

基于 DDPG 算法的海上无人救援技术研究

郑 帅¹ 贾宝柱^{1,2*} 张昆阳¹ 张 程¹

¹(大连海事大学轮机工程学院 辽宁 大连 116026)

²(广东海洋大学海运学院 广东 湛江 524088)

摘 要 针对海上无人救援过程中遇险目标的漂移及如何快速靠近的问题,提出一种基于深度强化学习理论的目标追踪算法,使无人搜救船在与环境交互的过程中学习到自动驾驶追踪漂移遇险目标的最优驾驶决策。在 SART 的辅助下,通过自主学习能够使搜救船以最短的时间追踪到漂移遇险目标。在 Gazebo 物理仿真器中建立三维仿真环境,基于 ROS 系统分别设计直线漂移轨迹和不规则漂移轨迹仿真实验,通过多次自主学习训练,验证所提方法的有效性。

关键词 深度强化学习 无人船 海上救援 目标追踪

中图分类号 TP249

文献标志码 A

DOI:10.3969/j.issn.1000-386x.2021.04.026

MARINE UNMANNED RESCUE TECHNOLOGY BASED ON DDPG ALGORITHM

Zheng Shuai¹ Jia Baozhu^{1,2*} Zhang Kunyang¹ Zhang Cheng¹

¹(Marine Engineering College, Dalian Maritime University, Dalian 116026, Liaoning, China)

²(College of Maritime, Guangdong Ocean University, Zhanjiang 524088, Guangdong, China)

Abstract Aiming at the problem of drifting distress target and the way of approaching quickly in the process of unmanned rescue at sea, a target tracking algorithm based on theory of deep reinforcement learning is proposed, which makes unmanned rescue vessel learn to autonomous driving to track drift target optimal decision during the interaction with environment. With the assistance of SART, the vessel got close to the drift distress target in shortest time through self-learning. A three-dimensional simulation environment was established in the Gazebo physics simulator. The simulation experiments of linear drift trajectory and irregular drift trajectory were designed respectively based on ROS. The effectiveness of the proposed method is verified through multiple independent learning and training.

Keywords Deep reinforcement learning Unmanned surface vehicle Maritime rescue Target tracking

0 引 言

海上应急救援往往受海况恶劣、时间紧迫等因素限制,救援成功的关键在于遇险目标位置的确定和如何快速靠近遇险目标。海上遇险目标会随着风、浪、流的作用而发生漂移,由于各种海洋漂浮物的受风面积与水下面积不尽相同,即使在同一风场与流场中,落水物的漂移轨迹也不相同^[1]。相关研究表明,落水人员在 20 °C 海水中最长存活时间为 16 小时 20 分钟,而在

0 °C 海水中最长存活时间仅为 12 分钟^[2],即使遇险人员能够利用救生艇、筏等延长存活时间,但也受到淡水、食物等补给限制。

目前,海上救援主要依靠人工的经验驾驶船舶应对海上多变的复杂环境追踪遇险目标,但是海上环境恶劣,使执行救援任务时有较高的安全风险。随着船舶无人驾驶技术的发展,海上无人救援研究受到了广泛重视^[3]。

针对遇险目标位置漂移问题,多数研究采用了预测遇险目标漂移轨迹的办法。旷芳芳等^[4]结合风场和

流场的数值模拟,给出了落水人员和救生筏的风致漂移系数,用于预测落水人员以及救生筏的实时位置。刘同木等^[5]基于 MMG 模型研究了风、浪、流的作用下船舶的漂移数学模型。Gao 等^[6]通过蒙特卡洛方法在疑似马航 370 航班遇难海域对其残骸的漂移轨迹进行预测。漂移模型预测方法的不足在于无法找到一个通用的预测模型对不同漂移目标的位置进行预测。同时,已有的多数研究工作仅仅关注预测方法准确度,却没有考虑遇险目标的快速追踪及靠近的问题。根据 SOLAS 公约规定,每艘救生艇、筏要求至少配备一台搜救雷达应答器(Search and Rescue Radar Transponder, SART),用来近距离确定遇难船舶、救生艇、救生筏,以及幸存者位置^[7]。因此,实际救援场景中更可靠的是根据 SART 所提供的实时目标位置信号开展营救。

当遇险目标的位置已知,就要考虑如何快速靠近遇险目标的问题。Park 等^[8]基于视觉的估计和制导方法,控制无人机追踪移动目标。李静等^[9]提出交通道路网络环境下的局部搜索树移动目标追踪算法。Woo 等^[10]设计了强化学习控制器,使无人船具备了自主路径跟踪的能力。本文尝试将 DDPG 深度强化学习算法应用于海上无人救援,结合 SART 所提供遇险目标位置信号,设计了基于机器人操作系统^[11](Robot Operating System, ROS)仿真实验平台。在 Gazebo 物理仿真器中模拟水文环境以及船舶的运动特性,使搜救无人船在感知环境信息的同时,通过对不同漂移轨迹的落水物进行追踪实验,训练其追踪漂移遇险目标的能力。

1 目标跟踪算法设计

1.1 船舶运动模型

一般情况下,船舶运动研究通常只考虑三个自由度的运动,通过 Gazebo 插件实现双体船操纵的非线性运动学模型,船舶操纵公式^[12]如下:

$$\mathbf{M}\mathbf{v} + \mathbf{C}(\mathbf{v})\mathbf{v} + \mathbf{D}(\mathbf{v})\mathbf{v} = \boldsymbol{\tau} + \boldsymbol{\tau}_{\text{env}} \quad (1)$$

$$\boldsymbol{\eta} = \mathbf{J}(\boldsymbol{\eta})\mathbf{v} \quad (2)$$

式中: \mathbf{M} 是惯性矩阵; $\mathbf{C}(\mathbf{v})$ 是科里奥利矩阵; $\mathbf{D}(\mathbf{v})$ 是阻尼矩阵; $\boldsymbol{\tau}$ 是推进系统产生的力和力矩向量; $\boldsymbol{\tau}_{\text{env}}$ 是通过 Gazebo 插件模拟的风、浪、流干扰力; 向量 $\boldsymbol{\eta}$ 描述了惯性参考系北(N)、东(E)中的位置和绕 Z 轴的转首角度 φ ; $\boldsymbol{\eta} = [x, y, \varphi]^T$ 和向量 \mathbf{v} 包含船体固定坐标系中的船体的纵向速度 u 、横向速度 v 和转首角速度 r , 即 $\mathbf{v} = [u, v, r]^T$ 。船体坐标系到惯性坐标系的坐标转

换关系为:

$$\mathbf{J}(\boldsymbol{\eta}) = \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

1.2 状态空间设计

为追踪到不同的遇险目标,使算法具有一定的泛化性,搜救船的环境模型需要考虑遇险目标的位置,得到遇险目标与搜救船的相对位置关系,这种关系包括二者之间的距离 d 以及船舶的首向与二者连线方向的夹角 φ ,环境模型状态如图 1 所示,定义状态 \mathbf{s} 为:

$$\mathbf{s} = [d, \varphi]^T \quad (4)$$

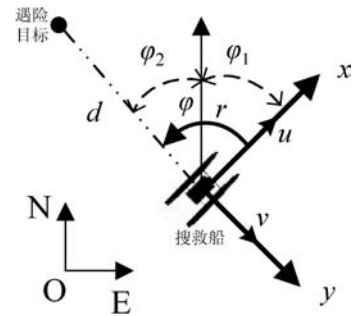


图 1 状态示意图

当搜救船捕获到 SART 反射的雷达信号,在雷达屏幕上形成 12 个一连串的光点,这种显示方式用来区别于其他雷达回波信号,其中最靠近雷达中心点的光点便是遇险目标的位置 (x_t, y_t) 。另外,通过船舶运动模型计算我们船的位置信息 (x_v, y_v) ,可以计算出二者之间的距离 d 。

$$d = \sqrt{(y_t - y_v)^2 + (x_t - x_v)^2} \quad (5)$$

$\varphi = \varphi_1 - \varphi_2$ ($-180 < \varphi, \varphi_1, \varphi_2 \leq 180$); φ_1 通过模型计算得出,其为首向角,即船首方向与正北方向的夹角; φ_2 为遇险目标和搜救船的相对方位,即二者连线与正北方向的夹角。

$$\varphi_2 = \arctan((x_t - x_v)/(y_t - y_v)) \quad (6)$$

1.3 动作空间设计

对环境状态的改变依赖于智能体通过驱动装置使船舶发生运动,双体船通过电机指令驱动两个螺旋桨产生差速推力来驱动搜救船移动。动作空间由一组电机指令组成,指令发布时间间隔为 1 s,定义动作空间 \mathbf{a} 为:

$$\mathbf{a} = \{[m+n, m-n]^T | m \in [-0.5, 0.5], n \in [-0.5, 0.5]\} \quad (7)$$

式中: m 表示双体船的双桨转速值; n 表示左右螺旋桨差动值。参考文献^[12]中的系柱实验得出的单桨推力与电机指令之间映射关系,得到搜救船的驱动力 $\boldsymbol{\tau}$,根据船舶操纵模型以及坐标转换公式可以计算出当前指令执行后船舶的位置姿态。

1.4 奖励函数设计

由于救援任务具有时间相关性,所以奖励函数的设计需要考虑时间与距离因素。奖励函数用 r_t 表示。

$$r_t = \begin{cases} 2 - 0.01 \times t & d \text{ 小于救援任务成功条件} \\ -2 & \text{搜救船触及搜救范围边界} \\ \xi(d_t - d_{t-1}) + \zeta(\varphi_t - \varphi_{t-1}) & \text{其他} \end{cases} \quad (8)$$

式中: t 表示时间环境反馈奖励值为 $(2 - 0.01 \times t)$,设计成为时间相关函数,目的是引导搜救船用尽量短的时间完成追踪任务; d_t 表示当前采样获得的搜救船与遇险目标的距离; d_{t-1} 表示上一次采样的距离。将上一次的距离与当前的距离做差,如果当前距离比上次距离短,差值为正,反之为负, φ 角同理,这种设计目的是引导搜救船朝着遇险目标的方向逐渐靠近,获得相对较高的奖励值。

2 DDPG 算法

环境观测值与动作值由深度确定性策略梯度下降(Deep Deterministic Policy Gradient, DDPG)^[13] 算法进行处理。DDPG 算法需要两套神经网络,分别为主网络与目标网络。每套网络都由一个 Actor(演员)网络与一个 Critic(评论家)网络构成。两套网络的网络结构完全相同,只是参数不同。

该算法是在梯度下降法的基础上,对 Actor-Critic 算法进行改进,将输出动作选择概率转变为输出确定动作值的强化学习算法,这个确定性的 zs915121 动作是随机策略梯度的极限值。Actor-Critic 算法框架由两个网络构成,包含了以求解值函数为核心的 Critic 网络,还有以策略为核心的 Actor 网络。其中 Actor 网络的输入为状态矩阵,输出确定的动作值 a_t ,而 Critic 网络的输入为动作值 a_t 与状态值 s_t ,输出动作的 Q 值, Q 值对当前回合的一系列动作策略做出评价,因此,训练该网络的过程是判断动作策略优劣的学习过程,通过训练,得出 Q 值最大的一系列动作策略。

每一步采集的样本 (s_t, a_t, r, s_{t+1}) 输入到主网络中训练,采样过程如图 2 所示,同时开辟一个记忆库来存储采样的结果,目标网络则随机抽取记忆库中的数据作为输入,记忆库的数据不断更新,切断了样本数据间的相关性。主网络的参数通过回合更新的方式,若干回合后,将主网络参数赋值给目标网络,进行目标网络的参数更新。

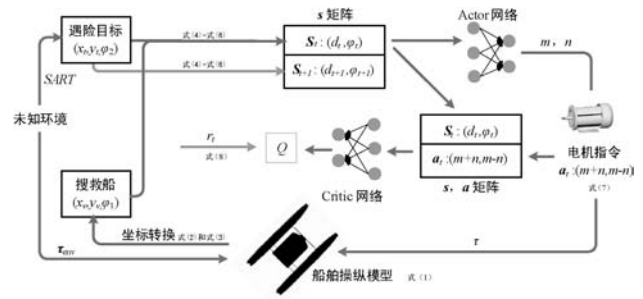


图 2 样本采集过程

算法具体流程如下:

(1) 构建 Actor 网络与 Critic 网络,主网络参数分别为 θ^μ 、 θ^Q ,并进行随机初始化。Critic 网络得到的 Q 值为 $Q(s, a | \theta^Q)$,Actor 网络得到状态到动作的映射 $\mu(s | \theta^\mu)$ 。

(2) 通过网络参数 θ^μ 、 θ^Q 来初始化所求解的目标网络的参数 $\theta^{\mu'}$ 、 $\theta^{Q'}$,同时开辟一个记忆库存储空间。

(3) 初始化状态 s_1 ,通过主网络加上高斯扰动 N_t ,从动作库中选择一个动作 a_t 进行探索:

$$a_t = \mu(s | \theta^\mu) + N_t \quad (9)$$

(4) 执行该动作,得到相应的奖励 r_t 和下一个状态 s_{t+1} ,并且形成元组 (s_t, a_t, r_t, s_{t+1}) ,作为一条采样值存入到记忆库存储空间中。

(5) 通过当前网络 $Q(s, a | \theta^Q)$ 进行估计,同时从记忆库中选取一小批元组数据,通过 Bellman 方程对 Q 值进行估计,估计结果表示为:

$$q(a_t) = r_t + \gamma Q(s_{t+1}, \mu(s_{t+1} | \theta^{\mu'}) | \theta^{Q'}) \quad (10)$$

式中: γ 表示折扣率,反映未来估计的奖励对当前决策的影响程度,如果 $\gamma = 0$,则不考虑未来任何价值;如果 $\gamma = 1$,则未来的价值不会随时间衰减。

将计算出的两个值做差,得到损失函数 TD_error。然后使用梯度下降法对 Critic 网络的参数进行更新。

(6) 主要采取策略梯度的方式进行 Actor 网络的更新,即:

$$\nabla_{\theta^\mu} J(\theta) \approx \frac{1}{N} \sum_i \nabla_a Q(s, a | \theta^Q) |_{s=s_i, a=\mu(s_i)} \times \nabla_{\theta^\mu} \mu(s | \theta^\mu) |_{s_i} \quad (11)$$

在得到策略梯度后,通过最大化期望奖励的方式对总奖励进行梯度上升更新参数。

(7) 若干回合后,将主网络的参数赋值给目标网络,对目标网络参数更新。

$$\begin{cases} \theta^{Q'} \leftarrow \kappa \theta^Q + (1 - \kappa) \theta^{Q'} \\ \theta^{\mu'} \leftarrow \kappa \theta^\mu + (1 - \kappa) \theta^{\mu'} \end{cases} \quad (12)$$

算法结构如图 3 所示。

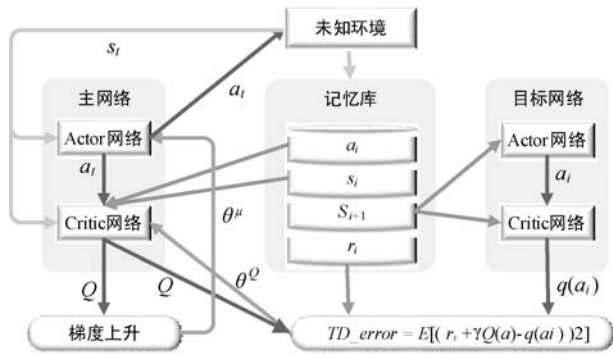


图3 DDPG 算法结构图

使用 TensorFlow 搭建 Actor 网络和 Critic 网络, Actor 网络的输入为状态矩阵, 网络结构有两个隐藏层, 每个隐藏层的节点分别为 400、300, 输出动作矩阵。将状态矩阵输入到 Critic 网络中, 该网络第二层有 400 个节点, 第三层有 300 个节点。动作矩阵也输入到 Critic 网络, 第二层有 300 个神经元节点, 并将状态空间矩阵输入的网络的第三层神经元与动作矩阵输入的网络的第二层神经元节点合并进行线性变换, 输入到第四层网络神经元节点上, 该层共有 300 个神经元节点, 最后输出动作的 Q 值。网络的所有神经元节点之间的连接方式均为全连接方式, 网络结构图如图 4 所示。

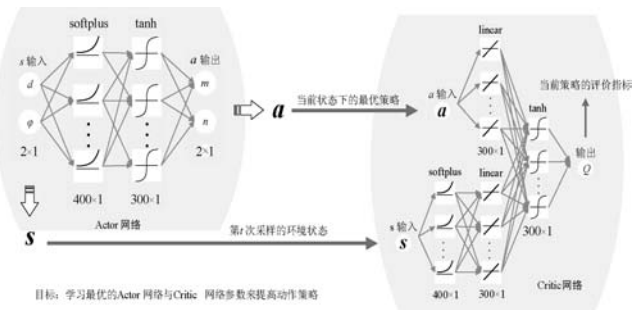


图4 Actor-Critic 网络结构图

3 实验

3.1 实验平台

实验设计基于 ROS, 在 Gazebo 物理仿真器中创建三维实验环境与智能体, 模拟海上救援场景, 设定环境场景边界为 50×50 的范围, 初始状态下搜救船位于 $(0,0)$ 的位置, 并在环境扰动的干扰下运动, 遇险目标位于 $(20,10)$ 的位置。当搜救船与遇险目标的距离小于 5 时, 视为满足成功救援的条件。Actor 网络的学习率为 0.000 1, Critic 网络的学习率为 0.001, 折扣因子为 0.99。每次实验的总回合数为 1 000, 每个回合最多采集 200 个样本数据, 由于搜救船执行电机指令需

要一定的时间, 设置样本采集的时间间隔为 1 s。记忆库的容积为 10 000, 批数据的容量为 64。奖励函数的系数为 ξ, ζ 均为 1, 目标网络更新参数 κ 为 0.01。

3.2 实验设计及结果分析

3.2.1 直线轨迹目标追踪

从图 5 中可以看出, 实验初期(前 200 回合), 由于探索的概率比较大, 这一阶段搜救船在探索环境, 采集环境样本数据, 很难准确追踪到遇险目标, 平均 Q 值 Q_a 也较低, 只有少数成功的情况, Q 值较高。从图 6 中可以看出, 这一阶段用时都相对较长。实验中期(200 ~ 500 回合), 随着样本数量的累积, 逐渐建立起了奖励函数与动作策略之间的联系, 搜救船从探索环境知识逐渐转为利用环境知识, 逐渐可以追踪到遇险目标, 算法快速收敛, 但是会走一些弯路。随着训练的不断深入, 搜救船会不断地调整自己的运动趋势, 以最短的时间靠近遇险目标, 最终 Q_a 稳定在 10, 平均时长 t_a 稳定在 70 s。从 Q 值以及时间的变化趋势来看, 算法符合救援过程中时间紧迫性的要求。

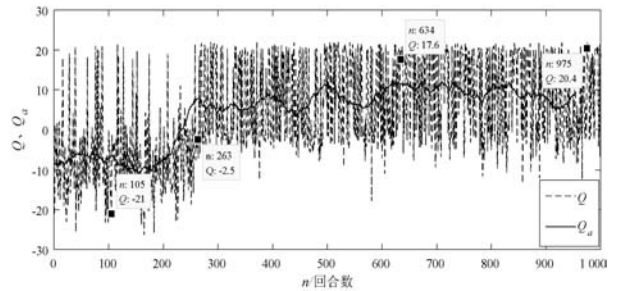


图5 Q 值以及平均 Q 值(Q_a) 变化曲线一

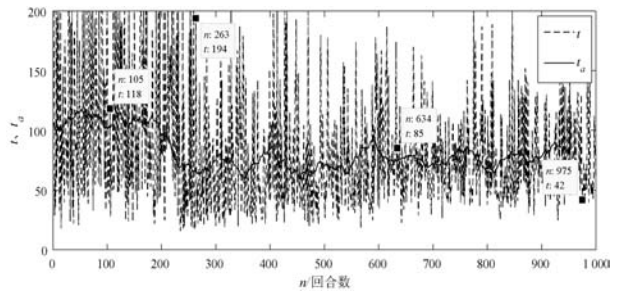


图6 回合时长以及平均时长(t_a) 变化曲线一

从三个阶段中选取第 105 回合、第 263 回合、第 634 回合, 以及第 975 回合训练过程进行分析, 如图 7 所示, 分别对应(a)、(b)、(c)、(d)的四幅轨迹图, 其中虚线轨迹为搜救船的运动轨迹, 实线轨迹为遇险目标的漂移轨迹。可以看出只有第 105 回合训练失败, 结合图 5 可以看出, 奖励值最低, 其他三个回合均成功追踪到了遇险目标。随着训练深入, 第 263、第 634、第 975 回合搜救船的轨迹渐短, Q 值越高, 结合图 6 可以看出所用的时间也越来越短。

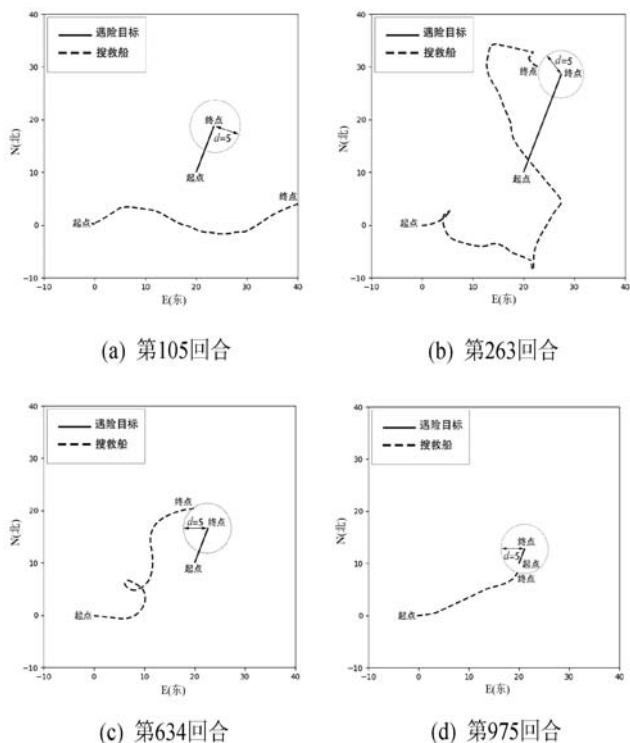


图 7 搜救船及遇险目标运动轨迹一

在 Gazebo 仿真环境中可以观察到以上四个回合结束时搜救船与遇险目标的相对位置如图 8 所示,其中黑色的圆点为回合结束时遇险目标的位置。

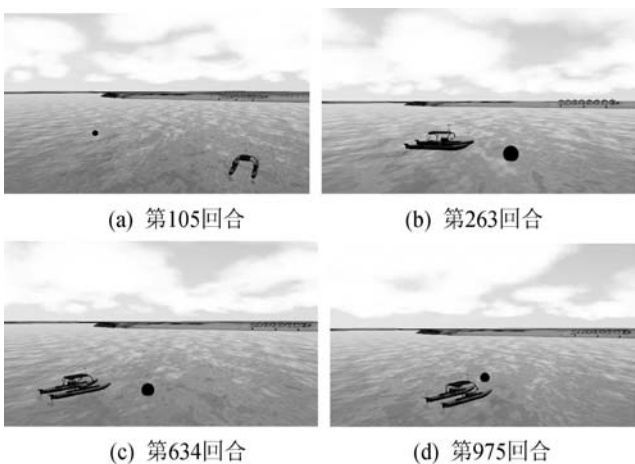


图 8 回合结束时 Gazebo 场景一

由图 9 可以直观地看出上述四个回合的样本的实时奖励值变化趋势。第 105 回合(曲线 a)奖励值多数情况下是负数,表明搜救船一直在远离遇险目标,因此 Q 值比较低,为 -21 。而第 975 回合(曲线 d)则每一个动作都是在靠近遇险目标,因此奖励值很高,高达 20.4 。其他两个回合(曲线 b 和曲线 c)均有正有负,表明追踪过程有远离的情况,这两个过程中的动作策略不是最优的策略。从图 7 的轨迹图中也可以看出这一点,证明奖励函数的设计可以引导搜救船快速追踪到救援目标,是有效的。

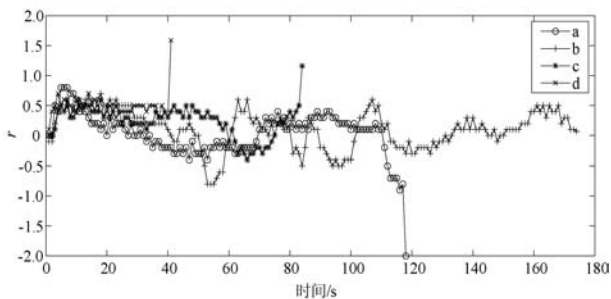


图 9 单个回合奖励值随时间的变化曲线对比一

3.2.2 不规则轨迹目标追踪

由于不同落水物体在海上的轨迹不同,为证明算法的有效性,补充了不规则轨迹落水物的追踪实验。从图 10 中可以看出与直线轨迹追踪实验结果相似,随着环境状态样本数量的增加,搜救船由探索环境转变为利用环境知识,从第 200 回合开始算法逐渐收敛;平均 Q 值(Q_a) 逐渐增加,最终稳定在 15 左右。从图 11 中可以看出完成任务所需的平均时间(t_a) 逐渐减少,用时较少的回合可达 28 s。由此可见,尽管不同的遇险目标漂移的轨迹不同,甚至轨迹变得不规则,搜救船仍然可以通过 DDPG 算法达到快速追踪到遇险目标的目的,算法仍能够收敛。

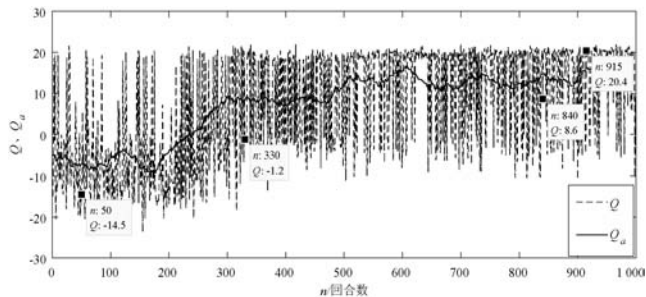


图 10 Q 值以及平均 Q 值(Q_a) 变化曲线二

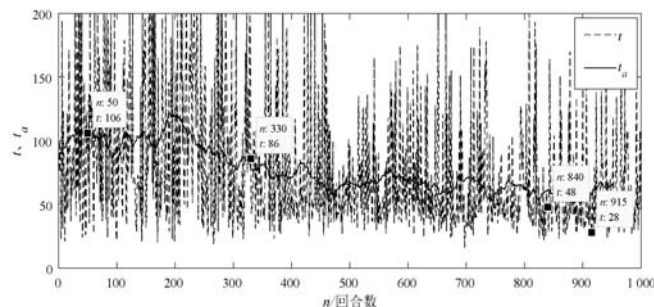


图 11 回合时长以及平均时长(t_a) 变化曲线二

选择第 50、第 330、第 840、第 915 回合,运动轨迹分别如图 12(a)、(b)、(c)、(d) 所示,同样虚线轨迹为搜救船的运动轨迹,实线轨迹为遇险目标的漂移轨迹。第 50 回合中搜救船没有能够追踪到遇险目标,而第 330、第 840、第 915 回合均完成了对遇险目标的追踪任务,并且第 915 回合的轨迹是最短的,用时最少, Q 值也是最高的。

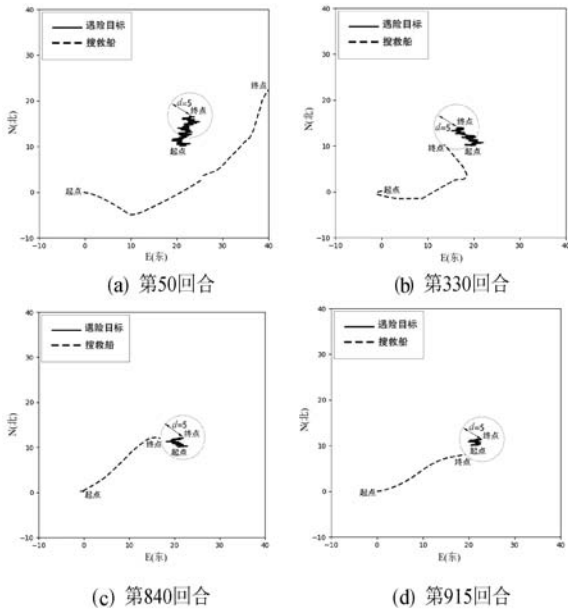


图 12 搜救船及遇险目标运动轨迹二

上述四个回合在回合结束时, Gazebo 中显示的相对位置如图 13 所示, 黑点表示回合结束时遇险目标的位置。

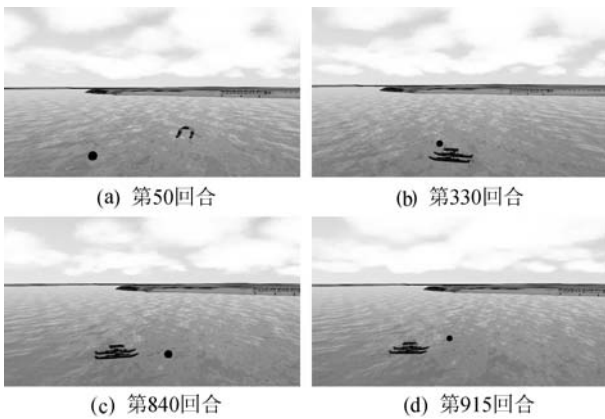


图 13 回合结束时 Gazebo 场景二

通过对以上四个回合中的每一次采样获得的奖励值变化进行分析, 如图 14 所示。从曲线 e 可以看出, 算法未建立起奖励函数与动作策略之间的关系, 与遇险目标的距离时远时近, 曲线在 0 附近波动。而曲线 g 和曲线 h 表示在这两个回合中每次采样的奖励值大多位于 0.5 以上, 表明搜救船与遇险目标正在快速靠近, 再一次说明奖励函数的设计是有效的。

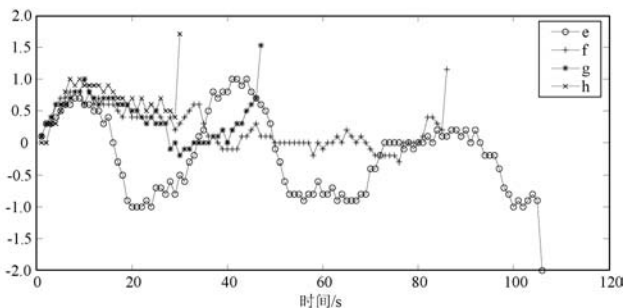


图 14 单个回合奖励值随时间的变化曲线对比二

4 结 语

在海上无人救援过程中, 针对遇险目标漂移问题, 本文提出一种基于 DDPG 算法的目标追踪方法, 本文所做的研究工作及相应结论可为实船实验提供参考。主要结论如下:

(1) 分析了强化学习算法应用于海上无人救援情景的技术难点, 在未知遇险目标的漂移模型情况下, 当获取遇险目标与自身相对位置时, 算法可以使搜救船自主做出恰当的操纵决策, 以较短的时间靠近遇险目标, 通过直线漂移轨迹的遇险目标追踪实验和不规则漂移轨迹的遇险目标追踪仿真实验得以验证。

(2) 将强化学习算法与海上实际救援情况相结合, 设计状态、动作空间、相应的奖励函数, 奖励函数的设计考虑了时间因素与距离因素。经过仿真实验验证, 奖励函数能够引导无人搜救船快速地追踪到遇险目标, 并且具有一定的泛化性, 符合海上搜救时间紧迫性的要求以及目标追踪距离逐渐缩短的要求。

参 考 文 献

[1] 钟铭. 国家海上搜救手册[M]. 大连:大连海事大学出版社,2011.

[2] 黄敏东. 论海上遇险黄金救援时间[J]. 世界海运,2014, 37(11):33-35.

[3] 宋杰, 闻佳. 无人船技术在海事的应用[J]. 中国海事, 2015(10):47-50.

[4] 旷芳芳, 靖春生, 张俊鹏. 基于观测和模型的风致漂移系数研究[J]. 应用海洋学学报,2017,36(1):41-48.

[5] 刘同木, 刘在科, 余建星, 等. 船舶漂移运动轨迹模型及其实船试验验证[J]. 船舶力学,2018,22(11):1315-1322.

[6] Gao J, Mu L, Bao X, et al. Drift analysis of MH370 debris in the southern indian ocean[J]. Frontiers of Earth Science, 2018,12(3):468-480.

[7] 周华峰, 王佐良. 一次救助经历和体会[J]. 航海技术, 2003(4):14.

[8] Park S, Jung D. Vision-based tracking of a ground-moving target with UAV[J]. International Journal of Aeronautical and Space Sciences,2019,(2):467-482.

[9] 李静, 王楠, 许铜华, 等. 基于局部搜索树的 UAV 与 UGS 协同移动目标追踪方法[J]. 电光与控制,2019,26(1):1-7.

[10] 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.

(2) 本文方法可以对普通知识图谱和领域性知识图谱进行补全,同时也可以应用到查询系统,比如建模三元组(查询,用户,文档),通过查找三元组中缺失的实体或关系,将缺失的实体或关系以三元组的形式添加到数据库中,从而对数据库进行补全扩充。

另外,针对得到的实体或关系与已有实体关系存在不相容或冲突问题:使用本文方法判断不同三元组得分,利用分数判断三元组正确性;使用实体或关系对齐模型来判断不同三元组的正确性,若三元组正确,则在知识图谱中可以共存,比如:姚明亦指大姚(别名)。

为提高三元组补全正确率,今后的研究将尝试修改卷积神经网络内部架构,对三元组特征提取作更深入的研究;将实体和关系的描述文本融入到卷积神经网络中,作为算法的输入。

参 考 文 献

- [1] 丁建辉,贾维嘉. 知识图谱补全算法综述[J]. 信息技术,2018(1):56-62.
- [2] 谢刚. 知识图谱精化研究综述[J]. 电子技术应用,2018,44(9):29-33,38.
- [3] 刘峤,李杨,杨段宏,等. 知识图谱构建技术综述[J]. 计算机研究与发展,2016,53(3):582-600.
- [4] Kim Y. Convolutional neural networks for sentence classification[EB]. arXiv:1408.5882,2014.
- [5] Zeng D, Liu K, Lai S, et al. Relation classification via convolutional deep neural network[C]//International Conference on Computational Linguistics,2014:2335-2344.
- [6] Nguyen D Q, Nguyen T D, Nguyen D Q, et al. A novel embedding model for knowledge base completion based on convolutional neural network[C]//The North American Chapter of the Association for Computational Linguistics. ACM, 2018:327-333.
- [7] Bordes A, Usunier N, Garcia-Duran A, et al. Translating embeddings for modeling multi-relational data[C]//Proceedings of the 26th International Conference on Neural Information Processing Systems. Curran Associates, 2013:2787-2795.
- [8] Wang Z, Zhang J W, Feng J L, et al. Knowledge graph embedding by translating on hyperplanes[C]//Proceedings of the 28th AAAI Conference on Artificial Intelligence. AAAI Press,2014:1112-1119.
- [9] Lin Y K, Liu Z Y, Sun M S, et al. Learning entity and relation embeddings for knowledge graph completion[C]//Proceedings of the 29th AAAI Conference on Artificial Intelligence. AAAI Press,2015:2181-2187.
- [10] Ji G, He S, Xu L, et al. Knowledge graph embedding via dynamic mapping matrix[C]//53rd Annual Meeting of the ACL. ACL,2015:687-696.
- [11] 刘知远,孙茂松,林衍凯,等. 知识表示学习研究进展[J]. 计算机研究与发展,2016,53(2):247-261.
- [12] 林海伦,王元卓,贾岩涛,等. 面向网络大数据的知识融合方法综述[J]. 计算机学报,2017(1):3-29.
- [13] Socher R, Chen D Q, Manning C D, et al. Reasoning with neural tensor networks for knowledge base completion[C]//Proceedings of the 26th International Conference on Neural Information Processing Systems. ACM,2013:926-934.
- [14] Xie R B, Liu Z Y, Jia J, et al. Representation learning of knowledge graph with entity descriptions[C]//Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence. AAAI Press,2016:2659-2665.
- [15] Hamaguchi T, Oiwa H, Shimbo M, et al. Knowledge transfer for out-of-knowledge-base entities: A graph neural network approach[C]//Proceedings of the 26th International Joint Conference on Artificial Intelligence. AAAI Press,2017:1802-1808.
- [16] Yang B, Yih W T, He X, et al. Embedding entities and relations for learning and inference in knowledge bases[EB]. arXiv:1412.6575,2014.
- [17] Trouillon T, Welbl J, Riedel S, et al. Complex embeddings for simple link prediction[C]//Proceedings of the 33rd International Conference on International Conference on Machine Learning,2016:2071-2080.
- [18] Dettmers T, Minervini P, Stenetorp P, et al. Convolutional 2D knowledge graph embeddings[C]//Proceedings of the AAAI,2018:1811-1818.
- [19] Kingma D, Ba J. Adam: A method for stochastic optimization[C]//Proceedings of the International Conference on Learning Representations,2015.
- [20] Toutanova K, Chen D. Observed versus latent features for knowledge base and text inference[C]//Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality,2015:57-66.

(上接第164页)

- [11] 张福海,李宁,袁儒鹏,等. 基于强化学习的机器人路径规划算法[J]. 华中科技大学学报(自然科学版),2018,46(12):65-70.
- [12] Sarda E I, Qu H, Bertaska I R, et al. Station-keeping control of an unmanned surface vehicle exposed to current and wind disturbances[J]. Ocean Engineering,2016,127:305-324.
- [13] Lillicrap T P, Hunt J J, Pritzel A, et al. Continuous control with deep reinforcement learning[EB]. arXiv:1509.02971,2015.