网络首发地址：https://kns.cnki.net/kcms/detail/42.1755.TJ.20210114.1026.003.html 期刊网址：www.ship-research.com

引用格式：祝亢, 黄珍, 王绪明. 基于深度强化学习的智能船舶航迹跟踪控制 [J]. 中国舰船研究, 2021, 16(1): 105–113.
ZHU K, HUANG Z, WANG . Tracking control of intelligent ship based on deep reinforcement learning[J]. Chinese Journal of Ship Research, 2021, 16(1): 105–113.

基于深度强化学习的智能船舶航迹跟踪控制

扫码阅读全文

祝亢[1], 黄珍[*1], 王绪明[2]

1 武汉理工大学 自动化学院, 湖北 武汉 430070

2 武汉理工大学 智能交通系统研究中心, 湖北 武汉 430063

摘　要:［**目的**］智能船舶的航迹跟踪控制问题往往面临着控制环境复杂、控制器稳定性不高以及大量的算法计算等问题。为实现对航迹跟踪的精准控制，提出一种引入深度强化学习技术的航向控制器。［**方法**］首先，结合视线（LOS）算法制导，以船舶的操纵特性和控制要求为基础，将航迹跟踪问题建模成马尔可夫决策过程，设计其状态空间、动作空间、奖励函数；然后，使用深度确定性策略梯度（DDPG）算法作为控制器的实现，采用离线学习方法对控制器进行训练；最后，将训练完成的控制器与 BP-PID 控制器进行对比研究，分析控制效果。［**结果**］仿真结果表明，设计的深度强化学习控制器可以从训练学习过程中快速收敛达到控制要求，训练后的网络与 BP-PID 控制器相比跟踪迅速，具有偏航误差小、舵角变化频率小等优点。［**结论**］研究成果可为智能船舶航迹跟踪控制提供参考。

关键词: 智能船舶；航迹跟踪控制；深度强化学习；视线导航法

中图分类号: U664.82　　**文献标志码:** A　　**DOI:** 10.19693/j.issn.1673-3185.01940

Tracking control of intelligent ship based on deep reinforcement learning

ZHU Kang[1], *HUANG Zhen*[*1], *WANG Xuming*[2]

1 School of Automation, Wuhan University of Technology, Wuhan 430070, China

2 Intelligent Transport System Research Center, Wuhan University of Technology, Wuhan 430063, China

Abstract: ［**Objectives**］The tracking control of intelligent ships often faces the problem of low controller stability in complex control environments and manual algorithmic computing. In order to achieve precise tracking control, this paper proposes a controller based on deep reinforcement learning (DRL). ［**Methods**］Guided by the line-of-sight (LOS) algorithm and based on the maneuvering characteristics and control requirements of ships, this paper formulates a path of Markov decision processes by following the control problem, designing its state space, action space and reward by applying a deep deterministic policy gradient (DDPG) algorithm to implement the controller. An off-line learning method was used to train the controller. After the training, a comparison was made with BP-PID control to analyze the control effects. ［**Results**］Simulation results show that the deep reinforcement learning (DRL) controller can rapidly converge from the training process to meet the control requirements, with the advantages of small yaw error, and a visible reduction in the frequency of changes of the rudder angle. ［**Conclusions**］The study results can provide a reference for the tracking control of intelligent ships.

Key words: intelligent ships; tracking control; deep reinforcement learning (DRL); line-of-sight algorithm

收稿日期: 2020–04–29　　**修回日期:** 2020–07–06　　**网络首发时间:** 2021–01–15 09:50

基金项目: 国家重点研发计划资助项目（2018YFB1601500）

作者简介: 祝亢, 女, 1998 年生, 硕士生。研究方向：船舶智能控制及其应用。E-mail: zk13972793427@163.com
黄珍, 女, 1974 年生, 博士, 教授。研究方向：智能控制理论及其应用研究。E-mail: h-zhen@163.com
王绪明, 男, 1964 年生, 博士, 研究员。研究方向：船舶智能化。E-mail: ted@whut.edu.cn

***通信作者:** 黄珍

0 引 言

目前，国内外对运载工具的研究正朝着智能化、无人化方向发展，智能船舶技术受到全球造船界与航运界的广泛关注。其以实现船舶航行环境的智能化、自主化发展为目标，深度融合传统船舶设计与制造技术以及现代信息通信与人工智能技术，包含智能航行、智能船用设备、智能船舶测试等多方面的研究[1]。其中，智能航行技术一直是保障船舶顺利完成货物运输、通信救助等任务的重要基础。要使船舶在面对多种复杂水域干扰的情况下也能遵守正常的通航秩序，安全地执行任务且保证完成效果，采取有效的控制手段精确进行航迹跟踪就显得尤为重要。

针对航迹跟踪的研究任务可以分为制导和控制 2 个方面。在制导方面，常由视线（line-of-sight, LOS）算法将路径跟踪问题转换为方便处理的动态误差控制问题；在控制方面，基于船舶的复杂非线性系统，常考虑使用 PID 等无模型控制方法，或采用模型线性化的方法来解决非线性模型在计算速率方面存在的问题。但对于复杂的环境，传统 PID 控制器不仅参数复杂，还不具备自适应学习能力。而最优控制、反馈线性化一类的控制算法通常需要建立精确的模型才能获得较高的控制精度。滑模控制虽然对模型精度要求不高，但其抖振问题难以消除[2]。即使存在一些自适应参数调节方法，如通过估计系统输出实现 PID 参数自整定的自适应 PID 控制方法，也会由于模型的不确定性和外界扰动，存在系统输出与真实输出的偏差[3]，又或者存在参数寻优时间过长的问题而影响控制的实时性。对于与模糊逻辑相结合的响应速度快、实时性好的 PID 自适应控制器[4]，其控制精度依赖于复杂的模糊规则库，致使整体计算复杂。

考虑到船舶的复杂非线性系统模型，和保障航迹跟踪控制的实时性时产生的大量参数整定和复杂计算等问题，本文将采用深度强化学习算法来研究智能船舶的轨迹跟踪问题。深度强化学习（deep reinforcement learning, DRL）是深度学习与强化学习的结合，其通过强化学习与环境探索得到优化的目标，而深度学习则给出运行的机制用于表征问题和解决问题。深度强化学习算法不依赖动力学模型和环境模型，不需要进行大量的算法计算，还具备自学习能力。Magalhães 等[5] 基于强化学习算法，使用 Q-learning 设计了一种监督开关器并应用到了无人水面艇，它能智能地切换控制器从而使无人艇的行驶状态符合多种环境与机动要求。2015 年，Mnih 等[6] 为解决复杂强化学习的稳定性问题，将强化学习与深度神经网络相结合，提出了深度 Q 学习（deep Q network, DQN）算法，该算法的提出代表了深度强化学习时代的到来。之后，在欠驱动无人驾驶船舶的航行避碰中也进行了相关应用[7]。

面对存在的大量参数整定、复杂算法计算等问题，为实现船舶航迹跟踪的精准控制，本文拟设计一种基于深度确定性策略梯度算法（deep deterministic policy gradient, DDPG）的深度强化学习航迹跟踪控制器，在 LOS 算法制导的基础上，对船舶航向进行控制以达到航迹跟踪效果。然后，根据实际船舶的操纵特性以及控制要求，将船舶路径跟踪问题建模成马尔可夫决策过程，设计相应的状态空间、动作空间与奖励函数，并采用离线学习方法对控制器进行学习训练。最后，通过仿真实验验证深度强化学习航迹控制器算法的有效性，并与 BP-PID 控制器算法的控制效果进行对比分析。

1 智能船舶航迹跟踪控制系统总体设计

1.1 LOS 算法制导

航迹跟踪控制系统包括制导和控制 2 个部分，其中制导部分一般是根据航迹信息和船舶当前状态确定所需的设定航向角值来进行工作。本文使用的 LOS 算法已被广泛运用于路径控制。LOS 算法可以在模型参数不确定的情况下，以及在复杂的操纵环境中与控制器结合，从而实现对模型的跟踪控制。视线法的导航原理是基于可变的半径与路径点附近生成的最小圆来产生期望航向，即 LOS 角。经过适当的控制，使当前船舶的航向与 LOS 角一致，即能达到航迹跟踪的效果[8]。

LOS 算法示意图如图 1 所示。假设当前跟踪路径点为$P_{k+1}(x_{k+1},y_{k+1})$，上一路径点为$P_k(x_k,y_k)$，以船舶所在位置$P_s(x_s,y_s)$为圆心，选择半径R_{Los}与路径P_kP_{k+1}相交，选取与P_{k+1}相近的点$P_{\mathrm{Los}}(x_{\mathrm{Los}},y_{\mathrm{Los}})$作为 LOS 点，当前船舶坐标到 LOS 点的方向矢量与x_0的夹角ψ_{Los}则为需要跟踪的 LOS 角。图中：d为当前船舶至跟踪路径的最短距离；ψ为当前航向角。

其中，半径R_{Los}的计算公式如式（1）和式（2）所示，为避免$R_{\min}$的计算出现零值，在最终的计算中加入了 2 倍的船长L_{pp}来进行处理[9]。

$$\begin{cases} a(t)=\sqrt{(x(t)-x_k)^2+(y(t)-y_k)^2} \\ b(t)=\sqrt{(x_{k+1}-x(t))^2+(y(t)-y_{k+1})^2} \\ c(t)=\sqrt{(x_{k+1}-x_k)^2+y_{k+1}-y_k)^2} \\ R_{\min}(t)=\sqrt{a(t)^2-\left(\dfrac{c(t)^2-b(t)^2+a(t)^2}{2c(t)}\right)^2} \end{cases} \tag{1}$$

$$R_{\text{Los}}=R_{\min}(t)+2L_{\text{pp}} \tag{2}$$

式中, 所计算的$R_{\min}$即为当前时刻 t 的航迹误差 ε, 也即图 1 中的d。

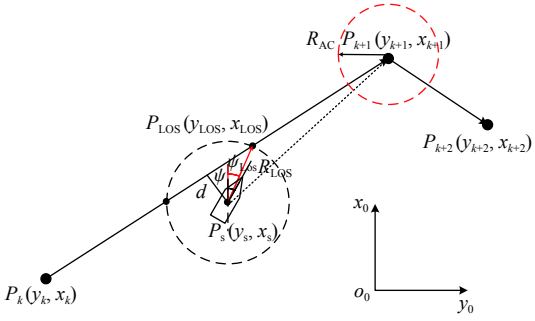


图 1　LOS 导航原理图

Fig. 1　Schematic diagram of LOS algorithm

船舶在沿着路径进行跟踪时, 若进入下一个航向点的一定范围内, 即以$P_{k+2}(x_{k+2},y_{k+2})$为圆心、R_{AC}为半径的接受圆内, 则更新当前航向点为下一航向点, 半径R_{AC}一般选取为 2 倍船长。

1.2　基于强化学习的控制过程设计

强化学习(reinforcement learning, RL)与深度学习同属机器学习范畴, 是机器学习的一个重要分支, 主要用来解决连续决策的问题, 是马尔可夫决策过程(Markov decision processes, MDP)问题[10]的一类重要解决方法。

此类问题均可模型化为 MDP 问题, 简单表示为四元组$<\boldsymbol{S},\boldsymbol{A},\boldsymbol{P},R>$。其中, $\boldsymbol{S}$为所有状态值的集合, 即状态空间; $\boldsymbol{A}$为动作值集合的动作空间; $\boldsymbol{P}$为状态转移概率矩阵, 即在t时刻状态为$\boldsymbol{S}_t=\boldsymbol{s}$的情况下选择动作值为$\boldsymbol{A}_t=\boldsymbol{a}$, 则$t+1$时刻产生状态为$\boldsymbol{s}_1$的概率$P^a_{ss_1}=\boldsymbol{P}[\boldsymbol{S}_{t+1}=\boldsymbol{s}_1|\boldsymbol{S}_t=\boldsymbol{s},\boldsymbol{A}_t=\boldsymbol{a}]$; $R=r(\boldsymbol{s},\boldsymbol{a})$为回报奖励函数, 用于评价在$\boldsymbol{s}$状态下选取动作值$\boldsymbol{a}$的好坏。航迹跟踪控制系统中的控制部分用 MDP 模型表示如图 2 所示。

如图 2 所示, 船舶智能体直接与当前控制环境进行交互而且不需要提前获取任何信息。在训练过程中, 船舶采取动作值$\boldsymbol{a}_t$与环境进行交互更新自己的状态$\boldsymbol{s}_t\rightarrow\boldsymbol{s}_{t+1}$, 并获得相应的奖励$r_{t+1}$, 之后, 继续采取下一动作与环境交互。在此过程中, 会产生大量的数据, 利用这些数据学习优化

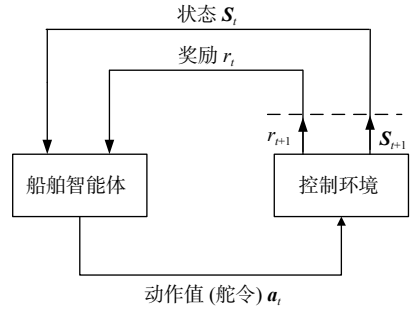


图 2　船舶控制的 MDP 模型

Fig. 2　MDP model of ship control

自身选择动作的策略 policy π。简单而言, 这是一个循环迭代的过程。在强化学习中, 训练的目标是找到一个最佳的控制策略 policy $\pi*$, 以使累积回报值R_t达到最大[11]。在下面的公式中, γ为折扣系数, 用来衡量未来回报在当前时期的价值比例, 设定$\gamma\in[0,1]$。

$$R_t=r_t+\gamma r_{t+1}+\gamma^2 r_{t+2}+\cdots=\sum_{k=1}^{\infty}\gamma^k r_{t+k+1} \tag{3}$$

Policy π可以使用 2 种值函数进行评估: 状态值函数$V^\pi(\boldsymbol{s}_t)$和动作值函数$Q^\pi(\boldsymbol{s}_t,\boldsymbol{a}_t)$。其中$V^\pi(\boldsymbol{s}_t)$为在遵循当前策略的状态下对累积回报值的期望, E为期望值; 类似地, $Q^\pi(\boldsymbol{s}_t,\boldsymbol{a}_t)$表示基于特定状态和动作情况$(\boldsymbol{s}_t,\boldsymbol{a}_t)$下对累积回报值的期望。

$$V^\pi(\boldsymbol{s}_t)=E_\pi[R_t|\boldsymbol{s}_t]=E_\pi\left[\sum_{k=1}^{\infty}\gamma^k r_{t+k+1}|\boldsymbol{s}_t\right] \tag{4}$$

$$Q^\pi(\boldsymbol{s}_t,\boldsymbol{a}_t)=E_\pi[R_t|\boldsymbol{s}_t,\boldsymbol{a}_t]=E_\pi\left[\sum_{k=1}^{\infty}\gamma^k r_{t+k+1}|\boldsymbol{s}_t,\boldsymbol{a}_t\right] \tag{5}$$

根据值函数和上述最佳控制策略 policy π^*的定义, 最佳 policy π^*总是满足以下条件:

$$\pi^*=\arg\max V^\pi(\boldsymbol{s}_t)=\arg\max Q^\pi(\boldsymbol{s}_t,\boldsymbol{a}_t) \tag{6}$$

1.3　航迹跟踪问题马尔可夫建模

从以上描述可以看出, 在基于强化学习的控制设计中, 马尔可夫建模过程的组件设计是最为关键的过程, 状态空间、动作空间和奖励的正确性对算法性能和收敛速度的影响很大。所以针对智能船舶的轨迹跟踪问题, 对其进行马尔可夫建模设计。

1) 状态空间设计。

根据制导采用的 LOS 算法, 要求当前航向角根据 LOS 角进行调节以达到跟踪效果。所以在选取状态时, 需考虑 LOS 算法中的输出参数, 包括目标航向ψ_{LOS}与实际航向ψ的差值e、航迹误差ε, 以及与航迹点距离误差ε^{d}。

对于船舶模型，每个时刻都可以获得当前船舶的纵荡速度u、横荡速度v、艏转向速度r和舵角δ。为使强化学习能实现高精度跟踪效果，快速适应多种环境的变换，除了选取当前时刻的状态值外，还加入了上一时刻的状态值进行比较，以及当前航向误差与上一时刻航向误差的差值$e(k-1)$，使当前状态能够更好地表示船舶是否在往误差变小的方向运行。最终，当前时刻t的状态空间可设计为

$$\begin{aligned}\boldsymbol{s}_t = [&e_t,\ \varepsilon_t,\ \varepsilon_t^{\mathrm{d}},\ u_t,\ v_t,\ r_t,\ \delta_t,\ e(k-1)_t,\\ &e_{t-1},\ \varepsilon_{t-1},\ \varepsilon_{t-1}^{\mathrm{d}},\ u_{t-1},\ v_{t-1},\ r_{t-1},\ \delta_{t-1}]\end{aligned} \tag{7}$$

2）动作空间设计。

针对航迹跟踪任务特点，以及 LOS 制导算法的原理，本文将重点研究对船舶航向，即舵角的控制，不考虑对船速与桨速的控制。动作空间只有舵令一个动作值，即δ，其值的选取需要根据实际船舶的控制要求进行约束，设定为在$(-35°,35°)$以内，最大舵速为 15.8 (°)/s。

3）奖励函数设计。

本文期望航向角越靠近 LOS 角奖励值越高，与目标航迹的误差越小奖励值越高。因此，设计的奖励函数为普遍形式，即分段函数：

$$r_t=\begin{cases}0, & \text{if } |e|\leqslant 0.1 \text{ rad}\\ -|e|-0.1\,|e(k-1)|-0.01\,|\varepsilon|, & \text{if } |e|>0.1 \text{ rad}\end{cases} \tag{8}$$

式中，$e(k-1)$为当前航向误差与上一时刻航向误差的差值。当差值大于0.1 rad时选择负值奖励，也可称之为惩罚值，是希望训练网络能尽快改变当前不佳的状态。将负值的选取与另一分段的0奖励值做明显对比，使其训练学习后可以更加快速地选择奖励值高的动作，从而达到最优效果。

1.4 控制系统总体方案

基于强化学习的智能船舶航迹控制系统总体框架如图 3 所示。LOS 算法根据船舶当前位置计算得到需要的航向以及航迹误差，在与船舶的状态信息整合成上述所示状态向量$\boldsymbol{s}_t$后输入进航迹控制器中，然后根据强化学习算法输出当前最优动作值$\boldsymbol{a}_t$给船舶执行，同时通过奖励函数r_t计算获得相应的奖励来进行自身参数迭代，以使航迹控制器具备自学习能力。

在将控制器投入实时控制之前，首先需要对控制器进行离线训练。设定规定次数的训练后，将获得的使累计回报值达到最大的网络参数进行存储整合，由此得到强化学习控制器，并应用于航迹跟踪的实时控制系统。

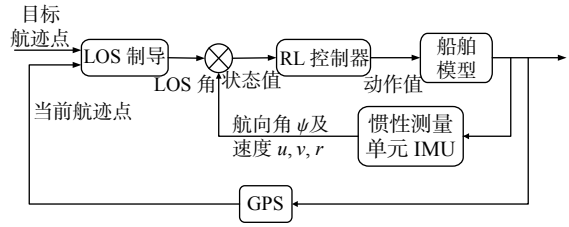


图 3 基于强化学习的智能船舶轨迹跟踪控制框图

Fig. 3 Block diagram of intelligent ship tracking control based on RL

要解决强化学习问题，目前有许多的算法、机制和网络结构可供选择，但这些方法都缺少可扩展的能力，并且仅限于处理低维问题。为此，Mnih 等[6]提出了一种可在强化学习问题中使用大规模神经网络的训练方法——DQN 算法，该算法成功结合了深度学习与强化学习，使强化学习也可以扩展处理一些高维状态、动作空间下的决策问题[12]。DQN 算法可解决因强化学习过程与神经网络逼近器对值函数逼近的训练相互干扰，而导致学习结果不稳定甚至是产生分歧的问题[13]，是深度强化学习领域的开创者。

DQN 算法显著提高了复杂强化学习问题的稳定性和性能，但因其使用的是离散的动作空间，故需要对输出的动作进行离散化，且只能从有限的动作值中选择最佳动作。对于船舶的轨迹跟踪问题，如果候选动作数量太少，就很难对智能体进行精确控制。为使算法满足船舶的操纵特性与要求，本文选择了一种适用于连续动作空间的深度强化学习算法，即基于 DDPG 的算法[14]来对智能船舶航迹跟踪控制器进行设计，该算法不仅可以在连续动作空间上进行操作，还可以高效精准地处理大量数据。

2 基于 DDPG 算法的控制器设计

2.1 DDPG 算法原理

DDPG 是 Lillicrap 等[14]将 DQN 算法应用于连续动作中而提出的一种基于确定性策略梯度的 Actor-Critic 框架无模型算法。DDPG 的基本框架如图 4 所示。

网络整体采用了 Actor-Critic 形式，同时具备基于值函数的神经网络和基于策略梯度的神经网络：Actor 网络的θ^{π}表示确定性策略函数$\boldsymbol{a}=\pi(\boldsymbol{s}|\theta^{\pi})$，Critic 网络的$\theta^{Q}$表示值函数$Q(\boldsymbol{s},\boldsymbol{a}|\theta^{Q})$。并且 DDPG 还借鉴了 DQN 技术，其通过采取经验池回放机制 (experience replay) 以及单独的目标网络来消除大规模神经网络带来的不稳定性。

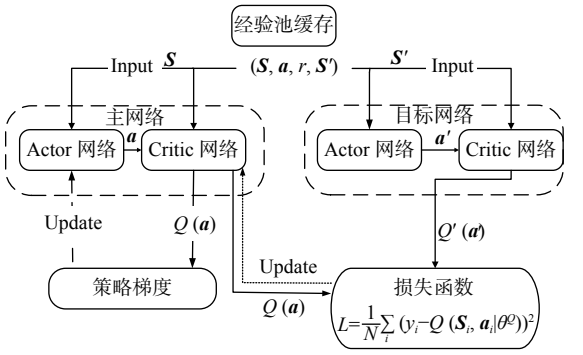


图 4　DDPG 基本框架

Fig. 4　Block diagram of DDPG

所谓经验池回放机制, 即在每个时间点都存储当前状态、动作等信息作为智能体的经验$\boldsymbol{e}_t=(\boldsymbol{s}_t,\boldsymbol{a}_t,r_t,\boldsymbol{s}_{t+1})$, 以此形成回放记忆序列$\boldsymbol{D}=\{e_1,\cdots,e_N\}$。在训练网络时, 从中随机提取 mini batch 数量的经验数据作为训练样本, 但重复使用历史数据的操作会增加数据的使用率, 也打乱了原始数据的顺序, 会降低数据之间的关联性。而目标网络则建立了 2 个结构一样的神经网络——用于更新神经网络参数的主网络和用于产生优化目标值的目标网络, 初始时, 将主网络参数赋予给目标网络, 然后主网络参数不断更新, 目标网络不变, 经过一段时间后, 再将主网络的参数赋予给目标网络。此循环操作可使优化目标值在一段时间内稳定不变, 从而使得算法性能更加稳定。

在训练过程中, 主网络中的 Actor 网络根据从经验池中随机选取的样本状态$\boldsymbol{s}$, 经过当前策略函数$\boldsymbol{a}=\pi(\boldsymbol{s}|\theta^{\pi})$选择出最优的动作值$\boldsymbol{a}$交予船舶智能体, 让其与环境交互后得到下一时刻的状态值$\boldsymbol{s}'$。而此时的 Critic 网络则接受当前的状态$\boldsymbol{s}$和动作值$\boldsymbol{a}$, 使用值函数$Q(\boldsymbol{s},\boldsymbol{a}|\theta^{Q})$评价当前状态的期望累计奖赏, 并用于更新 Actor 网络的参数。在目标网络中, 整体接收下一时刻的状态$\boldsymbol{s}'$, 经目标 Actor 网络选出动作后交予目标 Critic 获得目标期望值$Q'(\boldsymbol{a}')$, 然后, 再通过计算损失函数对主网络的 Critic 网络参数进行更新。对于主网络的 Actor 网络参数更新, Silver 等[15] 证实, 确定性策略的目标函数$J(\theta^{\pi})$采用π策略的梯度与 Q 函数采用π策略的期望梯度是等价的:

$$\frac{\partial J(\theta^{\pi})}{\partial\theta^{\pi}}=E_{\mathrm{s}}\left[\frac{\partial Q(\boldsymbol{s},\boldsymbol{a}|\theta^{Q})}{\partial\theta^{\pi}}\right] \tag{9}$$

根据确定性策略$\boldsymbol{a}=\pi(\boldsymbol{s}|\theta^{\pi})$, 得到 Actor 网络的梯度为:

$$\frac{\partial J(\theta^{\pi})}{\partial\theta^{\pi}}=E_{\mathrm{s}}\left[\frac{\partial Q(\boldsymbol{s},\boldsymbol{a}|\theta^{Q})}{\partial\boldsymbol{a}}\frac{\partial\pi(\boldsymbol{s}|\theta^{\pi})}{\partial\theta^{\pi}}\right] \tag{10}$$

$$\nabla_{\theta^{\pi}}J\approx\frac{1}{N}\sum_i(\nabla_{\boldsymbol{a}}Q(\boldsymbol{s},\boldsymbol{a}|\theta^{\pi})|_{\boldsymbol{s}=\boldsymbol{s}_i,\boldsymbol{a}=\pi(\boldsymbol{s}_i)}\cdot\nabla_{\theta^{\pi}}\pi(\boldsymbol{s}|\theta^{\pi})|_{\boldsymbol{s}=\boldsymbol{s}_i}) \tag{11}$$

另一方面, 对于 Critic 网络中的价值梯度:

$$\frac{\partial L(\theta^{Q})}{\partial\theta^{Q}}=E_{\boldsymbol{s},\boldsymbol{a},r,\boldsymbol{s}'\sim\boldsymbol{D}}\left[(\mathrm{Target}Q-Q(\boldsymbol{s},\boldsymbol{a}|\theta^{Q}))\frac{\partial Q(\boldsymbol{s},\boldsymbol{a}|\theta^{Q})}{\partial\theta^{Q}}\right] \tag{12}$$

$$\mathrm{Target}Q=r+\gamma Q'(\boldsymbol{s}',\pi(\boldsymbol{s}'|\theta^{\pi'})|\theta^{Q'}) \tag{13}$$

式中, $\theta^{\pi'}$和$\theta^{Q'}$分别为目标策略网络和目标值函数网络的网络参数。其中, 目标网络的更新方法与 DQN 算法中的不同, 在 DDPG 算法中, Actor-Critic 网络各自的目标网络参数是通过缓慢的变换方式更新, 也叫软更新。以此方式进一步增加学习过程的稳定性:

$$\theta^{Q'}=\tau\theta^{Q}+(1-\tau)\theta^{Q'} \tag{14}$$

$$\theta^{\pi'}=\tau\theta^{\pi}+(1-\tau)\theta^{\pi'} \tag{15}$$

式中, τ为学习率。

定义最小化损失函数来更新 Critic 网络参数, 其中, y_i为当前时刻状态动作估计值函数与目标网络得到的目标期望值间的误差:

$$L=\frac{1}{N}\sum_i(y_i-Q(\boldsymbol{s}_i,\boldsymbol{a}_i|\theta^{Q}))^2 \tag{16}$$

2.2　算法实现步骤

初始化 Actor-Critic 网络的参数, 将当前网络的参数赋予对应的目标网络; 设置经验池容量为 30 000 个, 软更新学习率为 0.01, 累计折扣系数设定为 0.9, 初始化经验池。训练的每回合步骤如下:

1) 初始化船舶环境;

2) 重复以下步骤直至到达设置的最大步长;

3) 在主网络中, Actor 网络获取此刻船舶的状态信息$\boldsymbol{s}_t$, 并根据当前的策略选取动作舵令δ_t给船舶执行, 即$\delta_t=\pi(\boldsymbol{s}_t|\theta^{\pi})$;

4) 船舶执行当前舵令后输出奖励r_t和下一个状态$\boldsymbol{s}_{t+1}$, Actor 网络再次获取该状态信息并选取下一舵令δ_{t+1};

5) 将此过程中产生的数据$(\boldsymbol{s}_t,\delta_t,r_t,\boldsymbol{s}_{t+1})$存储在经验池中, 以作为网络训练学习的数据集。当经验池存储满后, 再从第 1 个位置循环存储;

6) 从经验池中随机采样 N 个样本$(\boldsymbol{s}_t,\delta_t,r_t,\boldsymbol{s}_{t+1})$, 作为当前 Actor 网络和 Critic 网络的训练数据;

7) 通过损失函数更新 Critic 网络, 根据 Actor 网络的策略梯度更新当前 Actor 网络, 然后再对目标网络进行相应的软更新。

3 系统仿真与算法对比分析

3.1 仿真环境构建

为验证上述方法的有效性，基于 Python 环境进行了船舶航迹跟踪仿真实现。控制研究对象模型选用文献 [16-17] 中的单桨单舵 7 m KVLCC2 船模，建模采用三自由度模型（即纵荡、横荡和艏摇），具体建模过程参考文献 [16]。表 1 列出了船舶的一些主要参数。

表 1 KVLCC2 船舶参数

Table 1 Parameters of a KVLCC2 tanker

参数	数值	参数	数值
船长L_{pp}/m	7	方形系数C_b	0.809 8
船宽B_{wl}/m	1.168 8	浮心坐标/m	0.244 0
型深D/m	0.656 3	螺旋桨直径D_p/m	0.216 0
排水体积/m³	3.272 4	舵面积/m²	0.053 9

在所选用的 DDPG 控制器中，Crtic 网络和 Actor 网络的实现参数设置分别如表 2 和表 3 所示。

表 2 Critic 网络参数

Table 2 Critic network parameters

参数	赋值
输入层	状态向量$\boldsymbol{S}(t)$
第1个隐层	300
第1层激活函数	Relu
第2个隐层	200
第2层激活函数	Relu
输出层	动作$\delta(t)$
输出层激活函数	Tanh
参数初始化	Xavier初始化
学习率	0.000 1
优化器	Adam

表 3 Actor 网络参数

Table 3 Actor network parameters

参数	赋值
输入层	状态向量$\boldsymbol{S}(t)$，动作$\delta(t)$
第1个隐层	300
第1层激活函数	Relu
第2个隐层	200
第2层激活函数	Relu
输出层	$Q(\boldsymbol{S}(i),\delta(i))$
输出层激活函数	Linear
参数初始化	Xavier初始化
学习率	0.001
优化器	Adam

3.2 控制器离线学习

基于 DDPG 算法进行的离线训练学习设置如下：初始化网络参数以及经验缓存池，设计最大的训练回合为 2 000，每回合最大步长为 500，采样时间为 1 s。在规划训练期间所需跟踪的航迹时，为使控制器适应多种环境，以及考虑到 LOS 制导算法中对于航向控制的要求，依据文献 [18] 中的设计思想，根据拐角的变换，设计了多条三航迹点航线，每回合训练时随机选取一条进行航迹跟踪。

训练时，将数据存入经验池中，然后再从中随机采样一组数据进行训练，状态值及动作值均进行归一化处理，当达到最大步长或最终航迹点输出完成时，便停止这一回合，并计算当前回合的总回报奖励。当训练进行到 200, 300 和 500 回合时，其航向误差如图 5 所示。由图中可以看出，在训练时随着回合的增加，航向误差显著减小，控制算法不断收敛；当训练达到最大回合结束后，总奖励值是不断增加的。为使图像显示得更加清晰，截取了 200~500 回合的总回报奖励如图 6 所示。从中可以看出，在约 270 回合时算法基本收敛，展现了快速学习的过程。

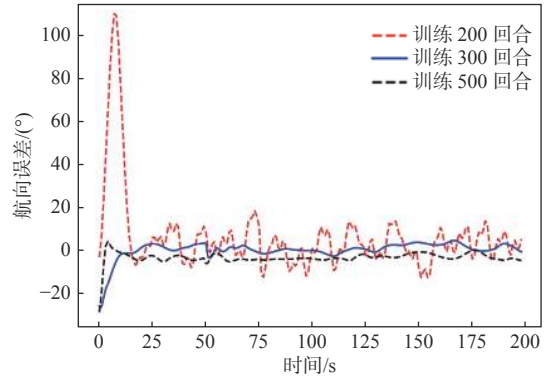


图 5 航向误差曲线

Fig. 5 Course error curves

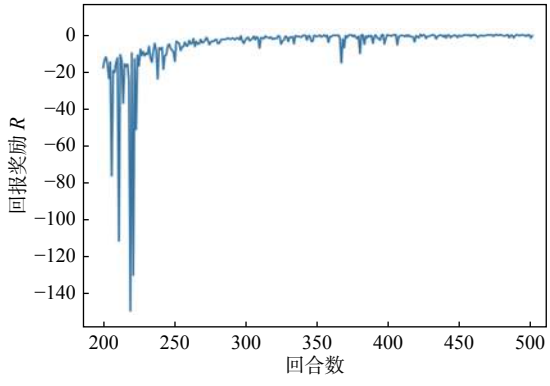


图 6 总回报奖励曲线

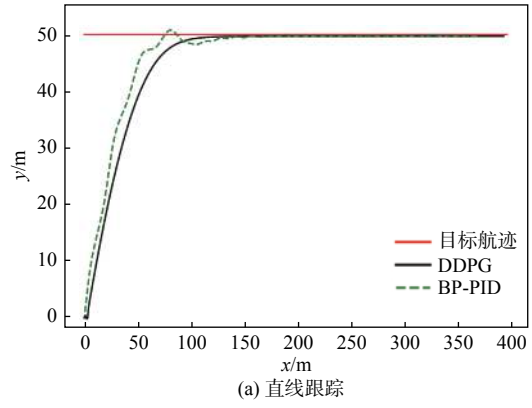
Fig. 6 Total reward curve

3.3　仿真实验设计及对比分析

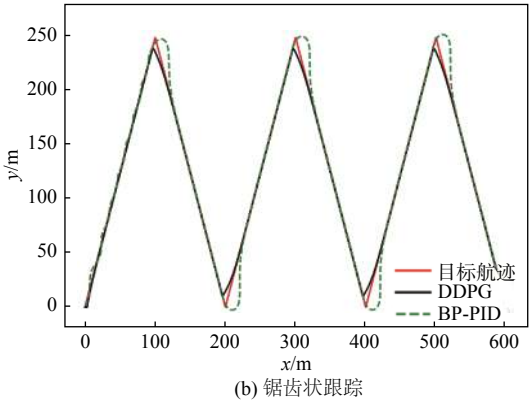
上述训练完成后, DDPG 控制器保存回报奖励函数最大的网络参数, 并将其应用于航迹跟踪仿真。为了验证 DDPG 控制器的可行性, 本文选用 BP-PID 控制器进行对比分析。

用于对比的 BP-PID 控制器选择使用输入层节点数为 4、隐含层节点数为 5、输出层节点数为 3 的 BP 神经网络对 PID 的 3 种参数进行选择, 其中学习率为 0.546, 动量因子为 0.79, 并参考文献 [19], 利用附加惯性项对神经网络进行优化。在相同的环境下, 将 DDPG 控制器与 BP-PID控制器进行仿真对比分析。仿真时, 船舶的初始状态为从原点 (0, 0) 出发, 初始航向为 45°, 初始航速也即纵荡速度u=1.179 m/s, 螺旋桨初始速度r=10.4 r/s。

仿真实验 1: 分别设计直线轨迹和锯齿状轨迹, 用以观察 2 种控制器对直线的跟踪效果和面对剧烈转角变化时的跟踪效果(图 7), 轨迹点坐标分别为 (0, 50), (400, 50) 和 (0, 0), (100, 250), (200, 0), (300, 250), (400, 0), (500, 250), (600, 0), 单位均为 m。



(a) 直线跟踪



(b) 锯齿状跟踪

图 7　航迹跟踪效果（实验 1）

Fig. 7　Tracking control result (experiment 1)

通过对 2 种类型轨迹跟踪的对比可以看出, 对于直线轨迹, DDPG 控制器能够更加快速地进行稳定跟踪, 在锯齿状轨迹转角跟踪时其效果也明显优于 BP-PID 控制器。对仿真过程中航向角的均方根误差(图 7(b))进行计算, 显示 BP-PID 控制器的数值达 61.017 8, 而 DDPG 控制器的仅为 10.018, 后者具有更加优秀的控制性能。

仿真实验 2: 为模拟传统船舶的航行轨迹, 设计轨迹点为 (0, 0), (100, 50), (150, 250), (400, 250), (450, 50), (550, 0) 的航迹进行跟踪。跟踪效果曲线和航向均方根误差(RMSE)的对比分别如图 8 和表 4 所示。

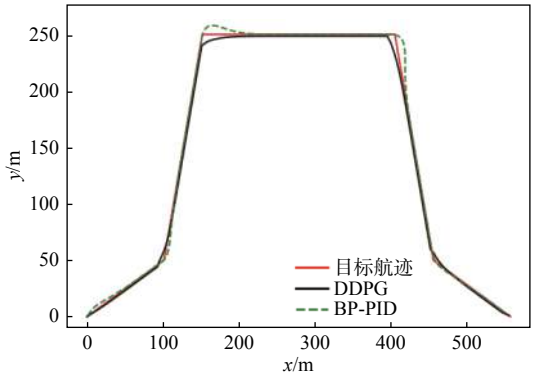


图 8　航迹跟踪结果（实验 2）

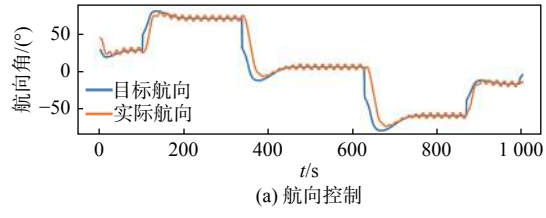
Fig. 8　Tracking control result (experiment 2)

表 4　控制性能指标

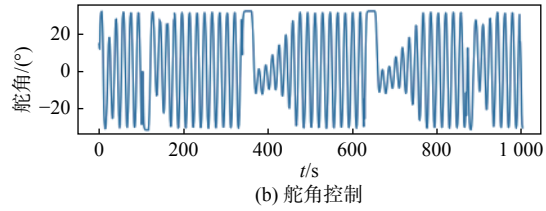
Table 4　Control performance

控制器	RMSE
BP-PID控制器	13.585 0
DDPG控制器	6.911 96

在此次仿真过程中, 进一步对比了 2 种控制器对于 LOS 角跟踪的效果以及舵角的变化频率, 结果分别如图 9 和图 10 所示。PID 经过 BP 神经网络参数整定后整体巡航时间约为 1 000 s, 而 DDPG 控制器的巡航时间则在此基础上缩短了 4%; 在转角处的航向跟踪中, DDPG 控制器在 20 s 内达到期望值, 而 BP-PID 的调节时间则约为 60 s, 且控制效果并不稳定, 舵角振动频率高。由此可见, 深度强化学习控制器可以很快地根据航迹变



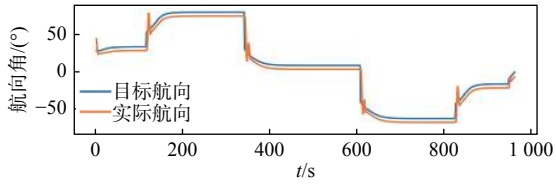
(a) 航向控制



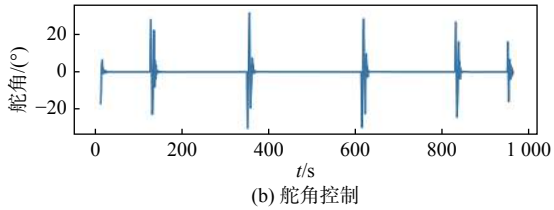
(b) 舵角控制

图 9　BP-PID 控制器控制效果

Fig. 9　Control result of BP-PID



(a) 航向控制



(b) 舵角控制

图 10 DDPG 控制器控制效果

Fig. 10 Control result of DDPG

化做出调整，减少了不必要的控制环节，调节时间短，控制效果稳定，舵角变化频率小，具有良好的控制性能。

4 结 语

本文针对船舶的航迹跟踪问题，提出了一种基于深度强化学习的航迹跟踪控制器设计思路。首先根据 LOS 算法制导，建立了航迹跟踪控制的马尔可夫模型，给出了基于 DDPG 控制器算法的程序实现；然后在 Python 环境中完成了船舶航迹跟踪控制系统仿真实验，并与 BP-PID 控制器进行了性能对比分析。

将航迹跟踪问题进行马尔可夫建模设计后，将控制器投入离线学习。通过对此过程的分析发现，DDPG 控制器在训练中能快速收敛达到控制要求，证明了设计的状态、动作空间以及奖励函数的可行性。并且航迹跟踪仿真对比结果也显示，DDPG 控制器能较快地应对航迹变化，控制效果稳定且舵角变化少，对于不同的轨迹要求适应性均相对良好。整体而言，基于深度强化学习的控制方法可以应用到船舶的航迹跟踪控制之中，在具有自适应稳定控制能力的情况下，不仅免去了复杂的控制计算，也保证了实时性，对船舶的智能控制具有一定的参考价值。

参考文献：

[1] 严新平, 刘佳仑, 范爱龙, 等. 智能船舶技术发展与趋势简述 [J]. 船舶工程, 2020, 42(3): 15–20.
YAN X P, LIU J L, FAN A L, et al. Development and trend of intelligent ship technology[J]. Ship Engineering, 2020, 42(3): 15–20 (in Chinese).

[2] 郭宝珠. 非线性系统的自抗扰控制引论 [J]. 数学建模及其应用, 2017, 6(1): 13–22, 52.
GUO B Z. An introduction to active disturbance rejection control for nonlinear systems[J]. Mathematical Modeling and its Applications, 2017, 6(1): 13–22, 52 (in Chinese).

[3] 张旋武, 谢磊, 初秀民, 等. 无人船路径跟随控制方法综述 [J]. 交通信息与安全, 2020, 38(1): 20–26.
ZHANG X W, XIE L, CHU X M, et al. An overview of path following control methods for unmanned surface vehicles[J]. Journal of Transport Information and Safety, 2020, 38(1): 20–26 (in Chinese).

[4] LIU S, XING B W, ZHU W L. A fusion fuzzy PID controller with real-time implementation on a ship course control system[C]//Proceedings of the 2015 23rd Mediterranean Conference on Control and Automation (MED). Torremolinos, Spain: IEEE, 2015.

[5] MAGALHÃES J, DAMAS B, LOBO V. Reinforcement learning: the application to autonomous biomimetic underwater vehicles control[J]. IOP Conference Series: Earth and Environmental Science, 2018, 172: 12–19.

[6] MNIH V, KAVUKCUOGLU K, SILVER D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540): 529–533.

[7] WOO J, KIM N. Vector field based guidance method for docking of an unmanned surface vehicle[C]//Proceedings of the 12th ISOPE Pacific/Asia Offshore Mechanics Symposium. Gold Coast, Australia: International Society of Offshore and Polar Engineers, 2016.

[8] 韩鹏, 刘志林, 周泽才, 等. 基于 LOS 法的自航模航迹跟踪控制算法实现 [J]. 应用科技, 2018, 45(3): 66–70.
HAN P, LIU Z L, ZHOU Z C, et al. Path tracking control algorithm based on LOS method for surface self-propulsion vessel[J]. Applied Science and Technology, 2018, 45(3): 66–70 (in Chinese).

[9] MOREIRA L, FOSSEN T I, SOARES C G. Path following control system for a tanker ship model[J]. Ocean Engineering, 2007, 34(14/15): 2074–2085.

[10] 任彧, 赵师涛. 磁导航 AGV 深度强化学习路径跟踪控制方法 [J]. 杭州电子科技大学学报, 2019, 39(2): 28–34.
REN Y, ZHAO S T. Deep reinforcement learning based path following control of magnetic navigation AGV[J]. Journal of Hangzhou Dianzi University, 2019, 39(2): 28–34 (in Chinese).

[11] CARRERAS M, RIDAO P, EL-FAKDI A. Semi-online neural Q_leaming for real-time robot learning[C]//Proceedings of 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems. Las Vegas, Nevada: IEEE, 2003: 662-667.

[12] 刘建伟, 高峰, 罗雄麟. 基于值函数和策略梯度的深度强化学习综述 [J]. 计算机学报, 2019, 42(6): 1406–1438.
LIU J W, GAO F, LUO X L. Survey of deep reinforcement learning based on value function and policy gradient[J]. Chinese Journal of Computers, 2019, 42(6):

1406–1438 (in Chinese).

[13] WOO J, YU C, KIM N. Deep reinforcement learning-based controller for path following of an unmanned surface vehicle[J]. Ocean Engineering, 2019, 183: 155–166.

[14] LILLICRAP T P, HUNT J J, PRITZEL A, et al. Continuous control with deep reinforcement learning[C]// Proceedings of the 4th International Conference on Learning Representations. San Juan, 2015: A187.

[15] SILVER D, LEVER G, HEESS N, et al. Deterministic policy gradient algorithms[C]//Proceedings of the 31st International Conference on Machine Learning. Beijing, China: ACM, 2014: I-387–I-395.

[16] YASUKAWA H, YOSHIMURA Y. Introduction of MMG standard method for ship maneuvering predictions[J]. Journal of Marine Science and Technology, 2015, 20(1): 37–52.

[17] LIU J L, QUADVLIEG F, HEKKENBERG R. Impacts of the rudder profile on manoeuvring performance of ships[J]. Ocean Engineering, 2016, 124: 226–240.

[18] 王艳. 无人船建模及路径跟踪控制 [D]. 杭州: 浙江大学, 2019.
WANG Y. Modeling and path tracking control of unmanned surface vessel[D]. Hangzhou: Zhejiang University, 2019 (in Chinese).

[19] 钟海鑫, 丘森辉, 罗晓曙, 等. 基于附加惯性项 BP 神经网络的四旋翼无人机姿态控制研究 [J]. 广西师范大学学报(自然科学版), 2017, 35(2): 24–31.
ZHONG H X, QIU S H, LUO X S, et al. Study of applying BP neural network with inertia term self-tuning to attitude stability of quadrotor unmanned aerial vehicle[J]. Journal of Guangxi Normal University (Natural Science Edition), 2017, 35(2): 24–31 (in Chinese).

(上接第 104 页)

[13] 袁宇祺. 多目标下的船舶智能避碰方法研究 [D]. 哈尔滨: 哈尔滨工程大学, 2019.
YUAN Y Q. Intelligent collision avoidance of surface vechiles under multi-objective[D]. Harbin: Harbin Engineering University, 2019 (in Chinese).

[14] HE Y X, HUANG L W, XIONG Y, et al. The research of ship ACA actions at different stages on head-on situation based on Cri and COLREGS[J]. Journal of Coastal Research, 2015, 73(Supp 1): 735–740.

[15] 王程博, 张新宇, 张加伟, 等. 未知环境中无人驾驶船舶智能避碰决策方法 [J]. 中国舰船研究, 2018, 13(6): 72–77.
WANG C B, ZHANG X Y, ZHANG J W, et al. Method for intelligent obstacle avoidance decision-making of unmanned vessel in unknown waters[J]. Chinese Journal of Ship Research, 2018, 13(6): 72–77 (in Chinese).

[16] SHEN H Q, HASHIMOTO H, MATSUDA A, et al. Automatic collision avoidance of multiple ships based on deep Q-learning[J]. Applied Ocean Research, 2019, 86: 268–288.

[17] 李丽娜, 陈国权, 李国定, 等. 船舶拟人智能避碰决策方法研究综述 [J]. 航海, 2014(2): 42–49.
LI L N, CHEN G Q, LI G D, et al. A review on PID-VCA system[J]. Navigation, 2014(2): 42–49 (in Chinese).

[18] 耿新力. 城区不确定环境下无人驾驶车辆行为决策方法研究 [D]. 合肥: 中国科学技术大学, 2017.
GENG X L. Research on behavior decision-making approaches for autonomous vehicle in urban uncertainty environments[D]. Hefei: University of Science and Technology of China, 2017 (in Chinese).

[19] DAVIS P V, DOVE M J, STOCKEL C T. A computer simulation of marine traffic using domains and arenas[J]. The Journal of Navigation, 1980, 33(2): 215–222.

[20] 苏开文. 船舶避碰专家系统的研究 [D]. 大连: 大连海事大学, 2007.
SU K W. A study on the expert system for ship collision avoidance[D]. Dalian: Dalian Maritime University, 2007 (in Chinese).

[21] FOSSEN T I. Guidance and control of ocean vehicles[M]. Chichester, New York: Wiley, 1994.

[22] 王鹏鲲, 陈国权, 李丽娜, 等. 船舶预测复航限制时间模型及算法验证 [J]. 中国航海, 2016, 39(1): 55–59.
WANG P K, CHEN G Q, LI L N, et al. Integrity verification of calculation model for forecasting recovery limited time of ship[J]. Navigation of China, 2016, 39(1): 55–59 (in Chinese).