中国科学: 信息科学 2022 年 第 52 卷 第 12 期: 2225–2238
SCIENTIA SINICA Informationis
《中国科学》杂志社
SCIENCE CHINA PRESS
非完全信息下人机合作对抗博弈专题 · 论文

分层决策多机空战对抗方法

王欢[1], 周旭[1], 邓亦敏[2], 刘小峰[1,3*]

1. 河海大学物联网工程学院, 常州 213022
2. 北京航空航天大学自动化科学与电气工程学院, 北京 100191
3. 江苏省特种机器人技术高校重点实验室, 常州 213022
* 通信作者. E-mail: xfliu@hhu.edu.cn

收稿日期: 2022–05–09; 修回日期: 2022–07–29; 接受日期: 2022–09–19; 网络出版日期: 2022–12–06

科技创新 2030—“新一代人工智能” 重大项目 (批准号: 2018AAA0100803) 资助

摘要 在空战研究领域, 战术决策旨在提高博弈对抗收益, 进而提升战机攻击效率. 现有战术决策算法大多基于规则方法设计, 当应用于多机空战的复杂环境时则存在设计难度大, 难以求解最优解等问题. 本文提出一种分层决策多机空战对抗方法, 首先, 在训练初始阶段借鉴已有人类专家经验, 指导模型训练; 其次, 根据战术动作类型设计分层动作决策网络, 降低动作决策空间维度; 最后, 将训练产生的对抗经验按阶段分解, 降低策略学习难度. 在多机空战仿真环境中进行了实验验证, 相比于现有多机空战决策方法, 本文提出的方法在训练收敛性和决策性能方面均具有更好的表现.

关键词 多机空战, 动作决策网络, 博弈, 分层强化学习, 决策收益

1 引言

无人机因其机动性强、成本低、可避免人员伤亡等优点, 已被广泛应用于各种监测、侦查等军事领域. 近年来, 在多次军事行动中, 无人机都发挥了重要作用. 世界各国纷纷投入大量科研力量进行无人机空战决策技术的相关研究, 如美国 Psibernetix 公司开发了基于人工智能的 “Alpha”AI 系统 [1], 该系统基于遗传算法进行战斗决策并在模拟空战中击败了美国空军退役上校; 美国国防高级研究计划局 (Defense Advanced Research Projects Agency, DARPA) 开启的 “进攻蜂群战术” (OFFSET) 项目 [2], 通过开发多样的无人机蜂群战术, 测试无人机蜂群协同行动能力, 以期在未来辅助小型地面部队在复杂城市环境中执行任务; 欧洲多国联合推动的 “未来战斗航空系统” (Future Combat Air System, FCAS) 项目, 旨在加强人机协同作战水平 [3]; 俄罗斯苏霍伊设计局研发了 S-70 “猎人” 无人机并着手研究该无人机与苏 -57 战机协同作战技术 [4]. 除此之外, 各国学者也在无人机空战决策方面做了大量研究, 如 McGrew [5] 将动态规划方法引入 1 对 1 空战博弈决策中提升对抗效率; 文献 [6] 中提出一种基于模糊

引用格式: 王欢, 周旭, 邓亦敏, 等. 分层决策多机空战对抗方法. 中国科学: 信息科学, 2022, 52: 2225–2238, doi: 10.1360/SSI-2022-0185
Wang H, Zhou X, Deng Y M, et al. A hierarchical decision-making method for multi-aircraft air combat confrontation (in Chinese). Sci Sin Inform, 2022, 52: 2225–2238, doi: 10.1360/SSI-2022-0185

规则的空战机动决策方法, 在策略上引导战机做出更有利的战术动作. 上述方案大都基于规则方法设计, 在多机博弈对抗环境中采用上述传统方法则存在求解难度高, 规则设计过于复杂等缺点.

深度强化学习技术已被逐渐应用于空战决策领域. 深度强化学习方法作为机器学习的方法之一, 具有适应性强、自主学习、不需要过高专业背景知识的特点. 本文提出分层决策多机空战对抗方法, 该方法针对多机空战场景采用分层动作决策网络降低决策动作空间维度; 引入经验分解变换机制对复杂任务经验进行分解变换同时结合专家经验指导, 提升训练效率; 最后, 在 MACA (multi-agent combat arena) 仿真环境 [7] 中验证了算法的有效性.

2 相关工作

针对如何在多机对抗环境中提高空战决策效率的问题, 本文结合强化学习技术与多机空战对抗特点, 提出一种分层决策多机空战对抗方法, 本节将先介绍无人机空战决策研究技术, 再介绍相关强化学习技术.

2.1 空战决策技术

空战决策是无人机空战博弈的核心问题. 目前广泛采用的是基于 OODA 回路 (Observation 观察、Orientation 判断、Decision 决策、Action 执行) 的决策理论. 无人机根据感知到的战场环境信息, 结合自身状态做出战术决策, 生成机动动作. 各国学者在空战决策问题的研究中提出了大量方法, 根据其求解思路的不同, 可分为基于对策理论的方法、基于专家系统的方法, 以及启发式学习的方法三类.

第一类是基于对策理论的方法. 如 Getz 等 [8] 提出了 “双目标对策” (two target game) 模型, 该模型中空战双方根据当前的战场态势不同, 都可成为追逐方或者逃逸方. Austin 等 [9] 提出的基于矩阵对策的机动决策方法. 这种矩阵对策方法首先将机动动作离散为由多个基本机动动作构成的机动动作库, 再利用矩阵描述双方所有可能的机动组合, 最后采用数值积分的方法对飞机的运动方程进行求解进而得到最优决策序列.

第二类是基于专家系统的方法. 如奚之飞等 [10] 将威力势场引入多机空战决策中, 提升了多机空战中的协同性能. 周文卿等 [11] 基于蚁群算法设计了一种多无人机自主控制算法以提高无人机集群在空战中的成功率. 严飞等 [12] 将协同粒子群算法和协同函数、协同变量结合解决多无人机同时攻击约束问题.

第三类是启发式学习的方法. 该类方法以强化学习的方法最为典型, 以网络拓扑形式模拟人类大脑神经元结构, 实现对非线性复杂关系的表征以及相应的学习能力. 如符小卫等 [13] 基于解耦多智能体深度确定性策略梯度算法提出一种多无人机协同追捕方案. Sun 等 [14] 提出一种多智能体分层策略梯度算法, 并通过自博弈训练学习得到超越人类专家认知的战术策略. 施伟等 [15] 提出一种近端策略优化空战决策算法, 提升了 3 对 3 空战环境战机的决策性能. 此外基于仿生学的相关算法, 如粒子群算法、遗传算法等, 也常常被用于空战决策领域.

上述三类方法存在下列问题.

基于对策理论的方法求解困难, 模型构建复杂, 适用于简单空战场景, 如追逃、拦截等; 基于专家系统的方法依赖于专家经验和决策水平, 对设计人员的专业背景要求较高, 当空战环境变得复杂多变时, 仅依据专家知识难以做出最优决策; 基于启发式学习方法的研究中, 许多工作主要用于追捕、拦截、一对一空战等简单场景, 对于复杂空战场景难以适用; 一些研究虽然被应用到了多机空战领域, 并对强化学习算法做出了改进, 但未能将已有人类专家经验策略与强化学习方法相结合, 导致策略学习

低效.

2.2 模仿学习

模仿学习也被称为基于演示的学习或者学徒学习, 机器学习中智能体与环境交互, 由环境反馈获得相应的奖励并更新网络, 但在初始探索过程中, 尤其在稀疏奖励环境中, 很难获得正向的奖励. 直观的做法是让机器学习人类的思想, 从已有的专家经验数据中学习专家策略. 如文献 [16] 中使用人类玩家对局数据和智能体自博弈数据结合训练生成的智能体 AlphaStar, 在星际争霸游戏中击败了职业玩家, 竞技水平超过 99.8% 的玩家; 文献 [17] 提出一种轻量级模仿学习框架, 使用相比 AlphaStar 更小的计算量训练出的多智能体, 依然能够在星际争霸游戏中达到顶级选手的表现; 文献 [18] 将模仿学习应用于无人驾驶环境, 使得智能体能够在高度交互环境中完成自主驾驶. 此外, 模仿学习在自主机械臂设计、无人潜水艇拦截、无人机导航等方面都有广泛的应用 [19, 20].

2.3 分层强化学习

分层强化学习是强化学习领域中的一个分支. 分层强化学习受启发于人类解决复杂问题的思想, 将复杂问题分解成若干子问题, 通过任务分解逐个解决子问题, 从而最终解决一个复杂问题. 这里的任务分解有两种方法: (1) 所有的子问题协同解决被分解的任务; (2) 把前一个子问题的结果作为下一个子问题解决方案的输入, 层层递进解决问题 [21~23]. 如 Wang 等 [24] 应用分层强化学习解决移动机器人导航问题, 解决了常规导航方法在复杂环境下导航效果不佳的问题; Yang 等 [25] 在 3 对 3 足球比赛环境中应用分层网络多智能体学习框架, 有效提升多智能体博弈对抗决策水平. 另外, 分层强化学习也应用在机械臂控制、星际争霸 2 游戏等场景中 [26, 27].

3 分层决策多机空战对抗方法

现有的空战决策方法大多基于规则设计, 少数采用了强化学习算法, 多应用于无人机追捕或 1 对 1 无人机对抗等简单场景中, 未能考虑专家经验对于训练过程的指导作用. 本文针对多机博弈对抗场景, 提出一种分层决策多机空战对抗方法, 简称 H-QMIX 方法.

3.1 总体设计

在多智能体空战中智能体无法观测到全局信息, 本文将多智能体空战决策任务考虑为一个部分可观测马尔可夫决策过程, 其由元组 $\langle S, A, P, r, Z, O, n, \gamma\rangle$ 组成. 令 $s \in S$ 表示环境的实际状态, 在每个时间步, 智能体 $g \in G \equiv \{1, \ldots, n\}$ 选择一个动作 $a^g \in A$, 共同组成一个联合动作 $a \in A \equiv A^g$, 进而通过状态转移函数 $P(s'|s,a) : S \times A \times S \to [0,1]$ 引起环境状态的变换. 整个过程中, 所有的智能体共享一个奖励函数 $r(s,a) : S \times A \to \mathbb{R}$, $\gamma \in [0,1)$ 代表折扣因子.

在部分可观测环境中, 每个智能体通过状态转移函数 $O(s,a) : S \times A \to Z$ 获得个体观测状态 $o \in Z$, 每个智能体的动作观测历史可表示为 $\tau^g \in T \equiv (Z \times A)^*$, 随机策略 $\Omega^g(a^g|\tau^g) : T \times A \to [0,1]$ 以此作为条件. 联合策略 Ω 的联合动作值函数可表示为

$$Q^{\pi}(s_t, a_t) = E_{s_{t+1:\infty}, a_{t+1:\infty}}[R_t|s_t, a_t], \tag{1}$$

其中, $R_t = \sum_{i=0}^{\infty}\gamma^i r_{t+i}$ 代表折扣奖励. 训练中每个智能体的策略学习仅以自身动作观察历史作为输入.

本文设计的分层决策多机空战对抗方法框架如图 1 所示, 主要包含专家经验池、飞行动作决策层、攻击动作决策层、环境交互部分、对抗经验分解变换部分. 训练开始从专家经验池 $\mathcal{B}$ 采样数据对两个动作决策层进行训练更新, 训练产生的飞行动作 a_f 与攻击动作 a_i 构成组合动作并输入环境交互部分; 环境交互部分根据输入的组合动作更新环境状态, 输出单步对抗经验存入 π^n, 同时输出用于飞行、攻击动作决策的观测状态 o_f, o_i, 如式 (2) 和 (3) 所示:

$$o_f = \{o^1, \ldots, o^n\}, \tag{2}$$

$$o_i^m = \omega^m \odot o_f, \tag{3}$$

其中, $\odot$ 为两数组对应元素乘积, N 为智能体总数, 系数数列 $\omega^m = \{\omega^{m,1}, \ldots, \omega^{m,n}, \ldots, \omega^{m,N}\}$, $o^1, \ldots, o^n$ 为各智能体当前观测状态, 当智能体 m 与 n 的距离小于攻击距离阈值时, $\omega^{m,n} = 1$, 反之, $\omega^{m,n} = 0$; 对抗经验分解变换部分将生成的对抗经验 π^n 按照变换规则分解后存入专家经验池 $\mathcal{B}$; 专家经验池中的经验在经验池存满后, 旧的经验将被逐步替换, 如此循环迭代, 直到训练结束. 飞行动作、攻击动作决策层均基于 QMIX [28] 值分解网络结构设计, 飞行动作决策层用于训练集中飞行动作层联合动作值函数 $Q_f^{\text{tot}}(\tau_f, a_f)$, 攻击动作决策层用于训练攻击动作层联合动作值函数 $Q_i^{\text{tot}}(\tau_i, a_i)$, 两者可表示为单个智能体值函数 $Q_g(\tau^j, a^j, \theta^j)$ 之和, 如式 (4) 所示:

$$Q^{\text{tot}}(\tau, a) = \sum_{j=1}^{n} Q_j(\tau^j, a^j, \theta^j). \tag{4}$$

同时需保证 Q^{tot} 满足式 (5):

$$\arg\max_a Q^{\text{tot}}(\tau, a) = \begin{pmatrix} \arg\max_{a^1} Q_1(\tau^1, a^1) \\ \cdots \\ \arg\max_{a^n} Q_n(\tau^n, a^n) \end{pmatrix}. \tag{5}$$

将式 (5) 转换为单调性约束, 如式 (6):

$$\frac{\partial Q_{\text{tot}}}{\partial Q_g} \geqslant 0, \quad \forall g \in G, \tag{6}$$

并使用混合网络实现. 其损失函数为

$$\mathcal{L}(\theta) = \sum_{i=1}^{b} \left[\left(y_i^{\text{tot}} - Q^{\text{tot}}(\tau, a, s; \theta) \right)^2 \right], \tag{7}$$

其中, b 为每次训练的采样批次大小, $y^{\text{tot}} = r + \gamma \times \max_{a'} Q^{\text{tot}}(\tau', a', s'; \theta^-)$, θ^- 代表目标网络参数. $Q_f^n(o_f{}^n, a_f{}^n)$ 和 $Q_i^n(o_i{}^n, a_i{}^n)$ 分别代表各个智能体用于生成飞行、攻击动作的动作值函数, 根据式 (1) 可计算得到.

3.2 分层动作决策网络

多机博弈对抗是一项复杂任务, 无人机需要在战场态势快速变化的情况下做出最优攻击决策. 本文提出将无人机决策控制任务按照特性分解为飞行动作 a_f 和攻击动作 a_i. 飞行动作负责完成无人机的飞行角度、速度、高度控制, 需要兼顾全局战场态势并躲避敌军导弹; 攻击动作仅完成是否攻击及攻

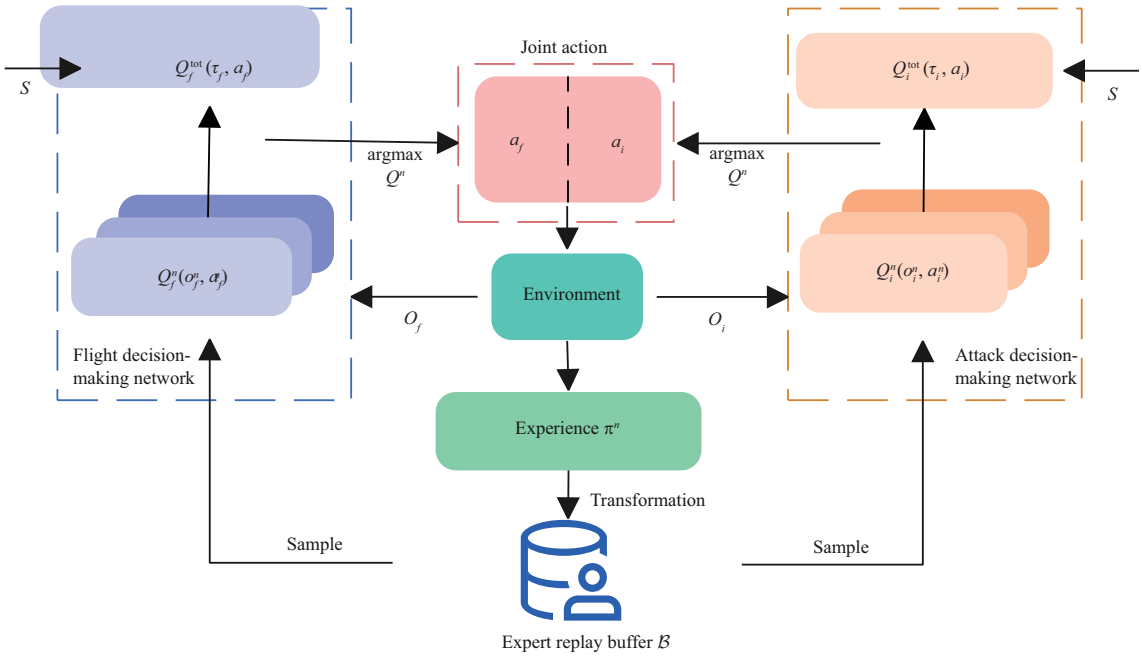

图 1 (网络版彩图) 算法整体框架

Figure 1 (Color online) Algorithm framework

击目标的选择, 考虑局部战场信息即可. 如图 1 所示, 飞行动作 a_f 由飞行决策层生成, 包含航向、高度和速度决策动作, 需要全面的战场信息 (o_f), 攻击动作 a_i 由攻击决策层生成, 包括是否攻击和攻击目标的编号, 仅考虑当前可攻击敌机的信息和附近友军信息 (o_i). 飞行与攻击动作构成组合动作控制无人机完成空战对抗任务.

3.3 专家经验指导

本文算法提出利用人类专家知识, 采用基于规则的方法编写无人机对抗决策规则并与敌机对抗生成专家对抗经验, 选择其中对抗胜利的经验构成专家经验池 $\mathcal{B}$ 指导模型训练. 基于规则的方法或许不能在所有对战情况下都获得胜利, 但生成的对抗经验能够在神经网络训练初期引导网络更新, 提升模型训练收敛速度. 随着训练进行, 专家经验池中旧的经验将逐步被训练过程中产生的高质量经验覆盖, 模型最终全部使用新生成的经验数据进行迭代训练至收敛.

3.4 经验分解变换

在多机博弈中获得胜利, 要求智能体在一段较长时间内连续完成高效的机动决策任务. 常规的方法直接训练智能体完成复杂任务, 存在两个问题: (1) 训练初期可用经验少, 高质量经验比例小; (2) 对抗回合经验序列长, 直接学习训练效率低. 针对此类问题, 文献 [29] 提出一种事后经验重放的方法, 针对采样的数据采用随机点方法分解经验, 未能考虑任务经验的具体阶段特性. 本文提出一种改进的经验分解变换方法, 在扩充初期训练经验数量的同时, 降低训练难度. 该方法借鉴人类学习复杂任务的过程, 将复杂任务分解为不同阶段任务进行学习, 如图 2 所示, 首先对于每回合对抗经验数据同步记录一条标志数据 d, d 中为 1 (如 d_i, d_j, d_k) 的位对应击落敌方战机的时刻, 其余位为 0; 下一步根据标志数据中为 1 位对应经验数据的位置 (如 e_i, e_j, e_k), 将数据分解变换为分解经验 1, 分解经验 2, 分解

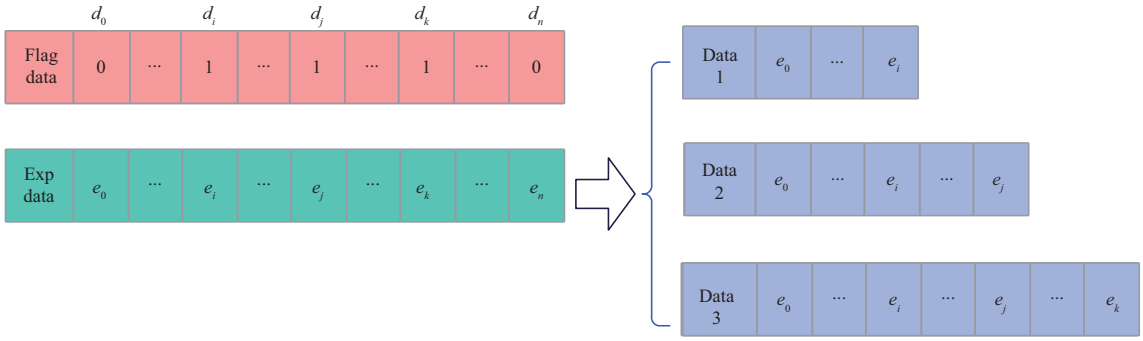

图 2　(网络版彩图) 经验分解变换示意图

Figure 2　(Color online) Schematic diagram of experience decomposition transformation

经验 3 并重新计算对应的奖励; 最后将原经验数据与分解经验数据共同存入专家经验池 $\mathcal{B}$, 用于模型训练. 该方法一方面扩充了每回合对战过程生成的经验, 另一方面将复杂的对战过程分解为几个阶段任务 (击毁不同数量战机), 降低了模型学习策略难度.

3.5　分层决策多机空战博弈决策算法

分层决策多机空战博弈算法具体如算法 1. 其中, 飞行决策网络与攻击决策网络利用专家经验池数据结合式 (7) 进行更新并通过 ϵ- 贪心算法 (如式 (8)) 来选择各自动作与环境交互生成新的对抗经验. 每个对抗回合产生的回合经验 π^n 采用式 (9) 进行分解变换得到包含子目标经验数据的集合 π^n_{sub}, 用于更新专家经验池. 式 (9) 中, π_j 为回合经验 π^n 中前 j 步经验的集合, d_j 为对抗数据中的标志数据位, 当 $d_j = 1$ 时, 表示此时刻击落了敌军战机.

$$a = \begin{cases} \text{argmax}_a Q(o,a), & \text{概率 } \varepsilon, \\ \text{随机选择动作}, & \text{概率 } 1-\varepsilon, \end{cases} \tag{8}$$

$$\pi^n \mapsto \pi^n_{\text{sub}} : \{\pi_j | d_j = 1, j \in [0, t_{\text{round}}]\}. \tag{9}$$

4　实验测试结果

本文实验平台为 “CETC-MACA 无人机多智能体强化学习平台”, 该平台是空战多智能体对抗算法研究、训练、测试和评估的环境, 支持红蓝双方多智能体算法在设定地图场景中进行空战博弈对抗. 实验服务器搭载的 CPU 为 Intel Xeon Silver 4210R, 显卡为 NVIDIA GeForce RTX 3080, 内存 64 GB.

4.1　多机对抗场景设计

多机对抗场景如图 3 所示, 场景中红蓝双方战机性能、数量相同, 红色战机 (己方) 采用本文提出算法进行动作决策, 蓝色战机 (敌机) 默认采用基于规则的方法完成动作决策. 对抗场景为长 10 km, 宽 10 km 的正方形公海区域. 在设计的对抗场景中, 我方战机巡逻中遭遇敌机, 在低空空域展开空战, 对敌军战机进行摧毁, 摧毁全部敌机或对抗时间结束时存活战机多者获胜, 实验采样步长为 1 s, 回合对抗时长为 600 s, 双方初始距离约 8 km, 设定航向水平向右为 0° 方向, 随顺时针方向增加, 战机主要性能及初始状态设定见表 1.

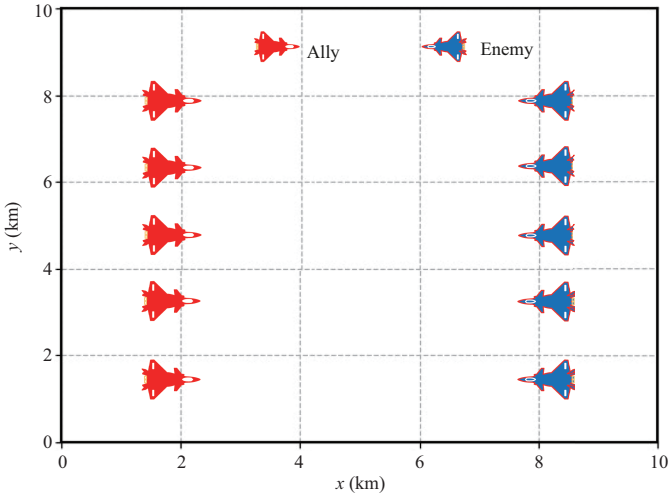

图 3 (网络版彩图) 对抗场景示意图

Figure 3 (Color online) Confrontation scenario

算法 1 分层决策多机空战对抗方法

输入: 空战对抗经验数据 $o_{f,t}$, $o_{i,t}$, s_t, $a_{f,t}$, $a_{i,t}$;

主迭代: 初始化 $Q_f^n(o_f^n, a_f^n)$, $Q_i^n(o_i^n, a_i^n)$, $Q_f^{\text{tot}}(\tau_f^n, a_f^n)$, $Q_i^{\text{tot}}(\tau_i^n, a_i^n)$, 专家经验池 $\mathcal{B}$, 迭代次数 T, $\varepsilon \in [0,1]$;

for epsoide = 1 to N **do**
  s_t, $o_{f,t}$, $o_{i,t}$, $a_{f,t}$, $a_{i,t}$, $r_{f,t}$, $r_{i,t}$ = env.reset();
  初始化回合对抗步长 t_{round};
  for each step $t = 1$ to T in the episode **do**
    if t mod $t_{\text{round}} \neq 0$ **then**
      从专家经验池采样作为网络输入, 采用 ϵ-greedy($Q_f^n(o_{f,t}^n, a_{f,t}^n)$), ϵ-greedy($Q_i^n(o_{i,t}^n, a_{i,t}^n)$) 贪心方法计算每个智能体的 $a_{f,t}^n$, $a_{a,t}^n$, 如式 (8);
      根据式 (2) 和 (3) 计算 s_{t+1}^n, $o_{f,t+1}^n$, $o_{i,t+1}^n$;
      计算每个智能体的 $r_{f,t}^n$, $r_{i,t}^n$;
      将数据 (s_{t+1}^n, $o_{f,t}^n$, $o_{i,t}^n$, $a_{f,t}^n$, $a_{i,t}^n$, $r_{f,t}^n$, $r_{i,t}^n$,$o_{f,t+1}^n$, $o_{i,t+1}^n$) 送入 π^n;
    else
      基于式 (9) 将回合对抗数据 π^n 分解为一组子目标经验数据 π_{sub}^n;
      将 π_{sub}^n 存入专家经验池 $\mathcal{B}$;
    end if
    if step t mod $t_{\text{train}} = 0$ **then**
      基于式 (7) 更新 $Q_f^n(o_f^n, a_f^n)$, $Q_i^n(o_i^n, a_i^n)$, $Q_f^{\text{tot}}(\tau_f^n, a_f^n)$, $Q_i^{\text{tot}}(\tau_i^n, a_i^n)$;
    end if
  end for
end for

输出: 更新后的网络.

4.2 模型构建

使用本文提出的分层决策多机空战博弈方法构建强化学习智能体, 按本文所述方法对强化学习中涉及相关要素进行定义.

(1) 状态设计. 状态包含己方和敌方战机及已发射导弹的信息. 由于雷达探测范围的限制, 敌方战机或者导弹的位置、航向或速度信息可能出现缺失, 缺失的信息做补零处理, 数据全部进行了去量纲

表 1 战机性能指标及初始状态

Table 1 Parameter and initial state of aircraft

Category	Value range	Category	Value range
Radar detection range	$[0, 1800]$ m	Flight speed range	$[50, 300]$ m/s
Number of missiles	6	Missile range	$[0, 1200]$ m
Ally's initial position	Left side of air combat area	Enemy's initial position	Right side of air combat area
Ally's initial speed	200 m/s	Enemy's initial speed	200 m/s
Ally's initial height	2000 m	Enemy's initial height	2000 m
Ally's initial course	0°	Enemy's initial course	180°

表 2 状态空间信息

Table 2 State space information

Entity name	State information
Ally	Position, course, speed, number of missiles
Ally's missle	Position, course, speed
Enemy	Position, course, speed
Enemy's missle	Position, course, speed

表 3 动作空间信息

Table 3 Action space information

Action	Value
Course	Turn left, hold on, turn right
Height	Pull up, hold on, dive
Speed	Decelerated flight, steady flight, accelerated flight
Attack	Not attack, target ID

表 4 奖励设计

Table 4 Reward shaping

Event	Reward	Event	Reward
Kill	10	Missile escaped	2
Out	−2	Detect the enemy	3
Win	30	Lose	−30
Draw	−6		

处理, 具体见表 2.

(2) 动作设计. 本文中战机决策动作包括飞行控制动作和攻击动作. 飞行控制动作包括航向、高度、速度; 攻击动作包括是否攻击及攻击敌机编号. 为降低决策动作空间维度, 本文对飞行控制动作进行离散化处理, 具体见表 3.

(3) 奖励设计. 奖励包含 2 个部分, 分别是侦测类奖励 r_d 和攻击类奖励 r_i. 其中, 飞行决策网络使用 r_d, r_i 两者之和进行训练, 攻击决策网络使用获得的攻击类奖励 r_i 进行训练. 具体奖励设计如表 4 所示.

表 5 神经网络超参数

Table 5 Neural network hyperparameters

Parameter	Value	Parameter	Value
Optimizer	Adam	Replay buffer size	3000
Learn rate	3E−4	Discount factor	0.96
Batch size	300	Initial expert sample size	1000

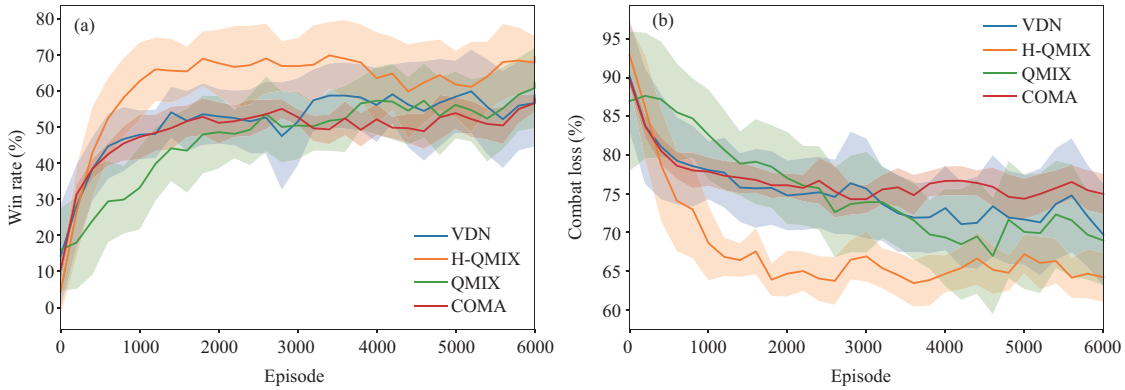

图 4 (网络版彩图) 算法对比图. (a) 胜率对比图; (b) 战损率对比图

Figure 4 (Color online) Comparison of (a) win rate and (b) combat loss

(4) 网络设计. 本文构建的神经网络包含两层并列结构神经网络, 即飞行、攻击决策层网络. 两个网络结构相似, 均采用双隐藏层网络结构, 输入状态在进入网络前需要经过归一化处理. 网络具体超参数如表 5 所示.

4.3 算法对比实验

为了验证本文算法的有效性, 采用上述设计方法构建模型与基线算法在均势对抗环境进行了对比实验. 对比算法为几种常见多智能体对抗算法 (VDN 算法、COMA 算法 [30]、QMIX 算法). 实验从胜率与战损率两个方面进行了对比, 战损率指战斗结束时损失战机数量的占比, 实验为 5 对 5 战斗场景, 共进行十轮训练, 训练对比结果如图 4 所示. 其中, 图 4(a) 为 5 对 5 胜率对比结果, 图 4(b) 为 5 对 5 战损率对比结果, H-QMIX 为本文提出的空战决策方法, 实线为多轮训练平均值, 相应颜色阴影部分为波动范围. 从对比结果可以看到, 本文提出的方法在训练中的胜率及战损率明显优于基线对比算法, 同时在训练过程中收敛速度也高于其他方法. H-QMIX 方法在训练 100 轮之后即开始收敛, 而 QMIX 与 VDN 方法在 300 轮训练之后才逐渐收敛, COMA 方法的收敛速度与 H-QMIX 相近, 但收敛结果较差.

为了测试算法最终性能, 实验在 5 对 5 对战场景下, 采用各算法训练最终模型进行了对抗测试, 共进行 100 轮测试, 每轮测试对局 500 局, 测试结果如表 6 所示. 从表中可以看出, H-QMIX 方法在胜率、战损率方面均优于对比算法, 稳定性也较好.

综合以上实验, 可以看出本文提出的方法在对抗胜率及战损率指标上均优于其他基线对比算法且训练收敛速度快, 性能相对稳定, 验证了算法的有效性.

表 6 算法测试结果

Table 6 Algorithm test results

Algorithm	Average win rate	Win rate std	Average combat loss	Combat loss std
H-QMIX	**0.647**	**0.036**	**0.695**	**0.020**
QMIX	0.588	0.103	0.725	0.055
VDN	0.612	0.032	0.759	0.022
COMA	0.516	0.028	0.809	0.015

表 7 消融实验设置

Table 7 Ablation experiment setting

Algorithm	Hierarchical network	Transformation	Expert learning
H-QMIX	✓	✓	✓
QMIX	×	×	×
H-QMIX-noH	×	✓	✓
H-QMIX-noED	✓	✓	×
H-QMIX-noET	✓	×	✓

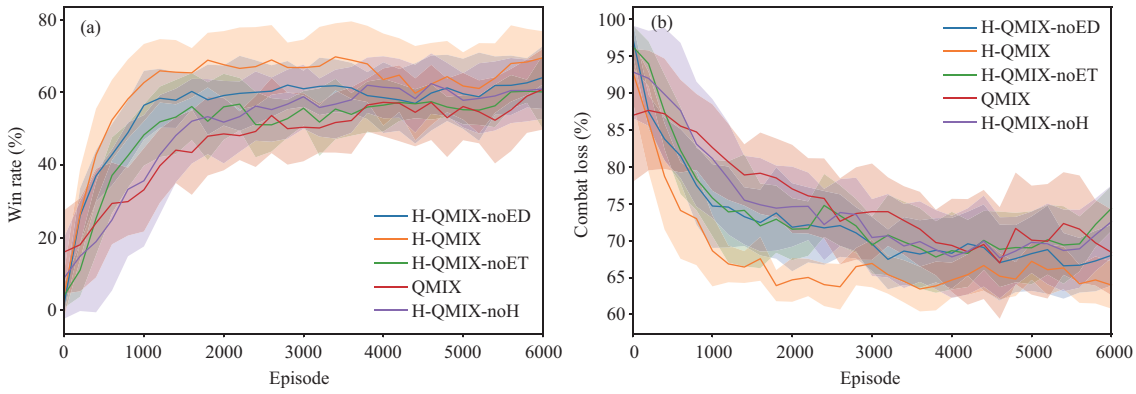

图 5 (网络版彩图) 消融实验结果. (a) 胜率; (b) 战损率

Figure 5 (Color online) Ablation experiment results. (a) Win rate; (b) combat loss

4.4 消融实验

本文提出的算法在 3 个方面对 QMIX 算法进行了改进. 为研究不同机制对算法性能改进的影响, 设计了消融实验, 在 H-QMIX 算法的基础之上单独去除一种改进, 在 5 对 5 对抗环境中比较训练效果. 三种对比算法设置如表 7 所示, 其中 ✓代表包含相应改进机制, × 代表不包含.

图 5 为消融实验算法对比结果, 图 5(a) 和 (b) 分别为消融实验训练过程胜率、战损率对比图. 从图中可以看出, 去除某种改进方法产生的影响有所不同. 从实验结果来看, 在算法稳定性方面, H-QMIX-noH, H-QMIX-noET, H-QMIX-noED 方法均优于 QMIX 方法; 在收敛速度方面, H-QMIX-noH, H-QMIX-noET, H-QMIX-noED 方法训练的模型在训练收敛速度方面均高于 QMIX 方法, 其中 H-QMIX-noET 算法收敛最快, H-QMIX-noH 收敛最慢, 说明经验分解对于算法收敛速度的提升贡献最小; 在胜率方面, H-QMIX-noH, H-QMIX-noET, H-QMIX-noED 算法高于 QMIX 方法, 三种改进方法

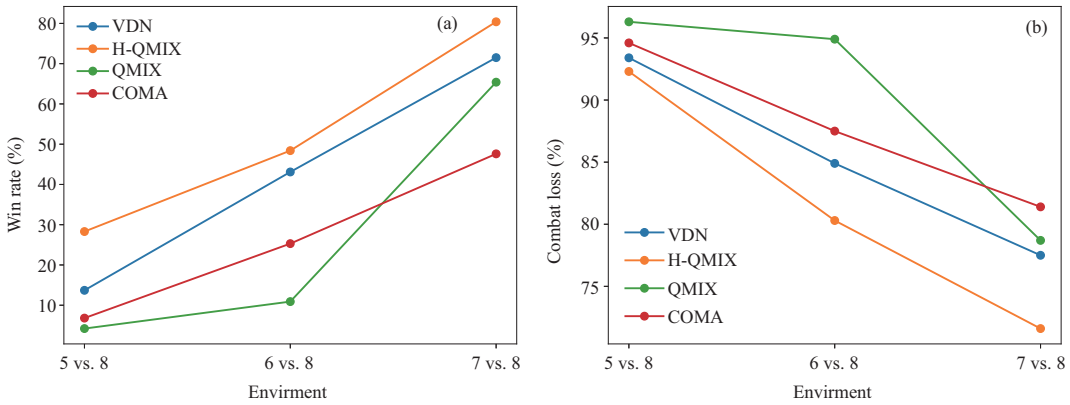

图 6 (网络版彩图) 劣势对抗实验结果. (a) 胜率对比图; (b) 战损率对比图

Figure 6 (Color online) Comparison of inferior countermeasure. (a) Win rate; (b) combat loss

对于模型胜率均有不同程度的提升, 其中 H-QMIX-noH, H-QMIX-noED 方法胜率相近, H-QMIX-noET 略高于 QMIX 算法; 在战损率方面, H-QMIX-noED 最优, H-QMIX-noH 和 H-QMIX-noET 方法稍低, 略优于 QMIX 方法, 说明专家经验学习对于降低战损率贡献最小. 总体而言, 本算法添加的改进在收敛速度、稳定性、胜率、战损率方面在大多数情况下均优于 QMIX 方法, 实验验证了算法各项改进的有效性, 适用于解决多机对抗决策问题.

4.5 劣势对抗测试

在实际对抗中, 敌方战机数量是事先未知的, 可能会出现我方处于劣势的对抗局面, 为测试本文算法在劣势对抗中的有效性, 本文在不同程度劣势对抗情况下 (5 对 8、6 对 8、7 对 8) 测试了算法性能.

图 6 为在不同程度劣势对抗环境下 (5 对 8、6 对 8、7 对 8) 进行算法对比测试的结果, 可以看到 H-QMIX 算法在所有劣势对抗测试下, 胜率与战损率均优于其他算法. 其中, 在 6 对 8 的劣势对抗下本文算法依然能够取得接近 50% 的胜率, 其他算法均低于 50%; 随着劣势程度减小, H-QMIX 的胜率呈现线性增长; QMIX 方法随着劣势程度减小, 性能提升波动较大, 而 H-QMIX 算法则相对稳定. 综上可以看到本文的方法在劣势 (战机数量不小于敌方 75%) 对抗中依然能取得高于 50% 的胜率, 体现了本文算法在劣势对抗中的有效性.

4.6 结果分析

通过分析实验结果数据, 总结多机博弈对抗过程中 H-QMIX 模型涌现出的对抗策略.

(1) 高速机动躲避导弹. 如图 7(a) 所示, 当己方战机感知敌方导弹逼近时, 会进行高速机动, 做出快速转向规避动作 (黑色虚线指示), 减少敌方导弹命中成功率.

(2) 高效攻击. 在训练开始, 决策模型控制的无人机攻击具有随机性, 导致经常进行无效攻击, 出现对抗未结束弹药已耗尽的情况. 随着训练的推进, 己方无人机学会了把握攻击机会, 如图 7(b) 所示, 在最有利的攻击位置发动攻击, 提高攻击成功率.

5 结束语

本文以提高多机空战决策效率为目标, 将分层决策思想与专家经验学习、经验分解变换方法相结合, 设计了基于分层强化学习的多机空战决策方法, 在 5 对 5 对抗环境下进行对比实验; 设计了消融

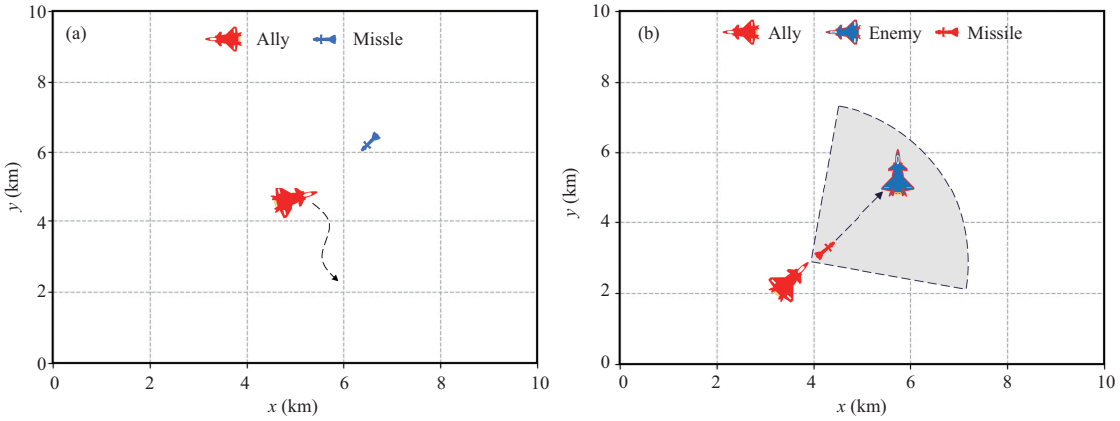

图 7 (网络版彩图) (a) 规避导弹示意图; (b) 高效攻击示意图

Figure 7 (Color online) (a) Missile avoidance; (b) efficient attack

实验, 分析不同算法改进机制对算法性能提升效果以及算法所表现出来的战法与策略; 针对几种劣势对抗场景测试了算法性能并总结了算法在对抗中表现出的行为策略. 实验结果表明, 本文提出的方法提升了空战决策模型训练收敛速度和决策性能, 为下一步设计在更复杂、更真实的多机对抗环境中进行对抗决策的方法提供借鉴.

本文重在探索多机对抗背景下如何提升无人机对抗算法模型训练学习效率及稳定性的方法, 所使用的仿真环境相对简单, 下一步将设计更加贴合实际空战场景的仿真环境, 用于算法的验证实验.

参考文献

1 Reilly M B, Ventre L. Beyond video games: new artificial intelligence beats tactical experts in combat simulation. UC Magazine, 2016. 231–232 [2016-06-27]. https://magazine.uc.edu/editors_picks/recent_features/alpha.html

2 Wang T, Li L, Jiang Q. Analysis on promoting the development of unmanned bee colony capability by "offensive bee colony enabling tactics" project. Tactical Missile Technol, 2020, 1: 33–38, 56 [王彤, 李磊, 蒋琪. "进攻性蜂群使能战术" 项目推进无人蜂群能力发展分析. 战术导弹技术, 2020, 1: 33–38, 56]

3 Linda K. France tests nEURON stealth combat drone with Rafale jets, AWACS Aircraft. Defense World, 2020. 111–112 [2020-02-22]. https://www.defenseworld.net/2020/02/22/france-tests-neuron-stealth-combat-drone-with-rafale-jets-awacs-aircraft.html

4 Yuri S. Russian heavy strike drone Okhotnik makes first flight. Defense Update, 2019. 127–128 [2019-08-19]. https://tass.com/defense/1071784

5 McGrew J S. Real-time maneuvering decisions for autonomous air combat. Dissertation for Ph.D. Degree. Cambridge: Massachusetts Institute of Technology, 2008. 91–104

6 Wu A, Yang R N, Liang X L, et al. Maneuvering decision of UAV in line of sight air combat based on fuzzy reasoning inference. J Nanjing Univ Aeronaut Astronaut, 2021, 53: 898–908 [吴傲, 杨任农, 梁晓龙, 等. 基于模糊推理的无人战斗机视距空战机动决策. 南京航空航天大学学报, 2021, 53: 898–908]

7 Gao F, Chen S, Li M, et al. MaCA: a multi-agent reinforcement learning platform for collective intelligence. In: Proceedings of 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS), Beijing, 2019. 108–111

8 Getz W M, Pachter M. Capturability in a two-target 'game of two cars'. J Guid Control, 1981, 4: 15–21

9 Austin F, Carbone G, Falco M, et al. Game theory for automated maneuvering during air-to-air combat. J Guid Control Dyn, 1990, 13: 1143–1149

10 Xi Z-F, Xu A, Kou Y-X, et al. Decision process of multi-aircraft cooperative air combat maneuver. Syst Eng Electron, 2020, 42: 381–389 [奚之飞, 徐安, 寇英信, 等. 多机协同空战机动决策流程. 系统工程与电子技术, 2020, 42: 381–389]

11 Zhou W Q, Zhu J H, Kuang M C. An unmanned air combat system based on swarm intelligence. Sci Sin Inform, 2020, 50: 363–374 [周文卿, 朱纪洪, 匡敏驰. 一种基于群体智能的无人空战系统. 中国科学: 信息科学, 2020, 50: 363–374]

12 Yan F, Zhu X P, Zhou Z, et al. Real-time task allocation for a heterogeneous multi-UAV simultaneous attack. Sci Sin Inform, 2019, 49: 555–569 [严飞, 祝小平, 周洲, 等. 考虑同时攻击约束的多异构无人机实时任务分配. 中国科学: 信息科学, 2019, 49: 555–569]

13 Fu X W, Wang H, Xu Z. Research on cooperative pursuit strategy for multi-UAVs based on DE-MADDPG algorithm. Acta Aeronautica Astronautica Sin, 2021, 42: 325311 [符小卫, 王辉, 徐哲. 基于 DE-MADDPG 的多无人机协同追捕策略研究. 航空学报, 2021, 42: 325311]

14 Sun Z, Piao H, Yang Z, et al. Multi-agent hierarchical policy gradient for Air Combat Tactics emergence via self-play. Eng Appl Artif Intell, 2021, 98: 104112

15 Shi W, Feng Y H, Cheng G Q, et al. Research on multi-aircraft cooperative air combat method based on deep reinforcement learning. Acta Autom Sin, 2021, 47: 1610–1623 [施伟, 冯旸赫, 程光权, 等. 基于深度强化学习的多机协同空战方法研究. 自动化学报, 2021, 47: 1610–1623]

16 Vinyals O, Babuschkin I, Czarnecki W M, et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. Nature, 2019, 575: 350–354

17 Wang X, Song J, Qi P, et al. SCC: an efficient deep reinforcement learning agent mastering the game of StarCraft II. In: Proceedings of the 38th International Conference on Machine Learning, 2021

18 Wang P, Liu D, Chen J. Decision making for autonomous driving via augmented adversarial inverse reinforcement learning. In: Proceedings of International Conference on Robotics and Automation (ICRA), Xi'an, 2021. 1036–1042

19 Xu Y Z, Wu H, You K Y, et al. A selected review of reinforcement learning-based control for autonomous underwater vehicles. Sci Sin Inform, 2020, 50: 1798–1816 [许雅筑, 武辉, 游科友, 等. 强化学习方法在自主水下机器人控制任务中的应用. 中国科学: 信息科学, 2020, 50: 1798–1816]

20 Gao X. Research of motion planning and compliant control for robot manipulators based on imitation learning. Dissertation for Ph.D. Degree. Wuhan: Wuhan University, 2021 [高霄. 基于模仿学习的机械臂运动规划与柔顺控制研究. 博士学位论文. 武汉: 武汉大学, 2021]

21 Sutton R S, Precup D, Singh S. Between MDPs and semi-MDPs: a framework for temporal abstraction in reinforcement learning. Artif Intell, 1999, 112: 181–211

22 Parr R, Russell S. Reinforcement learning with hierarchies of machines. In: Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems, Cambridge, 1998. 1043–1049

23 Dietterich T G. Hierarchical reinforcement learning with the MAXQ value function decomposition. J Artif Intell Res, 2000, 13: 227–303

24 Wang T, Li J, Song H Y, et al. Navigation method for mobile robot based on hierarchical deep reinforcement learning. Control Decision, 2022. doi: 10.13195/j.kzyjc.2021.1013 [王童, 李骜, 宋海荦, 等. 基于分层深度强化学习的移动机器人导航方法. 控制与决策, 2022. doi: 10.13195/j.kzyjc.2021.1013]

25 Yang J C, Igor B, Zha H Y. Hierarchical cooperative multi-agent reinforcement learning with skill discovery. In: Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, 2020. 1566–1574

26 Li C, Xia F, Martín-Martín R, et al. HRL4IN: hierarchical reinforcement learning for interactive navigation with mobile manipulators. In: Proceedings of the Conference on Robot Learning, 2020. 603–616

27 Pang Z J, Liu R Z, Meng Z Y, et al. On reinforcement learning for full-length game of starcraft. In: Proceedings of the AAAI Conference on Artificial Intelligence, Hawaii, 2019. 4691–4698

28 Tabish R, Mikayel S, Christian S, et al. 2018. QMIX: monotonic value function factorisation for deep multi-agent reinforcement learning. In Proceedings of the 35th International Conference on Machine Learning, Stockholm, 2018. 4295–4304

29 Andrychowicz M, Wolski F, Ray A, et al. Hindsight experience replay. In: Proceedings of the Neural Information Processing Systems, Long Beach, 2017. 5048–5058

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

A hierarchical decision-making method for multi-aircraft air combat confrontation

Huan WANG[1], Xu ZHOU[1], Yimin DENG[2] & Xiaofeng LIU[1,3*]

1. *College of IoT Engineering, Hohai University, Changzhou* 213022*, China*;
2. *School of Automation Science and Electrical Engineering, Beihang University, Beijing* 100191*, China*;
3. *Jiangsu Key Laboratory of Special Robot Technology, Changzhou* 213022*, China*
* Corresponding author. E-mail: xfliu@hhu.edu.cn

Abstract In air combat research, tactical decision-making aims to improve the gain of game confrontation and then the attack efficiency of one's own fighter aircraft. Most existing tactical decision-making algorithms are designed based on the rule-based approach, which brings difficulty to designing and solving the optimal solution for the complex environment of multi-aircraft air combat. This paper proposes a hierarchical decision-making multi-aircraft air combat method. First, we draw on the existing human expert experience in the initial stage of training to guide model training; second, we design a hierarchical action decision-making network according to the tactical action types to reduce the action decision space dimensions; and finally, we decompose the training-generated adversarial experience in stages to reduce the strategy learning difficulty. Experiments in a multi-aircraft air combat simulation environment demonstrate that the proposed method shows better performance regarding training convergence and decision-making performance compared with common multi-aircraft air combat decision-making methods.

Keywords multi-aircraft air combat, action decision-making network, game, hierarchical reinforcement learning, decision gain