基于多智能体强化学习的无人艇协同围捕方法

夏家伟[1], 朱旭芳[2†], 张建强[1], 罗亚松[1], 刘 忠[1]

(1. 海军工程大学 兵器工程学院，武汉 430033；2. 海军工程大学 电子工程学院，武汉 430033)

摘 要: 针对多无人艇对海上逃逸目标的围捕问题,提出一种基于多智能体强化学习的围捕算法. 首先,以无人艇协同进攻为背景建立无边界围捕问题的环境和运动学模型,并针对快速性和合围性的需求给出围捕成功的判定条件;然后,基于多智能体近端策略优化(MAPPO)算法建立马尔可夫决策过程框架,结合围捕任务需求分别设计兼具伸缩性和排列不变性的状态空间,围捕距离、方位解耦的动作空间,捕获奖励与步长奖励相结合的奖励函数;最后,采用集中式训练、分布式执行的架构完成对围捕策略的训练,训练时采用课程式学习训练技巧,无人艇群共享相同的策略并独立执行动作. 仿真实验表明,在无人艇起始数量不同的测试条件下,所提出方法在围捕成功率和时效性上相较于其他算法更具优势. 此外,当无人艇节点损毁时,剩余无人艇仍然具备继续执行围捕任务的能力,所提出方法鲁棒性强,具有在真实环境中部署应用的潜力.

关键词: 无人艇；多智能体；强化学习；深度学习；协同围捕；近端策略优化

中图分类号: TP249 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2022.0564

引用格式: 夏家伟, 朱旭芳, 张建强, 等. 基于多智能体强化学习的无人艇协同围捕方法 [J]. 控制与决策, 2023, 38(5): 1438-1447.

Research on cooperative hunting method of unmanned surface vehicle based on multi-agent reinforcement learning

XIA Jia-wei[1], *ZHU Xu-fang*[2†], *ZHANG Jian-qiang*[1], *LUO Ya-song*[1], *LIU Zhong*[1]

(1. College of Weaponry Engineering，Naval University of Engineering，Wuhan 430033，China；2. College of Electronic Engineering，Naval University of Engineering，Wuhan 430033，China)

Abstract: To solve the hunting problem of multi-USVs (unmanned surface vehicles) on the sea, a multi-agent reinforcement learning hunting algorithm is proposed. Firstly, the environmental and kinematic model of the boundary-free hunting problem is established based on the background of the cooperative attack of USVs, and the criteria for successful hunting are given according to the requirements of rapidity and encirclement. Then, a Markov decision process framework is established based on the multi-agent PPO (MAPPO) algorithm. The state-space with scalability and permutation invariant, an action space with decoupling of capture distance and azimuth, and a reward function combining capture reward and step reward are designed. Finally, the framework of centralized training and distributed execution is adopted to train the policy. During the training, the skills of curriculum learning are used to make the network converge quickly, and the USVs share the same strategy and execute the action independently. Simulation shows that the proposed method has more advantages than other algorithms in the hunting success rate and timeliness under different testing conditions. In addition, when some of the USVs are failed, the remaining USVs can continue the task, which proves strong robustness and potential for deployment in a real environment.

Keywords: USV；multi-agent；reinforcement learning；deep learning；cooperative hunting；proximal policy optimization

0 引 言

水面无人艇(unmanned surface vehicle, USV)作为一种小型海上无人任务平台,兼具高速智能、灵活隐蔽等特点,军事应用价值显著[1]. 随着无人装备的加速发展,无人作战将成为未来战争中的重要作战样式. 无人集群自主遂行任务的能力是提升其作战效

收稿日期: 2022-04-08；**录用日期:** 2022-09-05.

基金项目: 中国博士后科学基金项目(2016T45686)；湖北省自然科学基金项目(2018CFC865)；全军军事类研究项目(YJ2020B117).

责任编委: 杨涛.

†通讯作者. E-mail: 1580284687@qq.com.

*本文附带电子附录文件,可登录本刊官网该文“资源附件”区自行下载阅览.

能的关键[2], 多 USV 围捕作为无人协同攻击任务的典型应用, 是USV技术研究领域的热点之一.

通常围捕问题的求解可大致分为确定性方法和启发式方法. 确定性方法使用传统数学工具求解. Shneydor[3] 运用微分方程分析追捕者的轨迹, 计算并分析了成功捕获的条件; Sun 等[4] 使用基于可达性的方法建立了微分博弈模型, 计算时间最优轨迹和围捕策略. 受自然界中围捕行为的启发, 关于启发式方法的研究也相继展开. Muro 等[5] 基于对狼群的狩猎行为, 提出并验证了一种简易规则的围捕策略; Janosov 等[6] 受生物学中自然捕食系统的启发, 提出了集群追捕策略. 然而, 传统方法在设计围捕策略时, 往往对逃逸目标的运动策略作出单一假定, 但是在真实战场环境下己方很难获知逃逸目标的控制策略, 同时当环境模型发生变化时, 控制器参数难以快速适配, 具有一定局限性[7].

随着人工智能技术的快速发展, 多智能体强化学习 (multi-agent reinforcement learning, MARL) 为解决集群围捕问题提供了新的途径. MARL 是多智能体系统与深度强化学习 (deep reinforcement learning, DRL) 的领域结合, 可通过在一个公共环境中令智能体不断与环境进行交互试错, 使用深度学习来解决多个智能体的序列决策问题[8]. 学界针对 MARL 方法的围捕问题开展了大量探索性工作, Lowe 等[9] 提出了一种将 Actor-Critic 算法扩展至多智能体领域的方法, 并在围捕问题中得到验证. 符小卫等[7] 针对多无人机协同追捕问题, 提出了基于解耦多智能体深度确定性策略梯度算法, 在仿真环境中使用 4 架无人机实现对高速逃逸目标的围捕. Xu 等[10] 采用双向循环神经网络[11] 实现了智能体数量动态变化条件下的围捕任务. Wang 等[12] 针对协同围捕过程中的通信优化问题, 提出了一种基于学习的环状通信网络, 在降低通信开销的同时保证了较高的任务成功率. Hüttenrauch 等[13] 开展了同构智能体模型可伸缩性的研究, 所提出的算法在多对一和多对多围捕的测试场景中得到了验证. de Souza 等[14] 针对非完整性约束运动学模型, 使用 TD 3 算法[15] 实现了智能体的协同围捕, 并在真实场景中使用无人机进行了检验.

上述成果从强化学习算法、网络通信、特征工程和运动学模型等方面研究了围捕问题, 具有一定的指导意义, 但是在多 USV 协同的实际应用方面还存在探索的空间, 以下问题尚需进一步探讨: 1) 大部分研究引入了围捕边界约束[7,10,12-14], 当逃逸目标被追捕至区域边界时, 也视为围捕成功, 然而海洋环境通常不存在边界约束和机动范围限制. 2) 已有应用成果集中于无人机围捕领域[7,14], 而 USV 相较于无人机在机动能力和空间适用性上存在差异, 另外 USV 应用领域更注重围捕任务实现时 USV 集群对逃逸目标所形成的合围态势. 3) 对抗环境下要求 USV 集群执行协同任务时具备一定鲁棒性, 在少量节点损毁时需具备继续执行任务的能力, 但是目前关于节点损毁对围捕任务影响的研究尚不充分.

在上述问题的推动下, 本文首先建立无边界围捕环境模型, 结合战术应用背景给出 USV 围捕成功的判定条件, 在此基础上建立 USV 运动模型和逃逸目标的规避机动策略; 然后提出适用于 USV 围捕问题的状态特征和对应的状态空间设计, 围绕围捕成功判定条件设计相应的奖励函数和动作空间; 最后使用多智能体近端策略优化 (multi-agent PPO, MAPPO)[16] 算法以集中式训练分布式执行的架构来完成对围捕策略的训练, 实现 USV 协同围捕策略.

1　问题描述与建模

1.1　多 USV 围捕问题描述

任务场景描述如下: 作战海域中存在多艘同构 USV 和逃逸目标船只, 双方具有相反的战术目的. USV 间需要通过协同合作尽快对逃逸目标实现围捕, 而逃逸目标要躲避远离 USV 群.

现有研究通常认为当存在任意追击者与逃逸者的距离小于给定阈值时, 围捕任务即成功完成[9-10,12-14], 但是考虑到多角度饱和攻击等战术的运用, USV 的合围态势也应纳入考量. 围捕场景如图 1 所示.

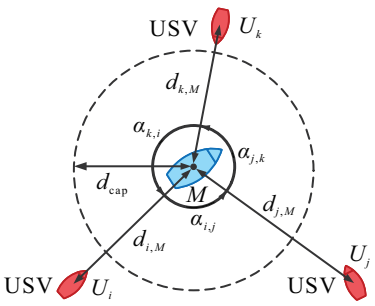


图 1　USV 围捕场景示意图

图 1 中: $U_i(i = 1, 2, \ldots, N)$ 为 USV, M 为逃逸目标, d_{cap} 为围捕半径, $d_{i,M}$ 为 U_i 与 M 的距离, 定义围捕角 $\alpha_{i,j}$ 为对逃逸目标 M 围夹的无人艇 U_i 与 U_j 相对 M 的夹角.

规定围捕成功需满足以下条件: 1) 存在任意 USV 与逃逸目标 M 的距离小于围捕半径 d_{cap}; 2) 相邻 USV 间的围捕角不大于 π.

围捕过程中需要满足以下约束: 1) 围捕任务在规定时间内完成, 且耗时越少越佳; 2) 围捕成功时各USV与逃逸目标M的距离尽量接近; 3) USV间不能发生碰撞.

1.2 USV运动学模型

如图2所示的USV坐标系, 建立USV的二阶运动学方程为

$$\begin{cases} \dot{v}_i = a_v, \\ \dot{\omega}_i = a_\omega, \\ \dot{x}_i = v_i \cos\psi_i, \\ \dot{y}_i = v_i \sin\psi_i, \\ \dot{\psi}_i = \omega_i. \end{cases} \tag{1}$$

其中: (x_i, y_i)为USV的位置; v_i、ψ_i、ω_i分别为USV的航速、航向和角速度, 航速v_i和角速度ω_i受到约束限制, 即$-\omega_{\max} \leqslant \omega_i \leqslant \omega_{\max}, 0 \leqslant v_i \leqslant v_{\max}$; a_v和a_ω分别为加速度和角加速度.

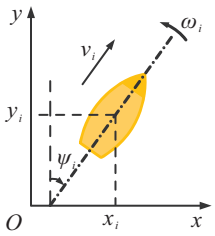


图 2 USV坐标系示意图

1.3 逃逸目标机动策略

本文借鉴路径规划中常用的人工势场法作为逃逸目标的逃逸策略. 假设各USV在逃逸目标的向量方向施加斥力, 各斥力分量随着二者间的距离增大而减小, 逃逸目标的速度方向向量$\vec{v}_M$表达式为

$$\vec{v}_M = \sum_i \left(\frac{\vec{p}_i - \vec{p}_M}{d_{i,M}^2} \right), \tag{2}$$

其中$\vec{p}_i(x_i, y_i)$和$\vec{p}_M(x_M, y_M)$分别为USV$_i$和逃逸目标M的位置.

2 多智能体强化学习围捕算法设计

USV协同围捕任务可描述为马尔可夫博弈(MDPs)[17], 用元组$\langle \mathcal{S}, \mathcal{A}, R, \mathcal{P}, \gamma \rangle$表示. 其中: $\mathcal{S}$为马尔可夫决策过程模型的状态集, $\mathcal{A} = A_1 \times A_2 \times \ldots \times A_N$为所有智能体联合动作集, $R = \{R_1, R_2, \ldots, R_N\}$为各智能体的奖励函数, $\mathcal{P}: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \boldsymbol{R}$为状态转移模型, γ为累计折扣奖励的衰减系数.

智能体的策略π是从状态到动作的映射概率, 在任意时刻t, 智能体i根据当前状态$s_t \in S_i$和策略$\pi(a|s)$选择动作$a_t \in A_i$, 并根据状态转移模型$\mathcal{P}$达到下一时刻状态$s_{t+1} \in S_i$, 同时获得奖励$r_{i,t} \in R_i$. 强化学习算法的目的是寻找最优策略π^*使得每个智能体的累计折扣奖励最大化, 有

$$R_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \gamma^{t+3} r_{i,3} + \ldots = \sum_{k=0}^{\infty} \gamma^k r_{i,t+k}. \tag{3}$$

本节给出了MAPPO多智能体强化学习算法的原理, 然后针对USV协同围捕问题, 详细设计了马尔可夫决策过程中的状态表示、奖励函数和动作空间, 最后介绍了算法实现流程.

2.1 MAPPO强化学习算法及原理

在多智能体系统中, 由于各智能体同时受到环境和其他智能体的影响, 使用单智能体强化学习算法的训练效果通常不理想, 需要使用联合行为值函数对智能体进行训练, MADDPG[9]、COMA[18]、QMIX[19]和MAPPO[16]等集中式训练分布式执行的方法被相继提出. 其中MAPPO (multi-agent proximal policy optimization) 是近端策略优化 (proximal policy optimization, PPO)[20-21]应用于多智能体任务的变种, 该算法采用Actor-Critic架构, 在有限算力的条件下具有较高的学习效率, 仅需进行少量超参数搜索工作便可达到主流MARL算法的性能[16], 在多种合作任务领域表现出了较高的水准, 因此本文采用MAPPO作为USV协同围捕问题的学习方法.

MAPPO算法是PPO算法多智能体形式的扩展, 具体方式是训练2个独立的网络, 分别为参数为θ的策略网络π_θ和参数为ϕ的价值网络V_ϕ, 它们对所有同构的智能体共享网络权重. 策略网络π_θ将智能体i的局部观测$s_{i,t}$映射至连续动作空间的分类分布; 价值网络$V_\phi: \mathcal{S} \to \boldsymbol{R}$将全局状态映射至状态价值的估计. 针对多智能体情况, 策略梯度损失函数$L^{\mathrm{CLIP}}(\theta)$改写为

$$L^{\mathrm{CLIP}}(\theta) = \frac{1}{BN} \sum_{i=1}^{B} \sum_{k=1}^{N} \min(c_{\theta,i}^{(k)} A_i^{(k)}, \mathrm{clip}(c_{\theta,i}^{(k)}, 1-\varepsilon, 1+\varepsilon) A_i^{(k)}). \tag{4}$$

其中: B和N分别为批处理数和集群数量, $c_{\theta,i}^{(k)} = \dfrac{\pi_\theta(a_i^{(k)}|s_i^{(k)})}{\pi_{\theta_{\mathrm{old}}}(a_i^{(k)}|s_i^{(k)})}$为多智能体形式的重要性采样权重, $A_i^{(k)}$为对应的优势函数[22].

相应地, 价值网络的损失函数$L^{\mathrm{CLIP}}(\phi)$为

$$L^{\mathrm{CLIP}}(\phi) = \frac{1}{BN} \sum_{i=1}^{B} \sum_{k=1}^{N} \max[(V_\phi(s_i^{(k)}) - \hat{R}_i)^2, (\mathrm{clip}(V_\phi(s_i^{(k)}), V_{\phi_{\mathrm{old}}}(s_i^{(k)}) - \varepsilon, V_{\phi_{\mathrm{old}}}(s_i^{(k)}) + \varepsilon) - \hat{R}_i)^2]. \tag{5}$$

2.2　状态空间设计

状态空间$\mathcal{S}$是智能体所能够获取到的全部信息,是智能体制定决策和评估其长期收益的依据,合理的状态空间设计能够确保MARL算法收敛. 针对围捕任务的特点,设计USV围捕状态特征并使用特征嵌入技术实现其伸缩和排列不变性.

2.2.1　特征设计

现有文献通常以智能体自身为参考系原点,选取逃逸目标和友方单位的距离、方位、速度等关系量作为特征[9-10,12-14]. 尽管上述特征包含了各实体间的位置关系,但是缺乏对“合围”状态的特征描述. 本文针对围捕成功条件,在位置关系特征的基础上加入了描述围捕角度的状态特征,建立了观测模型$\xi:\mathcal{S}\to\mathcal{O}$,根据全局状态$\boldsymbol{s}\{s_1,s_2,\dots,s_N\}$输出USV的全局观测$\boldsymbol{o}\{O^1,O^2,\dots,O^N\}$.

定义若两USV相对M的方位角间不存在其他USV,则称它们为相邻关系. 图3为U_i的状态特征示意图,其中U_j、U_k与U_i相邻$(i,j,k\in N)$.

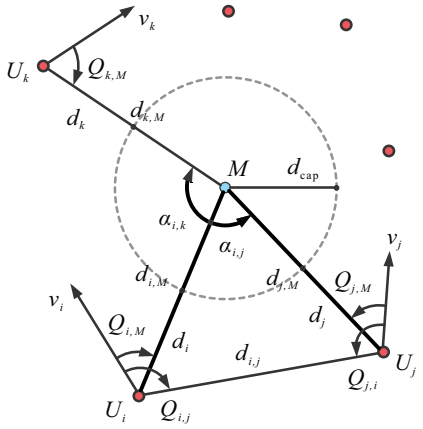


图3　USV状态特征示意图

定义U_i的状态特征O^i由本地特征o^i_{loc}、扩展特征o^i_{ext}和观测特征$o^{i,j}$组成.

o^i_{loc}描述USV自身的运动状态和对逃逸目标M的观测信息,包含逃逸目标M相对i的舷角$Q_{i,M}$及其变化率$\dot{Q}_{i,M}$;剩余捕获距离$d_i=d_{i,M}-d_{\text{cap}}$及其变化率$\dot{d}_i$;$i$的速度为$v_i$,加速度为$\dot{v}_i$.

o^i_{ext}描述USV集群相关的特征,包括i与相邻USV形成的围捕角差$\Delta\alpha_i=|\alpha_{i,j}|-|\alpha_{i,k}|$以及USV剩余捕获距离的均值$\bar{d}$.

$o^{i,j}$描述U_i对U_j的观测信息$(i\neq j)$,包括两艇间的距离$d_{i,j}$,相互间的舷角$Q_{i,j}$和$Q_{j,i}$,两USV剩余捕获距离差$\Delta d_{i,j}=d_i-d_j$,围捕角$\alpha_{i,j}$.

综上,U_i的状态特征O^i可表示为如下形式:

$$O^i=\{o^i_{\text{loc}},o^i_{\text{ext}},o^{i,j}\},\ i,j\in N,i\neq j, \tag{6}$$

$$\begin{cases} o^i_{\text{loc}}=\{Q_{i,M},\dot{Q}_{i,M},d_i,\dot{d}_i,v_i,\dot{v}_i\},\\ o^i_{\text{ext}}=\{\Delta\alpha,\bar{d}\},\\ o^{i,j}=\{d_{i,j},Q_{i,j},Q_{j,i},\Delta d_{i,j},\alpha_{i,j}\}. \end{cases} \tag{7}$$

2.2.2　伸缩和排列不变性设计

为了将多智能体的观测信息$o^{i,j}$输入网络,最常见的处理方式是将其串联为一维向量. 但是这一方法存在明显不足:观测信息的维度与集群数量成线性关系,使得对于每种无人艇起始数量的情况均需要独立训练策略网络. 因此,串联为一维向量的网络输入,无法灵活处理输入维度动态变化的应用场景,这一处理方式也忽略了集群系统中各智能体的排列不变性,导致网络模型缺乏伸缩性,难以扩展到大规模集群系统.

文献[14]试图通过对观测信息按照角度属性排序的方式来处理排列不变性,但是未解决伸缩性问题. 文献[23]将观测目标的空间位置映射至距离-方位二维网格中,同时兼顾了伸缩和排列不变的问题,但是栅格化的位置信息存在精度损失. 本文引入一种基于学习的状态特征表示方法[13],表达式如下:

$$\phi^{NN}(O^i)=\frac{1}{|O^i|}\sum_{o^{i,j}\in O^i}\phi(o^{i,j}). \tag{8}$$

其中:$\phi^{NN}(O^i)$为智能体i的状态特征O^i映射;$\phi(o^{i,j})$为智能体i对j的观测的$o^{i,j}$映射,采用单层前馈神经网络实现. $\phi(o^{i,j})$的表达式如下:

$$\phi(o^{i,j})=h(Wo^{i,j}+b). \tag{9}$$

其中:W和b为网络权重和偏置,h为激活函数.

图4为USV状态特征网络的结构. 首先,将USV的观测输入权重相同的前馈网络,得到各观测的特征向量;然后,对所有特征向量求均值,即可得到与观测数量无关的维度特征;最后,将USV的本地特征o^i_{loc}和扩展特征o^i_{ext}与其拼接,得到长度为72的一维特征向量.

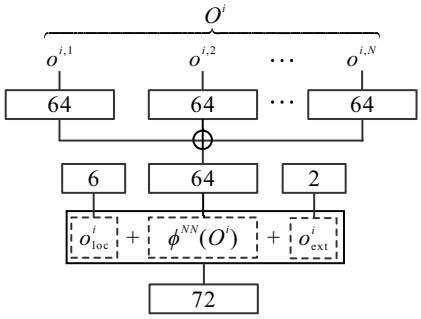


图4　USV状态特征网络示意图

该方法通过对特征取均值的方式解决了网络伸缩性和排列不变性的问题,$\phi(o^{i,j})$的权重也将在训练

中不断学习,从而实现观测信息特征的自动提取.

2.3 奖励函数设计

基于第1.1节中的围捕目标和约束条件,通过设计合适的奖励函数R来评估USV当前的状态. 在任务过程中,各USV在每个时间步长均会获取一次奖励,旨在引导USV追逐逃逸目标并通过协同合作形成围捕态势; 在任务完成时刻,所有USV均会获得一次性任务奖励. USV$_i$的奖励r_i由奖励函数R_i定义,有

$$R_i=\begin{cases} r_{\text{cap}},\ d_{i,M}\leqslant d_{\text{cap}};\\ r_{\text{help}},\ d_{j,M}\leqslant d_{\text{cap}},\exists j\neq i;\\ r_{\text{step}},\ \text{otherwise}. \end{cases}\tag{10}$$

当USV$_i$率先与逃逸目标的距离小于围捕距离d_{cap}时,获取捕获奖励r_{cap}; 当除USV$_i$外任意USV满足围捕距离条件时,获取协同奖励r_{help}; 其他情况时,获取步长奖励r_{step}.

步长奖励r_{step}由多种子奖励加权构成,即

$$r_{\text{step}}=\sum_k w_k r_k,\tag{11}$$

$$\sum_k w_k=1.\tag{12}$$

其中: w_k为各子奖励函数r_k的权重,r_k的定义如下:

1) 围捕距离奖励r_1为

$$r_1=-k_1 d_i.\tag{13}$$

为确保任务的时效性,每时间步长USV均会收到负回报. 式(13)中: r_1与剩余围捕距离d_i成线性关系,从而引导USV接近逃逸目标; k_1为调节系数.

2) 距离一致性奖励r_2为

$$r_2=\mathrm{e}^{-k_2(d_i-\bar{d})/\sigma(d)}-1.\tag{14}$$

为改善围捕过程中USV与整体“掉队”的情况,要求各USV与逃逸目标的距离接近. 建立了负指数形式的奖励函数,将r_2约束在$(-1,0]$间. 式(14)中: $\bar{d}$和$\sigma(d)$分别为所有USV围捕距离的均值和标准差,k_2为调节系数.

3) 围捕角度奖励r_3为

$$r_3=\min(\mathrm{e}^{-k_3|\alpha_{i,j}-2\pi/N|},\mathrm{e}^{-k_3|\alpha_{i,k}-2\pi/N|})-1.\tag{15}$$

理想状态下,USV集群形成对目标的包夹态势,此时相邻USV间的最优围捕角$\alpha^*=2\pi/N$. 根据USV与左右相邻的围捕角与α^*的差距,建立负指数形式的奖励函数,取二者最小值作为围捕角度奖励,$r_3\in(-1,0]$,k_3为调节系数.

4) 围捕角度一致性奖励r_4为

$$r_4=\mathrm{e}^{-k_4\Delta\alpha_i}-1.\tag{16}$$

当USV的相邻围捕角差$\Delta\alpha_i$为0时,可认为USV与逃逸目标的角度关系是局部最优的,建立负指数形式的奖励函数r_4描述USV的围捕角度一致性,$r_4\in(-1,0]$,k_4为调节系数.

5) 碰撞奖励r_5为

$$r_5=-\mathrm{e}^{-k_5 d_{\min}}.\tag{17}$$

建立负指数形式的奖励函数r_5描述USV间的碰撞风险,$d_{\min}=\min\limits_j d_{i,j}$为USV$_i$与其他USV最近的距离,$r_5\in(-1,0]$,$k_5$为调节系数.

综上,步长奖励r_{step}中的各项子奖励均设置为负值,且USV间形成的协同态势越接近理想状态时,r_{step}的值越趋近0,从而能够引导USV更新到较优的协同策略; 当围捕任务完成时,所有USV会得到正回报,使得USV集群达到快速围捕的目的.

2.4 动作空间设计

动作空间的设计应简单高效,从而有效降低训练难度并提高算法性能. 一般参照智能体运动学模型的控制维度设计动作空间. 完整性约束的运动学模型通常策略网络输出加速度矢量[9,12],无人机(UAV)、无人车(UGV)等非完整性约束的运动学模型可简化为一阶系统,输出速度和角速度量[14,24]. USV的控制响应较慢,通常采用二阶系统描述,控制量为加速度a_v和角加速度a_ω,然而a_v、a_ω与奖励函数R_i间存在复杂的耦合关系,直接使用a_v和a_ω作为网络输出量,网络学习效率和收敛性难以得到保证.

本文设计了一种适用于USV围捕问题的动作空间,将任务目标解耦为围捕距离控制和围捕角度控制,USV策略网络的输出$\boldsymbol{a}$由对应的距离控制量a_d和角度控制量a_α组成,具体如图5所示.

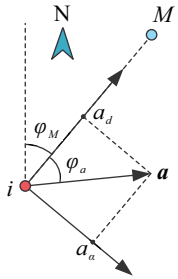


图5 USV动作空间设计示意图

定义距离控制量a_d为USV$_i$与逃逸目标M在连线方向上的控制分量,角度控制量a_α为垂直于a_d,以M为圆心顺时针方向上的控制分量,a_d、$a_\alpha\in[-1,1]$. 定义φ_M为M相对i的方位角,φ_a为$\boldsymbol{a}$相对iM方向的夹角. 定义期望航速v_{exp}和期望航向ψ_{exp}表达式为

$$\begin{cases} v_{\text{exp}}=v_{\max}\|a\|,\\ \psi_{\text{exp}}=\varphi_M+\varphi_a. \end{cases}\tag{18}$$

随即引入USV跟踪控制算法[25],输入期望航向$\psi_{\exp}$、期望航速$v_{\exp}$,间接获取USV控制量a_v和a_ω的控制值. 所提出方法将网络输出与围捕目标相结合,避免策略网络介入USV运动学控制与奖励函数间的复杂耦合关系,简化了系统的决策难度.

2.5　围捕算法流程设计

本文采用集中式训练、分布式执行的算法训练框架,该框架流程如图6所示.

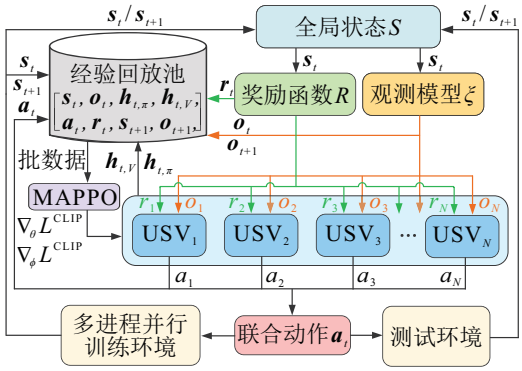


图 6　集中训练分布式执行框架

在训练阶段时,并行训练环境输出全局状态$\boldsymbol{s}_t$;然后,奖励函数R和观测模型ξ根据$\boldsymbol{s}_t$分别输出该回合各USV获取的全局奖励$\boldsymbol{r}_t$和全局观测$\boldsymbol{o}_t$,USV策略网络根据$\boldsymbol{o}_t$输出联合动作$\boldsymbol{a}_t$;最后,并行训练环境输入各USV的$\boldsymbol{a}_t$,给出更新后的全局状态$\boldsymbol{s}_{t+1}$,从而完成一次交互循环. 在每次循环过程中,中间状态数据以$[\boldsymbol{s}_t, \boldsymbol{o}_t, \boldsymbol{h}_{t,\pi}, \boldsymbol{h}_{t,V}, \boldsymbol{a}_t, \boldsymbol{r}_t, \boldsymbol{s}_{t+1}, \boldsymbol{o}_{t+1}]$的形式存储于经验回放池中(考虑到USV协同围捕决策过程具有强时序性,本文引入了RNN层,使得网络具有记忆能力,能够结合历史状态数据做出决策,$\boldsymbol{h}_{t,\pi}$和$\boldsymbol{h}_{t,V}$分别为策略网络和价值网络中的RNN状态数据),待收集到的经验达到指定容量时,计算策略网络和价值网络的更新梯度,并用Adam优化器更新网络权重.

测试阶段的流程相对简单,观测模型ξ根据全局状态$\boldsymbol{s}_t$输出各USV的全局观测$\boldsymbol{o}_t$;然后,USV根据$\boldsymbol{o}_t$输出联合动作$\boldsymbol{a}_t$;最后,测试环境根据$\boldsymbol{a}_t$更新全局状态,从而完成循环.

3　仿真实验和分析

本节建立仿真围捕环境,使用所提出多智能体强化学习围捕算法训练USV策略网络;然后分析该方法的有效性并与其他算法的性能指标作对比;最后验证该算法的鲁棒性以及课程式学习对网络训练的影响.

3.1　实验环境与参数设置

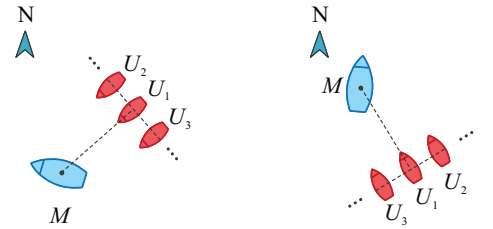
实验平台的硬件配置为CPU i9-10 980XE, GPU RTX 2 080 Ti, 内存128 GB, 使用Pytorch作为深度学习训练框架, OpenAI Gym作为强化学习环境框架.

特别地,对多USV协同围捕的实验条件设置作以下说明.

1) 所有USV均为同构无人艇,即性能参数一致,且USV间能够相互通信并共享逃逸目标的位置.

2) 由于海洋环境无需考虑地理边界约束,实验环境区域范围不设限,为确保存在可行解,只考虑高速围捕USV和低速逃逸目标的追逃对抗情景.

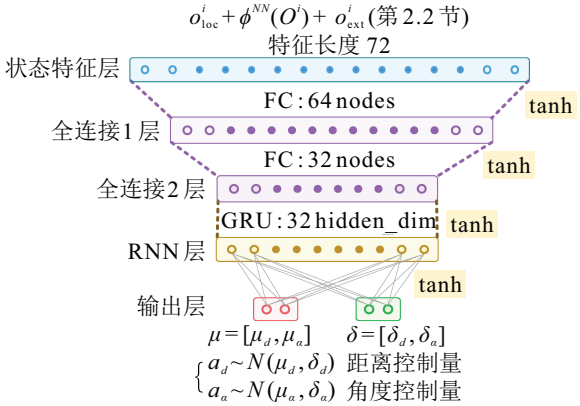
3) 为不失一般性,在每次实验开始时USV集群相对于逃逸目标的起始距离和方位随机生成,USV集群初始队形为面朝逃逸目标的单横队,如图7所示.



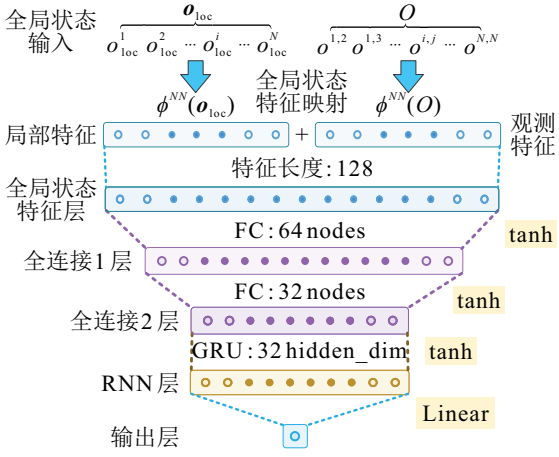
(a)　初始状态示意1　　(b)　初始状态示意2

图 7　多USV围捕环境初始状态示意图

4) 由于各USV具有同构性和相互替代性,将所有USV设置为共享相同的策略,MAPPO算法的策略



(a)　策略网络结构



(b)　价值网络估计

图 8　MAPPO算法网络结构设计

网络π_θ和价值网络V_ϕ的结构设计如图8所示. 图8中: 策略网络π_θ的输入为第2.2节中的状态特征, 经过全连接层和RNN层后输出距离控制量a_d、角度控制量a_α的高斯分布均值$\boldsymbol{\mu}=[\mu_d,\mu_\alpha]$以及标准差$\boldsymbol{\delta}=[\delta_d,\delta_\alpha]$. 在训练时, 控制量$a_d$和$a_\alpha$可通过对高斯分布采样获得; 测试时, a_d和a_α取值分别为μ_d和μ_α. 价值网络V_ϕ的输入为全局状态$\boldsymbol{o}_{\rm loc}$和O, 通过第2.2节状态特征映射得到长度均为64的局部特征和观测特征, 拼接后组成长度为128的全局状态特征, 后续网络结构与π_θ一致, 输出状态价值的估计值.

5) 在网络训练环节, 本文采用了课程式学习的训练技巧来缓解奖励稀疏的问题. 首先从简单的围捕任务开始训练, 然后逐步增加难度, 直至与实际难度一致. 决定围捕问题的难度因素包含逃逸目标相对速度和围捕半径, 本文采取在训练初期将目标速度降低, 围捕半径加大的方法, 从而使得USV能够较为容易地达到任务终止条件并鼓励USV在初始学习阶段时尝试探索更加复杂的协同行为. 具体参数设置如下: 围捕半径初始比例系数c_d为1.5, 逃逸目标航速初始比例系数c_v为0.2, 课程式学习训练比例c_p为0.5. 本文建立了线性难度的参数调度器, 能够根据当前训练周期动态调整训练难度. 以围捕半径的参数设置为例, 设当前训练步数为E_t, 参数调度器输出的实际围捕半径$d^*_{\rm cap}$表达式为

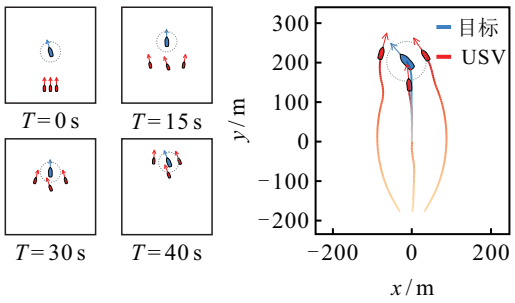
$$d^*_{\rm cap}=\begin{cases} d_{\rm cap}[c_d+(1-c_d){\rm ratio}], & E_t/E<c_p;\\ d_{\rm cap}, & {\rm otherwise}. \end{cases} \tag{19}$$

其中ratio$=E_t/c_pE$为比例系数.

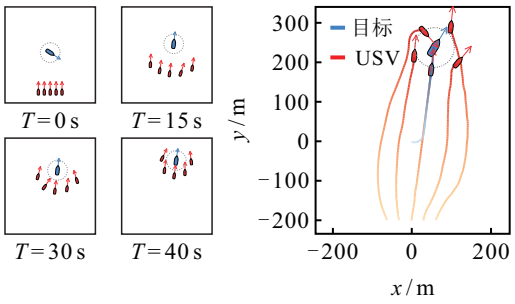
3.2 方法有效性分析

本节在USV起始数量不同的条件下分别测试协同围捕算法的有效性. 考虑到状态特征的设计具有可伸缩性, 因此理论上只需针对某一种USV数量进行训练, 其策略网络的权重便可兼容于任意数量的USV协同围捕任务, 但是出于算法性能对比的需要, 对于USV数量在3～9间的每种情况均独立完成训练, 分别对应7种情况下的最优策略.

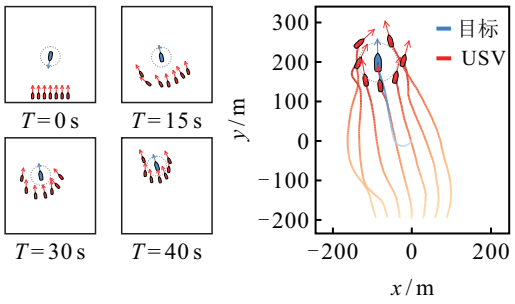
图9为USV数量分别为3、5和7时围捕过程的样本数据记录. 图9中: 蓝色和红色船只分别为逃逸目标和USV, 其中左侧的4幅小图为各时刻的围捕态势, 右侧大图为追逃双方围捕的全过程, 蓝色和红色渐变线分别为逃逸目标和USV的轨迹, 反映各USV的轨迹时序关系. 当任意USV进入逃逸目标的围捕半径(虚线区域表示)时, 围捕任务随即结束. 通过观察图9中的围捕轨迹, 可发现围捕成功时各USV与逃逸目标的距离一致接近, 同时USV间的围捕角度也趋于一致, USV通过协同合作实现了对目标的合围. 此外, 图9(c)中存在USV历史轨迹部分重叠的情况, 但是结合轨迹时序关系可知USV间未发生碰撞. 其他USV数量的协同围捕情况与上述描述基本一致, 由此可认为在USV起始数量不同的情况下, 所提出方法均能够对逃逸目标实施有效围捕.



(a) 3 USV协同围捕



(b) 5 USV协同围捕



(c) 7 USV协同围捕

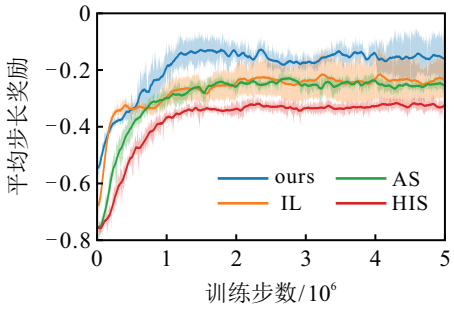
图9 不同数量USV协同围捕过程

3.3 性能对比分析

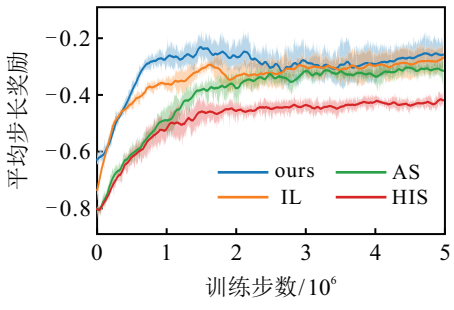
本文选取了3种其他算法作为对比, 分别为IL (independent learning)方法[20]、AS (angle sort)方法[14]和HIS (Histogram)方法[23]. IL方法使用传统单智能体的PPO算法, 训练时智能体价值网络的输入与策略网络相同, 为局部状态观测, 其他参数保持不变; AS方法将全部观测信息串联为一维特征, 这些特征包含USV对任意智能体或围捕目标的距离、夹角等参数. 为解决观测特征的排列不变性, 该方法按照角度属性对观测信息排序; HIS方法将观测目标的位置映射至距离-方位二维网格中, 使用网格图像的形式描述观测目标的空间位置特征和排列不变性. 实验使用的网格尺寸设置为12×12, 网格分辨率设置

为 20 m. 4 种方法均采用相同训练和环境参数, 针对 USV 数量 3～9 每种情况使用不同随机数种子各训练 5 次以确保结果的准确性.

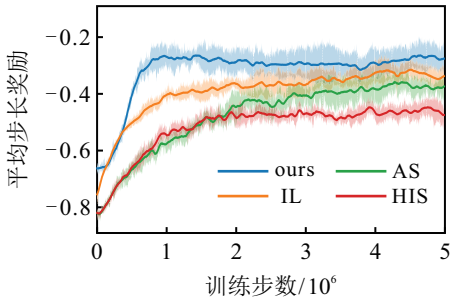
图 10 为不同 USV 数量下各算法的训练奖励曲线. 通过对比发现所提出方法能够迅速收敛, 并在多种算法中取得最佳的平均步长奖励, IL 方法次之, 仅在 USV 数量为 5 时能够接近所提出方法的性能, AS 和 HIS 方法收敛速度相对较慢, 在训练后半段平均奖励值仍然有上升趋势. 所提出方法与 IL 方法间的对比可反映出 MARL 算法的集中式训练分布式执行架构相对于单智能体算法的性能优势. 此外, 所提出方法在不同 USV 数量下均显著优于另外两种方法, 这可解释为 AS 的状态特征编码中的观测排序取决于观测目标 (其他 USV 和围捕目标) 与 USV 的夹角, USV 数量越多, 观测排序的序号变化会更加频繁, 这破坏了状态特征的排列不变性, 增大了网络的收敛难度; HIS 方法尽管编码了目标位置信息, 但是在栅格化编码的过程中会产生精度损失, 导致控制的精准度欠缺. 相比于 AS 与 HIS 方法, 本文针对围捕问题设计的基于学习的特征提取方法具有更佳的特征编码能力, 能够适应不同 USV 数量的围捕情况.



(a) 3 USV



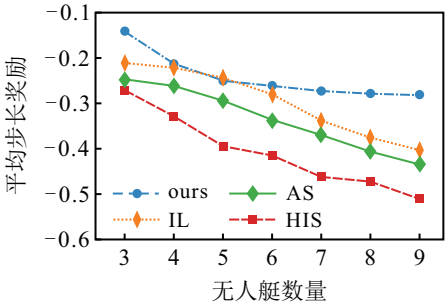
(b) 5 USV



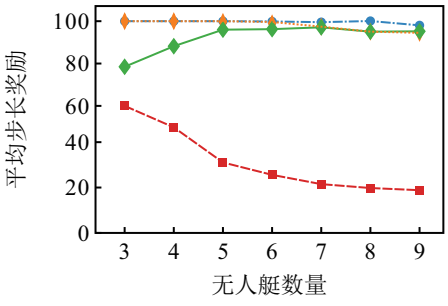
(c) 7 USV

图 10　不同算法训练奖励曲线对比

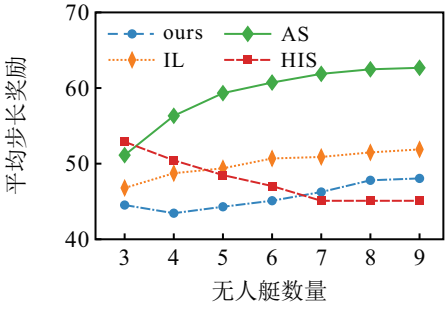
图 11 进一步展示了 4 种算法在 USV 各起始数量下的平均步长奖励、围捕成功率和平均捕获时间的变化情况. 由图 11(a) 可见, 随着 USV 数量的增多, 各算法的平均步长奖励均呈现下降趋势, 这种情况可解释为随着需要协同的智能体数量增加, USV 的协同难度与形成围捕队形奖励的获取难度也相应增加. 通过对比同一 USV 数量下的奖励可发现, 所提出方法优于其他方法, 与图 10 反映的结果相一致. 图 11(b) 为围捕成功率的变化, 随着 USV 数量的增加, 所提出方法围捕成功率基本保持接近 100 % 的水平, 其他方法均出现了成功率下降的情况, 其中 HIS 方法的下降最为显著, 原因是 HIS 的状态特征编码方法在 USV 数量增加时, 更易发生观测目标的距离和角度模糊, 增加了决策难度; AS 方法的成功率呈现过上升趋势, 可解释为 USV 数量增加降低了围捕难度, 然而数量的进一步增加也同时提高了 USV 协同的难



(a) 平均步长奖励



(b) 围捕成功率



(c) 平均捕获时间

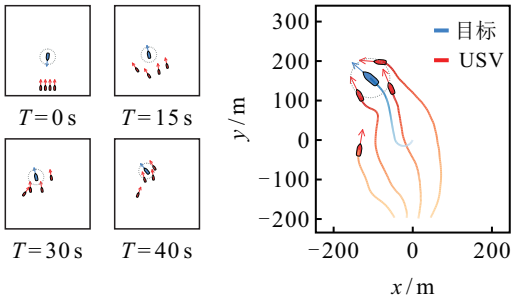
图 11　各算法在不同 USV 数量下的性能指标对比

度,因此成功率又小幅下降. 图11(c)反映了围捕时间随USV数量的变化情况,HIS方法在USV数量较多的情况下耗时较短,因为该方法出现了网络退化的现象,USV的策略均退化为径直追踪逃逸目标,这导致围捕成功率低,无法实现有效围捕. 其他方法的平均捕获时间总体呈上升趋势,可解释为USV数量越多,USV形成合围态势需要更多的占位机动时间. 通过比较可发现,所提出方法相对于另外两种方法在围捕时间上也具有较明显的优势.

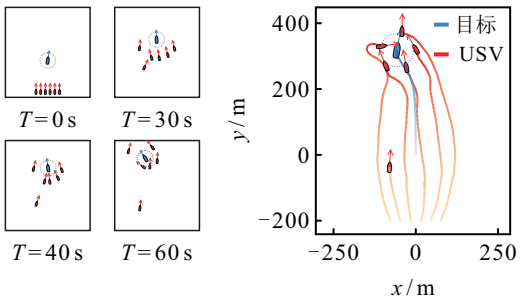
3.4 鲁棒性分析

为进一步探究算法鲁棒性,本文采取节点失效的方式模拟USV损毁,具体处理方法是在围捕过程中随机选取任意USV_i,使其油门和舵角控制设置为0,并切断其他USV对i的观测,验证剩余USV是否具备继续执行任务的能力.

实验选择了4 USV和6 USV两种情况,在$T = 20$ s时随机失效一个USV节点,抽取的围捕过程样本数据记录如图12所示.



(a) 4 USV协同围捕



(b) 6 USV协同围捕

图 12 发生节点损毁时USV协同围捕过程

图12(a)中:当左侧第1艘USV失效后,剩余USV仍然能够形成对目标的包围态势;图12(b)中:左侧第2艘USV失效后,在$T = 60$ s时刻观察到左侧第1艘USV自主完成了补位并行驶至逃逸目标前方实施堵截,以增加任务耗时为代价成功实现了对目标的围捕.

3.5 课程式学习效果分析

本节分析课程式学习训练技巧对网络训练的影响,实验设置USV数量为3. 作为对比的非课程式学习方法在训练时难度固定,即逃逸目标最大航速和围捕半径全程保持不变. 对比结果如图13所示.

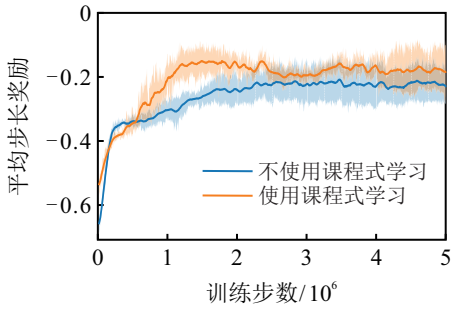


图 13 课程式学习训练对平均步长奖励影响

图13的结果表明,若不使用课程式学习,即便训练初期的奖励增长迅速,但是随后收敛速度显著降低;使用课程式学习训练方法的奖励曲线增长较为平缓,并最终能够保持在较佳的水准. 根据观察训练过程,上述现象可解释为训练初期,较低的任务难度可使得智能体具备充足的探索时间来学会协同包围逃逸目标,若训练初期目标全速逃逸,则USV需要同时兼顾学习追逐和包围两种行为,学习难度曲线较大. 因此,针对USV围捕问题使用课程式训练的学习技巧可在相同情况下获得较优的收敛速度和平均步长奖励.

4 结 论

本文提出了一种适用于USV协同围捕问题的多智能体强化学习方法. 首先,给出问题模型和约束,基于MAPPO建立马尔可夫博弈问题框架;然后,详细设计强化学习中的状态空间、奖励函数和动作空间等组成要素;最后,构建了仿真环境并完成围捕策略的训练. 仿真实验表明:1) 在不同数量的USV协同围捕任务中,所提出方法均可快速完成目标围捕且形成合围态势. 2) 相较于其他典型算法,所提出方法在学习效率、围捕平均时长和围捕成功率等方面更具优势. 3) 当部分节点损毁时,其他幸存无人艇节点仍然具备继续执行协同围捕任务的能力,所提出方法的鲁棒性较强,具有推广实际应用的潜力. 4) 使用课程式学习的训练技巧有利于提高USV协同围捕问题的学习效果,亦可为其他多智能体协同问题的网络训练提供参考.

参考文献(References)

[1] 王石, 张建强, 杨舒卉, 等. 国内外无人艇发展现状及典型作战应用研究[J]. 火力与指挥控制, 2019, 44(2): 11-15.
(Wang S, Zhang J Q, Yang S H, et al. Research on development status and combat applications of USVs in worldwide[J]. Fire Control & Command Control, 2019,

44(2): 11-15.)

[2] Kevin M, Mary J. Unmanned systems integrated roadmap 2017-2042[EB/OL]. (2018-08-30) [2022-0308]. http://defensedaily.com/wp-content/uploads/post_attachment/206477.pdf.

[3] Shneydor N A. Missile guidance and pursuit: Kinematics, dynamics and control[M]. Cambridge: Elsevier, 1998: 104-105.

[4] Sun W, Tsiotras P, Lolla T, et al. Multiple-pursuer/one-evader pursuit-evasion game in dynamic flowfields[J]. Journal of Guidance, Control, and Dynamics, 2017, 40(7): 1627-1637.

[5] Muro C, Escobedo R, Spector L, et al. Wolf-pack (Canis lupus) hunting strategies emerge from simple rules in computational simulations[J]. Behavioural Processes, 2011, 88(3): 192-197.

[6] Janosov M, Virágh C, Vásárhelyi G, et al. Group chasing tactics: How to catch a faster prey[J]. New Journal of Physics, 2017, 19(5): 053003.

[7] 符小卫, 王辉, 徐哲. 基于DE-MADDPG的多无人机协同追捕策略[J]. 航空学报, 2022, 43(5): 530-543.
(Fu X W, Wang H, Xu Z. Cooperative pursuit strategy for multi-UAVs based on DE-MADDPG algorithm[J]. Acta Aeronautica et Astronautica Sinica, 2022, 43(5): 530-543.)

[8] 王泊涵, 吴婷钰, 李文浩, 等. 基于多智能体强化学习的大规模无人机集群对抗[J]. 系统仿真学报, 2021, 33(8): 1739-1753.
(Wang B H, Wu T Y, Li W H, et al. Large-scale UAVs confrontation based on multi-agent reinforcement learningrevoke[J]. Journal of System Simulation, 2021, 33(8): 1739-1753.)

[9] Lowe R, Wu Y I, Tamar A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments[J]. Advances in Neural Information Processing Systems, 2017, 30: 6379-6390.

[10] Xu L, Hu B, Guan Z H, et al. Multi-agent deep reinforcement learning for pursuit-evasion game scalability[C]. Proceedings of Chinese Intelligent Systems Conference. Haikou, 2020: 658-669.

[11] Schuster M, Paliwal K K. Bidirectional recurrent neural networks[J]. IEEE Transactions on Signal Processing, 1997, 45(11): 2673-2681.

[12] Wang Y D, Dong L, Sun C Y. Cooperative control for multi-player pursuit-evasion games with reinforcement learning[J]. Neurocomputing, 2020, 412: 101-114.

[13] Hüttenrauch M, Šošić A, Neumann G. Deep reinforcement learning for swarm systems[J]. Journal of Machine Learning Research, 2019, 20(54): 1-31.

[14] de Souza C, Newbury R, Cosgun A, et al. Decentralized multi-agent pursuit using deep reinforcement learning[J]. IEEE Robotics and Automation Letters, 2021, 6(3): 4552-4559.

[15] Fujimoto S, Hoof H, Meger D, Addressing function approximation error in actor-critic methods[J/OL]. 2018, arXiv: 1802.09477.

[16] Yu C, Velu A, Vinitsky E, et al. The surprising effectiveness of MAPPO in cooperative, multi-agent games[J/OL]. 2021, arXiv: 2103.01955.

[17] Littman M L. Markov games as a framework for multi-agent reinforcement learning[C]. Proceedings of the International Conference on Machine Learning. San Francisco, 1994: 157-163.

[18] Foerster J, Farquhar G, Afouras T, et al. Counterfactual multi-agent policy gradients[C]. Proceedings of the AAAI Conference on Artificial Intelligence. New Orleans, 2018: 2974-2982.

[19] Rashid T, Samvelyan M, Schroeder C, et al. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning[C]. International Conference on Machine Learning. Stockholm, 2018: 4295-4304.

[20] Schulman J, Wolski F, Dhariwal P, et al. Proximal policy optimization algorithms[J/OL]. 2017, arXiv: 1707.06347.

[21] Schulman J, Levine S, Abbeel P, et al. Trust region policy optimization[C]. International Conference on Machine Learning. Lille, 2015: 1889-1897.

[22] Schulman J, Moritz P, Levine S, et al. High-dimensional continuous control using generalized advantage estimation[J/OL]. 2015, arXiv: 1506.02438.

[23] Hüttenrauch M, ŠošićA, Neumann G. Local communication protocols for learning complex swarm behaviors with deep reinforcement learning[C]. International Conference on Swarm Intelligence. Shanghai, 2018: 71-83.

[24] Xu G, Zhao Y, Liu H. Pursuit and evasion game between UVAs based on multi-agent reinforcement learning[C]. Chinese Automation Congress. Hangzhou, 2019: 1261-1266.

[25] 伊戈, 刘忠, 张建强, 等. 基于改进终端滑模控制的USV航向跟踪控制方法[J]. 电光与控制, 2020, 27(10): 12-16.
(Yi G, Liu Z, Zhang J Q, et al. A USV heading tracking control method based on improved terminal sliding mode control[J]. Electronics Optics & Control, 2020, 27(10): 12-16.)

作者简介

夏家伟 (1994−), 男, 博士生, 从事无人艇智能控制、多智能体强化学习等研究, E-mail: 491650471@qq.com;

朱旭芳 (1978−), 女, 副教授, 博士, 从事无人艇传感器技术、电路与系统等研究, E-mail: 1580284687@qq.com;

张建强 (1980−), 男, 副教授, 博士, 从事无人艇航行控制、通信与人工智能等研究, E-mail: jianqiang97176@163.com;

罗亚松 (1982−), 男, 副教授, 博士, 从事无人作战系统、舰艇指控系统等研究, E-mail: yours_baggio@sina.com;

刘忠 (1963−), 男, 教授, 博士生导师, 从事系统工程、无人系统作战与应用等研究, E-mail: liuz531@163.com.

(责任编辑：魏 冰)