http://bhxb.buaa.edu.cn jbuaa@buaa.edu.cn
DOI: 10.13700/j.bh.1001-5965.2024.0582

基于 AR-PPO 的端到端无人机自主导航方法

李涵玮[1]，李新凯[1,*]，张海[2]，姜凌峰[1]，孟月[3]，张宏立[3]

(1. 新疆大学 电气工程学院，乌鲁木齐 830017；2. 新疆大学 工程训练中心，乌鲁木齐 830017；
3. 新疆大学 智能科学与技术学院（未来技术学院），乌鲁木齐 830017)

摘　　要：针对基于深度强化学习 (DRL) 的无人机导航方法面临的部分环境不可观测、训练难度高等问题，提出一种基于动作屏蔽-循环-近端策略优化 (AR-PPO) 的端到端无人机自主导航方法。引入门控循环神经网络处理历史信息，为策略网络引入历史基准信息以解决复杂环境下部分信息不可观测问题；设计适用于连续动作空间强化学习任务的动作屏蔽方法，排除非法动作的干扰并加快模型训练速度；建立演员-评论家框架的三维连续动作空间 DRL 端到端模型并于训练环境进行训练。与主流 DRL 方法进行对比实验，AR-PPO 训练收敛速度最快，能取得最高奖励，在 4 种泛化测试环境中均取得了 85% 以上的成功率，在实机实验中能够安全地进行避障导航任务，表明所提方法能够有效实现无人机的精确避障与安全导航。

关 键 词：自主导航；深度强化学习；近端策略优化；复杂环境；无人机
中图分类号：V279; TP18
文献标志码：A　　**文章编号**：1001-5965(2026)11-3982-13

无人机作为一种能够飞行的特殊机器人，因其体积小、动作灵活、适宜在复杂环境下完成机动，从而广泛应用于各种军用及民用领域[1]。现有大多数导航方法均是基于地图的导航，包括 SLAM 建图、全局路径规划、局部路径规划、动作执行等方法[2]。这种基于地图的导航方法已广泛应用于各类移动机器人，但该方法严重依赖高精度的地图，各环节耦合性强，任何一个环节出错均会导致导航失败[3]。

由于无人机载荷的限制，使其难以独立实现快速精确的建图与规划，近年来，学者们开始开展不依赖地图的导航研究[4]，尝试用深度强化学习 (deep reinforcement learning, DRL) 构建导航策略，将机器人的运动策略用神经网络表示，通过机器人与环境不断交互试错的方式学习运动策略[5]。Singla 等[6] 使用深度 Q 网络（deep Q-network，DQN）并加入具有时间注意力的循环网络，使其算法拥有更高的避障能力；魏瑶等[7] 将门控循环单元（gated recurrent unit, GRU）引入双深度 Q 网络（double deep Q-network, DDQN）处理时间维度信息，提高了算法在部分可观测环境中的性能。但上述研究为无人机预设了简单的前进、后退、转向等离散动作，降低了训练难度。显然，离散动作空间使无人机的导航与避障能力受到了极大的限制，不利于无人机自主导航能力的提升。

部分学者将无人机拓展到三维连续动作空间，使其能够自由选择飞行速度或飞行距离，同时拥有更加平滑的飞行动作和更合理的飞行轨迹。张耀中等[8] 基于深度确定性策略梯度（deep deterministic policy gradient, DDPG）算法研究了无人机对敌方来袭目标的追击问题，经训练后无人机能较好地完成追击任务；唐蕾等[9] 使用改进双延迟深度确定性策

收稿日期：2024-08-02；**录用日期**：2024-10-30；**网络出版时间**：2024-11-07 14:00
网络出版地址：link.cnki.net/urlid/11.2625.V.20241107.1129.002
基金项目：国家自然科学基金 (62263030)；新疆维吾尔自治区自然科学基金 (2023D01C187)；新疆维吾尔自治区“天山英才”培养计划 (2023TSYCCX0037)
* **通信作者**. E-mail：lxk@xju.edu.cn

引用格式：李涵玮，李新凯，张海，等. 基于 AR-PPO 的端到端无人机自主导航方法 [J]. 北京航空航天大学学报，2026，52（11）：3982-3994.
Li H W，Li X K，Zhang H，et al. End-to-end autonomous navigation method for drones based on AR-PPO [J]. Journal of Beijing University of Aeronautics and Astronautics，2026，52（11）：3982-3994（in Chinese）.

略梯度(twin-delayed deep deterministic policy gradient, TD3)算法实现了无人机连续三维动作空间避障, 结果表明, 该算法效果优于 DDPG 算法; 张云燕等[10]使用 DDPG 算法实现了依靠深度相机的无人机连续三维动作空间避障, 实验证明 DDPG 算法的避障效果优于三维向量场直方图算法。上述研究都是基于无人机当前状态完全已知, 在简单环境下可行, 但在复杂场景中, 部分状态不可测时将导致无人机导航性能降低。

一些学者对考虑历史信息的无人机强化学习导航方法进行了研究。赵静等[11]基于近端策略优化(proximal policy optimization, PPO)算法实现了上述无人机避障方法, 但其动作空间为坐标轴方向的线速度, 无人机仅能面向初始方向飞行, 无法实现转向掉头等动作, 飞行场景为规则的管道之内, 飞行距离短, 场景简单。寇凯等[12]同样采用了上述方法, 但选择方形、柱形等简单障碍物, 无法充分表现无人机在复杂场景的自主避障优势。

综上所述, 行之有效的 DRL 无人机自主导航方法应当至少包含以下内容: 被设计于三维连续动作空间与观测空间下, 以获得更高的动作灵活性; 考虑历史信息, 以解决部分环境信息不可观测问题; 训练于复杂环境下并进行泛化实验, 以确保策略的有效性和泛用性。然而, 三维的任务空间、历史信息和复杂环境必然会增加训练时间与训练难度, 因此, 需要采取措施提高 DRL 算法的训练速度。

由此, 本文提出了一种适用于复杂场景基于动作屏蔽-循环-近端策略优化(action masking-temporal-proximal policy optimization, AR-PPO)的端到端自主导航方法, 主要创新有:

1) 在三维观测空间与动作空间中设计了连续动作的线速度、角速度控制方式, 赋予无人机良好的转向与掉头能力, 并拥有更加平滑的飞行轨迹和更高效的避障能力。

2) 为 PPO 模型引入 GRU 网络处理历史信息以解决飞行环境信息部分不可观测问题, 提高强化学习模型稳定性, 并使模型能够完成更加复杂的任务。

3) 为解决复杂环境带来的训练难度大和训练时间长等问题, 结合基于奖励的动作软约束与离散动作空间下的动作掩码硬约束, 设计了适用于连续动作空间的动作屏蔽方法, 通过减少非法动作的干扰以提高训练速度。

1　基于 DRL 的无人机导航策略

无人机导航是指无人机能够从起点出发, 依靠自身传感器执行动作躲避障碍物并安全地运动至目标位置。这一过程可以使用马尔可夫决策过程(Markov decision process, MDP)来描述, 然而, 无人机所搭载的传感器(如深度相机)仅能观测到正前方一定视角内的部分环境信息, 无人机侧面、后面、上下方和障碍物后等信息都难以获得。因此, 本文将无人机的导航任务定义为部分可观察马尔可夫决策过程 (partially observable Markov decision process, POMDP), 去除环境状态完全可观测的假设。具体用$(\boldsymbol{S},\boldsymbol{A},\boldsymbol{P},\boldsymbol{R},O,\boldsymbol{H})$表达, 其中: $\boldsymbol{S}$为无人机的状态空间, 采用由观测集合O接收基于隐藏状态$p(\boldsymbol{o}|\boldsymbol{s})$的观测结果; $\boldsymbol{A}$为无人机的动作空间, 包括无人机的不同执行动作; $\boldsymbol{P}$为状态转移概率矩阵, 由于无人机飞行存在干扰且运动环境未知, 因此, 设定$\boldsymbol{P}$未知; $\boldsymbol{R}$为无人机的动作奖励, 表示无人机在执行动作后根据环境信息和自身状态获取的奖励; O为无人机的观测集合, 包括环境信息、无人机状态、导航目标等。

在 POMDP 条件下, 状态转移概率矩阵$\boldsymbol{P}$与当前观测状态$\boldsymbol{o}_t$有关, $\boldsymbol{o}_t$与隐藏状态的转移函数$p(\boldsymbol{o}_t|\boldsymbol{s}_t)$相关, 包含当前状态$\boldsymbol{s}_t$的部分信息, 本文通过定义历史状态$\boldsymbol{H}$来解决无人机的部分可观测性问题, 在 POMDP 的基础上设置合理的动作奖励, 基于 DRL 的无人机导航策略求解式为

$$\pi^*_{\text{std}} = \arg\max_{\pi} \sum_{t} E_{(\boldsymbol{s}_t,\boldsymbol{a}_t)\sim\boldsymbol{P}}(r(\boldsymbol{s}_t,\boldsymbol{a}_t)) \tag{1}$$

式中: π为无人机的策略; E为期望; $\boldsymbol{s}_t$为第t步的状态; $\boldsymbol{a}_t$为无人机在$\boldsymbol{s}_t$状态下根据π策略选择的动作; $r(\boldsymbol{s}_t,\boldsymbol{a}_t)$为无人机在$\boldsymbol{s}_t$状态下执行$\boldsymbol{a}_t$动作获取的奖励; π^*_{std}为无人机经训练后得出使累计奖励最大的策略。

鉴于 PPO 强化学习算法在泛用性、训练速度、探索能力等方面的优势, 且其更加适用于连续动作空间强化学习问题[13], 本文以该算法开展无人机的导航问题研究。

2　基于 PPO 的改进算法设计

2.1　PPO 算法概述

PPO 算法是一种用于强化学习的优化算法。其由 OpenAI 在 2017 年提出, 主要用于解决 Policy Gradients 方法中训练不稳定和样本效率低的问题。PPO 算法具备策略梯度(policy gradient, PG)算法与信任域策略优化 (trust region policy optimization, TRPO)算法的部分优点, 采样数据过程与优化目标函数过程交替进行, 虽然标准的 PG 算法要求对每个数据样本执行一次梯度更新, 但 PPO 算法提出了新目标函数, 可以实现对数据样本的小批量更新。与传统

的 PG 算法相比，PPO 算法对样本的利用效率更高；与更加复杂的 TRPO 算法相比，PPO 算法的实现更为简单且效果相似[14]。

PPO 算法可以根据网络更新方式分为 PPO-Penalty 算法与 PPO-Clip 算法，由于 PPO-Clip 算法具有更优秀的性能，因此，本文选用 PPO-Clip 算法作为研究的核心算法。PPO-Clip 算法相对于其他强化学习算法的核心区别在于使用了策略损失函数：

$$\begin{cases} L^{\mathrm{CLIP}}(\theta)=\hat{E}_t\left(\min\left(\gamma_t(\theta)\hat{A}_t,\gamma_t'(\theta)\hat{A}_t\right)\right) \\ \gamma_t(\theta)=\dfrac{\pi_\theta(\boldsymbol{a}_t|\boldsymbol{s}_t)}{\pi_{\theta_{\mathrm{old}}}(\boldsymbol{a}_t|\boldsymbol{s}_t)} \\ \gamma_t'(\theta)=\mathrm{clip}(\gamma_t(\theta),1-\varepsilon,1+\varepsilon)\hat{A}_t \end{cases} \tag{2}$$

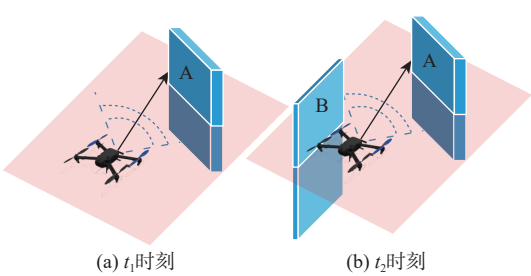
式中：ε为截断超参数，一般设置为 0.2；$\gamma_t(\theta)$表示新旧策略的差异，用来控制新策略的更新幅度，其值越趋近于 1，表明新旧策略差异越小；clip()为截断函数，将网络更新比例$\gamma_t(\theta)$限制在$[1-\varepsilon,1+\varepsilon]$区间内以确保算法的稳定性；$\hat{A}_t$为优势函数，表示在状态下动作相对均值的偏差，一般采用广义优势估计 (generalized advantage estimation, GAE) 来计算[15]：

$$\begin{cases} \hat{A}_t^{G_{\mathrm{AE}}(\gamma,\lambda)}=\displaystyle\sum_{l=0}^{\infty}(\gamma\lambda)^l\delta_{t+l}^V \\ \delta_t^V=r_t+\gamma V_\omega(\boldsymbol{s}_{t+1})-V_\omega(\boldsymbol{s}_t) \end{cases} \tag{3}$$

式中：V为价值估计函数，$V_\omega(\boldsymbol{s}_{t+1})$和$V_\omega(\boldsymbol{s}_t)$分别为下一时刻和当前时刻的预期累计奖励总和；$r_t$为 t 时刻智能体获得的奖励；γ为折扣因子。

2.2 门控循环网络

常规的 PPO 网络能够有效解决 MDP 问题，然而在 POMDP 条件下，无人机无法在当前时刻获取环境的所有信息，而环境信息的缺失将使无人机在复杂环境中难以得出行之有效的策略。如图 1 所示，无人机在t_1与t_2时刻仅能直接观测到障碍物 A，无法判断障碍物 B 的存在，因此，仅靠观测$\boldsymbol{o}_t$不足以得出最优的动作。



(a) t_1时刻　(b) t_2时刻

图 1　部分可观测状态

Fig. 1　Partially observable state

在环境部分可观测条件下，当前状态$\boldsymbol{s}_t$被观测$\boldsymbol{o}_t$所替代，观测$\boldsymbol{o}_t$中仅包含状态$\boldsymbol{s}_t$的部分信息。通过引入 GRU 对历史信息进行筛选与总结得出当前历史信息$\boldsymbol{h}_t$，强化学习模型能够依靠$\boldsymbol{h}_t$与$\boldsymbol{o}_t$得出组成更加完善的状态信息并实现更好的避障效果[16]，引入 GRU 后的 PPO 即为循环-近端策略优化 (recurrent-PPO, R-PPO)。

GRU 为一种常用的循环神经网络结构，为解决长期依赖问题而设计，在一定程度上解决了长短时记忆 (long-short term memory, LSTM) 网络存在的训练复杂、参数多等问题，相比于 LSTM，GRU 的参数更少，计算量更小，从而训练速度更快。同时，GRU 在一些任务上表现出色，能够更有效地处理数据的依赖关系，同时避免了 LSTM 中出现的梯度消失和梯度爆炸等问题[17]，GRU 前向计算式为

$$\begin{cases} \boldsymbol{z}_t=\sigma(\boldsymbol{W}_z\cdot[\boldsymbol{h}_{t-1},\boldsymbol{x}_t]+\boldsymbol{b}_z) \\ \boldsymbol{c}_t=\sigma(\boldsymbol{W}_c\cdot[\boldsymbol{h}_{t-1},\boldsymbol{x}_t]+\boldsymbol{b}_c) \\ \tilde{\boldsymbol{h}}_t=\tanh(\boldsymbol{W}_h\cdot[\boldsymbol{c}_t\odot\boldsymbol{h}_{t-1},\boldsymbol{x}_t]+\boldsymbol{b}_h) \\ \boldsymbol{h}_t=\boldsymbol{h}_{t-1}\odot\boldsymbol{z}_t+\tilde{\boldsymbol{h}}_t\odot(\boldsymbol{I}-\boldsymbol{z}_t) \end{cases} \tag{4}$$

式中：$\boldsymbol{W}$为权重矩阵；$\boldsymbol{b}$为偏置；σ与 tanh为激活函数；$\boldsymbol{I}$为全 1 向量；更新门$\boldsymbol{z}_t$控制时刻$t-1$的隐藏状态$\boldsymbol{h}_{t-1}$与时刻t的状态输入$\boldsymbol{x}_t$有多少保存至隐藏状态$\boldsymbol{h}_t$；重置门$\boldsymbol{c}_t$控制时刻$t-1$的隐藏状态$\boldsymbol{h}_{t-1}$有多少保存至候选隐藏状态$\tilde{\boldsymbol{h}}_t$；“⊙”表示向量元素相乘。GRU 网络结构如图 2 所示。

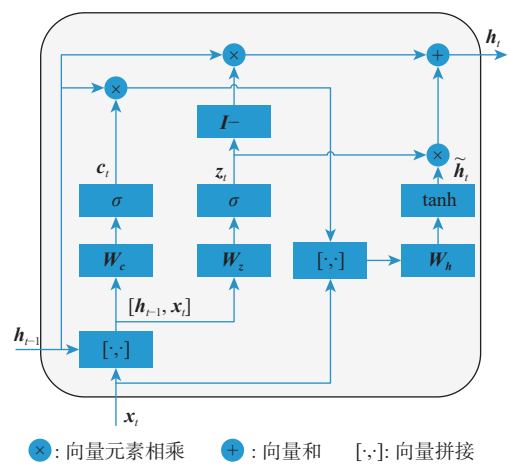


图 2　GRU 网络结构

Fig. 2　GRU network structure

2.3 动作屏蔽方法

在 DRL 任务场景中，动作是体现学习效果最直接的因素，直接影响了智能体的行为和环境状态的变化，而在实际应用中，智能体的可执行动作存在很多约束。为排除非法动作的干扰并提高训练效率和速度，需要设计适用于本文的动作屏蔽方法。

以无人机为例，其自身运动性能与环境障碍物都会限制其动作可行性。通常情况下，对动作的惩罚性奖励有助于约束无人机的无效动作，这种方式

被称为软约束。但采用奖励函数设计的方式来约束无人机会付出极大的训练代价并容易导致模型发散[18]。模型发散将会干扰网络的更新与策略的优化, 不断的动作重复同样使无人机无法探索更加有效的动作策略, 拖慢了训练效率。在离散动作强化学习任务中, 可以采用动作掩码的方法解决该问题, 但该方法无法应用于连续动作空间的强化学习任务中。动作掩码的基本思想是, 为 Actor 神经网络输出的动作概率加上一层掩码, 合法动作将正常输出对应的动作概率, 非法的动作将被掩码屏蔽。

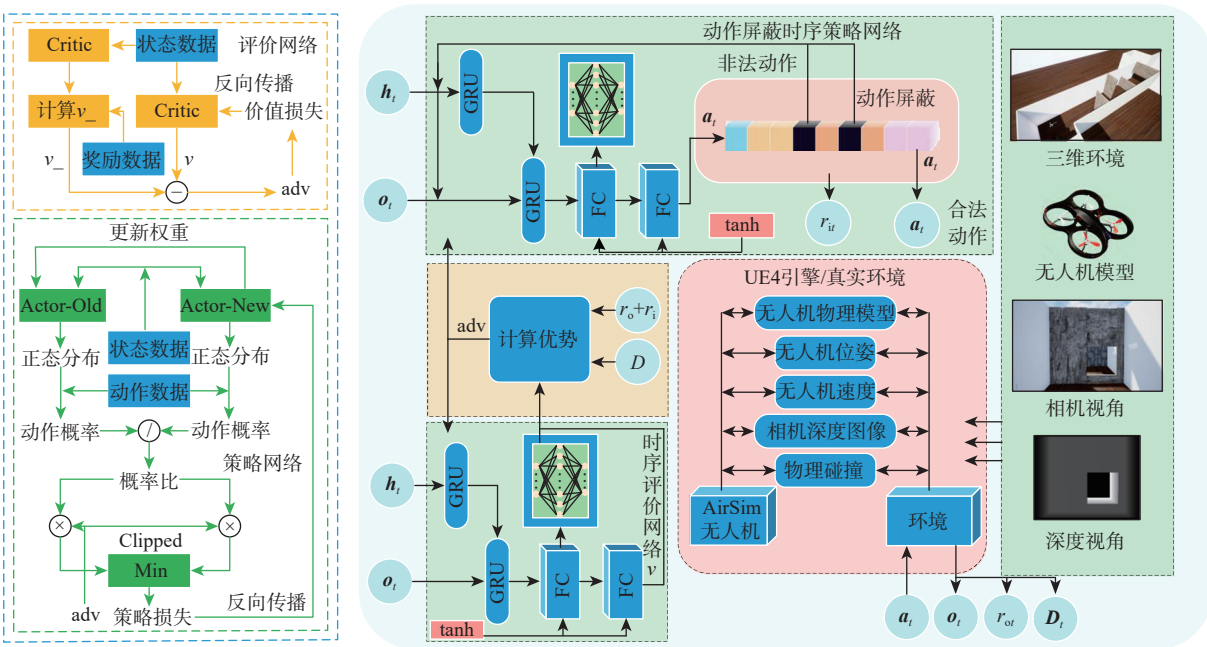
在本文的连续动作空间强化学习任务中, 结合动作掩码与奖励函数设计的基本思想, 提出了非法动作集合A_f与屏蔽概率分布ε_{a}以实现连续动作空间强化学习任务中的动作屏蔽方法, 将动作屏蔽加入 R-PPO 即为 AR-PPO。

非法动作集合A_f内包括绝对非法动作集合与概率非法动作集合, 绝对非法动作代表动作$\boldsymbol{a}_0$在当前状态$\boldsymbol{s}_0$下完全不可行或无效, 如撞向障碍物或飞向地图外等; 相对非法动作代表动作$\boldsymbol{a}_0$在当前状态$\boldsymbol{s}_0$下暂时可行, 但状态$\boldsymbol{s}_0,\boldsymbol{s}_1,\cdots,\boldsymbol{s}_t$下的动作$\boldsymbol{a}_0,\boldsymbol{a}_1,\cdots,\boldsymbol{a}_t$整体构成了连续的不可行动作或无效动作, 如原地转圈或反复转向。屏蔽概率分布f表示当前动作$\boldsymbol{a}_t$被屏蔽的概率, 绝对非法动作的屏蔽概率$f_0=1$, 相对非法动作的屏蔽概率$f_0\in(0,1)$, 屏蔽概率的计算式为

$$\begin{cases} f_{\mathrm{a}} = M(0,1)\sum_{t=0}^{n} f_t \\ f_t = F(\boldsymbol{a}_t|\boldsymbol{s}_t) \end{cases} \tag{5}$$

式中: $M(0,1)$为概率累计重置; f_{a}为t步动作后的屏蔽累计概率, 当累计概率为 1 时, 当前动作$\boldsymbol{a}_t$将被屏蔽并获得负奖励, 网络将重新输出动作直到动作合法。由此, 在训练过程中将屏蔽大量无效或重复的训练动作, 加快训练速度。

非法动作集合A_f中一部分非法动作由人工于训练前定义, 如约束无人机的平均速度, 限制转向速度; 另一部分需要依靠无人机的当前状态$\boldsymbol{s}_t$实时构建与更新, 如通过 AirSim 命令获取无人机当前动作的可行性或通过代码设计判断当前动作是否陷入循环。基于 AR-PPO 的导航方法示意图如图 3 所示, 其中, $v_$为网络预测价值, v 为目标价值, r_{i} 为内部奖励, r_{o} 为外部奖励, FC 表示全连接层, D 为无人机一项状态集合。



(a) 网络更新流程　　(b) 本文方法结构

图 3　AR-PPO 导航方法

Fig. 3　AR-PPO navigation method

图 3(b) 为本文方法的整体研究结构, 依靠 AR-PPO 算法在仿真环境中探索并收集数据更新网络, 策略网络与平均网络均为 GRU 时序网络连接 2 层线性层构成, 策略网络线性层后接入动作屏蔽算法。UE4 仿真环境与真实环境能够互相替换, 确保该算法能够同时用于仿真实验和实机实验。图 3(a) 为网络更新流程, 评价网络中 adv 即为式 (3) 计算所得优势值, 通过优势值计算损失函数进行反向传播实现网络更新; 策略网络则通过评价网络所得优势值与状态、动作数据经过式 (2) 计算损失函数进

行反向传播实现网络更新。

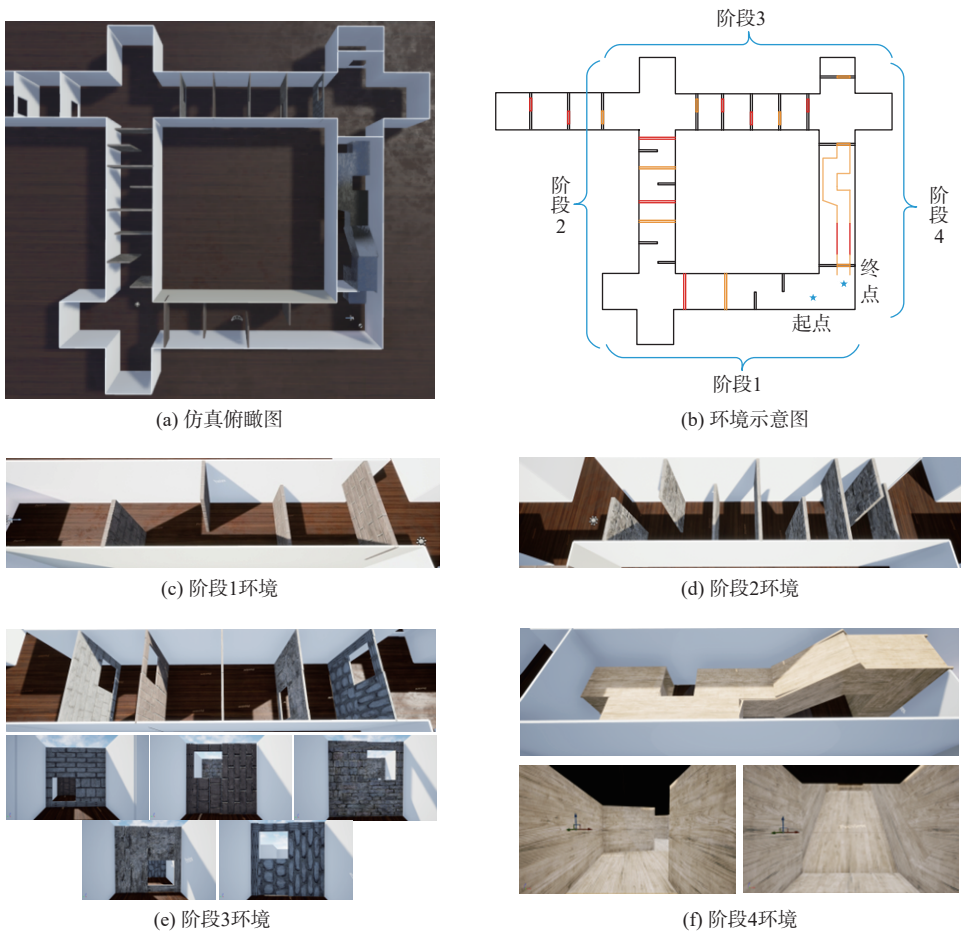
3 基于DRL的无人机导航方法设计

3.1 训练环境设计

无人机相比于无人车拥有更高维度的动作空间与状态空间，若使用常规的柱形或方形障碍物将会使无人机的z轴失去作用，因此，必须采用更加复杂的障碍物环境。

由于AirSim基于虚幻引擎的优越物理与视觉仿真效果，本文选用AirSim构建仿真环境。为加快训练速度和简化训练难度，训练环境被分为4个难度不同的训练阶段，但整体训练过程依然是连续的，无人机的每次训练都需要依次进行所有阶段的训练而不是对每个阶段单独训练。若单独对每个阶段训练，每阶段的最佳策略可能会影响其他阶段的策略，因此，需要整体训练以得出最为泛用的导航策略。

完整训练环境如图4所示，图4(b)平面示意图中红色线条代表该墙体、窗口、管道位于高处，无人机需要从下方绕行或从窗口或管道内穿越，黄色反之，无人机需要从上方绕行或穿越。无人机需要由起点开始依次经过4个阶段的不同障碍物，最终返回到达终点。阶段1采用较为简单的障碍物，训练初期易于获取的奖励将有利于有效导航策略的形成；阶段2将阶段1的障碍物缩短间隔并随机排列，提高了训练难度；阶段3采用较为困难的窗口障碍物，无人机的可通行区域更小；阶段4使用管道空间，无人机需要在狭窄的管道中精确执行动作以穿越管道。无人机完成某一阶段后需要自行根据环境变化改变飞行方向，确保自身飞向正确的方向而不是岔路。



(a) 仿真俯瞰图　(b) 环境示意图

(c) 阶段1环境　(d) 阶段2环境

(e) 阶段3环境　(f) 阶段4环境

图4 仿真训练环境

Fig. 4 Simulation training environment

3.2 观测空间设计

观测空间O中的数据主要可以分为两部分：通过深度相机获取环境信息和无人机飞行中的自身状态。本文采用双目深度相机作为观测环境的传感器。深度相机像素分辨率设置为256像素×144像素，视角场约为80°×60°，若将其直接作为观测量输入会导致训练难度过大，不利于实体无人机的部署，因此，将深度图像进行栅格化，每一栅格像素分辨率为32像素×24像素，为保证无人机正前方信息的完整提取并保留距离无人机最近的障碍物信息，采用栅格交点一定范围内的最小深度值，最终得出7×5的深度矩。

为避免利用位置信息需要全局定位的问题, 观测空间O中不包含无人机的全局坐标, 改为采用无人机与目标的相对信息, 常见的方案为: ①无人机与目标点的总距离d_{xyz}; ②无人机航向与目标方向的夹角θ_{xyz}; ③无人机与目标点的三轴方向的相对距离d_x、d_y、d_z。

相比于无人车, 无人机的自身状态需要考虑更高维度的相对信息, 采用以上无人车的常见方案所得到的信息均不够直观, 无人机需要较长时间来尝试与理解信息的含义。因此, 本文采用无人机航向与目标方向的水平夹角θ与高度差h作为观测信息。使用该方案能够减小无人机的训练难度, 在相同的测试环境下, 采用原始 PPO 使用以上观测空间方案的训练结果如图 5 所示。

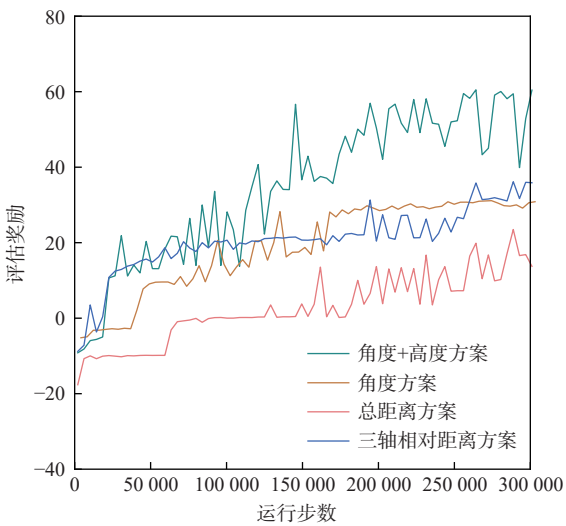


图 5　状态方案对比

Fig. 5　Comparison of state plans

由图 5 可以看出, 采用总距离的方案效果最差, 在训练过程中奖励一直低于其他方案; 仅使用角度的方案效果略好于总距离方案, 但状态信息的不足使其难以实现有效的飞行策略; 采用三轴相对距离的方案在训练前期效果较好, 但在训练中后期奖励增幅不大, 较多的状态量尽管包含了更多的环境信息, 但增大了强化学习训练的难度。采用水平夹角与高度差的方案效果最好, 所包含的环境信息足够完成导航任务, 训练中期就能够取得设计最大奖励。因此, 观测空间O设计为

$$\begin{cases} O = \{d_1, d_2, \cdots, d_n, \theta, h\} \\ \theta = \psi_{\text{yaw}} + \theta_{\text{goal}} \\ h = h_{\text{goal}} - h_{\text{uav}} \end{cases} \tag{6}$$

式中: d_i为经处理后的深度信息; ψ_{yaw}为无人机当前时刻偏航角; θ_{goal}为目标方向与无人机初始方向的夹角; h_{goal}为目标高度; h_{uav}为无人机当前高度。

3.3　动作空间设计

由于简单的离散动作空间在复杂训练环境中难以完成训练目标, 因此, 本文采用三维连续动作空间。连续动作空间允许无人机在一个连续的范围内选择动作, 而不是被限制在离散的动作集合中。这种灵活性使无人机可以更精细地调整其行为, 适应不同的环境和任务需求。无人机的动作空间$\boldsymbol{A}$可以采用速度控制、姿态控制、位置控制等方案。相比于其他方案, 速度控制更直接, 仅需调整速度即可改变无人机的运动状态, 操作相对简单, 运动轨迹更为平滑, 可以在空间中以各种曲线路径飞行。最重要的是速度控制不需要考虑无人机的具体的位置信息, 更适合快速应对变化。采用偏航角速度而不是直接指定偏航角度能够使飞行轨迹更加平滑, 无人机将能够实现平滑的转向和掉头动作。同时, 偏航角速度ω、线速度v_z与观测中的夹角θ、高度差h对应, 能够加快有效导航策略的学习速度。

综上所述, 本文采用速度控制连续动作空间设计, 动作$\boldsymbol{a}_t$由无人机机体坐标系下x轴方向的线速度v_x(单位: m/s)、偏航角速度ω(单位: rad/s), z轴方向的线速度v_z(单位: m/s)组成。

3.4　奖励函数设计

基于 DRL 算法的导航任务通常只设定到达目标的正奖励和碰撞的负奖励。然而, 这种奖励设计会产生奖励稀疏的问题, 在实际导航任务中, 如果目标点与起点之间的距离较远, 那么奖励稀疏问题就会更加显著影响训练样本效率和训练收敛速度[19]。

为解决奖励稀疏的问题, 本文在稀疏奖励基础上引入密集奖励, 密集奖励函数需要提供足够的反馈信息, 以便智能体可以准确地评估其行动的好坏。这要求奖励函数能够捕捉到任务的关键信息, 如智能体与目标之间的距离、避开障碍物的难易程度等。密集奖励函数还需要避免引入不良的副作用或不必要的偏差。如果奖励函数设计不当, 可能会导致智能体学到错误的策略或出现不稳定的学习行为。因此, 设计密集奖励函数需要仔细权衡各种因素, 以确保其在实际应用中的有效性和可靠性。

参考无人机的观测空间与动作空间设计奖励函数, 奖励函数设置如式 (7) 所示, 包括 2 个稀疏奖励与 1 个密集奖励。

$$r_t = \begin{cases} R_{\text{goal}} & d_t < d_{\text{r}} \\ R_{\text{done}} & \text{失败} \\ R_{\text{m}} + |h - h_{\text{r}}| + r_{\text{v}} & \text{其他} \end{cases} \tag{7}$$

2 个稀疏奖励分别为: 每当无人机通过训练环境中的一个阶段并与阶段目标点的距离d_t小于设

定值d_r时，无人机将获得一个较大的正奖励R_{goal}，即无人机在训练中最多可获得 3 次该奖励；当无人机与环境发生物理碰撞时认为本轮任务失败，无人机将获得一个较大的负奖励R_{done}并重置环境开始下一轮训练。

密集奖励包括恒定给予的负奖励R_m，无人机与目标高度差大于h_r时给予的负奖励$|h-h_r|$，无人机向目标前进时给予的正奖励r_v。

4 训练与结果

4.1 训练设置

在训练流程中，将奖励函数计算所得奖励定义为外部奖励，即由外部环境获取的奖励；动作屏蔽所产生奖励为内部奖励，即由算法自身计算所得奖励，2 种奖励均参与网络更新，但内部奖励不计入测试环境的结果输出。无人机的每一步动作都需要将自身状态、相机信息、目标位置输入强化学习网络，网络根据当前策略输出动作并验证其合法性，将合法动作命令发布给无人机执行并计算奖励。最终，当无人机完成一定步数的训练后，将收集的训练数据用于网络更新。AR-PPO 算法伪代码如下：

输入： 目标位置 Goal、初始位置 Pose、最大步数 Step。

输出： 实现研究任务的无人机避障模型。

1: 初始化网络与环境

2: for s = 0 to Step do

3: 重置 Airsim 环境,获取$\boldsymbol{o}_t$

4: 初始化历史信息$\boldsymbol{h}_t$

5: for t = 0 to 4 096 do

6: 根据 Done 选择是否重置环境并重新获取$\boldsymbol{o}_t$与重置$\boldsymbol{h}_t$

7: 根据$\boldsymbol{o}_t$与$\boldsymbol{h}_t$从策略网络获取动作$\boldsymbol{a}_t$与$\boldsymbol{h}_{t+1}$

8: 通过动作屏蔽算法判断$\boldsymbol{a}_t$是否合法并得出内源奖励r_{it}

9: 执行或屏蔽$\boldsymbol{a}_t$并获取环境奖励$\boldsymbol{r}_{ot}$与观测$\boldsymbol{o}_{t+1}$

10: 根据 Airsim 判断无人机是否 Done

11: 存储$\boldsymbol{o}_t$、$\boldsymbol{a}_t$、r_{it}、r_{ot}、$\boldsymbol{h}_t$、$\boldsymbol{o}_{t+1}$、$\boldsymbol{h}_{t+1}$数据

12: end for

13: 存储无人机飞行轨迹

14: 通过式 (3) 计算价值函数$V_{\phi k}$优势估计$\hat{A}_t$

15: 通过优势估计$\hat{A}_t$计算评价网络损失函数并反向传播

16: 通过式 (2) 计算策略网络损失函数并反向传播

17: 再次执行 4 096 步循环并更新网络

18: for i = 0 to 5 do

19: 重置 Airsim 环境

20: 在环境中测试策略θ_{k+1}，记录测试奖励

21: end for

22: 计算 5 次测试的平均奖励与方差并绘制奖励曲线

23: 保存强化学习模型

24: end for

在训练环境中进行实验以验证算法。模型训练在一台操作系统为 Ubuntu 20.04 LTS，搭载 GTX 3060 显卡、i7-12700H CPU、32 GB 内存的计算机中进行，激活函数设置为 tanh。AR-PPO 算法超参数与各算法超参数如表 1 所示，AR-PPO 算法超参数主要采用 Open-AI 推荐参数，SAC、TD3、DDPG 超参数则采用 ElegantRL 小雅强化学习库推荐设置，确保各算法超参数设置合理以提高训练速度和稳定性。

表 1 模型超参数设置

Table 1 Model hyperparameters setting

算法	学习率	折扣率	Horizon	经验池	探索噪声	策略噪声	软更新	裁剪系数	广义优势估计平滑函数	熵系数	小批次
AR-PPO/PPO	2×10^{-4}	0.97	4 096					0.25	0.95	0.01	512
SAC	1×10^{-3}	0.97	4 096	1×10^{6}			5×10^{-3}				
TD3	1×10^{-4}	0.97	4 096	1×10^{6}	0.05	0.10					
DDPG	1×10^{-4}	0.97	4 096	1×10^{6}	0.05						

在训练环境中训练 10^6 步，每轮训练开始时无人机会被重置于起始位置并重新起飞，当无人机到达目标点、发生碰撞、单轮步数到达 600 步时视为本轮结束。每进行 8 192 步训练，将模型调整为评估状态以测试当前策略性能，并保存训练结果。

4.2 训练结果

为在训练阶段评估 AR-PPO 算法，选择目前较为常见的 DRL 算法进行对比实验，使用与 AR-PPO 算法相同的奖励函数、训练设计，每种算法训练 10^6 步，对比算法均采用 ElegantRL 小雅强化学习开源库实现，选择对比算法包括 DDPG 算法、TD3 算法、SAC 算法。实验每进行 8 192 步，将模型设置为评估模式进行测试，取测试中的奖励平均值并记录均差。训练效果如图 6 所示，在训练结束时，只

有 AR-PPO 算法能够取得最高奖励, 这说明在相同的训练环境中, AR-PPO 算法的训练效率优于其他算法。

在训练过程中, 使用 AirSim 保存无人机训练过程中的飞行轨迹, 采用 AR-PPO 算法训练过程中无人机轨迹图如图 7 所示, 由于实际仿真环境中墙体厚度的存在, 因此, 图中轨迹与墙体之间有一定误差。

图 7(a)中红色路径为训练过程中无人机经过的全部路径, 颜色越深说明经过次数越多。可以看出, 无人机在训练的第 1、第 2 阶段探索了所有岔路, 这说明此时无人机仍不能正确处理目标方向与远离障碍物之间的取舍。而在第 3、第 4 阶段无人机的航行轨迹被限制在一个个窗口与管道之间, 无人机需要谨慎调整自身姿态和飞行速度, 因此, 第 3、第 4 阶段轨迹细长。图 7(b)为训练完成后将模型进行多次测试所得无人机飞行轨迹, 测试中无人机飞行策略稳定、动作较为平稳, 能够安全完成导航任务。

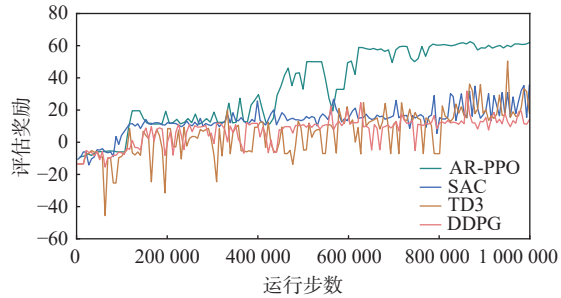
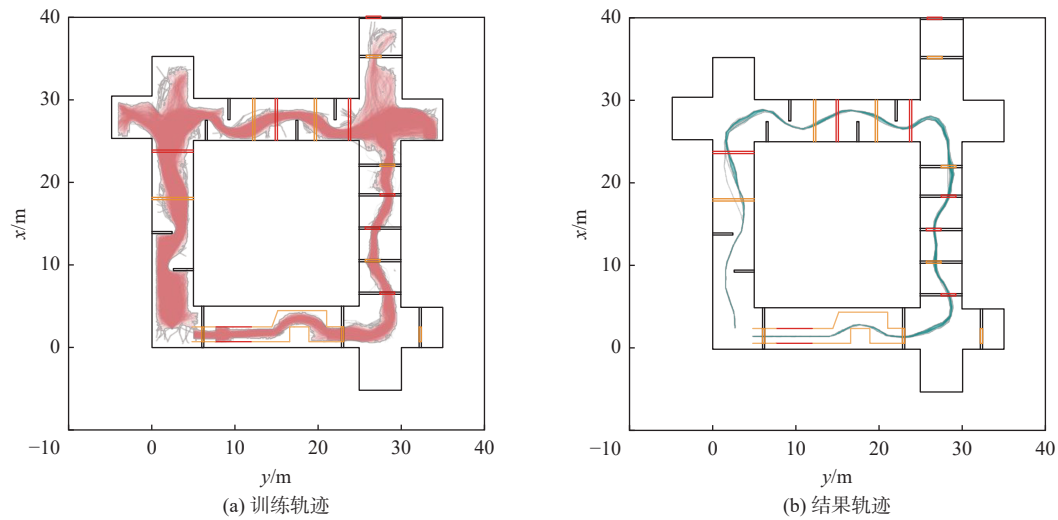


图 6　训练效果对比

Fig. 6　Comparison of training effects



(a) 训练轨迹　　(b) 结果轨迹

图 7　训练轨迹与结果轨迹

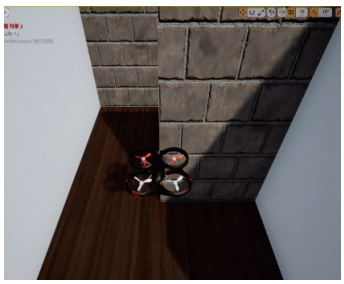
Fig. 7　Training trajectory and result trajectory

将无人机固定于 3 个选定位置并进行强化学习训练, 保存动作屏蔽算法输出的合法动作与被屏蔽的非法动作, 连接动作起始点与目标点, 得出如图 8 所示的动作屏蔽图。图中, x 轴方向为无人机朝向, 浅红色线条为会撞向障碍物的非法动作, 浅蓝色线条为可执行的合法动作。在训练过程中, 非法动作会被动作屏蔽算法拦截并给予负奖励。只有合法动作会被真正输入至无人机并执行, 由此节约大量的试错时间和重复探索时间, 加快训练速度。

4.3　泛化实验

将 AR-PPO 算法与对比算法训练所得模型应用于不同的测试环境, 以验证各模型的泛化能力, 每种算法模型进行 100 次测试, 实验过程中各模型取得结果如表 2 所示, 4 种测试环境与算法测试轨迹如图 9 所示, 图中仅展示对比算法中成功率最高的一种算法。

由表 2 可知, AR-PPO 算法在不同环境中表现最好, 能够在与训练环境不同的场景中达到 85% 以上的导航成功率, ; SAC 算法能够在环境 1、环境 2、环境 3 中到达目标点, 成功率低于 AR-PPO 算法但高于其他算法, 环境 4 中则只能飞行 7 m, 无法穿过狭窄通道; TD3 算法成功率低于 SAC 算法但优于 DDPG 算法; DDPG 算法在各个环境成功率都最低, 表现最差。其中, 环境 4 仅有 AR-PPO 算法能够成功抵达目标, 这说明对比算法在训练中未能学习到如何穿越狭窄通道或无法将策略迁移至未知环境。图 9 中仅展示 AR-PPO 算法与对比算法中最佳算法的飞行轨迹, 环境 1 除各算法均能到达靠近目标点的位置, 但不能稳定越过最后 2 道障碍; 环境 2 中 TD3 算法到达目标点概率较低, SAC 算法则能够以 2 条不同路径到达目标点, 但飞行速度较慢; 环境 3 各算法均能够以一定概率到达目标点, 但只有 AR-PPO 算法拥有稳定的策略以同一轨迹到达目标点; 环境 4 除 AR-PPO 算法以外, 对比算法无



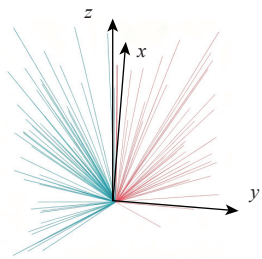
(a1) 右侧障碍环境



(b1) 上方障碍环境

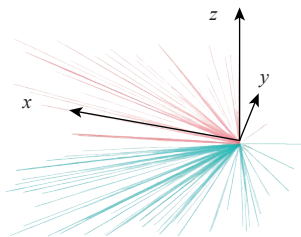


(c1) 窗口障碍环境



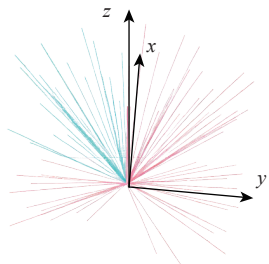
(a2) 动作结果

(a) 右侧障碍环境与动作结果



(b2) 动作结果

(b) 上方障碍环境与动作结果



(c2) 动作结果

(c) 窗口障碍环境与动作结果

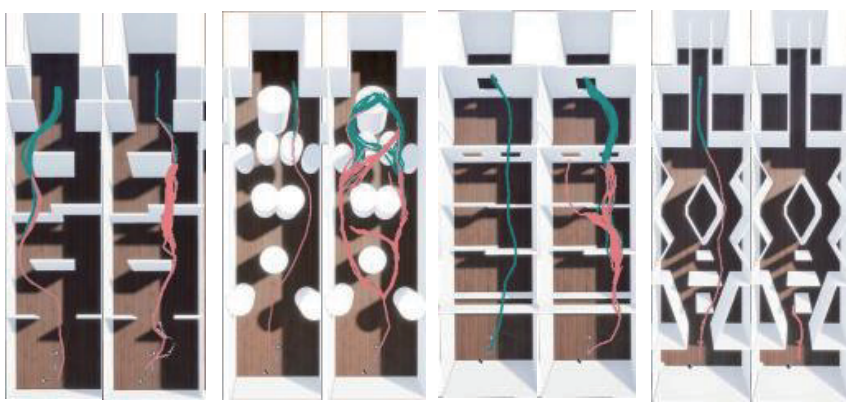
图 8 动作屏蔽效果

Fig. 8 Action block effect

表 2 实验结果

Table 2 Experimental results

环境	目标成功到达率/%				最高单步平均奖励				飞行距离众数/m			
	AR-PPO	DDPG	TD3	SAC	AR-PPO	DDPG	TD3	SAC	AR-PPO	DDPG	TD3	SAC
环境1	85	9	48	44	0.70	0.18	0.42	0.31	30	12	30	26
环境2	96	4	36	64	0.77	0.18	0.38	0.68	30	27	18	30
环境3	100	21	51	57	0.82	0.42	0.65	0.67	30	16	30	30
环境4	92	0	0	0	0.82	0.14	0.14	0.14	30	7	7	7



(a1) AR-PPO (a2) TD3

(a) 环境1

(b1) AR-PPO (b2) SAC

(b) 环境2

(c1) AR-PPO (c2) SAC

(c) 环境3

(d1) AR-PPO (d2) SAC

(d) 环境4

未到达目标点 到达目标点

图 9 迁移实验轨迹

Fig. 9 Migration experiment trajectory

法越过第 1 道障碍，这表明其不能在狭窄通道中完成导航任务，AR-PPO 算法则能够在狭长的通道中谨慎地调整姿态并完成飞行任务。

以上实验结果表明，由 AR-PPO 算法在训练环境中所得的强化学习模型在未经调整的情况下，能够直接部署于新的未知环境实现有效的避障与导航，即该模型具有良好的泛化性能。

4.4 消融实验

设计消融实验评估 AR-PPO 算法各组件的影响，将原始 PPO 作为基准模型作为对比，R-PPO 算

法为移除动作屏蔽的时序 PPO 算法, A-PPO 算法为移除 GRU 的动作屏蔽 PPO 算法, 于训练场景中将模型分别进行训练, 训练设置与上文相同, 最终获得奖励曲线如图 10 所示, 将训练后算法模型应用于测试环境验证模型导航能力, 各算法所得结果如表 3 所示。

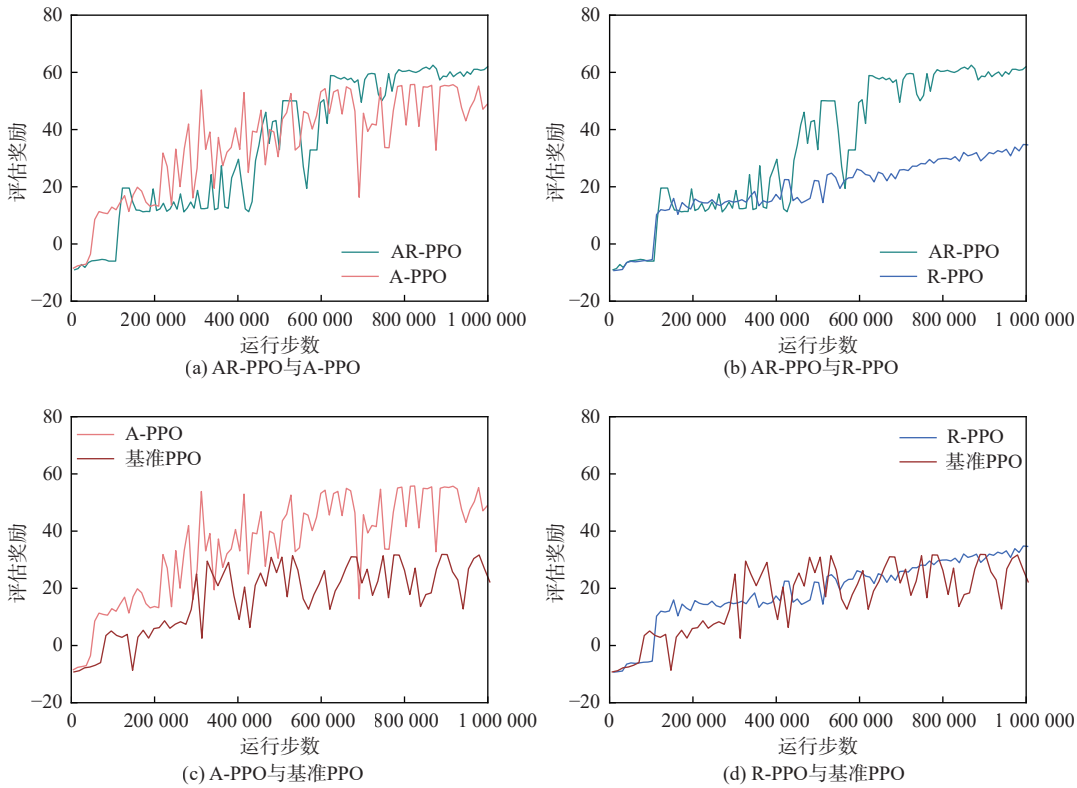


图 10　消融实验

Fig. 10　Experimental of ablation

表 3　消融对比结果

Table 3　Comparison results of ablation

环境	目标成功到达率/%			最高单步平均奖励			飞行距离众数/m		
	AR-PPO	A-PPO	R-PPO	AR-PPO	A-PPO	R-PPO	AR-PPO	A-PPO	R-PPO
环境1	85	47	87	0.70	0.71	0.68	30	26	30
环境2	96	64	84	0.77	0.79	0.75	30	30	30
环境3	100	73	24	0.82	0.81	0.65	30	30	25
环境4	92	3	0	0.82	0.47	0.14	30	7	14

由图 10 可得, 仅加入动作屏蔽的 A-PPO 算法在训练前期与中期均能以最少的训练步数取得最高的累计奖励, 训练速度快于作为基准的原始 PPO 算法; 加入动作屏蔽与 GRU 的 AR-PPO 算法取得高奖励所需要的步数同样高于移除动作屏蔽的 R-PPO 算法, 由此可以得出, 动作屏蔽能够有效减少非法动作的干扰并提高训练速度。在训练后期, A-PPO 算法的累计奖励曲线高于 R-PPO 算法与基准 PPO 算法但低于 AR-PPO 算法, 奖励曲线反复震荡不平稳, 阴影部分面积大, 累计奖励均差大, 模型策略不稳定, 这说明动作屏蔽能够加快训练速度, 但不能使模型在部分可观测环境下稳定收敛。仅加入 GRU 的 R-PPO 算法在训练前中期均最慢, 同样, AR-PPO 算法的训练速度在前中期慢于移除 GRU 的 A-PPO 算法, 由此可得 GRU 的加入拖慢了网络的训练速度。在训练后期, R-PPO 算法的阴影区域明显小于未加入 GRU 的基准 PPO 算法; AR-PPO 算法的阴影区域同样小于移除 GRU 的 A-PPO 算法且曲线振荡幅度较小, 能够稳定地取得较高累计奖励, 证明 GRU 的加入能够提高模型的稳定性, 使模型在部分可观测环境下完成训练目标。

由表 3 可知, A-PPO 算法在各个环境均能够取得一定的成功率且单步奖励更高, 这表明 A-PPO 算法能够快速取得较高的奖励但策略不够稳定; R-PPO 算法在环境 1、环境 2 中成功率与 AR-PPO 算法相近, 高于 A-PPO 算法, 由于该模型在 10^6 步的

训练中并未完成完整的训练流程，在环境3和环境4中导航效果较差。这表明R-PPO算法所得模型策略稳定，但训练速度较慢。测试实验结果能够印证奖励曲线所得结论。综上，实验结果与测试结果表明动作屏蔽与GRU均在AR-PPO算法起到重要作用，其负面效果均在可接受范围内或能够被其他组件所弥补。

4.5 实机测试

将仿真训练环境的模型移植到实际环境进行实机测试是一个具有挑战性的研究，而无人机所面临的坠毁风险更不利于实机测试的实现。相比于仿真环境，实际环境下无人机的导航将面临更多的困难。在实机测试中，无人机在起飞、导航、降落环节均有可能出现问题，实际相机所得深度信息精度与准确性相比仿真相机较低。但AR-PPO算法仍然能够顺利完成避障任务。

因此，考虑到飞行场地大小、实机不确定性和测试风险，实验中搭建的障碍物较为典型，实体无人机采用常见四旋翼无人机机体，搭载D435I深度相机，PIXhawk V5飞控，Nvidia NX机载电脑，安装Ubuntu 18.04系统。将训练所得AR-PPO模型部署于机载电脑中进行实机测试，测试结果如图11所示。

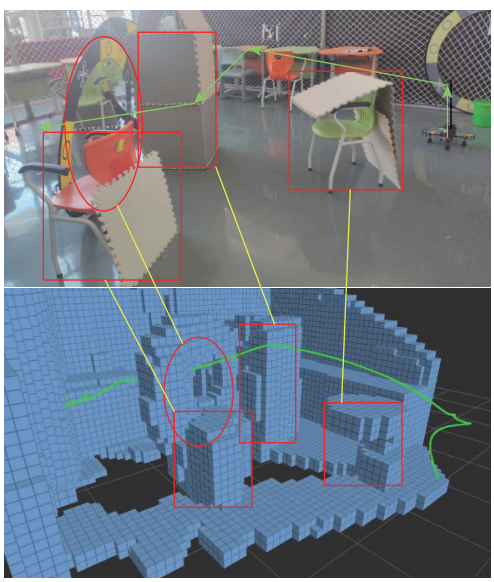


图11 实机测试环境

Fig. 11 Real machine testing environment

图11中所建三维点云地图是通过D435i深度相机生成，绿色轨迹由无人机实时位置生成，仅用于展示实体无人机在实际环境中的飞行轨迹，不会参与导航策略，有必要说明，无人机在实际飞行中的观测信息比生成的点云地图更为精准。由飞行轨迹可知，将强化学习模型部署于实体无人机后，无人机能够依靠训练所得策略在实际环境下安全完成避障任务。

5 结 论

1）本文提出了一种基于PPO的AR-PPO改进算法，通过引入GRU和动作屏蔽方法设计提高了强化学习模型的策略稳定性与训练效率。

2）设计适用于无人机自主导航的训练环境、观测空间、动作空间、奖励函数，通过改进奖励函数，选择合理的观测空间状态量改善训练效果。

3）通过AirSim仿真训练，无人机在复杂场景中能够迅速获得稳定可行的强化学习策略。将该强化学习模型部署于其他环境中测试，结果表明，该策略能够在不同环境中展现优秀的导航效果。

4）通过消融实验验证AR-PPO算法各组件效果，结果证明各组件能够协同提高算法的训练效率和学习能力。

5）将仿真训练所得强化学习模型部署于实体无人机进行实际测试，结果证明本文方法在实际环境下能够顺利完成避障导航任务。

必须指出，由于缺少全局地图，在新的未知环境中如何提高飞行策略的最优性仍需进一步研究。

参考文献（References）

[1] Han S F, He Y, Fang H, Recent development in automatic guidance and autonomous vehicle for agriculture: a review [J]. Journal of Zhejiang University, 2018, 44(4): 381-391.

[2] 徐国艳, 熊绎维, 周彬, 等. 基于PPO的移动平台自主导航[J]. 北京航空航天大学学报, 2022, 48(11): 2138-2145.
Xu G Y, Xiong Y W, Zhou B, et al. Autonomous navigation based on PPO for mobile platform[J]. Journal of Beijing University of Aeronautics and Astronautics, 2022, 48(11): 2138-2145(in Chinese).

[3] 董豪, 杨静, 李少波, 等. 基于深度强化学习的机器人运动控制研究进展[J]. 控制与决策, 2022, 37(2): 278-292.
Dong H, Yang J, Li S B, et al. Research progress of robot motion control based on deep reinforcement learning[J]. Control and Decision, 2022, 37(2): 278-292(in Chinese).

[4] 户高铭, 蔡克卫, 王芳, 等. 基于深度强化学习的无地图移动机器人导航[J]. 控制与决策, 2024, 39(3): 985-993.
Hu G M, Cai K W, Wang F, et al. Mapless navigation based on deep reinforcement learning for mobile robots[J]. Control and Decision, 2024, 39(3): 985-993(in Chinese).

[5] Marchesini E, Farinelli A. Discrete deep reinforcement learning for mapless navigation[C]//Proceedings of the IEEE International Conference on Robotics and Automation. Piscataway: IEEE Press, 2020: 10688-10694.

[6] Singla A, Padakandla S, Bhatnagar S. Memory-based deep reinforcement learning for obstacle avoidance in UAV with limited environment knowledge[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 22(1): 107-118.

[7] 魏瑶, 刘志成, 蔡彬, 等. 基于深度循环双 Q 网络的无人机避障算法研究[J]. 西北工业大学学报, 2022, 40(5): 970-979.
Wei Y, Liu Z C, Cai B, et al. Study on UAV obstacle avoidance algorithm based on deep recurrent double Q network[J]. Journal of Northwestern Polytechnical University, 2022, 40(5): 970-979(in Chinese).

[8] 张耀中, 许佳林, 姚康佳, 等. 基于 DDPG 算法的无人机集群追击任务[J]. 航空学报, 2020, 41(10): 309-321.
Zhang Y Z, Xu J L, Yao K J, et al. Pursuit missions for UAV swarms based on DDPG algorithm[J]. Acta Aeronautica et Astronautica Sinica, 2020, 41(10): 309-321(in Chinese).

[9] 唐蕾, 刘广钟. 改进 TD3 算法在四旋翼无人机避障中的应用[J]. 计算机工程与应用, 2021, 57(11): 254-259.
Tang L, Liu G Z. Application for improved TD3 algorithm in obstacle avoidance of quad-rotor UAV[J]. Computer Engineering and Applications, 2021, 57(11): 254-259(in Chinese).

[10] 张云燕, 魏瑶, 刘昊, 等. 基于深度强化学习的端到端无人机避障决策[J]. 西北工业大学学报, 2022, 40(5): 1055-1064.
Zhang Y Y, Wei Y, Liu H, et al. End-to-end UAV obstacle avoidance decision based on deep reinforcement learning[J]. Journal of Northwestern Polytechnical University, 2022, 40(5): 1055-1064(in Chinese).

[11] 赵静, 裴子楠, 姜斌, 等. 基于深度强化学习的无人机虚拟管道视觉避障[J]. 自动化学报, 2024, 50(11): 2245-2258.
Zhao J, Pei Z N, Jiang B, et al. Virtual tube visual obstacle avoidance for UAV based on deep reinforcement learning[J]. Acta Automatica Sinica, 2024, 50(11): 2245-2258(in Chinese).

[12] 寇凯, 杨刚, 张文启, 等. 基于 SAC 的无人机自主导航方法研究[J]. 西北工业大学学报, 2024, 42(2): 310-318.
Kou K, Yang G, Zhang W Q, et al. Exploring UAV autonomous navigation algorithm based on soft actor-critic[J]. Journal of Northwestern Polytechnical University, 2024, 42(2): 310-318(in Chinese).

[13] Yu C, Velu A, Vinitsky E, et al. The surprising effectiveness of PPO in cooperative multi-agent games[C]//Proceedings of the Advances in Neural Information Processing Systems 35. San Diego: Neural Information Processing Systems Foundation, Inc., 2022: 24611-24624.

[14] Dey R, Salem F M. Gate-variants of gated recurrent unit (GRU) neural networks[C]//Proceedings of the IEEE 60th International Midwest Symposium on Circuits and Systems. Piscataway: IEEE Press, 2017: 1597-1600.

[15] Schulman J, Wolski F, Dhariwal P, et al. Proximal policy optimization algorithms[EB/OL]. (2017-07-17)[2024-07-20]. https://arxiv.org/pdf/1707.06347v1.

[16] Baker B, Kanitscheider I, Markov T, et al. Emergent tool use from multi-agent autocurricula[C]//Proceedings of the International Conference on Learning Representations. Appleton: ICLR, 2020.

[17] Ouahouah S, Bagaa M, Prados-Garzon J, et al. Deep-reinforcement-learning-based collision avoidance in UAV environment[J]. IEEE Internet of Things Journal, 2022, 9(6): 4015-4030.

[18] Xue Y T, Chen W S. A UAV navigation approach based on deep reinforcement learning in large cluttered 3D environments[J]. IEEE Transactions on Vehicular Technology, 2023, 72(3): 3001-3014.

[19] Engstrom L, Ilyas A, Santurkar S, et al. Implementation matters in deep RL: a case study on PPO and TRPO[C]//Proceedings of the International Conference on Learning Representations. Appleton: ICLR, 2020.

End-to-end autonomous navigation method for drones based on AR-PPO

LI Hanwei[1], LI Xinkai[1,*], ZHANG Hai[2], JIANG Lingfeng[1], MENG Yue[3], ZHANG Hongli[3]

(1. School of Electrical Engineering, Xinjiang University, Urumqi 830017, China;
2. Engineering Training Center, Xinjiang University, Urumqi 830017, China;
3. School of Intelligent Science and Technology (School of Future Technology), Xinjiang University, Urumqi 830017, China)

Abstract: We propose an end-to-end autonomous navigation method for drones based on action masking-temporal-proximal policy optimization (AR-PPO) to address problems with deep reinforcement learning (DRL) algorithms for drone navigation, such as partial observability of the environment and high training difficulty. First, we introduce a gated recurrent network to process historical information, allowing the policy network to incorporate historical baseline information to tackle the problem of partial observability in complex environments. Second, we design an action masking method suitable for continuous action space reinforcement learning tasks, which excludes the interference of illegal actions and accelerates the model training speed. Then, we establish an actor-critic framework for a three-dimensional continuous action space DRL end-to-end model, which directly outputs control actions based on sensor perception results. Finally, we conduct comparative experiments with mainstream DRL methods. With an over 85% success rate in four generalized test contexts, the AR-PPO approach has the fastest training convergence speed and the largest rewards. Drones can safely perform obstacle avoidance navigation in real machine experiments. The results show that the proposed method effectively enables precise obstacle avoidance and safe navigation for drones.

Keywords: autonomous navigation; deep reinforcement learning; proximal policy optimization; complex environment; drone

Received: 2024-08-02; **Accepted**: 2024-10-30; **Published Online**: 2024-11-07 14:00

URL: link.cnki.net/urlid/11.2625.V.20241107.1129.002

Foundation items: National Natural Science Foundation of China (62263030); Natural Science Foundation of Xinjiang Uygur Autonomous Region (2023D01C187); Tianshan Talents Training Program of Xinjiang Uygur Autonomous Region (2023TSYCCX0037)

* **Corresponding author**. E-mail: lxk@xju.edu.cn