doi:10.3969/j.issn.1003-3106.2025.11.009
引用格式:苑司宇,康国钦,郑学强,等.面向指挥决策的 DRA-MADDPG 协同控制方法[J].无线电工程,2025,55(11):2218-2226.[YUAN Siyu, KANG Guoqin, ZHENG Xueqiang, et al. DRA-MADDPG Cooperative Control Method for Command Decision-making[J]. Radio Engineering, 2025, 55(11): 2218-2226.]

面向指挥决策的 DRA-MADDPG 协同控制方法

苑司宇[1],康国钦[2],郑学强[3],周强强[1]
(1.国防科技大学,湖北 武汉 430035;
2.信息支援部队工程大学,湖北 武汉 430035;
3.陆军工程大学,江苏 南京 210001)

摘　要:随着人工智能等技术的发展,多智能体如无人机群等的实际应用领域逐渐广泛。多智能体深度确定性策略(Multi-Agent Deep Deterministic Policy Gradient,MADDPG)算法旨在解决多智能体在协作环境中的协同配合问题,凭借其独特的 Actor-Critic 架构已成为多智能体领域主流的应用算法之一。针对指挥决策中多智能体协同任务存在的角色分工模糊、信息过载导致的算法策略收敛较慢等问题,提出了一种引入动态角色注意力(Dynamic Role Attention,DRA)机制的改进 MADDPG 算法——DRA-MADDPG。该算法在 Actor-Critic 架构中嵌入了 DRA 模块,通过动态调整智能体对不同角色同伴的关注权重,来实现分工协作的精准优化。具体而言,定义了指挥任务的角色集合与阶段划分,进而构建角色协同矩阵和阶段调整系数;在 Critic 网络中设计 DRA 模块,依托角色相关性与任务阶段来计算权重并筛选关键信息;改进了 Actor 网络,结合角色职责生成针对性的动作。仿真实验表明,与 MADDPG 相比,DRA-MADDPG 的训练累积回报曲线下面积(Area Under the Curve,AUC)提升了 2.4%,任务完成耗时降低了 19.3%,且通过训练回报曲线对比分析可知,DRA-MADDPG 对于短期训练拥有更好的学习效率。证明了该方法适用于复杂指挥决策场景,为多智能体协同提供了一种相对高效的解决方案。

关键词:指挥决策;多智能体强化学习;多智能体深度确定性策略;动态角色注意力;协同控制
中图分类号:E9;TP391.9　　**文献标志码**:A　　**开放科学(资源服务)标识码(OSID)**:
文章编号:1003-3106(2025)11-2218-09

DRA-MADDPG Cooperative Control Method for Command Decision-making

YUAN Siyu[1], KANG Guoqin[2], ZHENG Xueqiang[3], ZHOU Qiangqiang[1]
(1. *National University of Defense Technology*, *Wuhan* 430035, *China*;
2. *Information Support Force Engineering University*, *Wuhan* 430035, *China*;
3. *Army Engineering University of PLA*, *Nanjing* 210001, *China*)

Abstract: With the development of technologies such as artificial intelligence, multi-agents (e.g., unmanned aerial vehicle swarms) have been increasingly applied in practical combat operations. The Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm, designed to solve the coordination problems of multi-agents in cooperative environments, has become one of the mainstream applied algorithms in the multi-agent field owing to its unique Actor-Critic framework. To address the problems in multi-agent collaborative tasks during command and decision-making—including ambiguous role division and slow convergence of the algorithm's policy caused by information overload—an improved MADDPG algorithm incorporating a Dynamic Role Attention (DRA) mechanism, namely DRA-MADDPG, is proposed. This algorithm embeds a DRA module into the Actor-Critic framework, and achieves accurate optimization of division of labor and collaboration by dynamically adjusting the attention weights of each agent towards peers with different roles. Specifically, the role set (reconnaissance, assault, command) and phase division (exploration→execution→encirclement) for command tasks are defined, and on this basis, a role coordination matrix and phase adjustment coefficients are constructed. A DRA module is designed in the Critic network to calculate weights and filter key information by leveraging role relevance and task phases. Additionally, the Actor network is improved to generate targeted actions by integrating role responsibilities. Simulation experiments show that compared with MADDPG, the Area Under the Curve (AUC) of the cumulative training reward of DRA-MADDPG increases by 2.4%, and the task completion time decreases by 19.3%. Furthermore, comparative analysis of training reward curves reveals that DRA-MADDPG exhibits better learning efficiency in short-term training. It is demonstrated that this method is suitable for

收稿日期:2025-08-27

complex command and decision-making scenarios and provides a relatively efficient solution for multi-agent coordination.

Keywords: command and decision-making; multi-agent reinforcement learning; MADDPG; DRA; cooperative control

0 引言

多智能体强化学习是人工智能领域的重要技术之一,它具有自主学习、分布协调和组织的能力,通过与其他智能体的协作配合,规划自己的行为,改变自己的状态信息,最终高效地完成任务[1]。目前,在无人机群围捕、多梯队攻防等指挥决策场景中,多智能体强化学习的应用日益广泛,而 MADDPG 算法凭借其独特的“集中训练、分布执行”框架成为当前研究的热点,国内外不少学者已经针对 MADDPG 算法存在的收敛慢及信用分配问题提出了改进方法。

邹长杰等[1]提出了分组学习策略,通过循环神经网络(Recurrent Neural Network,RNN)预测分组矩阵,在组内进行共享经验,同时引入信息微量使其在所有智能体间传递全局信息,相比于 MADDPG 的训练时间减少了 12%~17%;刘峰等[2]提出了 Att-MADDPG 方法,通过注意力机制增强智能体之间的相互关注,优化了无人机群围捕控制的性能;Foerster 等[3]提出的 COMA 算法通过反事实基线和中心评论家模式解决了多智能体信用分配问题,该算法通过反事实基线来评估单个智能体的贡献,同时保持其他智能体的行动不变,显著提高了学习效率;贾思雨等[4]针对 MADDPG 的收敛问题,引入了碰撞区域重点训练、经验池分离和优先经验回放机制,使多机器人路径规划任务成功率提升 21%~32%;符小卫等[5]提出的 DE-MADDPG 算法通过解耦方式设计了全局和局部 2 种奖励函数,使得无人机在追捕任务中比 MADDPG 更快地收敛,有效协调了多无人机的协同行为;孙彧等[6]将现有算法划分为无关联型、通信规则型、互相协作型和建模学习型 4 类,其中 MADDPG 因“集中训练、分布执行”框架,被归为互相协作型核心算法,虽能缓解环境非平稳性,但仍存在智能体数量多时收敛慢、信度分配难的共性问题。

现有研究表明,MADDPG 及其改进算法在多智能体协同决策中展现出显著优势,但应用于指挥决策领域仍存在一定的局限性:一是对于智能体在执行任务时的角色分工相对模糊;二是对于 MADDPG 算法在智能体的数量较多时收敛较慢的问题上仍具有可优化空间;三是对于指挥决策任务中从观察到执行的动态阶段变化,智能体策略难以快速适配阶段目标。

针对以上分析,本文提出了一种引入 DRA 机制的 MADDPG 改进算法——DRA-MADDPG,旨在让智能体能够更好地适配指挥决策任务的不同阶段变化,为提高多智能体在实际应用中的效能提供解决方案。

1 问题描述

1.1 指挥决策环境建模

1.1.1 任务场景描述

为便于对比 DRA-MADDPG 算法相较于 MADDPG 算法更适用于指挥决策类任务,本文以无人机围捕行动为作为任务场景,在二维空间中,部署有初始位置随机的 N 个围捕无人机 $U_i(i=1,2,\cdots,N)$ 与 1 个动态逃逸目标 T。假设各个围捕无人机之间可以通过通信网络 W 实时共享状态,且假设任务环境中不存在电子干扰设备,即通信稳定。任务目标设置为在有限时间 t 内,围捕无人机会根据角色分工形成以 T 为中心、半径 R 的包围圈。包围圈默认为理想状态,即负责实施围捕的无人机均匀分布在包围圈上[2]。以 $window=50$ km、$N=4$ 为例,任务场景如图 1 所示。

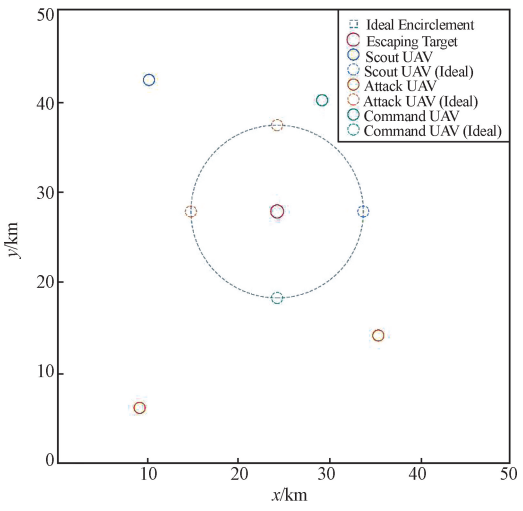

图 1 任务场景

Fig. 1 Task scenario

1.1.2 角色与阶段设计

(1)设置角色集合 $\xi=\{r_1,r_2,r_3\}$,其中各角色职责分工概括如下:

r_1(探测角色)负责感知目标位置,不断扩大搜索范围;r_2(执行角色)负责快速接近目标,对目标实施包围,不断缩小包围圈;r_3(调度角色)负责整合全

局信息,协调 DRA 变化。此设计是基于文献[7]中通过分层强化学习框架验证了“全局控制+局部执行”的层级逻辑可显著提升异构多智能体的协同效率[7],为调度角色主导全局、探测/执行角色执行局部任务的分工设计理念提供了理论参考。

(2)设置任务阶段集合 $\phi=\{\phi_1,\phi_2,\phi_3\}$,各个阶段设置如下:$\phi_1$(探索阶段)由探测角色主导,自任务开始至目标出现在探测角色探测范围时结束,并转入执行阶段;ϕ_2(执行阶段)由调度角色主导,控制角色之间的协同配合,直至目标与执行角色之间距离处于有效执行距离时结束,并转入包围阶段;ϕ_3(包围阶段)由执行角色主导,实施执行并形成包围圈[8]。

设计探测、执行、调度 3 类角色及探索→执行→包围三阶段,是基于指挥决策场景的协同需求。文献[9]指出,多智能体对抗中“奖励稀疏”会导致策略收敛慢,因此通过阶段划分,为不同阶段匹配核心角色并设计针对性奖励,可以有效缓解该问题[9]。

1.2 数学模型构建

1.2.1 状态与动作空间设置

围捕无人机 U_i 的状态 $o_i=[x_i,y_i,v_i,\varphi_i,r_i,s_i]$,其中$(x_i,y_i)$为位置,$v_i$ 为围捕无人机飞行速度,φ_i 为航向角,$r_i\in\xi$ 为角色,$s_t\in S$ 为当前阶段;目标 T 的状态 $o_T=[x_T,y_T,v_T,\varphi_T]$,其中$(x_T,y_T)$为位置,$v_T$ 为逃逸目标速度,φ_T 为航向角且服从均匀分布;无人机的控制输入 $u_i\in[-\omega_0,\omega_0]$($\omega_0=0.5$ rad/s 为无人机的最大角速度),其决定了无人机的航向调整 $\dot{\varphi}_i=u_i$[10-12]。

1.2.2 无人机运动学方程

围捕无人机的运动学方程为:

$$\begin{cases}\dot{x}_i = v_i\cos(\varphi_i)\\ \dot{y}_i = v_i\sin(\varphi_i)\\ \dot{\varphi}_i = u_i\end{cases}, \tag{1}$$

式中:u_i 为围捕无人机的角速度的大小,v_i 为围捕无人机的速度大小,是一个固定的值,即在飞行过程中不改变。此外,文献[13]基于阿波罗奥尼斯圆(Apollonius Circle)和几何规律研究了多追捕者-单逃跑者追逃问题能够成功实现捕获目标的约束条件为速度比 $\lambda\geqslant\sin(\pi/N)$,因此本文同文献[5]设置围捕无人机速度 v_i 与逃逸目标速度 v_T 的速度比为 $\lambda\in[\sin(\pi/N),1)$[5,13]。

1.2.3 任务目标函数

任务目标为最小化包围完成的时间 t,为实现任务目标需要同时满足以下 2 个条件:

① 围捕无人机构成的包围圈的紧凑度:$\max_i\|(x_i,y_i)-(x_T,y_T)\|\leqslant R$;

② 特定角色需要满足的协同约束:探测角色有效捕捉到目标的距离为 $d_i\leqslant 3R$;执行角色的有效执行距离为 $d_i\leqslant 1.5R$。

2 DRA-MADDPG 算法设计

2.1 多智能体强化学习算法

多智能体强化学习是一种研究多个智能体在共享环境中通过交互及协作(或竞争)来优化各自策略,从而实现各自或全局目标的强化学习[13]。多智能体强化学习的数学框架是基于马尔可夫博弈(Markov Game),是基于马尔可夫决策过程(Markov Decision Process,MDP)的扩展。一般在包含 N 个智能体的系统中将马尔可夫博弈定义为元组形式[14]:

$$\langle S,A_1,A_2,\cdots,A_N,r_1,r_2,\cdots,r_N,P,\gamma\rangle, \tag{2}$$

式中:S 表示环境的状态空间,A_i 表示第 i 个智能体的动作空间,$r_i:S\times A_1\times A_2\times\cdots\times A_N\to\mathbb{R}$表示第 i 个智能体的即时奖励,此奖励取决于所有智能体的动作;$P:S\times A_1\times A_2\times\cdots\times A_N\to S$ 表示状态转移的概率,下一状态由当前状态和所有智能体的动作决定;$\gamma\in[0,1)$表示折扣因子,即未来奖励的权重。

在马尔可夫博弈中,每个智能体所能获得的奖励即个体 Q 值函数 $Q_i^{\pi}(s,a_1,a_2,\cdots,a_N)$是在联合策略 $\pi(a|s)=\prod_{i\in N}\pi_i(a_i|s)$下,所有智能体在状态 s 采取联合动作$(a_1,a_2,\cdots,a_N)$后,智能体 i 所能够获得的累积折扣奖励期望。个体 Q 值函数的贝尔曼更新方程如下:

$$Q_i^{\pi}(s,a) = r_i(s,a)+\gamma\sum_{s'\in S}P(s'|s,a)\times \sum_{a'}\left(\prod_{j=1}^{N}\pi_j(a'_j|s')\right)Q_i^{\pi}(s',a'), \tag{3}$$

式中:$a=(a_1,a_2,\cdots,a_N)$表示所有智能体的联合动作,$r_i(s,a)$为智能体 i 在状态 s 和联合动作 a 下获得的即时奖励,$P(s'|s,a)$表示在联合动作 a 的作用下从状态 s 转移到状态 s'的概率,a'为下一时刻的联合动作。

通常多智能体强化学习需要完成的任务类型,可分为完全合作、完全竞争和混合类型[15]。而在指挥决策场景中则通常表现为完全合作模式,如本文的无人机围捕任务,即所有智能体都需要围绕统一目标,通过角色分工与信息交互从而实现全局最优。

2.2 MADDPG 算法

MADDPG 算法是一种基于深度确定性策略梯

度(Deep Deterministic Policy Gradient,DDPG)算法扩展得到的多智能体强化学习算法[16]。MADDPG算法采用的是Actor-Critic框架,对多智能体主要采用集中式训练,分布式执行,如图2所示[16]。

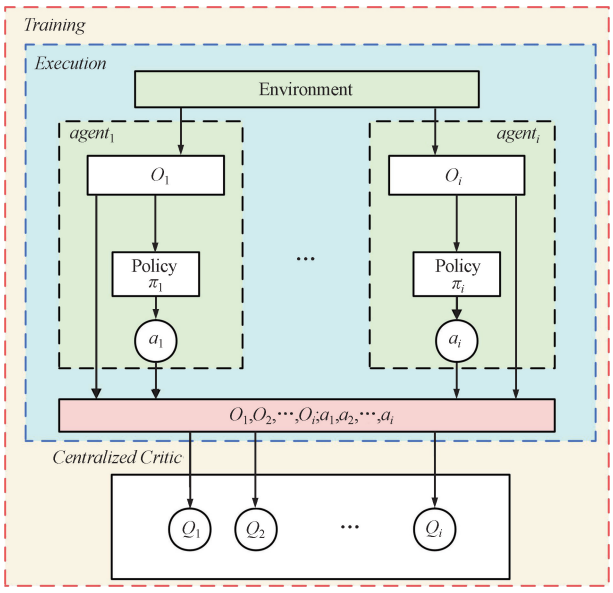

图2 MADDPG算法框架

Fig. 2 Framework of MADDPG algorithm

Actor网络使用局部观测信息,而Critic网络则整合全局信息进行学习[17]。Actor网络更新策略为μ_θ函数,$agent_i$通过确定性行为策略μ_θ进行行为选择:

$$a_i = \mu_\theta(O_i) + N_{\text{noise}}, \tag{4}$$

式中:O_i表示$agent_i$的观测值,包括自己状态和其他智能体的状态。Critic网络的优化通过最小化Critic网络的损失函数$L(\theta_i)$来进行价值评估[15]:

$$L(\theta_i)=\mathbb{E}[(y-Q_i^{\mu'}(O',a_1',a_2',\cdots,a_n'))^2], \tag{5}$$

$$y = r_i + \gamma Q_i^{\mu}(O,a_1,a_2,\cdots,a_n)\mid_{a_j'=\mu_j'(O_j)}, \tag{6}$$

式中:每次$agent_i$根据自己的观测值和其他所有智能体的行为计算目标函数y的值。Critic扩展为可以利用其他智能体的策略进行学习,这点的进一步改进就是每个智能体对其他智能体的策略进行一个函数逼近[18]。

2.3 DRA-MADDPG算法

DRA-MADDPG算法是在MADDPG算法的基础上引入DRA模块进行改进的算法,目的是使Critic和Actor网络能够根据智能体的角色及任务阶段,调整对不同智能体的信息的关注权重。改进后的算法框架如图3所示,DRA-MADDPG算法与MADDPG算法的关键差异在于:

① 在Critic网络的输入中增加了"角色-阶段注意力分布c_i",从而实现对关键角色状态和动作的加权。

② 在Actor网络的输入中引入了自身角色及阶段信息,以确保生成的动作能够符合角色职责。

③ 在策略梯度中增加与角色注意力相关的优化项,进一步强化角色协同。

DRA模块的创新点在于"角色-阶段2个维度的动态调整",文献[19]虽然在MADDPG的Acror-Critic网络中引入了自注意力,但其仅是基于智能体之间的距离计算来固定权重。

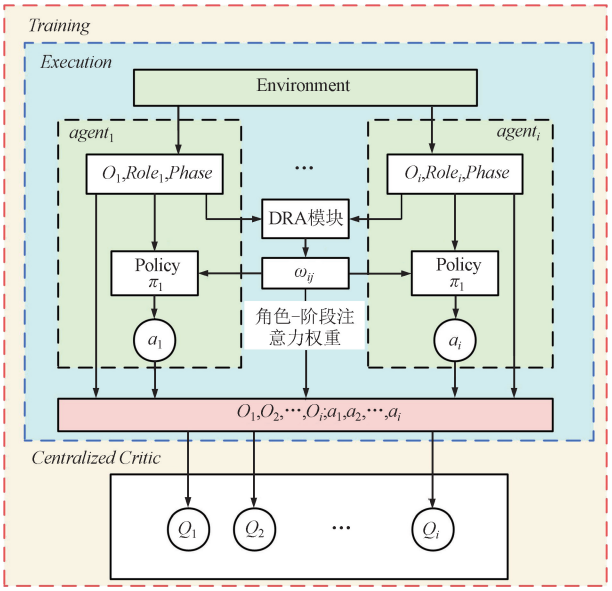

图3 DRA-MADDPG算法框架

Fig. 3 Framework of DRA-MADDPG algorithm

2.3.1 角色协同矩阵与阶段系数设计

根据第2.1.2所定义的角色集合$\xi=\{r_1,r_2,r_3\}$及职责分工形成角色协同矩阵$\boldsymbol{C}\in\mathbb{R}^{3\times3}$,用以量化不同角色之间的协同强度,本文采用先验知识及逻辑初始化协同矩阵中各个元素值。

$$\boldsymbol{C}=\begin{bmatrix}(r_1,r_1) & (r_1,r_2) & (r_1,r_3)\\ (r_2,r_1) & (r_2,r_2) & (r_2,r_3)\\ (r_3,r_1) & (r_3,r_2) & (r_3,r_3)\end{bmatrix}=\begin{bmatrix}0 & 0.6 & 0.8\\ 0.6 & 0 & 1.0\\ 0.8 & 1.0 & 0\end{bmatrix}。 \tag{7}$$

矩阵中$\boldsymbol{C}[r_i,r_j]$越大,表明角色r_i与角色r_j之间的协同强度越大。

依据2.1.2的任务阶段划分$\phi=\{\phi_1,\phi_2,\phi_3\}$及任务阶段的核心目标与角色职责的匹配关系,量化阶段调整系数$\alpha(\phi_t,r_j)$,让DRA机制更贴合不同阶段的任务重点,其根本作用是根据当前阶段放大关键角色的权重。

$$\alpha(\phi_t,r_j)=\begin{cases}1.2,\phi_t\text{ 为探索阶段且 }r_j=r_1\text{(探测)}\\1.2,\phi_t\text{ 为执行阶段且 }r_j=r_4\text{(调度)}\\1.2,\phi_t\text{ 为包围阶段且 }r_j=r_2\text{(执行)}\\1.0,\text{其他}\end{cases}。\tag{8}$$

2.3.2 计算 DRA 权重

现给出基于 DRA 机制的注意力分布的计算过程：

① 通过角色协同矩阵 $\boldsymbol{C}$ 及阶段系数 $\alpha(\phi_t,r_j)$ 计算出围捕无人机之间的角色-阶段相关性系数 $Sim_{i,j}=\boldsymbol{C}[r_i,r_j]\cdot\alpha(\phi_t,r_j)$ 是角色固有的价值与阶段的临时需求的乘积，这与强化学习的“状态-动作-奖励”的逻辑一致，其中 $\boldsymbol{C}[r_i,r_j]$ 表示角色 r_i 对 r_j 的长期依赖，$\alpha(\phi_t,r_j)$ 表示角色 r_j 在当前阶段 ϕ_t 的短期重要性。

② 引入 softmax 函数对角色-阶段相关性系数进行归一化处理，得到注意力权重系数 $\omega_{i,j}$。其表示智能体 i 对智能体 j 的注意力权重，权重越高，表明 j 的信息对 i 的决策越重要。

$$\omega_{i,j}=\text{softmax}(Sim_{i,j})=\frac{e^{Sim_{i,j}}}{\sum_{k=1}^{N}e^{Sim_{i,k}}}。\tag{9}$$

③ 根据注意力权重系数对所有围捕无人机的状态/动作进行加权求和，计算注意力分布 c_i：

$$c_i=\sum_{j=1}^{N}\omega_{i,j}\cdot p_j,\tag{10}$$

式中：$p_j=[x_j,y_j,v_j,\varphi_j,a_j]$ 为智能体 j 的状态及动作信息。

2.4 DRA-MADDPG 网络结构优化

2.4.1 Critic 网络优化

DRA-MADDPG 算法为了实现对角色相关信息的侧重，其 Critic 网络在 MADDPG 的基础上引入了动态角色的注意力分布 c_i，其 Critic 网络的输入为 (s,a,c_i)，输出加权后的动作价值为 $Q_i^{\text{DRA}}(s,a,c_i)$：

$$Q_i^{\text{DRA}}(s,a,c_i)=Q_{\theta_i}(s,a,c_i)。\tag{11}$$

因为角色在不同阶段的价值不同，角色与角色之间的协同比重不同，所以这并非是单一智能体行动的结果，由此可知 Critic 网络拟合的是全局值函数，故 s 为全局状态，a 为联合动作，c_i 为注意力分布。策略通过 MADDPG 双网络进行更新，Critic 网络的损失函数 $L_{\text{DRA}}(\theta_i)$ 以及目标值函数 y_i^{DRA} 设置如下：

$$L_{\text{DRA}}(\theta_i)=\mathbb{E}[(y_i^{\text{DRA}}-Q_i^{\text{DRA}}(s,a,c_i;\theta_i))^2],\tag{12}$$

$$y_i^{\text{DRA}}=r_i+\gamma Q_i^{\text{DRA}'}(s',a',c'_i;\theta'_i)\mid a'_j=\pi'_j(o'_j,r_j,\phi'_t),\tag{13}$$

式中：θ_i 表示网络参数(权重和偏置)。如此设置是因为目标 Actor 在生成动作时需要考虑角色 r_j 和阶段 ϕ'_t，才能够确保下一个状态的价值评估仍然受到角色的引导。

2.4.2 Actor 网络优化

DRA-MADDPG 算法的 Actor 网络输入增加了角色 r_i、阶段 ϕ_t 及注意力分布 c_i，输出动作 $a_i=\pi_i(o_i,r_i,\phi_t,c_i;\theta_i)$。其策略梯度在 MADDPG 的基础上，增加与角色注意力相关的梯度项：

$$\nabla_{\phi_i}J_{\text{DRA}}(\phi_i)=\underbrace{\mathbb{E}[\nabla_{\phi_i}\pi_i(a_i\mid o_i)\cdot\nabla_{a_i}Q_i^{\pi}(s,a;\theta_i)]}_{\text{MADDPG}}+\underbrace{\mathbb{E}[\nabla_{\phi_i}\pi_i(a_i\mid r_i,\phi_t,c_i)\cdot\nabla_{a_i}Q_i^{\text{DRA}}(s,a,c_i;\theta_i)]}_{\text{DRA}},\tag{14}$$

式中：MADDPG 的基础部分用于保证单智能体的基本策略学习，DRA 部分则是为了强化多个智能体之间为满足角色和阶段需求而进行协同优化的能力。$\nabla_{\phi_i}\pi_i(a_i\mid r_i,\phi_t,c_i)$ 是 Actor 网络参数在 DRA 条件下对动作的梯度，表示参数如何变化能使得动作更加符合当前角色的职责以及阶段需求；$\nabla_{a_i}Q_i^{\text{DRA}}(s,a,c_i;\theta_i)$ 是 Critic 网络在引入注意力分布 c_i 后的动作梯度，表示当前动作在协同场景下的价值。

DRA-MADDPG 算法的伪代码如算法 1 所示。

算法 1 DRA-MADDPG 算法伪代码

初始化环境参数、智能体角色集合 ξ、任务阶段集合 ϕ；
初始化 Actor 网络 π_i、Critic 网络 Q，及目标网络 π_i'、Q'；
创建经验回放池 D；
for *episode* = 1 to *Max-Episode* do
　初始化环境状态 S，随机分配智能体角色 $r_i\in\xi$，获取初始阶段 $\phi_i\in\phi$；
　for t = 1 to *Max-Step* do
　基于当前状态 S、角色 r_i、阶段 ϕ，通过 DRA 模块计算注意力权重 ω；
　各智能体 i：$a_i=\pi_i(s_i,r_i,\phi,\omega)$ $(s_i\in S)$，组成动作集 A；
　执行 A 与环境交互得到新状态 S'、即时奖励 r_R，终止信号 *done*；
　将$\langle S,r_i,\phi,\omega,A,r_R,S',done\rangle$存入 D；
　$S\leftarrow S'$；
　#网络更新
　If $|D|\geq BatchSize$：
　从 D 采样批量数据$\{\langle S,r_i,\phi,\omega,A,r_R,S',done\rangle\}$；
　for 智能体 i = 1 to N do
　基于 DRA 机制计算 $Q_i^{\text{DRA}}(s,a,c_i)$；
　设置目标Q：$y_i^{\text{DRA}}=r_i+\gamma Q_i^{\text{DRA}'}(s',a',c'_i;\theta'_i)\mid a'_j=\pi'_j(o'_j,r_j,\phi'_t)$；
　基于损失函数 $L(\theta)$ 更新 Critic；
　基于策略梯度 ∇J 更新 Actor；

反向传播更新 π_i 的参数；
end for
$\pi_i' \leftarrow \tau\pi_i + (1-\tau)\pi_i'$；
$Q' \leftarrow \tau Q + (1-\tau)Q'$；
end for
end for

3 仿真实验

为验证 DRA-MADDPG 算法的有效性，选取 4 架不同角色且初始位置随机的围捕无人机对单一目标进行围捕的仿真实验，并对比 MADDPG 算法进行训练，测试相关性能指标。仿真环境是基于 Python 语言编写，调试软件为 PyCharm 2024.3.1.1，深度学习环境采用 PyTorch 2.8.0+cu126，计算机配置为 CPU 11th Gen Intel(R) Core(TM) i5-11400H，GPU NVIDIA GeForce RTX 3050，内存 16 GB，CUDA 12.7。

3.1 实验设置

(1)环境与角色配置

围捕无人机及逃逸目标的初始位置随机部署在二维矩形区域：$window=50$ km。围捕方部署了 4 架无人机 $N=4$，分为 1 架探测角色、2 架执行角色、1 架调度角色，且飞行速度固定。围捕无人机速度为 $v_{r_i}(i=1,2,3)$，探测无人机 r_1 角色的有效探测半径为 $3R$；执行无人机 r_2 角色的有效执行距离为 $1.5R$；目标无人机速度 v_e 为固定值。速度比 $\lambda=\dfrac{v_{r_i}}{v_e}$ 均满足 $\lambda\in[\sin(\pi/N),1)$，且 $v_{r_2}>v_{r_1}>v_{r_3}$，捕获条件设置为形成包围圈且持续时间超过 10 s。实验设定的训练参数如表 1 所示[6]。

表 1 训练超参数设置表

Tab. 1 Training hyper parameters setting

训练参数	值
折扣因子 λ	0.9
惯性更新率 τ	0.01
经验池大小 D	30 000
批样本数 *BatchSize*	64
仿真时间步长 ΔT	0.1
Critic 网络学习率 α_Q	0.002
Actor 网络学习率 α_U	0.001

续表

训练参数	值
回合数 *Max-Episode*	2 000
单回合最大时间步长 *Max-Step*	1 500

其中，惯性更新率 τ 的选取遵循 DDPG 类算法的通用设置(τ 取 0.001～0.01)以确保目标网络稳定性；经验池大小 $D=30\ 000$ 相较于 GAED-MADDPG[1]($D=25\ 000$)更大是由于 DRA-MADDPG 需要存储“角色+阶段”的额外信息，需要更大的样本池覆盖所有角色-阶段组合；Critic/Actor 网络的学习率设置参考了文献[17]中 MADDPG 的经典参数，使 Actor 学习率略低，确保策略更新稳定；Critic 学习率略高，加速价值评估的收敛。

(2)奖励函数设计

全局奖励 r_g：

$$r_g=\underbrace{\omega_t\cdot e^{(-\eta\cdot\max d_i)}}_{r_t}+\underbrace{\omega_c\cdot\left[e^{\left(-\beta\left|\frac{1}{N}\sum_{i=1}^{N}d_i-R\right|\right)}-\gamma\cdot\delta_{\text{gap}}\right]}_{r_c},\quad(15)$$

公式分为两部分，分别是锁定目标奖励 r_t 以及形成包围圈奖励 r_c，其中 d_i 指围捕无人机与 c 目标之间的距离($i=1,2,\cdots,N$)，$\eta=0.005$，$\beta=0.01$，且加权系数 ω_t 和 ω_c 满足 $\omega_t+\omega_c=1$。另外示性函数 δ_{gap} 形式如下：

$$\delta_{\text{gap}}=\begin{cases}1,\text{包围圈存在明显间隙}\\0,\text{包围圈紧凑，符合理想状态}\end{cases}。\quad(16)$$

局部奖励 r_l：

$$r_l=\underbrace{\alpha_1\cdot r_{\text{Role}}}_{\text{角色奖励}}+\underbrace{\alpha_2\cdot\left[-10\sum_j e^{\left(-\frac{(d_{ij})^2}{(2d_s)^2}\right)}\right]}_{\text{避碰奖励}}+\underbrace{\alpha_3\cdot(-d_i)}_{\text{阶段引导奖励}},\quad(17)$$

公式分为三部分，分别为角色奖励、避碰奖励以及阶段引导奖励，其中加权系数 α_1、α_2、α_3 满足 $\alpha_1+\alpha_2+\alpha_3=1$，$d_s$ 为围捕无人机之间的安全距离。另外角色奖励 r_{Role} 设置如下：

$$\text{探测角色：}r_{\text{探测}}=\begin{cases}e^{\left(\frac{3R}{d_i}\right)^2}, & d_i<3R\\ -e^{\left(\frac{d_i}{3R}\right)^2}, & d_i\geqslant 3R\end{cases},\quad(18)$$

$$\text{执行角色：}r_{\text{执行}}=e^{(-\mu\cdot[d_i(t)-d_i(t-1)])},\quad R<d_i(t)<d_i(t-1),\quad(19)$$

$$\text{调度角色：}r_{\text{调度}}=\begin{cases}0.8\cdot r_{\text{探测}}+0.2\cdot r_{\text{执行}}, \overline{d_i}\geqslant 2.5R\\ 0.5\cdot r_{\text{探测}}+0.5\cdot r_{\text{执行}}, 2R\leqslant\overline{d_i}<2.5R\\ 0.2\cdot r_{\text{探测}}+0.8\cdot r_{\text{执行}}, \overline{d_i}<2R\end{cases},\tag{20}$$

式中：$\overline{d_i}=\dfrac{1}{N}\sum_{i=1}^{N}d_i(i=1,2,3,\cdots,N)$。

3.2 训练过程与稳定性分析

MADDPG 算法及 DRA-MADDPG 算法的训练过程如图 4、图 5 所示，可以直观地看到训练效果。

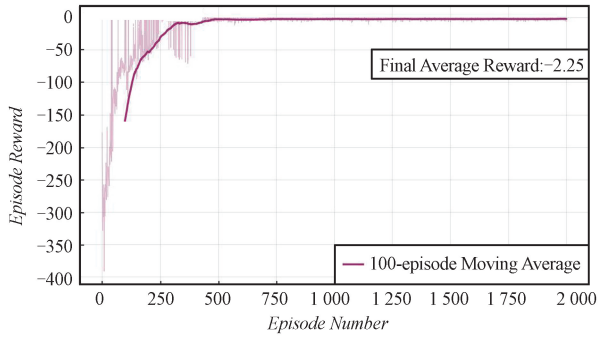

图 4 MADDPG 算法训练过程

Fig. 4 Trainingprocess of MADDPG algorithm

由图 4 可知，MADDPG 在约 500 回合近似收敛并趋于稳定，整体训练曲线比较流畅；由图 5 可知，DRA-MADDPG 算法在 220~375 回合虽然因为探索噪声及目标网络更新延迟而出现短暂震荡，但并未导致长期性能骤降或无法收敛的情况，其震荡后能快速恢复并在 520 回合近似收敛，表明算法对训练过程中的固有扰动具有良好的适应与恢复能力，而非仅在理想无波动训练环境下有效。

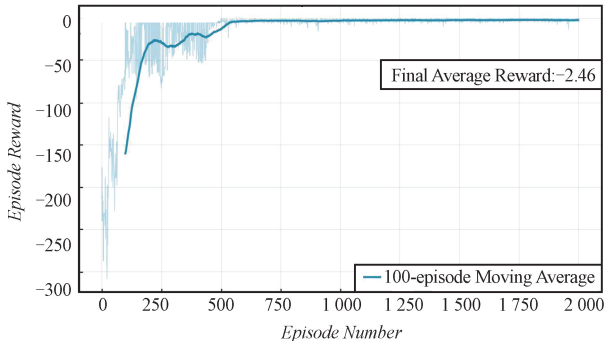

图 5 DRA-MADDPG 算法训练过程

Fig. 5 Training process of DRA-MADDPG algorithm

经过图 4 和图 5 的直观对比，可以看出二者在长期稳态回报上接近，但 DRA-MADDPG 更早进入稳态区间：在 *Reward* = −25 时，DRA-MADDPG 仅需约 200 回合，而 MADDPG 需要约 250 回合，表明 DRA-MADDPG 算法的学习速度更快。DRA-MADDPG 的训练累计回报 AUC 高于 MADDPG，表明其拥有出更出色的任务完成效率。AUC 对比数据如表 2 所示。

表 2 AUC 对比数据

Tab. 2 AUC comparison data

算法	原始 *AUC*	位移后 *AUC*	归一化 *AUC*[0,1]	区间平均回报(0~2 000)
DRA-MADDPG	−26 263.3	773 736.7	0.967 171	−13.131 7
MADDPG	−44 681.9	755 318.1	0.944 148	−22.341

3.3 动态协同围捕任务仿真验证

图 6 为在 DRA-MADDPG 算法达到稳态性能后抽取了一个样本的动态协同围捕轨迹图，可以看到围捕无人机并没有达到理想包围圈(即围捕无人机平均分布在包围圈上)，这可能是由于实验设计的围捕无人机数量偏少，且围捕无人机的初始位置设置的是随机分布(为了模拟无人机空中警戒巡逻的位置随机性)，从而导致包围圈形成的不够理想，但可以看出训练基本上达到了预期目标，围捕无人机能够对动态目标进行围捕。本实验与现有相关研究(如文献[2,5])等的不同之处在于，本实验所设置的逃逸目标并非单纯的匀速直线或按照既定策略进行逃逸，而是将逃逸目标设计为不固定运动方向，不固定运动速度的复杂情况，更加贴合实战。

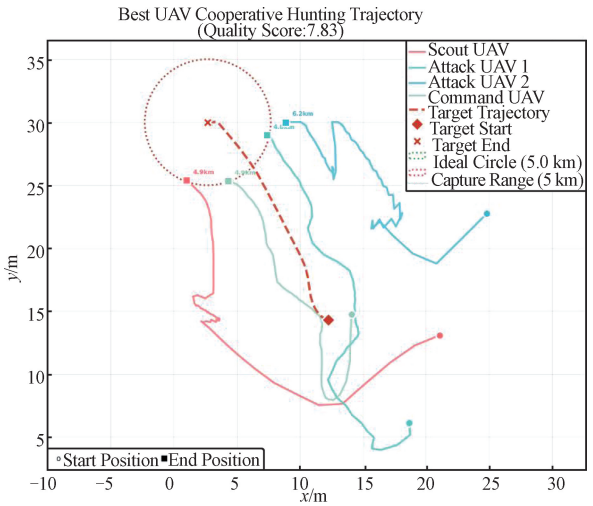

图 6 动态协同围捕轨迹

Fig. 6 Dynamic cooperative pursuit trajectory

图 7 为 MADDPG 与 DRA-MADDPG 算法的任务完成度对比,能够直观反映每一回合的任务完成度,可以看出 DRA-MADDPG 算法在前期表现更优,但在中期出现震荡,因此中期的完成度低于 MADDPG,最终 2 种算法都达到近似稳态性能,都能够在实验条件下成功完成无人机围捕任务。

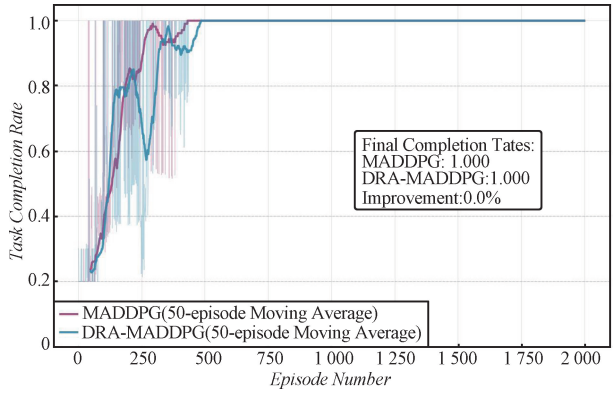

图 7　任务完成度对比

Fig. 7　Comparison of task completion rate

图 8 是当 2 种算法收敛后,随机抽样一次任务成功的样本在任务各个阶段耗时对比。可以看出,DRA-MADDPG 算法在无人机围捕任务耗时相较于 MADDPG 算法,探索阶段降低了 21.5%,包围阶段降低了 22.1%,围捕阶段降低了 15.8%,总时间消耗降低了 19.3%,未出现因阶段切换导致的协同中断或效率骤降。表明其在阶段目标变化的场景下,能够快速适配协同策略,任务执行鲁棒性更优且在无人机动态协同围捕行动中使用 DRA-MADDPG 算法能够更快地实现任务目标,这在指挥决策任务场景中至关重要。

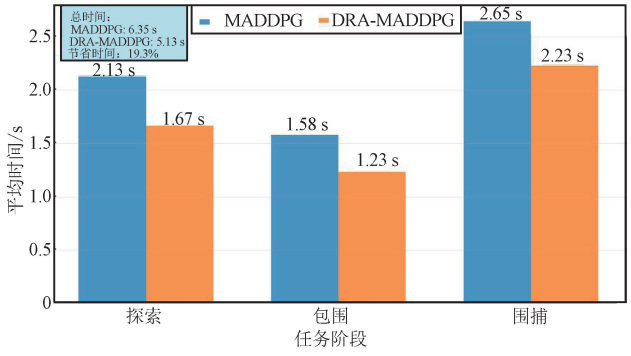

图 8　任务完成耗时对比

Fig. 8　Comparison of task completion time

4　结论

通过无人机动态协同围捕仿真实验验证了 DRA-MADDPG 算法的相关性能,经过对实验结果的综合分析,本文认为在 MADDPG 算法的基础上引入 DRA 机制能够更好地贴合指挥决策过程,可以解决不同角色多智能体的协同、阶段目标的灵活转换以及 MADDPG 算法易信息过载的问题。具体结论如下:

① DRA-MADDPG 算法较 MADDPG 算法在训练前期收敛速度更快且任务完成度更高,能够更早地进入稳态区间。

② DRA-MADDPG 算法较 MADDPG 算法训练累计回报 AUC 提升了 2.4%,略高于 MADDPG,表明其拥有出更出色的综合性能。

③ DRA-MADDPG 算法在无人机围捕任务的耗时相较于 MADDPG 算法降低了 19.3%,表明在无人机围捕行动中使用 DRA-MADDPG 算法能够更快地实现任务目标。

在下一步工作中,可以考虑进一步优化 DRA-MADDPG 算法,提高其复杂环境的适用度,将其拓展至三维领域或其他基于角色-阶段划分智能体的指挥活动中,用以提升指挥效能及任务完成效果。

参考文献

[1] 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.

[2] 刘峰,魏瑞轩,丁超,等. 面向多机协同的 Att-MADDPG 围捕控制方法设计[J]. 空军工程大学学报(自然科学版),2021,22(3):9-14.

[3] FOERSTER J N, FARQUHAR G, AFOURAS T, et al. Counterfactual Multi-agent Policy Gradients[C]// Proceedings of the AAAI Conference on Artificial Intelligence. New Orleans: AAAI Press,2018:2974-2982.

[4] 贾思雨,毕凌滔,曹扬,等. 基于改进 MADDPG 的多机器人路径规划方法研究[J]. 计算机仿真,2024,41(8): 458-465.

[5] 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.

[6] 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.

[7] 畅鑫,李艳斌,刘东辉. 基于分层强化学习的多智能体博弈策略生成方法[J]. 无线电工程,2024,54(6):1361-1367.

[8] 张建东,王鼎涵,杨啟明,等. 基于分层强化学习的无人机空战多维决策[J]. 兵工学报,2023,44(6):1547-1563.

[9] 刘东辉,郑赢营,畅鑫,等. 基于静态博弈和遗传算法的多智能体博弈策略生成方法[J]. 无线电工程,2024,54(6):1355-1360.

[10] 李波,越凯强,甘志刚,等. 基于 MADDPG 的多无人机协同任务决策[J]. 宇航学报,2021,42(6):757-765.

[11] 孙懿豪,闫超,相晓嘉,等. 基于分层强化学习的多无人机协同围捕方法[J]. 控制理论与应用,2025,42(1):96-108.

[12] 轩书哲,柯良军. 基于多智能体强化学习的无人机集群攻防对抗策略研究[J]. 无线电工程,2021,51(5):360-366.

[13] 周浦城,洪炳镕,王月海. 动态环境下多机器人合作追捕研究[J]. 机器人,2005(4):289-295.

[14] 李茹杨,彭慧民,李仁刚,等. 强化学习算法与应用综述[J]. 计算机系统应用,2020,29(12):13-25.

[15] 高阳,陈世福,陆鑫. 强化学习研究综述[J]. 自动化学报,2004,30(1):86-100.

[16] LOWE R, WU Y, TAMAR A, et al. Multi-agent Actor-Critic for Mixed Cooperative-Competitive Environments[C]// Advances in Neural Information Processing Systems. Long Beach: [s. n.],2017:6379-6390.

[17] 杜威,丁世飞. 多智能体强化学习综述[J]. 计算机科学,2019,46(8):1-8.

[18] 梁宸. 基于强化学习的多智能体协作策略研究[D]. 沈阳:沈阳理工大学,2020.

[19] 殷宇维,王凡,丁录顺,等. 基于 MADDPG 的多无人战车协同突防决策方法研究[J]. 指挥控制与仿真,2025,47(3):40-49.

作者简介

苑司宇 男,(2001—)。主要研究方向:5G 及 AI 技术应用。

康国钦 男,(1981—),博士,副教授。主要研究方向:网电安全与电磁频谱管理。

郑学强 男,(1981—),博士,副教授。主要研究方向:短波通信、认知无线网络、智能通信。

周强强 男,(1987—)。主要研究方向:电磁频谱管理。