or without disturbances. A traditional PID course autopilot is also designed to compare with the proposed RL method. Finally, the simulation results demonstrate that the RL autopilot, which is almost not influenced by disturbances, can accurately and rapidly track order course signal without overshoot and excess actions. Obviously, its performance is better than the traditional PID autopilot. Furthermore, the proposed RL method shows good tracking performance and strong robustness. A limitation of the method is required to train neural network offline. Our future efforts will be made to extend our method to online learning algorithm.

Acknowledgment. This work is supported by the National Natural Science Foundation of China (Grant Nos. 61374114, 51579024 and 51179019), the Applied Basic Research Program of Ministry of Transport of P. R. China (Grant Nos. 2011-329-225-390 and 2013-329-225-270), the Doctoral Research Foundation of Liaoning Province (Grant No. 20141102), Fundamental Research Funds for the Central Universities (Grant Nos. 3132014321 and 3132015001). The authors also gratefully acknowledge the valuable comments and suggestions of the anonymous reviewers, which have improved this paper.

REFERENCES

[1] M. Breivik and T. I. Fossen, Path following of straight lines and circles for marine surface vessels, *Proc. of IFAC Conf. Contr. Appl. Marine Systems*, Ancona, Italy, 2004.

[2] T. I. Fossen, *Guidance and Control of Ocean Vehicles*, Wiley, New York, 1994.

[3] J. F. Li, T. S. Li, Z. Z. Fan, R. X. Bu, Q. Li and J. Q. Hu, Direct adaptive NN control of ship course autopilot with input saturation, *The 4th International Workshop on Advanced Computational Intelligence*, pp.655-661, 2011.

[4] K. W. Yu, R. C. Hwang and J. G. Hsieh, Automatic ship handling of the maritime search mission using a self-tuning fuzzy gain scheduling PD controller, *Journal of Navigation*, vol.52, no.3, pp.378-387, 1999.

[5] R. H. Li, T. S. Li, Q. L. Zheng and Q. Li, Ship tracking control based on linear active disturbance rejection control, *The 3rd International Conference on Intelligent Control and Information Processing*, pp.201-205, 2012.

[6] J. Kober and J. Peters, Reinforcement learning in robotics: A survey, *Springer Tracts in Advanced Robotics*, vol.32, no.11, pp.1238-1274, 2014.

[7] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, G. Ostrovski et al., Human-level control through deep reinforcement learning, *Nature*, vol.518, no.7540, pp.529-533, 2015.

[8] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd Edition, MIT Press, 2015.

[9] X. L. Jia and Y. S. Yang, *Ship Motion Mathematical Model*, Dalian Maritime University, Dalian, 1999.

[10] J. Q. Hu, *Clonal Selection Optimization Based Adaptive Control for Ship Steering*, Ph.D. Thesis, Dalian Maritime University, 2008.

[11] R. S. Sutton, Learning to predict by the methods of temporal differences, *Machine Learning*, vol.3, no.1, pp.9-44, 1988.

[12] R. X. Wang, L. Sun and X. G. Ruan, Reinforcement learning based on internally recurrent net, *Control Engineering of China*, vol.12, no.2. pp.138-140, 2005.

[13] J. Si and Y. T. Wang, Online learning control by association and reinforcement, *IEEE Trans. Neural Networks*, vol.12, no.2, pp.264-276, 2001.