文章编号:1671-4598(2024)02-0156-06 DOI:10.16526/j.cnki.11-4762/tp.2024.02.023 中图分类号:V211 文献标识码:A

基于 EPF-MADDPG 算法的多导弹机动策略研究

聂文川，樊志强
（中国电科智能科技研究院，北京 100083）

摘要：随着人工智能研究的进一步加深以及在俄乌战场上相关技术的大放异彩，其在军事领域扮演的角色越来越重要；针对日益复杂的战场环境，当前的导弹突防领域存在着信息维度高、指挥反应缓慢、突防机动战术不够灵活等问题；提出了一种基于多智能体深度确定性策略梯度（MADDPG）的训练方法，用以快速制定导弹攻击机动方案，协助军事指挥官进行战场决策；同时改进算法的经验回放策略，添加经验池筛选机制缩短训练的时长，达到现实场景中的快速反应需求；通过设置多目标快速拦截策略，仿真验证了所设计的方法能够突防的机动策略优势，通过协作智能地对目标进行突防打击，并通过比较，验证了该方法相较其他算法可以提升 8%的收敛速度以及 10%的成功率。

关键词：多智能体；MADDPG；强化学习；协同机动突防；导弹机动

Research on Multi Missile Maneuvering Strategy Based on MADDPG Algorithm

NIE Wenchuan，FAN Zhiqiang
（Intelligent Technology Research Institute，China Electronics Technology Group Corporation，Beijing 100083，China）

Abstract：In recent years，with the further deepening of artificial intelligence research and the shine of related technologies on the battlefield of Russia and Ukraine，it has become more and more important in the military field. In view of increasingly complex battlefield environment，current missile penetration field has problems such as high information dimension，slow command response，and inflexible penetration maneuver tactics. A training method based on multi-agent deep deterministic strategy gradient（MADDPG）is proposed to quickly generate missile attack maneuver schemes to assist commanders in making battlefield decisions. At the same time，the experience playback strategy of the algorithm is improved，and the experience pool filtering mechanism is added to shorten the training time and meet the rapid response requirements in real scenarios. By setting the multi-target rapid interception strategy，the simulation verifies that the maneuvering strategy advantages of the designed method can penetrate defense，intelligently and collaboratively strike the target. Compared with other algorithms，the method can improve the convergence speed of 8% and success rate of 10%.

Keywords：multi-agent；MADDPG；reinforcement learning；coordinated mobile penetration；missile maneuvering

0 引言

随着我军不断地信息化改革，研究人员探索了众多的人工智能技术[1-5]。强化学习技术近年来逐渐火热，强化学习是可以自学习的，它适用于决策，已经应用于许多领域，如流量控制、无人机控制、网络构建等[6-9]。博弈是指一个理性的人或团队从选择行为或策略，到最终获取相应的利益。强化学习算法通过博弈对抗中产生的回报来优化策略选择。强化学习的最主流应用仍在游戏领域中，近年来，强化学习征服了象棋、围棋等完全信息游戏，以及扑克等不完全信息游戏，在电子游戏竞赛中的战争迷雾和复杂状态空间以及动作空间的游戏，如 Dota、星际争霸等[10-12]，人类玩家也逐渐被强化学习算法超越，而这就是算法有效性最强有力的体现。

本文基于现实海上反舰场景中导弹机动的强化学习进行了研究，将他们迁移到仿真的场景中，尤其是导弹集群反舰任务。针对异构多智能体博弈对抗的情况，本文将 MADDPG（multi-agent deep deterministic strategy gradient）算法应用到多智能体弹群反舰任务的场景中，通过分析在巨大状态空间和动作空间的收敛速度，聚焦真实报酬稀疏的问题。同时，通过设计仿真实验来验证算法的有效性。

1 场景分析与数学模型

复杂对抗场景一直是强化学习的热点和难点之一。随

收稿日期:2023-09-08； **修回日期**:2023-11-15。

作者简介:聂文川(1996-),男,硕士研究生。

引用格式:聂文川,樊志强. 基于 EPF-MADDPG 算法的多导弹机动策略研究[J]. 计算机测量与控制,2024,32(2):156-161,212.

着深度强化学习的发展，该算法应用到了各种场景。然而目前的主流应用是在围棋等游戏领域[13-15]，一个重要的原因就是游戏场景具有现成的游戏环境和自洽的规则以及奖惩机制，便于强化学习的应用。但在自动驾驶等真实场景中，由于仿真环境的仿真完成度和奖惩机制的不确定性，无法实现强化学习算法。因此在仿真系统中，仍需大量的工作来促进强化学习算法的进步，同时在类似的仿真系统中，强化学习算法本身也有很大的发展潜力。在历史上，计算机的发展首先运用于军事领域，用来协助人类计算以及密码破译，在现代，人工智能依旧可以运用在军事领域。基于强化学习的博弈对抗推理是维持军队战斗力的重要手段之一。近年来，军事象棋推演成为人们普遍关注的热点，人工智能在推理和分析的过程中起到重要的作用。本文将强化学习应用于多智能体博弈对抗仿真系统中，选取了红蓝两面异构的博弈对抗场景，即近海反舰作战场景，红色攻击和蓝色防御，实现仿真作战。本文将多智能体强化学习算法应用于异构多智能体系统，增强了智能体之间的协作性，提高了算法的能力。

1.1　弹群协作的特点

导弹集群协作智能化具有以下 4 个重要特点：

1）去中心化：任何一枚导弹的消失或者功能丧失，弹群的目标依然可以有序实现[16]。同时每一颗导弹都可以协作其他导弹实现战术目标。

2）自主性：战场态势瞬息万变，依赖指挥官根据战场态势进行决断势必会浪费宝贵的作战时间，甚至可能错过稍纵即逝的机会。因此为了节省人为决策消耗的时间，飞行期间导弹采取的一切机动操作均可进行自主判断并及时决策，且弹群内的所有导弹只控制自身飞行，但可以观察其他导弹位置，不对其他导弹产生影响。

3）高动态：导弹需要根据战场态势变化做出快速响应。传统预先规划的形式已经无法满足现在瞬息万变的战场环境，而导弹的作战时间非常短暂，因此要求弹群在收集到战场态势信息后迅速做出决策。

4）自治化：所有的导弹形成一个稳定的集群，并且各自承担相应的功能，当某一导弹丧失功能造成集群结构的缺失后，其他导弹应及时调整并重新构成稳定的集群结构[17-18]。

综合来看，目前多弹头集群协同突防技术的研究仍处于初级阶段，因为该技术要求各个弹头具有高度自主性，面临复杂任务可以快速响应，因此对于弹载计算机的要求较高[19]。

1.2　导弹运动学模型

导弹的运动学方程为：

$$\begin{cases} x_i = v_i \cos\psi_i \\ y_i = v_i \sin\psi_i \\ \psi_i = \omega_i \end{cases} \tag{1}$$

式（1）中，$i=p$，e；ω_i 为拦截导弹或突防导弹的角速度大小；v_i 为拦截导弹或突防导弹的速度，其为一个固定值，即导弹在飞行过程中的速度不改变。

导弹的运动控制变量约束为：

$$\begin{cases} -\omega_{p\max} \leqslant \omega_p \leqslant \omega_{p\max} \\ -\omega_{e\max} \leqslant \omega_e \leqslant \omega_{e\max} \end{cases} \tag{2}$$

式（2）中，$\omega_{p\max}$，$\omega_{e\max}$分别为拦截导弹和突防导弹的最大角速度，其计算方程为：

$$\begin{cases} \omega_{i\max} T = \psi_i \\ n_{i\max} g = \dfrac{v^2}{r_{i\min}} \\ r_{is} \sin\Delta\psi_i \approx \dfrac{v_i \Delta T}{2} \end{cases} \tag{3}$$

式（3）中，$i=p$，e；ΔT 为方针的时间步长；r_i 为导弹的机动半径；$r_{i\min}$为导弹的最小机动半径；$\Delta\psi_i$ 为 ΔT 时间内的航向最大转弯角；$n_{i\max}$为导弹的最大侧向过载。因此，由式（4）可得最大角速度的确定公式为：

$$\omega_{i\max} = \frac{\arcsin\left(\dfrac{\Delta T n_{i\max} g}{2v_i}\right)}{\Delta T} \tag{4}$$

拦截捕获条件为式（5），在拦截半径范围内，即我方导弹进入敌方拦截导弹的作用范围，便会被拦截捕获。

$$\| r_{pe}^n \| \leqslant r_c, \exists n \in (1, N) \tag{5}$$

式（5）中，r_c 为拦截导弹的爆炸半径（捕获半径），$\| r_{pe}^n \|$为拦截导弹 n 到突防导弹的二维矢量的 2 范数距离。

由于本文假定的突防问题是在有限的二维平面内进行的，因此导弹在设定的环境边界内运动需要满足式（6）：

$$\begin{cases} x_{\min} \leqslant x_p^n \leqslant x_{\max} \\ y_{\min} \leqslant y_p^n \leqslant y_{\max} \end{cases} \tag{6}$$

式（6）中，$n=[1,\ldots,4]$；$x_{\min}$、$x_{\max}$分别为环境边界，本文的边界为$-250\sim250$；$y_{\min}$、$y_{\max}$分别为环境边界，本文的边界为$-250\sim250$。

在研究中，定义速度比为拦截导弹的最大速度与突防导弹的最大速度之比：

$$\lambda = \frac{v_{p\max}}{v_{e\max}} \tag{7}$$

双方导弹各具优势，拦截方导弹数量多于突防方导弹，并且突防方导弹的最大速度高于拦截方导弹。在文献[20]中，通过使用阿波罗奥尼斯圆和几何规律，研究了追捕者—逃跑者追逃问题的成功捕获约束条件。具体而言，当追捕者的速度比 $\lambda < \sin\dfrac{\pi}{N}$时，追捕者很难追上逃跑者。为了增加可研究性，本文设定了速度比的约束条件，如式（8）所示：

$$\lambda \in \left[\sin\frac{\pi}{n}, 1\right) \tag{8}$$

2　约束与算法设计实现

2.1　导弹突防场景下的约束

在弹群攻防对抗的场景中，除了双方弹群之间的对抗，

弹群内部的导弹也需要协同完成任务，使得场景要素更加复杂，且对抗双方的对抗性更强。针对在作战空域内的多导弹协同攻防对抗场景，本文将对抗场景的预设为：作战空域内同时存在多颗拦截导弹和突防导弹，双方具有相反的战术目标。拦截导弹的目标是追击并拦截突防导弹，而突防导弹的目标是尽可能地突破拦截导弹的封锁，或者尽可能地保护其他导弹进行突防。弹群突防的对抗场景如图 1 所示。

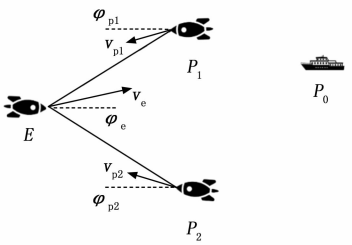

图 1 导弹追捕场景

图 1 中，E 为进攻方导弹，P_1 P_2 为拦截方导弹，P_0 为进攻方目标（拦截方保护目标；v_e 为进攻方导弹的速度大小及方向，v_{p1} v_{p2} 为拦截方导弹的速度大小及方向；φ_e 为进攻方的导弹的速度航向角，φ_{p1} φ_{p2} 为拦截防导弹的速度航向角。针对以上导弹集群攻防问题描述构造弹群攻防博弈数学模型[25]，建立了有控制约束的多无人机追捕对抗零和微分博弈模型。

考虑到我们简化的二维平面区域的追逃博弈，可以使用直角坐标系来表示对抗双方导弹的实时运动状态。图 2 展示了数学几何模型。

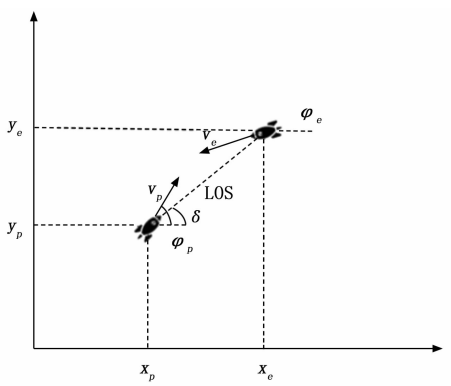

图 2 导弹运动模型

图 2 中，δ 为目标视线（LOS，line of sight）的夹角—视线角，目标视线指攻防导弹 E 的射线，（x_{pn}，y_{pn}）（$n=1, 2, \cdots, N$）、（x_e，y_e）分别为攻击方导弹和拦截方导弹的位置坐标。

拦截方导弹的目标是通过最短时间内拦截目标。而突防导弹的目标是躲避拦截导弹，以避免在作战时间段内被导弹拦截捕获。或者尽可能延迟其他突防导弹被拦截的时间。攻防双方博弈标准微分博弈数学描述为：

$$T_c = f[v_{p1}, \psi_{p1}, L^1, \cdots, v_{pm}, \psi_{pm}, L^n, \cdots, v_{pN}, \psi_{pN}, L^N, v_e, \psi_e] \tag{9}$$

式（9）中，L^n（$n=1, 2, \cdots, N$）为拦截导弹 n 到突防导弹的距离；T_c 为拦截导弹 P 拦截突防导弹 E 的时刻。其中导弹拦截的最优时刻是 T_{cmin}，导弹突防的最优时刻是 T_{cmax}。

2.2 奖励设计

奖励设计是指导增强学习算法性能改进的重要组成部分。攻防双方之间的对抗最终结果只有一个真正的奖赏。在引导智能体产生足够智能的策略中，人工设计的内在回报是关键。本文设计了攻击方导弹、拦截方导弹的奖励，以指导其各自的策略。突防导弹根据爆炸时距离目标位置扣 10 分到加 10 分不等，给予随距离变化的负奖励，系数为 0.001，并引导突防方导弹尽快获得正奖励；当拦截导弹处于拦截任务时，拦截成功的目标越多，得到的奖励越多，以引导拦截导弹尽可能同时拦截多个突防导弹。同时，为了防止进攻方导弹耗尽燃料，将给予随着时间变化的负奖励。攻击上，歼敌航母加 50 分，引爆多个拦截导弹加 5 到 20 分，自身损坏扣 5 分。这种设置是鼓励进攻方导弹重视协作的重要性，引导导弹进行掩护任务。同时，为鼓励导弹进攻敌方航母，将距敌航母的距离设为正奖励，系数为 0.000 000 1。防御方面，拦截方将敌导弹和航空母舰的距离作为负奖励，系数为 0.000 000 1，可防止导弹太近。

2.3 EPF-MADDPG 算法结构及优化

2.3.1 MADDPG 算法

MADDPG 算法是一种针对多智能体协同决策的强化学习算法，在导弹协同领域具有以下优势：1）基于策略梯度的方法，能够有效地处理非线性、高维、连续的动作空间，更适合于导弹协同问题；2）可以学习合作策略，MADDPG 算法可以学习到智能体之间的合作策略，从而在导弹协同中实现协同作战和任务分配，提高协同效率和任务完成率[21-23]。而其他单智能体算法往往只能处理独立策略的问题；3）具有策略共享机制，MADDPG 算法具有策略共享机制，能够让智能体之间共享策略信息，提高学习效率并减少训练时间；4）具有经验回放机制：MADDPG 算法还具有经验回放机制，能够利用过去的经验进行学习，减小样本相关性，提高算法的稳定性和收敛性。综上所述，MADDPG 算法在导弹协同相比其他方法具有更好的学习效果、更高的协同效率和任务完成率。

“集中训练，分散执行”是一种方法，它在训练阶段集中资源进行模型学习和优化，然后在执行阶段将训练好的模型分散到不同计算节点或设备上进行并行计算和推理。这样做可以通过训练学习得到最优的训练策略，使算法得到高效灵活地执行。在运行该算法时，利用智能体的观测信息可以求出最优解，从而得出想要的最优策略。

在“集中训练”阶段，为了计算出更精确的 Q 值反馈给“表演者”网络，可以根据 DDPG 算法平台添加额外数据，包括其他智能体的运动状态、观察值或动作。智能体

还可以根据其他智能体的动作价值以及自身的观察值和动作来判断当前输出动作的价值。

“分散执行”是指在训练完成后，每个 Actor 根据自身的观测值选择适当的动作，无需其他智能体的动作信息。在 MADDPG 算法中，“表演者”网络和“评论家”网络协同工作。每个智能体都有自己的“表演者”网络，用于输出确定的动作。然而，“评论家”网络不仅考虑自身的观测状态和动作，同时也要考虑其他智能体的动作信息。每个智能体都有一个中心化的“评论家”网络，该网络同时接收所有智能体的“表演者”网络生成的数据。[24]。

2.3.2　基于经验池筛选机制的算法策略改进

采取原始 MADDPG 算法时，每一个评论家都需要观察到所有 agent 的状态，而对于本文中涉及的大量不确定 agent 的场景，不是特别适用，而且当按 agent 数量特别多时，状态空间太过于巨大，导致难以收敛。同时每一个 agent 都对应了一个评论家和表演者网络，数量多时，存在大量的模型，增加算法的计算时间。

针对上述问题，设计基于经验池筛选的 EPF-MADDPG 算法。从两个方面对算法进行改进：1）引入长短期记忆（LSTM）网络保存过往训练信息；2）加入阈值筛选机制对算法经验回放策略做出调整。

MADDPG 算法的经验回放策略没有考虑到动作前后的相关性，在遇到从未见过的情况时，往往需要大量的尝试才能学习到最优动作。LSTM 网络主要用于处理环境状态信息的输入，基于“门”来控制信息的丢弃或增加，从而实现遗忘或者记忆的功能，达到缓解梯度消失的作用。

LSTM 网络中的遗忘门、记忆门以及输出门是 LSTM 神经网络中的 3 种门控机制，用于控制输入、输出和忘记之前的信息。其中，遗忘门用于决定之前输入的信息被遗忘的程度；输入门用于控制新输入信息的加入程度；输出门用于控制当前状态的输出程度。网络的整体结构如图 3 所示。

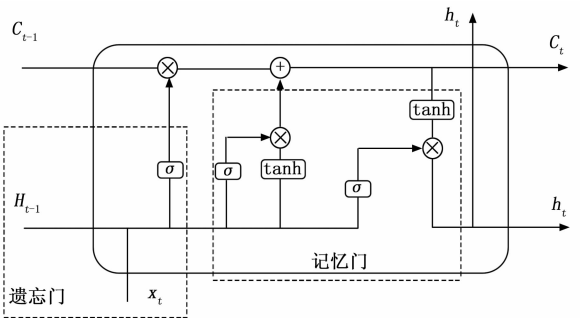

图 3　LSTM 网络结构

遗忘门：控制历史状态流经当前状态后允许多少进入当前状态的门控设备。

记忆门：控制从当前状态向长期记忆中存储哪些信息的门控设备。

输出门：控制从长期记忆中向当前状态输出哪些信息的门控设备。

经验池阈值的设置由预训练决定，将预训练的样本数据按照从大到小依次排列为一序列，序列样本总数为 n，设定参数 α 代表正式训练时使用序列样本的比例，选取 $\alpha \times n$ 位置的样本所对应的值作为预值。为设立合理的阈值进行预训练，按优先级从高到低的顺序对数据列表进行排序，然后从高斯随机数值生成器中获取一个 0～1 之间的随机数 α，其中 α 在 0～1 之间取值的概率呈正态分布，这样就可以尽可能取到中间的数值，避免出现接近 0 或接近 1 的极端情况。

对于正式训练的样本数据，只有大于预设阈值的样本才会放入经验池中。在基于经验池筛选的 MADDPG 算法中，采用纯粹贪婪优先方法对样本进行排序，确保被采样的频率在继承优先级上是单调的。同时在排序好的样本队列中加入均匀随机采样，避免了高优先级产生的过拟合问题。

2.3.3　算法框架实现

本文采用的 MADDPG 算法框架如图 4 所示。在训练过程中，首先初始化整体的状态和策略网络。智能体根据当前时刻的状态输入 Actor 网络，生成对应的动作。环境返回智能体执行当前动作时所获得的奖励和转移到的下一状态。智能体将生成的四元组数据存储到经验回放缓存中，以备后续的“表演者”网络和“评论家”网络更新时使用。然后智能体从缓存池中采样多个批次的机动轨迹，每一条机动轨迹是智能体与环境进一步交互得的。输入 Actor 网络进行训练的数据是智能体当前时刻的状态。智能体利用已更新的模型与环境进行下一步的交互，然后利用生成的数据更新经验回放缓存池。当然，每个智能体都有自己的“表演者”网络和“评论家”网络，还有一个所有智能体共有的“评论家”网络，每个智能体自身的“评论家”网络学习单个智能体每轮训练的期望收益，所有智能体共有的“评论家”网络学习团队的期望收益。

下面是本文的整体算法设计。

初始化全局 EvalCritic 网络 Q_ψ^κ

初始化全局 TargetCritic 网络 $Q_\psi^{\kappa\prime}$，并拷贝 Q_ψ^κ 参数

初始化每个智能体的 EvalActor 网络 Q_θ^π 和 EvalCritic 网络 Q_θ^π

初始化每个智能体的 TargetActor 网络 $Q_\theta^{\pi\prime}$ 和 TargetCritic 网络 $Q_\theta^{\pi\prime}$，并拷贝 Q_θ^π 和 Q_θ^π 参数

Forepisode = 1 to MaxEpisode do

在设定的范围内随机初始化突防导弹、拦截导弹的初始状态

Fort = 1 to MaxStep do

获得仿真环境初始状态 s^t

对于每颗导弹 i，选择动作 $a_i^t = \pi\theta_i(o_i^t)$

执行动作 $a^t = [a_1^t, a_2^t, \cdots, a_N^t]$

获得全局奖励 r_g^t 和局部奖励 r_i^t

存储元组 $(s^t, a^t, r_1^t, r_g^t, s^{t+1})$ 进入经验池

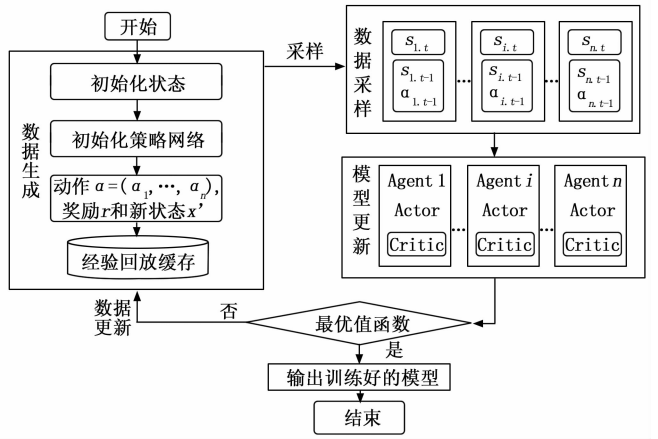

图 4 MADDPG 算法框架

/ * 全局 Critic 网络更新 * /

从经验池随机抽样(s^j,a^j,r_g^j,s'^j)组成样本

$(a'_1,a'_2,\cdots,a'_N)=(\pi'\ \theta_i(o'^j_i),\cdots,\pi'\ \theta_N(o'^j_N))$

$y_g=\ r_g+\gamma Q_{\psi'}^g(s',a'_1,a'_2,\cdots,a'_N)\ |_{a'_i=\pi'_i(o'_i)}$

计算损失$\frac{1}{S}\sum(y_g-Q_\psi^g(s^j,a'_1,a'_2,\cdots,a'_N))^2$

刷新 argetCritic 网络 $Q_\psi^g{}'$参数

/ * Actor 网络和局部 Critic 网络更新 * /

For Agenti=1 to N do

从经验池随机抽样 K 个样本(s^j,a^j,r_g^j,s'^j)组成批样本

计算 $y^j=r_i^j+\gamma Q_{\varphi'}^{\pi'}(o'^j,\pi'_\theta(o'^j_i))$

计算损失$\frac{1}{S}\ \sum(y_g-Q_\phi i^\pi(o^j,a_i^j))^2$，刷新 EvalCritic 网络 Q_ϕ^g 的参数

更新 EvalActor 网络 Q_θ^π参数

刷新智能体 Agenti TargetActor 网络 $Q_{\theta'}^\pi{}'$和 TargetCritic 网络 $Q_{\theta'}^\pi{}'$

End For

End For

End For

3 实验结果

为了验证所提方法的优越性，本实验的硬件配置为，CPU：Intel® Core™ i7-13700KF CPU @4.20 GHz；内存：32 G；显卡：Geforce RTX4070Ti（12 G 内存）上，基于 Windows10 平台，显存位宽为 64 位 DDRM。

图 5 为在不同范围的仿真场景下的所有智能体的算法回报，图中随着场景的一步步扩大，算法收敛得到的回报也逐渐提高，说明在更大的作战范围中突防导弹可以更好地达到任务目标，拦截方导弹在更小的作战范围内，拦截的成功率就越高。同时在 1 000 * 1 000（km）以后，场景得到的回报提升就不再显著。

本文实验针对海域上的导弹集群攻防博弈情形进行了设计。假设在某海域中，我方发射两枚导弹对敌方航母发起打击，在相对坐标 1 000 * 1 000（km）的区域内敌方发射三枚拦截导弹，实施突防策略。为了使实验具有可操作性，设定突防导弹的机动能力比拦截导弹的机动能力大，同时规定为距离的安全约束，当其中有一个拦截导弹靠近了突防导弹该距离约束值内，追捕成功，博弈结束。为加速收敛，忽略 z 轴的动力学模型，得到一个平面内二维的博弈场景[26-27]。实验设计的训练参数如表 1 所示。

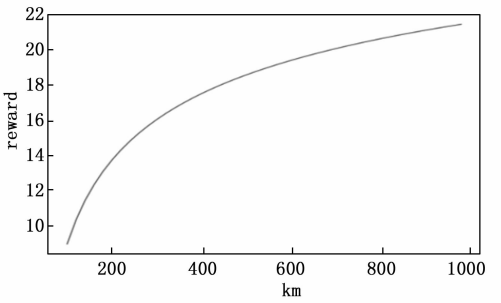

图 5 范围－回报变化

表 1 算法训练超参

训练参数	数值
折扣因子 γ	0.9
经验池大小 M	30 000
批样本数 Batchsize	128
Critic 网络学习率 α_C	0.002
Acrtor 网络学习率 α_A	0.001
回合数目 $Episode_{\max}$	60 000
单回合最大步长 $Step_{\max}$	1 500

首先，实验分析了该场景下 MADDPG 算法的收敛性。图 6 为 MADDPG 与 DQN 算法的回报奖励，其中 DQN 的学习率设置为 0.001，采用批量梯度下降的方式进行学习，经验池大小与批样本数与 EPF-MADDPG 算法保持一致。经过 14 000 轮的训练后，网络的 loss 值逐渐降低，且趋于稳定，说明网络收敛，各个智能体都能产生更合适的动作。从图 5、6 中可以看出，MADDPG 算法相较于 DQN 算法具有更快的收敛速度，以及更优秀的回报奖励。同时，各单元参与者网络的下降趋势相似，关键网络的下降趋势也相似。

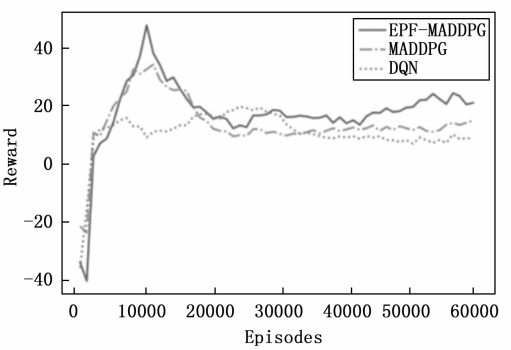

图 6 各个算法回报

同时，根据图 6 所示，基于经验池筛选策略的 MADDPG 算法耗时明显低于传统的 MADDPG 算法，其最大时延为 320 ms，而 DQN 算法需要 400 ms。EPF-MADDPG 相较于 DQN 算法提升了 8%左右，满足实际场景中的实时性需求。

随着不断地训练，敌方智能体也会学习到一些策略，这就会导致回报的下降，但这也会促进我方智能体的学习，最后收敛到一个稳定的回报。

随着不断地学习，智能体会逐渐学习到一些策略，用来欺骗敌方。图 7 中是智能体的行动轨迹，我们可以看到智能体会做出“假动作”诱使敌方智能体做出错误的判断，并加速通过速度优势越过拦截导弹的拦截。

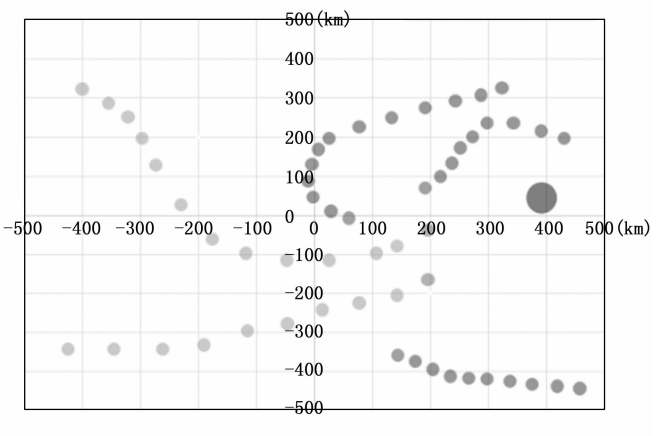

图 7　智能体机动行为

并且经过训练的智能体也表现出协作的特征，图 8 中颜色较深的智能体作为诱饵，吸引了敌方 3 枚导弹的围追堵截，通过消灭拦截方的 3 枚导弹，为己方的突防导弹创造了条件，另一枚导弹最后顺利完成任务目标。

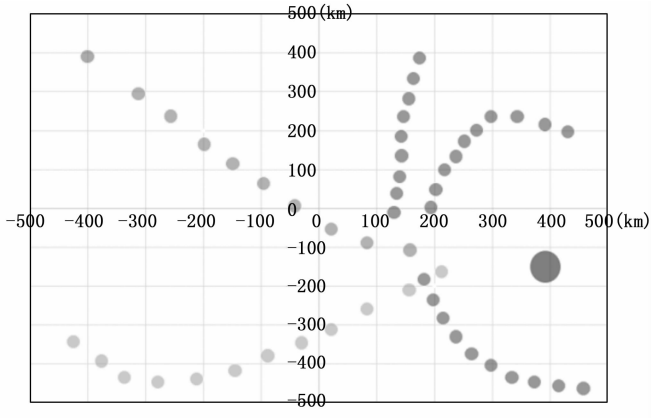

图 8　智能体协同行为

实验经过 100 次仿真模拟，如表 2 所示，经过 EPF-MADDPG 算法训练的突防方导弹胜率可以达到 73%，实验结果表明，训练出来的协同突防策略具有明显合作以及欺骗对手的行为，突防导弹不仅简单的依靠速度进行突防，同时表现出一些高级的协同行为，极大提高了突防策略的训练效率。

表 2　各个算法突防成功率

算法	成功率/%
DQN	63
MADDPG	69
EPF-MADDPG	73

4　结束语

为了在仿真环境中实现多智能体对抗的智能决策，提出了一种基于 MADDPG 的异构多智能体对抗决策算法，辅助决策者进行导弹集群突防方案的制订，并且在方案执行的过程中具有一定的自主决策能力。为了进一步地验证 MADDPG 算法对于导弹突防场景的可行性，本文还从仿真的角度进行验证，经过基于经验池筛选策略的 MADDPG 算法计算的突防策略成功率达到 73%。

本文还存在待改进的方面：首先，对于导弹突防任务来讲，不仅有同波次导弹间的协同配合，同时还应有多波次导弹的协同配合，对于任务分解规划，以及战场态势的侦察获取，还需要进行深入的研究改进，得到一个简单易行的方法；其次，本算法的仿真业务场景具有特殊性，仍需进行改进学习，在不同环境不同维度进行推演验证。

参考文献：

[1] 王　星，郝泽龙，周一鹏. 美国智能导弹空战体系结构与技术[J]. 飞航导弹，2021 (11)：91－97.

[2] 徐国奇，洪昭斌，陈水宣，等. 采用 DDPG 算法的弹道导弹突防诱饵分布空域[J]. 厦门理工学院学报，2022，30 (1)：34－41.

[3] 李　刚，王蜀杰，李兴格. 地空导弹突防技术综述[J]. 飞航导弹，2019 (8)：35－38.

[4] 王　芳，涂震飚，魏佳宁. 战术导弹协同突防关键技术研究[J]. 战术导弹技术，2013 (3)：13－17.

[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)：522－535.

[6] CHEN W，NIE J. A MADDPG-based multi-agent antagonistic algorithm for sea battlefield confrontation[J]. Multimedia Systems，2022，29 (5)：2991－3000.

[7] 李乔扬，陈桂明，许令亮. 弹道导弹突防技术现状及智能化发展趋势[J]. 飞航导弹，2020 (7)：56－61.

[8] 槐泽鹏，梁雪超，王洪波，等. 多弹协同及其智能化发展研究[J]. 战术导弹技术，2019 (5)：77－85.

[9] 陈中原，韦文书，陈万春. 基于强化学习的多发导弹协同攻击智能制导律[J]. 兵工学报，2021，42 (8)：1638－1647.

[10] 张冬呈. 多智能体强化学习在随机博弈中的研究[D]. 成都：电子科技大学，2021.

[11] 张　克，刘永才，关世义. 多智能体系统在导弹攻防对抗仿真中应用的可行性研究[J]. 战术导弹技术，2001 (6)：59－65.

(下转第 212 页)