http://bhxb.buaa.edu.cn jbuaa@buaa.edu.cn
DOI: 10.13700/j.bh.1001-5965.2020.0600

无人集群系统行为决策学习奖励机制

张婷婷[1,2,3,*]，蓝羽石[2]，宋爱国[3]

(1. 陆军工程大学 指挥控制工程学院，南京 210017； 2. 中国电子科技集团公司第二十八研究所，南京 210017；
3. 东南大学 仪器科学与工程学院，南京 210096)

摘 要：未来作战的发展方向是由多智能体系统构成的无人集群系统通过智能体之间自主协同来完成作战任务。由于每个智能体自主采取行为和改变状态，增加了智能群体行为策略训练的不稳定性。通过先验约束条件和智能体间的同构特性增强奖励信号的实时性，提高训练效率和学习的稳定性。采用动作空间边界碰撞惩罚、智能体间时空距离约束满足程度奖励；通过智能体在群体中的关系特性，增加智能体之间经验共享，进一步优化学习效率。在实验中，将先验增强的奖励机制和经验共享应用到多智能体深度确定性策略梯度(MADDPG)算法中验证其有效性。结果表明，学习收敛性和稳定性有大幅提高，从而提升了无人集群系统行为学习效率。

关 键 词：无人集群系统；MADDPG 算法；对抗任务；行为决策；奖励机制
中图分类号：TP181
文献标志码：A **文章编号**：1001-5965(2021)12-2442-10

无人集群系统是近年来国内外军事领域发展的重要作战系统，推动无人作战样式由“单平台遥控作战”向“智能集群作战”发展[1]。例如，无人机集群作战是无人集群系统典型的作战样式。无人集群系统可以看作是由若干同构或者异构的无人装备通过自组织构成的智能群体，形成分布式感知、目标识别、自主决策及协同规划与攻击能力，具有交互学习和智能涌现的群体智能特征[2]。人类期望无人集群有自学习、自决策的自主作战能力，随着人工智能技术的发展，无人系统行为自主决策成为可能[3]。无人集群系统往往面临对抗任务，在此类情况下，实现各个无人执行模块高效准确地协同完成既定任务，亟须研究构建在对抗环境中无人系统协同完成任务的高效行为决策方法，如何提高自主行为决策效率是关键问题。目前，勘测、侦察及公共安全等领域所采用的大多既定环境和任务规划下的协同操作策略，缺乏对抗任务下多无人系统自适应感知与自主协同的行为生成策略。在双方对抗任务中，从单个无人系统视角看，其他协同无人系统也是动态变化的，行为是未知的，导致执行环境动态性增强，增加无人系统适应动态环境的不确定性和行为自主决策学习的复杂度，改变传统单智能体学习所依赖的环境转移的不确定性，导致智能体学习的复杂度。

目前，多智能体深度强化学习用于无人集群系统自主行为策略学习是主流的方法。无人系统通过试探和奖励反馈形成决策行为。在设计时，通常会精心设计信息丰富的奖励功能，以引导无人系统正确的行为策略。对于许多实际问题来说，定义一个好的奖励函数并非易事。例如，只在无人系统成功完成任务时奖励，为了完成这个任

收稿日期：2020-10-23；**录用日期**：2021-04-23；**网络出版时间**：2021-05-21 14:14
网络出版地址：kns.cnki.net/kcms/detail/11.2625.V.20210520.1657.002.html
基金项目：国家自然科学基金(61802428)；中国博士后科学基金(2019M651991)；军委科技委国防科技基金(2019-JCJQJJ-014)
* **通信作者**. E-mail: 101101964@seu.edu.cn

引用格式：*张婷婷，蓝羽石，宋爱国. 无人集群系统行为决策学习奖励机制[J]. 北京航空航天大学学报，2021，47(12)：2442-2451.*
ZHANG T T, LAN Y S, SONG A G. Behavioral decision learning reward mechanism of unmanned swarm system [J]. Journal of Beijing University of Aeronautics and Astronautics, 2021, 47(12): 2442-2451(in Chinese).

务需要长时间的试探行动过程,那么奖励就变得很少,在疏松奖励情况下,无人系统策略学习效率非常低。本文增加动作空间边界碰撞惩罚、智能体间时空距离约束满足程度奖励; 同时通过智能体在群体中的关系特性,增加智能体间经验共享,进一步优化学习效率。在实验中将先验增强的奖励机制和经验共享应用到多智能体深度确定性策略梯度(Multi-Agent Deep Deterministic Policy Gradient,MADDPG) 算法中,多智能体行为学习效率显著提升。

1　相关工作

2017 年,谷歌的 DeepMind 团队开创性地提出 MADDPG 算法[4],实现多智能体在协同与对抗的复杂场景中的自主行为决策学习,该算法考虑到智能体之间的协同与对抗关系,设计协同与对抗关系奖励函数。另外,该算法对所有智能体策略进行估计,训练时充分利用全局信息,执行时策略只用局部信息,以缓解执行环境不稳定问题。利用该算法可以解决连续动作空间的无人集群自主对抗策略生成问题。

MADDPG 算法虽然解决了多 Agent 环境的不稳定问题,但解优化性能不好。深度强化学习中最大的难点是对领域问题求解时奖励函数的设计,扩展至多智能体场景时,这一问题更加显著,直接决定了智能体是否能学到目标策略,并影响算法的收敛性和最终的实现效果。近年来,诸多学者围绕该问题进行了研究。文献[5]提出了一种带有网络参数共享机制的 MADDPG 算法,在此基础上,针对多智能体合作场景中奖励函数设计难题,提出了一种基于群体目标状态的奖励函数,并进一步把带优先级的经验重放方法引入多智能体领域,训练出了稳定的协同策略。文献[6]提出了一种基于赫布迹和行动者-评价者框架的多智能体强化学习方法,利用赫布迹加强游动策略的学习记忆能力,基于同构思想实现了多智能体的分布式学习。文献[7]提出了一种改进的多目标追踪方法,基于追踪智能体和目标智能体数量及其环境信息建立任务分配模型,运用匈牙利算法根据距离效益矩阵对其进行求解,得到多个追踪智能体的任务分配情况,并以缩短目标智能体的追踪路径为优化目标进行任务分工,同时利用多智能体协同强化学习算法使多个智能体在相同环境中不断重复执行探索—积累—学习—决策过程,最终根据经验数据更新策略完成多目标追踪任务。文献[8]提出一种基于 MADDPG 的改进算法——GAED-MADDPG,解决了多智能体强化学习算法收敛时间过长和可能无法收敛的问题。文献[9]提出了基于并行优先经验回放机制的 MADDPG 算法(PPER-MADDPG) ,采用并行方法完成经验回放池数据采样,并在采样过程中引入优先回放机制,实现经验数据并行流动,数据处理模型并行工作,经验数据优先回放,提升了 MADDPG 算法性能。文献[10]在基于 MADDPG 算法的基础上,设计了一种 CGF 空战策略生成算法,为了提高空战策略生成算法的效率,提出了一种基于潜力的奖励形成方法,得到的策略具有较好的收敛性和较好的空战性能。文献[11]提出了一种基于经典 MDRL 算法的 MADDPG 并行评价方法(MADDPG-PC) ,引入了一种策略平滑技术来减小学习策略的方差,提高了多智能体协同竞争环境下训练的稳定性和性能。文献[12]针对 MADDPG 算法学习效率低、收敛速度慢的问题,研究了一种优先体验重放(PER) 机制,提出了一种优先体验重放 MADDPG(PER-MADDPG) 算法,基于时间差(TD) 误差,设计了优先级评估功能,以确定从回放缓冲区中优先采样的体验,解决了智能体学习效率低、算法收敛速度慢的问题。

通过实验发现,仅采用现有的 MADDPG 算法用于无人集群系统协同对抗行为策略生成,奖励为稀疏奖励,在训练过程中,奖励信号变化不明显,导致智能体采用策略梯度算法进行探索时成功样本数量少,需要长时间训练才能达到最优策略,算法的收敛性表现较差,从而很难真正意义上实现对抗任务下无人集群系统自主行为快速学习,需要提高算法收敛效率。

本文改进 MADDPG 算法的奖励机制,提出 Per-Distance 奖励机制。①引入动作空间边界的惩罚、智能体时空距离惩罚,解决延迟奖励问题,以提高无人集群系统行为学习效率; ②通过智能体在群体中的关系特性,增加智能体间经验共享,提高无人集群系统合作学习效率。通过实验验证该方法提高了行为学习收敛速度,使其更加稳定。从而提高对抗任务下无人集群行为决策学习的效率。

2　问题描述

针对无人集群系统用多智能体强化学习算法解决行为策略学习,存在奖励稀松、学习效率低下的问题,本文采用 MADDPG 算法为学习算法,重新设计奖励函数,用于红方无人机群协同围捕蓝

方无人机任务案例中，以无人机为智能体实体，验证无人机群协同自主围捕行为效率。

2.1 无人机运动学模型

无人机集群协同围捕是集群作战的典型样式，作战空域内存在多个捕食者和逃逸者，两方无人机具有相反的战术目的，捕食者要追击捕食逃逸者，而逃逸者要躲避远离捕食者追踪。多无人机的捕食-逃逸场景如图 1 所示。

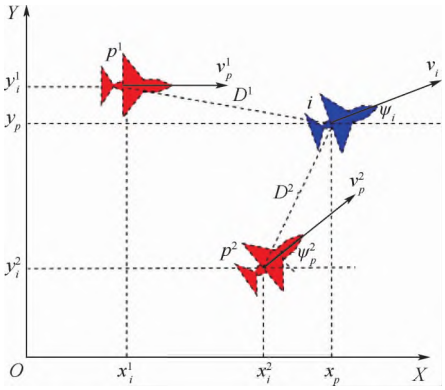

图 1 捕食-逃逸几何模型

Fig. 1 Geometric predation-escape model

本文假定捕食-逃逸问题在有限的二维平面内进行，图 1 为二维平面区域的捕食-逃逸对抗的笛卡儿直角坐标系，捕食者 i 和逃逸者 p 的速度分别为 v_i、v_p，速度航向角分别为 ψ_i、ψ_p。捕食者无人机的运动学方程为

$$\begin{cases}\dot{x} = v\cos\psi \\ \dot{y} = v\sin\psi \\ \dot{v} = a\end{cases} \tag{1}$$

式中：a 为捕食者加速度；v 为捕食者速度。两者的上限是无人机运动约束。

捕食者的行为决策目标是以最短时间捕获逃逸者，逃逸者的行为决策目标是远离捕食者，避免在预设的作战时间内被追捕或尽可能延迟被追捕到的时间。捕食-逃逸对抗数学描述为

$$\begin{cases}T_{\min} = f_i(s_i^1, s_i^2, \cdots, s_i^n) \\ T_{\max} = f_p(x_p, y_p, v_p, a_p, D^1, \cdots, D^n)\end{cases} \tag{2}$$

式中：$T_{\min}$为捕食者的最优行为决策目标；$T_{\max}$为逃逸者的最优行为决策目标；$s_i^n = (x_i^n, y_i^n, v_i^n, a_i^n)$为捕食者 i 的状态信息，包括坐标(x_i^n, y_i^n)、速度 v_i^n 和加速度 a_i^n，捕食者联合状态$(s_i^1, s_i^2, \cdots, s_i^n)$构成环境状态信息；$D^n$为捕食者 n 到逃逸者的距离，本文将捕食者和逃逸者作为智能体形状大小忽略不计，视为质点，捕获条件为$D^n = 0$。捕食者根据环境状态信息$(s_i^1, s_i^2, \cdots, s_i^n)$和逃逸者目标位置信息$(x_p, y_p)$来计算相对距离$D^n$，并以此进行决策输出相应的动作 $(\Delta x, \Delta y)$，Δx 为捕食者关于 X 方向上的动作输出，Δy 为捕食者关于 Y 方向上的动作输出，$(\Delta x, \Delta y)$ 为捕食者的速度v_i关于 2 次动作间隔时间的积分结果。逃逸者的动作输出同理。加速度控制规则为捕食成功后$(D = 0)$，捕食者加速度减小，逃逸者加速度增大。

捕食者和逃逸者的运动需满足边界约束：

$$\begin{cases}x_{\min} < x_p, x_e < x_{\max} \\ y_{\min} < y_p, y_e < y_{\max}\end{cases} \tag{3}$$

式中：$x_{\min}$、$y_{\min}$分别为环境边界的最小横纵坐标；$x_{\max}$、$y_{\max}$分别为环境边界的最大横纵坐标。当捕食者和逃逸者触碰到边界速度降为零。

2.2 MADDPG 算法

2.2.1 算法核心思想

在多智能体强化学习训练过程中，每个智能体的动作是实时变换的，从单智能体视角观测到的环境是不断变化的，从而造成学习算法收敛性差是多智能体深度强化学习当下的困境。为解决该问题，MADDPG 算法引入中心化训练、分布式执行的方法，采用 Actor-Critic（动作-评价）网络更新策略，以解决训练不稳定性问题。具体方法是：在智能体训练时，将其他智能体的动作信息 Actor 加入到环境状态中，在 t 时刻，添加所有智能体的执行动作，作为下一个时刻 t的环境状态，加入可以观察全局的 Critic 网络来指导 Actor 网络训练。测试时只使用有局部观测的 Actor 采取行动，将不稳定的环境状态变为稳定的环境状态，降低多智能体行为决策的复杂度。

2.2.2 基本假设

MADDPG 算法遵循马尔可夫决策过程[13]，可以定义为一个多元组 $\langle S, A_1, A_2, \cdots, A_n, R_1, R_2, \cdots, R_n, T, O, \gamma\rangle$。智能体所处的环境中包含了 n 个智能体，S 为环境的状态空间。$A_i(i = 1, \cdots, n)$ 表示单个智能体 i 的动作空间，而 $A_1 \times A_2 \times \cdots \times A_n$ 表示所有智能体的联合动作空间。R_i 由一系列的r_i求和而成，表示智能体 i 的奖励总额，r_i为多智能体执行联合动作 A 时，从状态$s \in S$ 转移到状态 $s' \in S$ 时智能体 i 所获得的即时奖励。T: $S \times A \times S \rightarrow [0, 1]$为状态转移函数，表示多智能体在状态 S 下，执行联合动作 A 后转移到状态 S'的概率分布。$o_i \in O$ 为智能体 i 对环境的观测值，观测属性又可以分为部分观测和完全观测。γ 为折扣因子，用于调节长期奖励与即时奖励之间的权重。在多智能环境中，状态转移是所有智能体共同行动的结果。n 个智能体根据自身的观测值o_i及所获得的即时奖励r_i做出行为决策a_i，共同输出联合

动作 A 促使环境状态 S 发生转移。因此,智能体的奖励与联合策略有关。所有智能体的参数集合为 $\theta=\{\theta_1,\theta_2,\cdots,\theta_n\}$。假设智能体每次采用的确定性策略为 μ,每一步的动作都可以通过公式 $a_t=\mu(S_t)$ 获得,而执行某一策略后获得的奖励,奖励值大小由 Q 函数决定,实现确定通信方式下多智能体的竞争、合作博弈。

算法运行条件为: ①学习策略基于单 Agent 视角观测信息; ②Agent 自身的行为仅仅取决于策略; ③Agent 之间的通信为全联通模式。

2.2.3　算法执行

算法执行过程如图 2 所示。区别于共享环境下 Agent 视角,每个 Agent 的输入状态不一样,每个执行者与环境交互,无需关注其他 Agent 状态,环境输出下一个全信息状态 S_{all} 后,执行者 Actor_1 和 Actor_2 只能获取自己能够观测到的部分状态信息 s_1、s_2,分别执行图 2 中绿色线标识的循环部分。训练过程中,评论家 Critic_1 和 Critic_2 可以获得全信息状态,同时还能获得所有 Agent 采取的策略动作 a_1、a_2。

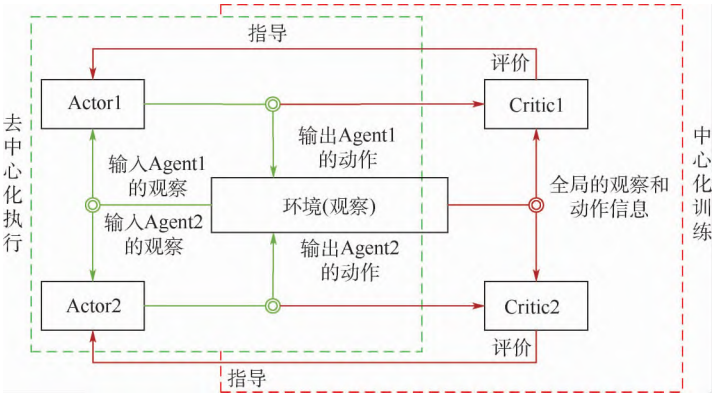

图 2　MADDPG 算法训练执行视图

Fig. 2　Training execution of MADDPG algorithm

2.2.4　Actor-Critic 网络更新策略

如图 3 所示,在环境 Evns 中,智能体由 Actor 网络和 Critic 网络构成,这 2 个网络又分别包含目标网络(target-net)和估计网络(eval-net)。Actor 网络是卷积神经网络对策略函数 π 的模拟,参数为 θ^{π}。Critic 网络是卷积神经网络对奖励函数 Q 的模拟,参数为 θ^{Q}。

Actor 网络表示为

$$\begin{cases}\text{eval-net}: (s\mid\theta^{\pi})\\ \text{target-net}: (s\mid\theta^{\pi'})\end{cases}\tag{4}$$

Critic 网络表示为

$$\begin{cases}\text{eval-net}: (a\mid\theta^{Q})\\ \text{target-net}: (a\mid\theta^{Q'})\end{cases}\tag{5}$$

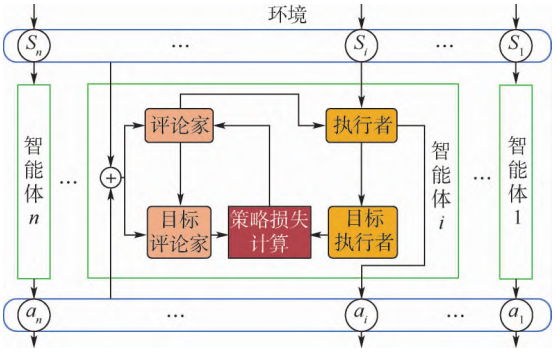

图 3　MADDPG 算法训练框架

Fig. 3　Training framework of MADDPG algorithm

Actor 网络看不到全部的环境状态信息,不知晓其他智能体的策略,但是每个智能体的 Actor 网络有一个拥有全部视角的导师 Critic 网络,该导师可以观测到所有信息,并指导对应的 Actor 网络优化策略。在训练过程中,只需要估计网络(eval-net)的参数,目标网络(target-net)参数每隔一定时间直接复制估计网络参数[14]。依据 MADDPG 算法,训练时引入可观察全局的 Critic 网络来指导 Actor 网络训练,执行时仅使用有局部观测的 Actor 网络采取行动[4]。智能体 i 采取的动作 $a_i^j=\pi_i^j s_i^j$,获得经验 $(a_i^j,\pi_i^j,s_i^{j+1},r_i^j)$ 存储起来。等到所有智能体与环境交互后,每个智能体将缓冲区存储的行为经验加入到策略网络 Critic 中进行训练,Critic 网络中添加所有的智能体的观测状态信息和行为动作,定义如下:

$$Q=Q\{S_j,a_1,a_2,\cdots,a_n,\theta^{Q}\}$$

式中:

$$S_j=\{s_1^j,s_2^j,\cdots,s_n^j\}\tag{6}$$

由此知道了所有智能体的动作,即使策略发生变化,那么环境也是静止的,随即通过梯度下降更新每个 Agent 的行为者 Actor 网络参数:

$$\nabla\theta^{\pi}J(\theta_i)=\frac{1}{k}\sum_{j=1}^{k}\nabla\theta^{\pi}\pi(S,\theta^{\pi})\,\nabla_a Q(S_j,a_1,a_2,\cdots,a_n,\theta^{Q})\tag{7}$$

3 奖励机制的改进

3.1 奖励函数设置机制

在对抗任务中,多智能体的奖励不仅取决于自身策略,也取决于对手学习到的对抗策略,两者的策略学习速度未必同步,导致其奖励未必会持续升高,甚至出现波动和震荡[15]。由于智能体之间存在关系结构的约束,会对策略学习产生影响。如何设计一个无人集群系统合适的奖励信号来解决竞争对抗环境中智能体快速学习和稳定收敛,就成为了一个关键问题。

3.2 MADDPG 算法奖励机制缺陷的实例

在捕食者 i 和逃逸者 p 实例中,MADDPG 算法将对抗双方分别标识为捕食者和逃逸者,捕食者奖励机制是碰撞时碰撞者奖励值 +10,不碰撞的时间内,惩罚值 -1,逃逸者与此相反。这种奖励机制的好处就是捕食者和逃逸者的奖励值绝对值大小相同,双方的策略相反,因此二者的学习速度会逐渐达到同步,缓解了式(7)中 $\nabla\theta^{\pi}$ 估计值方差波动和震荡,如表1所示。

表1 奖励机制设置

Table 1 Reward mechanism setting

执行者	碰撞	不碰撞
捕食者	+10	-1
逃逸者	-10	+1

通过实验发现,MADDPG 算法奖励信号太过疏松,最优解收敛效率低,即学习效率低,需要增加即时奖励信号,以增强学习效率。

3.3 基于环境信息的显式 Per-Distance 奖励机制

智能体和环境、其他智能之间的某些关系可以显式地描述,免除学习,并能够提供及时的学习信号。本文对 MADDPG 算法奖励机制进行改进,提出 Per-Distance 的智能体和环境之间显式关系的奖励机制、智能体之间关系的经验共享奖励信号,以提高学习的稳定性和效率。

3.3.1 智能体和环境关系的先验奖励信号

智能体执行早期获得的成功样本很少,导致经验池缺乏足够的学习经验用来调整策略。在 Critic 网络更新时,大部分时间里,捕食者回报值为 $r_i^j = -1$,逃逸者的回报值为 $r_p^k = 1$,Loss 为 $\alpha[R(s,a) + \gamma\max Q'(s',a') - Q(s,a)]$ 几乎没有变化(α 为学习率),智能体做任意动作后的奖励值是相同的,Critic 网络无法区分动作优劣,奖励函数不稳定,训练收敛速度很慢。为此,加入越界约束、智能体之间距离约束等先验价值,以提升奖励函数的收敛效率。

MADDPG 算法的 3V1 围猎场景下,捕食者与逃逸者之间的距离为

$$D(i,p) = \sqrt{(x_i - x_p)^2 + (y_i - y_p)^2} \tag{8}$$

当捕食者与逃逸者之间的距离等于 0 时,捕食者会收获较大的奖励,同时,逃逸者返回的奖励值为负。碰撞奖励为

$$C = \begin{cases} 10 & \text{碰撞} \\ -1 & \text{不碰撞} \end{cases} \tag{9}$$

实验发现,智能体经常出现越界情况,为提高计算效率,尽量保证智能体在设置的运行范围内产生对抗行为,对智能体越界想法进行限制。增加边界奖励,具体做法为:对逃出边界的智能体,施加较大的惩罚,惩罚大小取决于远离边界的程度。Per-Distance 奖励机制中增加边界奖励 B,保证智能体在环境范围内运动,不产生越界逃逸行为。设 (x_i,y_i) 为智能体 i 在二维环境中的坐标,0.9 为智能体 i 的直径,如果该智能体离边界的最大距离小于 0.9,则认为是超出边界,边界奖励值 $B=0$。如果智能体离边界的最大距离大于 0.9,边界奖励值 B 给定一个智能体与边界距离有关的动态奖励值 $(\max(x_i,y_i) - 0.9)m$,为任意给定的权重值,起到放大系数的作用,本文实验 $m=200$。

以捕食者为例,边界奖励 B 为

$$B = \begin{cases} 0 & \max(x_i,y_i) < 0.9 \\ (\max(x_i,y_i) - 0.9)200 & \text{其他} \end{cases} \tag{10}$$

在 MADDPG 算法中,捕食者的奖励公式为

$$r_i = B + C \tag{11}$$

被捕食的奖励公式为

$$r_p = B - C \tag{12}$$

3.3.2 智能体之间关系的先验奖励信号

MADDPG 算法的奖励机制,对抗双方的距离设定仅有 2 种状态,即 $D(i,p) > 0$ 不碰撞或 $D(i,p) \leqslant 0$ 碰撞。真实情况是:大部分时间对抗双方处于 $D(i,p) > 0$ 不碰撞状态,需要很长时间才能训练得到最优策略,造成延迟奖励。

为解决延迟奖励问题,Per-Distance 奖励机制中增加智能体之间独立计算的距离参数。不再是 $D(i,p) > 0$ 不碰撞或 $D(i,p) \leqslant 0$ 碰撞 2 种状态下的奖励值,改为根据距离可变动态设置奖励值,增加距离参数 $D(i,p)$ 表示每个捕食者与逃逸者之间的距离,距离值在 $(-1,1)$ 区间内变化,距离越大奖励 r_i 越小,实现通过距离参数引导智能体快速地发生碰撞,以解决原算法中因距离状态过少

而产生的奖励延迟问题。通过实验调参发现,距离参数 0.1 为最优,此时有利于 Per-Distance 奖励机制的稳定。

捕食者 i 的奖励机制为

$$r_i = -0.1D(i,p) + B + C \tag{13}$$

相对于捕食者,逃逸者是反向奖励,只要逃离距离自己最近的捕食者,决策就是成功的,因此逃逸者只需要计算与自己距离最近的捕食者的距离并计算回报值。

逃逸者 p 的奖励机制为

$$r_p = 0.1\min(D(i,p)) + B - C \tag{14}$$

在实验中发现,增加智能体之间距离动态关系的奖励机制,奖励值随捕食者和逃逸者之间的距离变化,奖励信号明显,智能体行为策略对应的动作区分明显,有利于奖励值收敛。说明考虑智能之间的关系对性能提高有明显影响。

3.3.3　增加智能体间经验值共享

MADDPG 算法中,回报值 Critic 网络中共享容易造成最大值回报值的智能体的行为在群体中扩散,使得其他智能采取类似的策略。为了避免智能体策略的相似性,可以采取以下策略:

1) 集中式训练,分布式执行模式的 MADDPG 算法是各智能体执行的队长,分配每个智能体行为。在 Per-Distance 奖励机制下,如果智能体的动作趋同,则同一动作的奖励值会降低,避免所有智能体采取同一行为去抓捕逃逸个体,防止整体抓捕效率下降。

2) 随机化最大回报,将最大回报加上随机数 $r' \in (-1,1)$,每个智能体的回报值修改为 $r'r$。

通过增加其他智能体对当前智能体的影响,同时,把价值引入到学习信号中,最终使得 Critic 网络在策略更新时能更好地识别出不同动作值之间奖励值的差异,提高学习的稳定性。

4　实验结果与分析

4.1　实验环境

实验设计是将原算法和改进后算法奖励曲线和智能体实际表现进行对比分析。

实验软件环境为 Windows10 操作系统; 硬件环境为英特尔至强 E78880v3 * 2 型处理器、NVIDIA GTX 1080TI* 3、64 GB 内存; 测试环境为 OpenAI-gym,隐藏层为 2 层、隐藏单元个数为 64 的全连接神经网络构成的 Actor、Critic 网络及对应的目标网络和估计网络。

模型超参数设计为: Actor 网络与 Critic 网络均采用采用全连接 4 层神经网络结构,隐藏层神经元数量为 64。每个 Actor 网络拥有单独的 Critic 网络,实验中发现,由于引入了距离参数,捕食者的距离参数值为负值,探索初期回报值将长时间停留在负数区间,而神经网络中的激活函数的负数域非常小,不利于训练,将神经网络输出层的激活函数去掉。

4.2　实验场景

Simple_tag 实验场景是在对抗任务下智能体自主行为仿真,实验空间为二维有界密闭空间,包含 4 个智能体,其中,3 个捕食者(蓝色),1 个逃逸者(红色),实验场景描述如表 2 所示。

表 2　3V1 对抗实验场景

Table 2　Experimental scenario of 3 versus 1 confrontation

场景名称	是否对抗	智能体数量	获胜条件
Simple_tag	是	3 红 VS 1 蓝	蓝: 蓝色 Agent 尽可能避免与 3 个红色 Agent 的碰撞 红: 3 个红色 Agent 之间尽可能协同与蓝色 Agent 发生碰撞

实验参数设置如下:

1) 捕食者和逃逸者作为智能形状大小忽略不计,视为质点。

2) 坐标轴上智能体的运动范围为 [0,20]。

3) 3 个捕食者合作共同追捕 1 个逃逸者。

4) 捕获者速度上限为 1.0/s,加速度上限为 0.5/s^2,逃逸者速度上限为 1.3/s,加速度上限为 0.7/s^2。

5) 当捕食者和逃逸者发生碰撞即间距为 0 时,视为捕获者捕获成功,逃逸者失败。

6) 碰撞规则为碰撞后捕食者加速度减小,逃逸者加速度增大。

7) 捕食者和逃逸者触碰到边界速度降为零。

4.3　实验分析

在不降低捕食者的捕获效果情形下,提升算法收敛速度和稳定性。捕食者分别将 Per-Distance 奖励机制放入 MADDPG 算法和 DDPG 算法进行学习训练。表 3 为经过 20 000 轮训练后,多智能体学习到的策略在 1 000 轮、60 000 步随机实验下,捕食者执行每步动作后的平均碰撞次数。与原算法相比,看到加入 Per-Distance 奖励机制捕获效果更好,MADDPG 相较于原算法平均碰撞次数提高了 3.3%,DDPG 算法平均碰撞次数提升幅度较小为 1.7%。

对上述实验进行 40 000 轮的训练,利用 TensorFlow 的可视化工具 TensorBoard 描绘出捕食者和逃逸者的奖励值与训练次数之间的关系,对比

MADDPG算法奖励机制和改进后的Per-Distance奖励机制的关系曲线。

图4~图6分别为捕食者1、捕食者2、捕食者3的奖励函数曲线。可以看出,引入距离参数,随道捕食者与逃逸者的距离增大,捕食者的奖励回报值减小,导致整体回报值呈降低趋势,奖励函数曲线下移。随着碰撞次数的增多,更多的直接奖励值开始叠加,使得奖励曲线下降趋势减缓并稳定下来。在改变奖励机制后,算法的收敛速度有较大

表3 平均每步碰撞次数

Table 3 Average number of collisions per step

算法	奖励机制是否改进	平均碰撞次数
MADDPG	是	0.538
	否	0.521
DDPG	是	0.532
	否	0.523

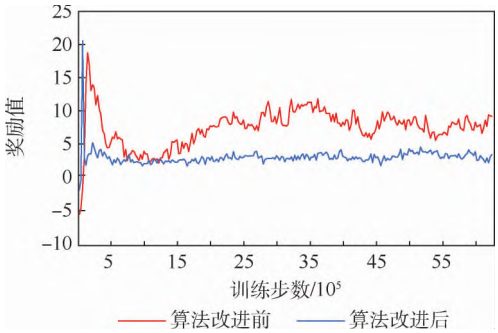

图4 捕食者1奖励函数曲线

Fig. 4 Curves of Predator 1 reward function

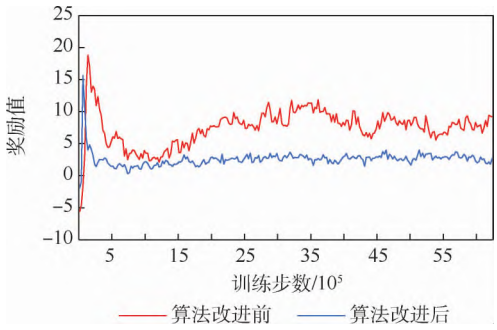

图5 捕食者2奖励函数曲线

Fig. 5 Curves of Predator 2 reward function

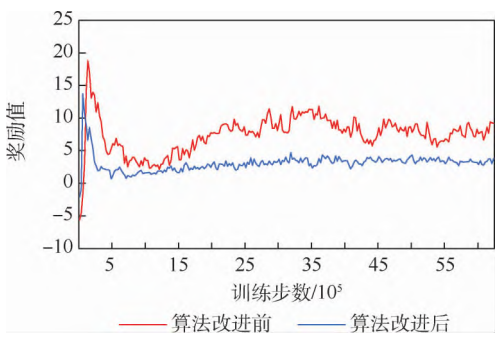

图6 捕食者3奖励函数曲线

Fig. 6 Curves of Predator 3 reward function

提升,在5 000轮左右奖励值趋于平稳,奖励值在[2,4]区间内缓慢波动。通过实验证明捕食者奖励函数的收敛性、算法的稳定性提升十分明显。

如图7所示,由于捕食者捕获效果的提升,逃逸者获得的负奖励(也称为惩罚)大大增加,导致奖励函数值减小。相较于捕食者收敛速度的明显改善,逃逸者奖励值的收敛速度改善效果不够突出,这是因为逃逸者要计算与捕食者中的最小距离,当离自己最近的捕食者更换时,策略网络要重新计算最小距离,更新步长较大,收敛性会打折扣。

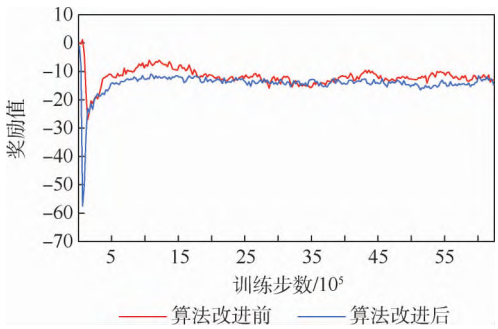

图7 逃逸者奖励函数曲线

Fig. 7 Curves of escaper reward function

2种算法下,逃逸者的奖励值拐点都处于1 200 000步左右。在原来奖励机制下,逃逸者奖励函数的稳定性较差,奖励值在[-6,-15]的较大域值内上下浮动,函数曲线震荡幅度很大。引入Per-Distance奖励机制后,曲线波动幅度见减小,函数值在[-12,-16]的区间内变化,收敛性也有所提升。新的奖励机制对于逃逸者函数也有改进作用。

对上述实验中所有智能体的奖励值进行叠加,绘制出奖励值总和与训练轮步数的曲线,如图8所示。图8对比很明显,红色曲线显示原算法奖励值曲线在大范围波动,收敛性不好,引入Per-Distance奖励机制后,蓝色曲线显示奖励值曲线较早地进入到小区间波动,算法收敛性及稳定性得到了显著提升。

此外,为了进一步地评估Per-Distance奖励机制下算法的有效性,又与PES-MADDPG算法进行了奖励值与训练步数比较,如图9所示,依然是引入了Per-Distance奖励机制的PD-MADDPG算法奖励值收敛速度快,更快地趋于稳定。

上述实验中看到,在Per-Distance奖励机制下,智能体行为策略对应的动作区分明显,有利于奖励值收敛,说明考虑智能体的先验知识和智能体之间的关系对性能提高有明显影响。

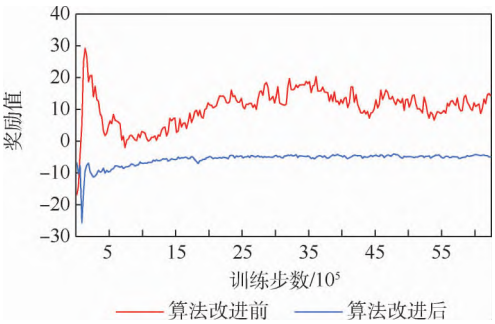

图 8　奖励函数曲线总和

Fig. 8　Reward function curve sum

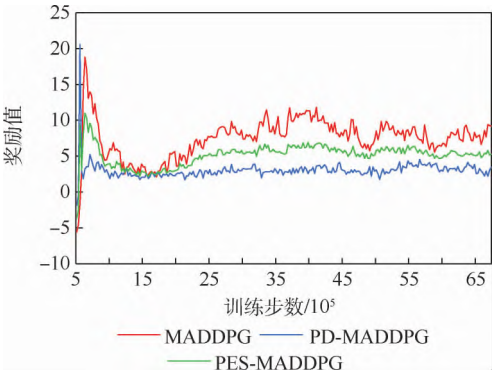

图 9　MADDPG、PD-MADDPG、PES-MADDPG 算法奖励函数收敛性对比

Fig. 9　Reward function convergence comparison among MADDPG, PD-MADDPG and PES-MADDPG algorithms

4.4　SwarmFlow 仿真平台

在陆军工程大学控制技术与智能系统实验室自主开发的智能陆战协同对抗仿真平台 SwarmFlow 上训练改进后的算法，加载山地三维地图。图 10 为 3 架捕食者无人机围捕 1 架逃逸者无人机，实施一次围捕任务时三维可视化效果及围捕航迹图。图 11 和图 12 分别为捕食者无人机和逃逸者无人机一次任务的航迹。

图 13 为 SwarmFlow 仿真平台展示的陆战场景下，引入 Per-Distance 奖励机制的 MADDPG 算法，智能体 3V1 最终围捕效果。该算法可以推广至更多的智能体，算法对集群自主系统亦具有适应性。随着集群数量的增加，状态空间指数级增加，行为策略学习训练时间很长。图 14 展示了集群智能体 20V6 的围捕效果。

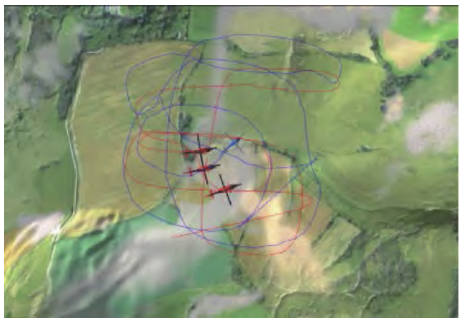

图 10　对抗任务下双方航迹

Fig. 10　Track map of both parties under confrontation mission

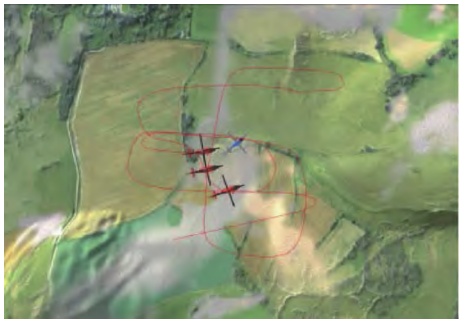

图 11　捕食者无人机航迹

Fig. 11　Predator UAV track map

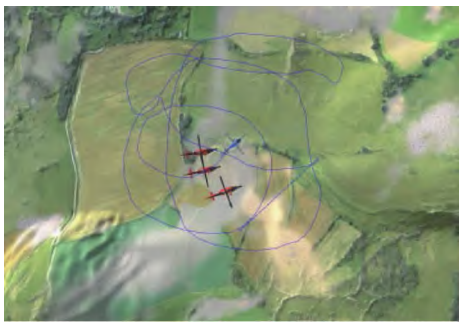

图 12　逃逸者无人机航迹

Fig. 12　Escaper UAV track map

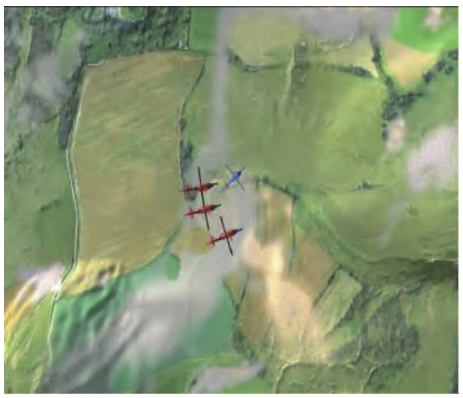

图 13　智能体 3V1 围捕结果

Fig. 13　Result of agent 3V1 roundup

图 14　智能体 20V6 围捕结果

Fig. 14　Result of agent 20V6 roundup

5 结束语

目前,将多智能体强化学习算法用于无人系统自主行为决策研究,最大的问题是算法收敛速度慢,使得在无人集群系统中的应用效果较差。

为了解决这一问题,本文着重研究了 MADDPG 算法的奖励机制,引入距离参数,提出 Per-Distance 奖励机制。在对抗任务下,改变回报值共享的方式,将对抗双方的距离奖励传递给执行智能体,解决延迟奖励问题。通过 3V1 围猎场景的仿真实验验证了改进的奖励机制实用性和优越性,提高了对抗任务下无人集群系统的群体行为策略学习效率。该算法可应用于集群对抗任务。

后期的研究中,考虑如何实现提升大规模集群行为决策效率问题。

参考文献(References)

[1] 张婷婷,宋爱国,蓝羽石. 集群无人系统自适应结构建模与预测[J]. 中国科学: 信息科学,2020,50(1):347-362.
ZHANG T T,SONG A G,LAN Y S. Adaptive structure modeling and prediction of cluster unmanned system [J]. Chinese Science: Information Science,2020,50(1):347-362(in Chinese).

[2] 孙长银,穆朝絮. 多智能体深度强化学习的若干关键科学问题[J]. 自动化学报,2020,46(7):1301-1309.
SUN C Y,MU C X. Important scientific problems of multi-agent deep reinforcement learning [J]. Journal of Automatica Sinica,2020,46(7):1301-1309(in Chinese).

[3] 陈杰. 多智能体系统中的几个问题[J]. 中国科学人,2019,12(1):40-43.
CHEN J. Several problems in multi-agent system [J]. Scientific Chinese,2019,12(1):40-43(in Chinese).

[4] LOWE R,WU Y I,TAMAR A,et al. Multi-agent actor-critic for mixed cooperative-competitive environments[EB/OL]. (2020-03-14) [2020-03-22]. http://arxiv.org/abs/1706.02275.

[5] 许诺,杨振伟. 稀疏奖励下基于 MADDPG 算法的多智能体协同[J]. 现代计算机,2020(15):47-51.
XU N,YANG Z W. Multi-agent collaboration based on MADDPG algorithm under sparse reward [J]. Modern Computer,2020(15):47-51(in Chinese).

[6] 杨慧慧,黄万荣,敖富江. 基于强化学习的鱼群自组织行为模拟[J]. 国防科技大学学报,2020,42(1):194-202.
YANG H H,HUANG W R,AO F J. Simulation on self-organization behaviors of fish school based on reinforcement learning [J]. Journal of National University of Defense Technology,2020,42(1):194-202(in Chinese).

[7] 王毅然,经小川,贾福凯,等. 基于多智能体协同强化学习的多目标追踪方法[J]. 计算机工程,2020,46(11):90-96.
WANG Y R,JING X C,JIA F K,et al. Multi-target tracking method based on multi-agent collaborative reinforcement learning[J]. Computer Engineering,2020,46(11):90-96(in Chinese).

[8] 邹长杰,郑皎凌,张中雷. 基于 GAED-MADDPG 多智能体强化学习的协作策略研究[J]. 计算机应用研究,2020,37(12):3656-3661.
ZOU C J,ZHENG J L,ZHANG Z L. Research on collaborative strategy based on GAED-MADDPG multi-agent reinforcement learning [J]. Application Research of Computers,2020,37(12):3656-3661(in Chinese).

[9] 高昂,董志明,李亮,等. MADDPG 算法并行优先经验回放机制[J]. 系统工程与电子技术,2021,43(2):420-433.
GAO A,DONG Z M,LI L,et al. Parallel priority experience replay mechanism algorithm of MADDPG [J]. Systems Engineering and Electronics,2021,43(2):420-433(in Chinese).

[10] WEIREN K,DEYUN Z,ZHEN Y. Air combat strategies generation of CGF based on MADDPG and reward shaping [C]//2020 International Conference on Computer Vision,Image and Deep Learning (CVIDL). Piscataway: IEEE Press,2020:651-655.

[11] SUN Y,LAI J,CAO L,et al. A novel multi-agent parallel-critic network architecture for cooperative-competitive reinforcement learning [J]. IEEE Access,2020,8:135605-135616.

[12] ZHU P,DAI W,YAO W,et al. Multi-robot flocking control based on deep reinforcement learning [J]. IEEE Access,2020,8:150397-150406.

[13] VAN OTTERLO M,WIREING M. Reinforcement learning and Markov decision processes [M]//WIREING M,VAN OTTERLO M. Reinforcement learning. Berlin: Springer,2012:3-42.

[14] 陈亮,梁宸,张景异,等. Actor-Critic 框架下一种基于改进 DDPG 的多智能体强化学习算法[J]. 控制与决策,2021,36(1):75-82.
CHEN L,LIANG C,ZHANG J Y,et al. A multi-agent reinforcement learning algorithm based on improved DDPG under actor critical framework [J]. Control and Decision,2021,36(1):75-82(in Chinese).

[15] 孙彧,曹雷,陈希亮,等. 多智能体深度强化学习研究综述[J]. 计算机工程与应用,2020,56(5):13-24.
SUN Y,CAO L,CHEN X L,et al. A review of multi-agent deep reinforcement learning research [J]. Computer Engineering and Application,2020,56(5):13-24(in Chinese).

Behavioral decision learning reward mechanism of unmanned swarm system

ZHANG Tingting[1,2,3,*], LAN Yushi[2], SONG Aiguo[3]

(1. School of Command and Control Engineering, Army Engineering University of PLA, Nanjing 210017, China;
2. The 28th Research Institute of China Electronics Technology Group Corporation, Nanjing 210017, China;
3. School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China)

Abstract: Unmanned swarm system is composed of a multi-agent system, which can meet task requirements through autonomous and cooperative behavior. The instability of agent training is increased because agents adopt behavior and change states autonomously. In this paper, the prior constraints and the isomorphism between agents are used to enhance the real-time performance of reward signals and improve the efficiency of training and the stability of learning. Specifically, it includes the punishment of action space boundary collision and the reward for the satisfaction degree of the space-time distance constraint between agents. At the same time, through the relationship characteristics of agents in the group, experience sharing among agents is increased to further optimize the learning efficiency. In the experiment, the prior enhanced reward mechanism and experience sharing are applied to the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm to verify its effectiveness. It is observed that the learning convergence and stability are greatly improved, and thus the behavior learning efficiency of unmanned swarm system is enhanced.

Keywords: unmanned swarm system; Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm; confrontation mission; behavioral decision; reward mechanism

Received: 2020-10-23; **Accepted**: 2021-04-23; **Published online**: 2021-05-21 14:14

URL: kns.cnki.net/kcms/detail/11.2625.V.20210520.1657.002.html

Foundation items: National Natural Science Foundation of China (61802428); China Postdoctoral Science Foundation (2019M651991); National Defense Science and Technology Project Fund of Science and Technology Commission of the Military Commission (2019-JCJQJJ-014)

* **Corresponding author**. E-mail: 101101964@seu.edu.cn