自动化学报
ACTA AUTOMATICA SINICA

基于行为预测和策略融合的轨道博弈决策方法

王英杰 袁利 黄煌 耿远卓

A Decision Method for Orbital Game Based on Behavior Prediction and Strategy Fusion

WANG Ying-Jie, YUAN Li, HUANG Huang, GENG Yuan-Zhuo

在线阅读 View online: https://doi.org/10.16383/j.aas.c250268

您可能感兴趣的其他文章

基于终端诱导强化学习的航天器轨道追逃博弈

Terminal-guidance Based Reinforcement-learning for Orbital Pursuit-evasion Game of the Spacecraft

自动化学报. 2023, 49(5): 974-984 https://doi.org/10.16383/j.aas.c220204

基于距离信息的追逃策略: 信念状态连续随机博弈

Distance Information Based Pursuit-evasion Strategy: Continuous Stochastic Game With Belief State

自动化学报. 2024, 50(4): 828-840 https://doi.org/10.16383/j.aas.c230018

基于深度强化学习的无人机自主感知规划控制策略

Autonomous Perception-Planning-Control Strategy Based on Deep Reinforcement Learning for Unmanned Aerial Vehicles

自动化学报. 2025, 51(6): 1305-1319 https://doi.org/10.16383/j.aas.c240639

基于微分博弈的追逃问题最优策略设计

Design of Optimal Strategies for the Pursuit-evasion Problem Based on Differential Game

自动化学报. 2021, 47(8): 1840-1854 https://doi.org/10.16383/j.aas.c200979

融合进化算法和深度强化学习的飞行器制导控制一体化

Integrated Guidance and Control of Flight Vehicles by Fusing Evolutionary Algorithms and Deep Reinforcement Learning

自动化学报. 2026, 52(1): 52-64 https://doi.org/10.16383/j.aas.c250278

深度强化学习联合回归目标定位

Union Regression Object Localization Based on Deep Reinforcement Learning

自动化学报. 2023, 49(5): 1089-1098 https://doi.org/10.16383/j.aas.c200045

基于行为预测和策略融合的轨道博弈决策方法

王英杰 [1,2] 袁利 [2,3] 黄煌 [1,2] 耿远卓 [1,2]

摘 要 轨道追逃博弈中逃逸策略的高度未知性与行为多样性, 给追踪策略的泛化能力带来严峻挑战. 深度强化学习虽可提升追踪星的博弈效能, 但当逃逸策略偏离训练分布时, 策略网络易产生次优甚至失效的决策. 为此, 提出一种基于行为预测和策略融合的轨道博弈决策方法. 在训练阶段, 首先采用"预测制导 + 人工势场法"构建多样化逃逸策略集. 随后在传统演员−评论家训练框架的基础上, 通过引入预测网络构建预测器−演员−评论家算法, 针对每类逃逸策略分别训练以获得对应的追踪子策略. 其中预测网络用于估计逃逸星动作, 并通过预测结果与真实动作的相似性衡量子策略与未知逃逸策略的匹配度. 在执行阶段, 策略融合器以逃逸星历史动作与各追踪子策略的预测结果为输入, 动态计算匹配度并选择最优子策略进行博弈决策. 实验结果表明, 预测网络能有效评估追踪子策略对未知逃逸策略的适应性, 策略融合器可显著提升追踪星面对多样化逃逸策略的泛化能力与可靠性.

关键词 轨道追逃博弈; 深度强化学习; 行为预测; 策略融合

引用格式 王英杰, 袁利, 黄煌, 耿远卓. 基于行为预测和策略融合的轨道博弈决策方法. 自动化学报, 2026, **52**(3): 451−462

DOI 10.16383/j.aas.c250268 **CSTR** 32138.14.j.aas.c250268

A Decision Method for Orbital Game Based on Behavior Prediction and Strategy Fusion

WANG Ying-Jie[1,2] YUAN Li[2,3] HUANG Huang[1,2] GENG Yuan-Zhuo[1,2]

Abstract The high uncertainty and behavioral diversity of evasion strategies in the orbital pursuit-evasion game pose significant challenges to the generalization capability of pursuit strategies. Although deep reinforcement learning can enhance the pursuer′s performance, the policy network often produces suboptimal or even invalid decisions when facing evasion strategies that deviate from the training distribution. To address this issue, this paper proposes a decision method for orbital game based on behavior prediction and strategy fusion, named predictor-actor-critic with fusion. During the training phase, a set of diverse evasion strategies is modeled using a prediction-guided approach combined with the artificial potential field method. Based on the traditional actor-critic framework, a predictor-actor-critic algorithm is developed by introducing a prediction network, and a corresponding pursuit sub-policy is trained for each type of evasion strategy. The prediction network estimates the evader′s actions, and the similarity between predicted and actual actions is used to quantify the matching degree between each sub-policy and the unknown evasion strategy. During the execution phase, the fusion module takes the evader′s historical actions and pursuit sub-policies′ prediction outputs as input, dynamically evaluates matching degree, and selects the most appropriate sub-policy for decision-making. Experimental results demonstrate that the prediction network effectively evaluates the adaptability of sub-policy to unknown evasion strategies, and the fusion module significantly enhances the generalization capability and reliability of the pursuer when confronted with diverse evasion strategies.

Keywords orbital pursuit-evasion game; deep reinforcement learning; behavior prediction; strategy fusion

Citation Wang Ying-Jie, Yuan Li, Huang Huang, Geng Yuan-Zhuo. A decision method for orbital game based on behavior prediction and strategy fusion. *Acta Automatica Sinica*, 2026, **52**(3): 451−462

随着航天任务的多样化与轨道资源的日益紧张, 近地轨道空间呈现愈发复杂的态势. 自 2014 年以来, 美国"地球同步轨道空间态势感知计划 (geosynchronous space situational awareness program, GSSAP)"中的卫星频繁对地球同步轨道上的各国航天器实施抵近侦察[1]. 2021 年 7 月与 10 月, 面对主动接近的"星链-1095"和"星链-2305", 中国空间站组合体在地面指控下相继实施紧急避碰控制, 有效规避了潜在碰撞风险[2].

在此类具有强实时性和高不确定性的空间轨道

收稿日期 2025-06-19 录用日期 2025-09-14
Manuscript received June 19, 2025; accepted September 14, 2025
国家自然科学基金 (62303047, U21B6001), 空间智能控制技术全国重点实验室开放基金课题 (2024-CXPT-GF-JJ-012-05) 资助
Supported by National Natural Science Foundation of China (62303047, U21B6001) and National Key Laboratory of Space Intelligent Control (2024-CXPT-GF-JJ-012-05)
本文责任编委 王卓
Recommended by Associate Editor WANG Zhuo
1. 北京控制工程研究所 北京 100094 2. 空间智能控制技术全国重点实验室 北京 100094 3. 中国空间技术研究院 北京 100094
1. Beijing Institute of Control Engineering, Beijing 100094 2. National Key Laboratory of Space Intelligent Control, Beijing 100094 3. China Academy of Space Technology, Beijing 100094

博弈任务中, 传统依赖地面测控指令的“星地大回路”控制模式, 因响应滞后与自主性不足, 难以满足任务需求. 因此, 亟须发展面向空间轨道博弈的自主决策方法, 以提升航天器的智能应对与快速反应能力.

微分对策理论源于双边最优控制理论, 其研究动态环境下多决策者基于各自利益选择最优策略的博弈过程, 可描述受微分方程约束的连续时间系统中的博弈竞争问题, 目前已广泛应用于轨道追逃场景中[3−5]. 该方法通常将航天器推进系统建模成连续小推力模型, 然而对于百公里量级的轨道追逃任务, 其更适合采用具有大推力输出的脉冲推进方式.

航天器可达域[6−8] 是指在给定航天器初始时刻的位置与速度以及最大机动能力约束条件下, 终端时刻其所有可能到达位置的集合. 可达域是机动航天器态势分析的有效工具, 具有出色的可视化效果和较强的可解释性. 然而当航天器采用脉冲推进方式时, 该方法通常只能用于单步分析, 得到单步最优解. 在多轮决策博弈中, 该方法难以从全局目标的角度进行优化求解.

强化学习通过智能体在环境中不断探索产生数据, 根据环境的反馈迭代学习, 进而习得最大化累积回报的最优策略, 目前已在轨道追逃问题中得到初步应用. 针对“一对一”轨道追逃问题, 文献 [9] 充分考虑追逃双方的燃料、推力、决策周期、运动范围等实际约束条件, 把基于 CW (Clohessy Wiltshire) 方程预测的双星终端时刻相对误差引入奖励函数, 有效引导追踪星在指定时刻进入逃逸星的安全接近区. 文献 [10] 提出基于事后经验回放 (hindsight experience replay, HER) 的分层网络结构, 上层网络依据可达域分析进行子目标的制定, 底层网络进行轨道控制. 文献 [11] 在决策步之间进行轨道外推, 并根据外推状态生成奖励信号, 有效提升学习训练的收敛性. 针对带有锥形成像区约束的追逃博弈问题, 文献 [12] 将追逃双方的相对距离和太阳光照角信息统一表征为追踪星成像区相对于逃逸星的最短距离, 有效削减了奖励函数参数. 文献 [13] 依据逃逸星感知范围, 将轨道追逃问题建模为远距离交会段和近距离博弈段: 前者通过遗传算法优化脉冲序列, 后者则基于强化学习训练追踪策略. 针对“多对一”轨道追逃问题, 文献 [14−16] 采用多智能体强化学习方法有效激发追踪星之间的协同行为.

深度强化学习通过学习训练对追踪星策略进行赋能. 然而, 当逃逸策略偏离训练分布时, 追踪星策略网络所生成的决策动作容易出现次优甚至失效的情况, 泛化能力显著下降, 该问题因深度神经网络的黑箱特性而进一步加剧. 由于缺乏对策略内部机制的解释方法, 因此难以有效判断当前追踪策略对未知逃逸策略的适应能力. 为提升深度强化学习策略的可靠性与泛化能力, 近年来研究者开始关注如何引入可解释机制, 以辅助理解和评估策略网络的行为边界与决策合理性. 面向深度强化学习的可解释研究方法大致可分为事后可解释性方法和模型内置可解释性方法[17]: 前者在策略网络训练完成后, 通过分析网络关注的关键特征来溯源其行为[18−19], 可提升用户对模型的信任度. 然而, 该类方法仅能评估特定状态或决策步骤的重要性, 难以在模型实时运行过程中判断当前策略对未知对手行为的适应性. 后者则通过引入白盒自解释模型 (如决策树[20−21]、IF-THEN 规则[22] 和有限状态机[23]) 对深度神经网络进行拟合, 以提升决策模型的透明性与解释能力. 但在处理大规模决策问题时, 传统自解释模型性能往往不及深度网络模型, 且模型规模扩大后, 其结构复杂性 (如决策树的节点数) 也将显著增加, 反而削弱了可解释性.

针对上述问题, 本文提出一种基于行为预测和策略融合的轨道博弈决策方法 PACF (predictor-actor-critic with fusion). 在训练阶段, 基于现有演员−评论家 (actor-critic, AC[24]) 框架, 通过引入预测网络构建预测器−演员−评论家 (predictor-actor-critic, PAC) 算法用于追踪策略训练. 其中, 预测网络实时估计逃逸星的动作行为, 依据预测结果与真实动作的相似性, 量化追踪策略与未知逃逸策略之间的匹配度. 逃逸星策略采用“预测制导 + 人工势场法”建模, 通过调节人工势场 (artificial potential field, APF) 中的引力系数与斥力系数, 构建具备多样行为特征的逃逸策略集. 针对每类逃逸策略, 分别采用 PAC 方法训练得到一个对应的追踪子策略, 构成追踪子策略集合. 在实际任务执行阶段, 追踪星通过策略融合器 (fusion, F) 引入匹配度驱动机制, 依据各子策略对逃逸星历史动作序列的估计准确性动态计算其匹配度, 并据此选择匹配度最高的子策略用于当前博弈决策.

基于上述设计, 本文的主要贡献如下: 1) 匹配度评估方法. 针对深度强化学习可解释性不足的问题, 首次将行为预测与匹配度量化引入轨道追逃博弈中, 通过预测网络 (predictor, P) 实时估计逃逸星动作并计算与真实动作的相似性, 建立可在线反映策略适应性的指标, 在不降低神经网络性能的前提下显著提升策略的可解释性与可靠性. 2) 匹配度驱动的策略融合方法. 针对逃逸策略的多样化特点, 提出基于匹配度驱动的策略融合器 (F), 在执行阶

段动态选择最契合当前逃逸行为的子策略, 显著提升追踪策略面对未知逃逸对手的泛化能力.

1　问题描述与建模

本文研究场景为三颗同构追踪星在百公里量级范围内协同拦截一颗逃逸星. 假定逃逸星初始轨道为地球同步轨道 (geosynchronous orbit, GSO), 以逃逸星初始位置为原点, 建立相对轨道坐标系, $\boldsymbol{z}_o$ 轴指向地心方向, $\boldsymbol{y}_o$ 轴沿轨道面负法线方向, $\boldsymbol{x}_o = \boldsymbol{y}_o \times \boldsymbol{z}_o$. 对于航天器 $i \in \{P_1,\ P_2,\ P_3,\ E\}$, P_j 代表第 j 颗追踪星, E 代表逃逸星, 其位置表示为 $\boldsymbol{r}_i = [x_i,\ y_i,\ z_i]^{\mathrm{T}}$, 速度表示为 $\boldsymbol{v}_i = [\dot{x}_i,\ \dot{y}_i,\ \dot{z}_i]^{\mathrm{T}}$, 状态表示为 $\boldsymbol{X}_i = [x_i,\ y_i,\ z_i,\ \dot{x}_i,\ \dot{y}_i,\ \dot{z}_i]^{\mathrm{T}}$.

假定三颗追踪星与逃逸星均采用固定且一致的周期 T 进行脉冲推力控制. 航天器在每个脉冲机动时刻 t_k 可获得瞬时脉冲速度增量:

$$\boldsymbol{X}_i(t_k^+) = \boldsymbol{X}_i(t_k^-) + \begin{bmatrix} \boldsymbol{0}_{3\times 1} \\ \boldsymbol{a}_i(t_k) \end{bmatrix} \tag{1}$$

式中 $\boldsymbol{a}_i(t_k) = [a_i^x(t_k),\ a_i^y(t_k),\ a_i^z(t_k)]^{\mathrm{T}}$ 为航天器在 t_k 时刻的三轴脉冲控制量, $\boldsymbol{X}_i(t_k^-)$、$\boldsymbol{X}_i(t_k^+)$ 分别表示脉冲前、后的状态.

在脉冲推力控制过程中, 所有航天器均受到单次最大脉冲控制约束, 即:

$$\left\|\boldsymbol{a}_{P_j}(t_k)\right\| \le \Delta V_P,\ \forall j \in \{1,\ 2,\ 3\},\quad \forall k \in \{0,\ 1,\ \cdots,\ K-1\} \tag{2}$$

$$\left\|\boldsymbol{a}_E(t_k)\right\| \le \Delta V_E,\ \forall k \in \{0,\ 1,\ \cdots,\ K-1\} \tag{3}$$

式中 K 为追踪星和逃逸星的脉冲机动总次数; $\boldsymbol{a}_{P_j}(t_k)$ 和 $\boldsymbol{a}_E(t_k)$ 分别表示 t_k 时刻第 j 颗追踪星和逃逸星的脉冲控制量; ΔV_P 和 ΔV_E 分别代表追踪星和逃逸星的单步最大脉冲控制约束. 在本文所研究的场景中, 逃逸星具备更强的机动能力, 即 $\Delta V_E > \Delta V_P$. 因此, 三颗追踪星需通过协同机动拦截逃逸星.

在两次脉冲间隔期间, 航天器处于无控漂移状态. 由于各航天器相对于参考点的距离远小于 GSO 轨道半径, 因此可采用线性化的 CW 相对运动方程[14] 对无控漂移过程进行建模, 其状态转移关系如下:

$$\boldsymbol{X}_i(t_{k+1}^-) = \begin{bmatrix} \boldsymbol{\Phi}_{rr} & \boldsymbol{\Phi}_{rv} \\ \boldsymbol{\Phi}_{vr} & \boldsymbol{\Phi}_{vv} \end{bmatrix} \boldsymbol{X}_i(t_k^+) \tag{4}$$

式中相邻两次脉冲机动时刻满足 $T = t_{k+1}^- - t_k^+$; $\boldsymbol{\Phi}_{rr}$、$\boldsymbol{\Phi}_{rv}$、$\boldsymbol{\Phi}_{vr}$ 和 $\boldsymbol{\Phi}_{vv}$ 为 CW 方程中的状态转移子矩阵, 均为 3×3 维度的矩阵块.

在轨道追逃博弈任务中, 追踪星与逃逸星分别优化各自的脉冲控制策略, 以实现拦截或逃逸的既定目标. 当任一追踪星抵近逃逸星安全距离 r_c 时, 判定追踪任务成功. 追踪星的终端目标集定义为:

$$\Lambda = \left\{ (\boldsymbol{X}_P(t),\ \boldsymbol{X}_E(t)) \,\middle|\, \min_{j\in\{1,\ 2,\ 3\}} \left\|\boldsymbol{r}_{P_j}(t) - \boldsymbol{r}_E(t)\right\| \le r_c \right\} \tag{5}$$

其中 $\boldsymbol{X}_P(t) = \{\boldsymbol{X}_{P_1}(t),\ \boldsymbol{X}_{P_2}(t),\ \boldsymbol{X}_{P_3}(t)\}$; $\boldsymbol{X}_{P_j}$ 和 $\boldsymbol{X}_E$ 分别表示第 j 颗追踪星和逃逸星的状态向量; $\boldsymbol{r}_{P_j}$ 和 $\boldsymbol{r}_E$ 分别表示第 j 颗追踪星和逃逸星的位置向量.

当航天器状态首次进入上述终端目标集时, 追逃博弈任务结束, 定义终端时刻为:

$$t_f := \min\left\{ t \in \mathbf{R}^+ \,|\, (\boldsymbol{X}_P(t),\ \boldsymbol{X}_E(t)) \in \Lambda \right\} \tag{6}$$

考虑任务实际性, 追逃博弈不应无限进行. 设定终端时刻最大上限 $t_{\max} = KT$, 若 $t_f > t_{\max}$, 则视为逃逸星成功脱离追踪.

在轨道博弈问题中, 追踪星的目标为尽可能在短时间内抵近逃逸星安全距离; 相反, 逃逸星的目标为尽可能延长终端时间, 避免被拦截. 因此, 双方的目标函数可由下式表示:

$$\begin{cases} \min\limits_{\boldsymbol{a}_P} \max\limits_{\boldsymbol{a}_E} t_f = f\left(\boldsymbol{X}_P(t_0),\ \boldsymbol{X}_E(t_0),\ \boldsymbol{a}_P,\ \boldsymbol{a}_E\right) \\ \boldsymbol{a}_P = \{\boldsymbol{a}_{P_1},\ \boldsymbol{a}_{P_2},\ \boldsymbol{a}_{P_3}\} \\ \boldsymbol{a}_{P_j} = \left\{\boldsymbol{a}_{P_j}(t_0),\ \boldsymbol{a}_{P_j}(t_1),\ \cdots,\ \boldsymbol{a}_{P_j}(t_f)\right\} \\ \boldsymbol{a}_E = \{\boldsymbol{a}_E(t_0),\ \boldsymbol{a}_E(t_1),\ \cdots,\ \boldsymbol{a}_E(t_f)\} \end{cases} \tag{7}$$

式中 $\boldsymbol{a}_{P_j}$ 和 $\boldsymbol{a}_E$ 分别表示第 j 颗追踪星和逃逸星的脉冲控制序列; 函数 $f(\cdot)$ 表示终端时间计算函数.

相较于面内机动, 面外机动通常会带来更高的速度增量消耗. 为简化问题分析并降低计算复杂度, 文献 [11, 15] 将轨道追逃问题建模为二维轨道面内场景. 基于此, 本文主要聚焦于面内机动策略研究.

2　基于 PAC 算法的追踪子策略训练方法

针对轨道追逃任务中目标行为具有未知性与多样性的挑战, 本文提出一种基于行为预测和策略融合的决策方法 PACF, 其结构如图 1 所示. 在训练阶段, 采用 PAC 算法开展博弈训练, 构建追踪子策略集合 $\Pi = \{\mathcal{P}_n\}_{n=1}^N$. 在执行阶段, 当面对未知逃逸策略时, 策略融合器 (F) 根据各子策略的匹配度动态选择最优子策略用于博弈决策.

2.1　PAC 算法设计

本文在现有 AC 框架基础上, 提出 PAC 算法. 该算法在价值网络 $\hat{Q}_{\boldsymbol{\phi}}$ 和策略网络 $\pi_{\boldsymbol{\theta}}$ 之外, 引入预

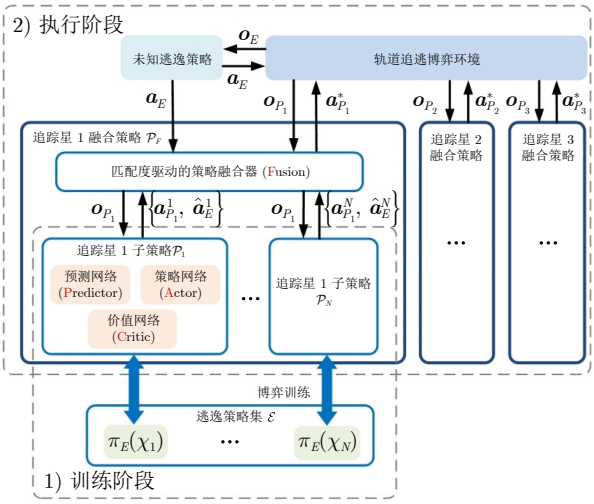


图 1 基于行为预测和策略融合的轨道博弈决策方法

Fig.1 A decision method for orbital game based on behavior prediction and strategy fusion

测网络 $P_{\boldsymbol{\psi}}$, 用于实时估计逃逸星的动作量 $\hat{\boldsymbol{a}}_E$, 并通过比较预测量 $\hat{\boldsymbol{a}}_E$ 和真实动作量 $\boldsymbol{a}_E$ 的相似性, 从而评估追踪策略与未知逃逸策略的匹配度. 上述三类网络均采用多层前馈神经网络进行建模, 其参数在训练初始阶段随机初始化, 并在离线训练阶段依据损失函数 L 完成优化更新.

第 j 颗追踪星的观测量 $\boldsymbol{o}_{P_j}$ 包含所有航天器的状态向量, 具体形式为 $\boldsymbol{o}_{P_j}=[\boldsymbol{X}_{P_j}^{\mathrm{T}}, \boldsymbol{X}_E^{\mathrm{T}}, \boldsymbol{X}_{P_1}^{\mathrm{T}}, \cdots, \boldsymbol{X}_{P_J}^{\mathrm{T}}]^{\mathrm{T}}$, 其中 J 代表追踪星总数. 逃逸星的观测量 $\boldsymbol{o}_E$ 同样包含所有航天器的状态向量. 第 j 颗追踪星的预测网络 $P_{\boldsymbol{\psi}_j}$ 根据观测量 $\boldsymbol{o}_{P_j}$ 预测逃逸星动作 $\hat{\boldsymbol{a}}_E$. 相应地, 第 j 颗追踪星的策略网络 $\pi_{\boldsymbol{\theta}_j}$ 输入为动作估计量 $\hat{\boldsymbol{a}}_E$ 和观测量 $\boldsymbol{o}_{P_j}$, 输出为自身的三轴脉冲控制量 $\boldsymbol{a}_{P_j}=[a_{P_j}^x, a_{P_j}^y, a_{P_j}^z]^{\mathrm{T}}$, 实现分布式决策执行. 价值网络 $\hat{Q}_{\boldsymbol{\phi}_j}$ 采用集中式架构, 输入包括自身观测量 $\boldsymbol{o}_{P_j}$ 与所有航天器的动作量, 最终输出动作–价值函数估计.

本文采用多智能体双延迟深度确定性策略梯度 (multi-agent twin delayed deep deterministic policy gradient, MATD3) 算法进行学习训练. 相较于深度确定性策略梯度 (deep deterministic policy gradient, DDPG) 算法[25], 双延迟深度确定性策略梯度 (twin delayed deep deterministic policy gradient, TD3) 算法[26] 额外引入了一组价值网络, 以减少动作–价值函数估计中的高估偏差, 从而提升训练的稳定性. 在此基础上, 本文进一步引入集中式训练、分布式执行 (centralized training and decentralized execution, CTDE) 框架, 构建 MATD3 算法.

价值网络的损失函数设计为动作–价值函数估计与其期望值的均方误差:

$$\begin{cases} L\left(\boldsymbol{\phi}_j^1\right)=\mathrm{E}_{\boldsymbol{o},\,\boldsymbol{a}}\left[\left(\hat{Q}_1-y_j\right)^2\right] \\ L\left(\boldsymbol{\phi}_j^2\right)=\mathrm{E}_{\boldsymbol{o},\,\boldsymbol{a}}\left[\left(\hat{Q}_2-y_j\right)^2\right] \\ \hat{Q}_1=\hat{Q}_{\boldsymbol{\phi}_j^1}\left(\boldsymbol{o}_{P_j},\,\boldsymbol{a}_{P_j},\,\boldsymbol{a}_{P_1},\,\cdots,\,\boldsymbol{a}_{P_J}\right) \\ \hat{Q}_2=\hat{Q}_{\boldsymbol{\phi}_j^2}\left(\boldsymbol{o}_{P_j},\,\boldsymbol{a}_{P_j},\,\boldsymbol{a}_{P_1},\,\cdots,\,\boldsymbol{a}_{P_J}\right) \\ y_j=r_j+\gamma\min\limits_{l=1,\,2}\hat{Q}_{\boldsymbol{\phi}_j^{l-}}\left(\boldsymbol{o}'_{P_j},\,\boldsymbol{a}'_{P_j},\,\boldsymbol{a}'_{P_1},\,\cdots,\,\boldsymbol{a}'_{P_J}\right) \\ \boldsymbol{a}'_{P_k}=\pi_{\boldsymbol{\theta}_k^-}\left(\boldsymbol{o}'_{P_k},\,\hat{\boldsymbol{a}}'_E\right) \\ \hat{\boldsymbol{a}}'_E=P_{\boldsymbol{\psi}_k}\left(\boldsymbol{o}'_{P_k}\right) \end{cases} \tag{8}$$

式中 $\boldsymbol{\phi}_j^1$ 和 $\boldsymbol{\phi}_j^2$ 为价值网络参数, $\boldsymbol{\phi}_j^{1-}$ 和 $\boldsymbol{\phi}_j^{2-}$ 为目标价值网络参数; $\hat{Q}_1$ 和 $\hat{Q}_2$ 分别为两个价值网络对动作–价值函数的估计值; y_j 是动作–价值函数期望值, 通过贝尔曼方程求解; r_j 为瞬时奖励; γ 为衰减率; $\boldsymbol{a}'_{P_k}$ 为目标策略网络基于下一时刻观测量 $\boldsymbol{o}'_{P_k}$ 和逃逸星动作预测 $\boldsymbol{a}'_E$ 所输出的动作量.

策略网络的目标为最大化累积回报, 其损失函数设计为动作–价值函数估计的相反数:

$$\begin{cases} L(\boldsymbol{\theta}_j)=\mathrm{E}_{\boldsymbol{o},\,\boldsymbol{a}}\left[-\hat{Q}_{\boldsymbol{\phi}_j^1}\left(\boldsymbol{o}_{P_j},\,\bar{\boldsymbol{a}}_{P_j},\,\boldsymbol{a}_{P_1},\,\cdots,\,\boldsymbol{a}_{P_J}\right)\right] \\ \bar{\boldsymbol{a}}_{P_j}=\pi_{\boldsymbol{\theta}_j}\left(\boldsymbol{o}_{P_j},\,\hat{\boldsymbol{a}}_E\right) \\ \hat{\boldsymbol{a}}_E=P_{\boldsymbol{\psi}_j}\left(\boldsymbol{o}_{P_j}\right) \end{cases} \tag{9}$$

式中 $\boldsymbol{\theta}_j$ 为策略网络参数; $\bar{\boldsymbol{a}}_{P_j}$ 为当前策略网络基于观测量 $\boldsymbol{o}_{P_j}$ 和预测量 $\hat{\boldsymbol{a}}_E$ 生成的新动作量.

目标策略网络和目标价值网络均采用软更新方式:

$$\begin{cases} \boldsymbol{\theta}_j^-=\tau\times\boldsymbol{\theta}_j+(1-\tau)\times\boldsymbol{\theta}_j^- \\ \boldsymbol{\phi}_j^{1-}=\tau\times\boldsymbol{\phi}_j^1+(1-\tau)\times\boldsymbol{\phi}_j^{1-} \\ \boldsymbol{\phi}_j^{2-}=\tau\times\boldsymbol{\phi}_j^2+(1-\tau)\times\boldsymbol{\phi}_j^{2-} \end{cases} \tag{10}$$

式中 $\tau\in(0,\,1)$ 为软更新速率, $\boldsymbol{\theta}_j^-$ 为目标策略网络参数.

预测网络的损失函数设计为逃逸星动作估计量 $\hat{\boldsymbol{a}}_E$ 与其真实动作量 $\boldsymbol{a}_E$ 的均方误差:

$$\begin{cases} L\left(\boldsymbol{\psi}_j\right)=\mathrm{E}_{\boldsymbol{o},\,\boldsymbol{a}}\left[\left(\hat{\boldsymbol{a}}_E-\boldsymbol{a}_E\right)^2\right] \\ \hat{\boldsymbol{a}}_E=P_{\boldsymbol{\psi}_j}\left(\boldsymbol{o}_{P_j}\right) \end{cases} \tag{11}$$

式中 $\boldsymbol{\psi}_j$ 为预测网络参数.

PAC 算法采用离线策略训练模式. 在训练阶

段, 智能体与环境交互生成经验数据 $\{\boldsymbol{o},\ \boldsymbol{a},\ r,\ \boldsymbol{o}'\}$, 并将其存储于经验回放池中. 每次网络更新时, 从经验回放池中随机采样进行学习训练. 为保证经验回放池中数据的多样性, 训练过程中引入动作噪声以增强智能体的探索能力, 防止策略过早陷入局部最优, 从而提升训练的稳定性和泛化性能.

为提升追踪策略的泛化性, 三颗同构追踪星采用结构与权值参数相同的神经网络. 每一颗追踪星仅需调整网络输入顺序, 即可获得自身动作量.

网络训练完成后, 可在轨道追逃博弈过程中实时比较逃逸星动作的预测值 $\hat{\boldsymbol{a}}_E$ 与真实动作量 $\boldsymbol{a}_E$ 之间的相似性, 计算追踪策略与未知逃逸策略之间的单步匹配度. 匹配度越高, 说明追踪策略对未知逃逸策略的适应性越强.

具体而言, 首先计算 $\hat{\boldsymbol{a}}_E$ 与 $\boldsymbol{a}_E$ 两个向量之间的余弦相似度 $S_{\cos} \in [-1,\ 1]$, 用于衡量动作方向的一致性; 随后计算两向量在模长上的差异度 $\Delta l \in [0,\ 1]$, 以刻画动作幅度的接近程度; 最终融合二者, 得到归一化的单步匹配度指标 $M \in [0,\ 1]$. 相关计算公式如下:

$$S_{\cos}\left(\hat{\boldsymbol{a}}_E,\ \boldsymbol{a}_E\right) = \frac{\hat{\boldsymbol{a}}_E \cdot \boldsymbol{a}_E}{\|\hat{\boldsymbol{a}}_E\| \cdot \|\boldsymbol{a}_E\|} \tag{12}$$

$$\Delta l\left(\hat{\boldsymbol{a}}_E,\ \boldsymbol{a}_E\right) = \frac{\left|\|\hat{\boldsymbol{a}}_E\| - \|\boldsymbol{a}_E\|\right|}{\|\hat{\boldsymbol{a}}_E\| + \|\boldsymbol{a}_E\|} \tag{13}$$

$$M = \frac{S_{\cos} + 1}{2} \cdot (1 - \Delta l) \tag{14}$$

上述匹配度评估方法需要获取逃逸星当前时刻的真实动作量 $\boldsymbol{a}_E$, 然而在实际任务中该信息通常难以直接获得. 因此本文在第 3 节提出基于逃逸星历史动作信息的替代方案.

为评估追踪策略对未知逃逸策略的适应性, 本文基于预测网络输出 $\hat{\boldsymbol{a}}_E$ 与真实动作 $\boldsymbol{a}_E$ 之间的相似性构建了匹配度评估方法. 该方法符合深度学习中以监督信号指导策略学习的一般原理. 由于预测网络与策略网络采用相同的数据集进行训练, 理论上在面对未知逃逸策略时二者应具有相近的泛化能力. 当预测网络能够准确估计当前逃逸策略的动作指令时, 说明其在该类逃逸策略上的拟合能力较强, 从而可以推断出策略网络生成的动作具有较高的适应性; 反之, 若预测网络的输出与真实动作存在较大偏差, 则说明其对当前逃逸策略的拟合能力较弱, 相应地, 策略网络在该类逃逸策略上的响应效果可能不佳.

2.2　奖励函数设计

在多航天器轨道追逃场景中, 追踪星需维持恰当的编队构型, 以实现对机动能力更强的逃逸星的有效拦截. 为此, 本文设计了一种综合个体收益与集体收益的追踪星奖励函数, 其由三部分组成:

1) 距离奖励 $R^j_{\text{dist}}(t_k)$. 该奖励用于引导各追踪星通过协同机动缩短其与逃逸星的相对距离之和, 定义如下:

$$R^j_{\text{dist}}(t_k) = -\left(D_1(t_k) + D_2(t_k) + D_3(t_k)\right) \tag{15}$$

式中 $D_j(t_k)$ 表示第 j 颗追踪星与逃逸星的相对距离.

2) 燃料消耗奖励 $R^j_{\text{fuel}}(t_k)$. 该奖励用于引导追踪星尽可能减少自身脉冲消耗, 以实现节约推进剂的目的, 定义如下:

$$R^j_{\text{fuel}}(t_k) = -\left\|\boldsymbol{a}_{P_j}(t_k)\right\| \tag{16}$$

其中 $\boldsymbol{a}_{P_j}(t_k) = \left[a^x_{P_j}(t_k),\ a^y_{P_j}(t_k),\ a^z_{P_j}(t_k)\right]^{\mathrm{T}}$ 为第 j 颗追踪星的脉冲控制量.

3) 终端奖励 $R^j_{\text{ter}}(t_k)$. 当任一追踪星抵近逃逸星安全距离, 为鼓励协同完成任务, 给予所有追踪星一个全额正向奖励:

$$R^j_{\text{ter}}(t_k) = \begin{cases} +C, & \min\limits_{l=1,\,2,\,3} D_l(t_k) \le r_c \\ 0, & \text{其他} \end{cases} \tag{17}$$

式中常数 $C > 0$ 表示固定的终端奖励值.

最终, 第 j 颗追踪星的综合奖励由上述三项加权求和得到:

$$R_j(t) = \alpha_1 R^j_{\text{dist}}(t) + \alpha_2 R^j_{\text{fuel}}(t) + \alpha_3 R^j_{\text{ter}}(t) \tag{18}$$

式中 α_1、α_2 和 α_3 分别为对应奖励项的比例系数, 用于调控各奖励在总收益中的贡献比例.

2.3　逃逸策略建模

本文采用“预测制导 + 人工势场法”建模逃逸策略, 作为追踪星策略训练的陪练对象. 逃逸星在对追踪星实施规避的同时, 还需尽可能保持在其初始轨道附近. 为此, 设定逃逸星初始轨道位置为目标点, 并将所有追踪星视为需要规避的动态障碍物.

APF 方法已广泛应用于交会对接[27]、编队飞行[28−29] 等航天器轨迹规划任务中. 该方法通过在目标位置附近构建引力场, 同时在障碍物周围构建斥力场, 从而引导智能体趋近目标、避开障碍.

对于目标点 $\boldsymbol{q}_f$, 引力场对智能体当前位置 $\boldsymbol{q}$ 的引力势能函数定义如下:

$$U_{\text{att}}(\boldsymbol{q}) = \begin{cases} \dfrac{1}{2}\xi\|\boldsymbol{q} - \boldsymbol{q}_f\|^2, & \|\boldsymbol{q} - \boldsymbol{q}_f\| \le d_g \\ d_g\xi\,\|\boldsymbol{q} - \boldsymbol{q}_f\| - \dfrac{1}{2}\xi d_g^2, & \|\boldsymbol{q} - \boldsymbol{q}_f\| > d_g \end{cases} \tag{19}$$

式中 ξ 为引力系数, d_g 为引力场分段阈值.

为避免与障碍物发生碰撞, 在每一个障碍物 $\mathcal{O}_j$ 的周围构建斥力场, 其势能函数为:

$$U_{\text{rep}}^{j}(\boldsymbol{q})=\begin{cases}\dfrac{1}{2}\eta\left(\dfrac{1}{\rho_j(\boldsymbol{q})}-\dfrac{1}{\rho_0}\right)^2, & \rho_j(\boldsymbol{q})\le\rho_0\\ 0, & \rho_j(\boldsymbol{q})>\rho_0\end{cases}\tag{20}$$

式中 $\rho_j(\boldsymbol{q})$ 表示智能体与障碍物 $\mathcal{O}_j$ 的相对距离, η 为斥力系数, ρ_0 为斥力场影响半径.

总势能函数由引力场与所有斥力场叠加而成:

$$U(\boldsymbol{q})=U_{\text{att}}(\boldsymbol{q})+\sum_{j=1}^{N_{\text{obs}}}U_{\text{rep}}^{j}(\boldsymbol{q})\tag{21}$$

式中 N_{obs} 表示障碍物总数.

由势能场产生的人工势场力为:

$$\boldsymbol{F}(\boldsymbol{q})=-\nabla U(\boldsymbol{q})\tag{22}$$

智能体沿人工势场力方向移动, 即可实现向目标点的无碰撞运动.

本文采用"预测制导 + 人工势场法"建模逃逸策略, 其速度增量的生成过程如下:

1) 根据当前时刻 t_k 各航天器的位置 $\boldsymbol{r}_i(t_k^-)$ 与速度 $\boldsymbol{v}_i(t_k^-)$, 采用 CW 方程预测下一决策时刻的无控漂移位置 $\hat{\boldsymbol{r}}_i(t_{k+1})$:

$$\hat{\boldsymbol{r}}_i(t_{k+1})=\boldsymbol{\Phi}_{rr}\boldsymbol{r}_i(t_k^-)+\boldsymbol{\Phi}_{rv}\boldsymbol{v}_i(t_k^-)\tag{23}$$

2) 将逃逸星初始轨道位置 $\boldsymbol{r}_E(t_0)$ 设为目标点, 将所有追踪星预测位置 $\hat{\boldsymbol{r}}_{P_j}(t_{k+1})$ 设为障碍物, 依次构建引力场与斥力场. 逃逸星根据自身预测位置 $\hat{\boldsymbol{r}}_E(t_{k+1})$, 利用式 (22) 计算 t_{k+1} 时刻的期望机动方向 $\Delta\vec{\boldsymbol{r}}_E(t_{k+1})$.

3) 根据 CW 方程求解 t_k 时刻的期望速度增量方向 $\vec{\boldsymbol{a}}_E(t_k)$:

$$\vec{\boldsymbol{a}}_E(t_k)=\boldsymbol{\Phi}_{rv}^{-1}\Delta\vec{\boldsymbol{r}}_E(t_{k+1})\tag{24}$$

4) 逃逸星沿期望方向 $\vec{\boldsymbol{a}}_E(t_k)$ 执行最大幅值速度脉冲, 其最终速度脉冲 $\boldsymbol{a}_E(t_k)$ 可表示为:

$$\boldsymbol{a}_E(t_k)=\Delta V_E\frac{\vec{\boldsymbol{a}}_E(t_k)}{\|\vec{\boldsymbol{a}}_E(t_k)\|}\tag{25}$$

本文定义参数比值 $\chi=\eta/\xi$ 为规避强度因子, 用于量化逃逸策略的行为倾向. 该因子越小, 说明逃逸星越偏向于回归初始轨道位置, 表现出更为保守的策略特征; 反之, χ 越大, 表示逃逸星对规避追踪星的权重越高, 规避行为越激进. 当引力系数 ξ 设置为 0, $\chi\to\infty$, 逃逸星在此情况下将不再考虑回归初始轨道位置, 仅专注于规避拦截.

轨道博弈决策方法 PACF 的结构如图 1 所示. 在训练阶段, 首先通过调节规避强度因子 χ, 构建具备差异化行为特征的逃逸策略集 $\mathcal{E}=\{\pi_E(\chi_n)\}_{n=1}^N$. 然后基于 PAC 算法分别与逃逸策略 $\pi_E(\chi_n)$ 开展博弈训练, 得到追踪策略 $\mathcal{P}_n$, 进而构成追踪子策略集 $\Pi=\{\mathcal{P}_n\}_{n=1}^N$. 其中, 每个子策略 $\mathcal{P}_n$ 能针对性地应对逃逸策略 $\pi_E(\chi_n)$, 以实现最优拦截效果.

3 匹配度驱动的追踪策略融合方法

为提升追踪星面对多样化逃逸策略的泛化能力, 本文在追踪子策略集 $\Pi=\{\mathcal{P}_n\}_{n=1}^N$ 的基础上构建匹配度驱动的策略融合器. 追踪子策略集 Π 与融合器共同构成执行阶段的融合策略 $\mathcal{P}_F$, 其结构如图 1 所示. 以追踪星 P_1 为例, 融合策略 $\mathcal{P}_F$ 的执行过程如下: 各追踪子策略 $\mathcal{P}_n$ 的预测网络 $P_{\boldsymbol{\psi}_n}$ 与策略网络 $\pi_{\boldsymbol{\theta}_n}$ 根据感知输入 $\boldsymbol{o}_{P_1}$, 分别输出逃逸动作估计 $\hat{\boldsymbol{a}}_E^n$ 和追踪动作 $\boldsymbol{a}_{P_1}^n$, 并将结果反馈至策略融合器. 策略融合器通过计算预测结果 $\hat{\boldsymbol{a}}_E^n$ 与真实动作 $\boldsymbol{a}_E$ 之间的匹配度, 评估各子策略对当前未知逃逸策略的适应性. 最终融合器选择匹配度最高的子策略 $\mathcal{P}^*$, 执行其输出的动作 $\boldsymbol{a}_{P_1}^*$.

由于追踪星难以获得逃逸星在当前时刻 t_k 的真实动作, 因而无法直接计算该时刻的单步匹配度 $M_n(t_k)$. 为此, 本文引入基于历史信息的平均匹配度 $\bar{M}_n(t_k)$ 作为替代指标, 用于评估子策略 $\mathcal{P}_n$ 对未知逃逸策略的适应性.

本文采用滑动加权平均机制对匹配度进行动态更新, 具体如下: 初始化阶段设定所有子策略的平均匹配度为常数 $\bar{M}_0$. 之后第 n 个子策略的平均匹配度 $\bar{M}_n(t_k)$ 按照如下递推公式更新:

$$\begin{cases}\bar{M}_n(t_0)=\bar{M}_0\\ \bar{M}_n(t_k)=\lambda\bar{M}_n(t_{k-1})+(1-\lambda)M_n(t_{k-1})\\ M_n(t_{k-1})=M\left(\boldsymbol{a}_E(t_{k-1}),\ \hat{\boldsymbol{a}}_E^n(t_{k-1})\right)\end{cases}\tag{26}$$

式中 $\lambda\in(0,1)$ 为平滑系数, 用于在历史统计信息与最新预测结果之间进行平衡. 当 λ 较大时, 匹配度对单步预测误差的敏感性降低, 但响应速度变慢; 当 λ 较小时, 响应速度加快, 但对预测网络的波动更敏感. 函数 $M(\cdot,\cdot)$ 表示单步匹配度计算公式, 具体定义见式 (14). 逃逸星上一时刻的真实动作 $\boldsymbol{a}_E(t_{k-1})$ 可利用当前时刻和上一时刻的状态信息进行反解, 计算如下:

$$\begin{cases}\hat{\boldsymbol{r}}_E(t_k^-)=\boldsymbol{\Phi}_{rr}\boldsymbol{r}_E(t_{k-1}^-)+\boldsymbol{\Phi}_{rv}\boldsymbol{v}_E(t_{k-1}^-)\\ \boldsymbol{a}_E(t_{k-1})=\boldsymbol{\Phi}_{rv}^{-1}\left(\boldsymbol{r}_E(t_k^-)-\hat{\boldsymbol{r}}_E(t_k^-)\right)\end{cases}\tag{27}$$

式中 $\hat{\boldsymbol{r}}_E(t_k^-)$ 为依据上一时刻状态信息的外推结果.

为避免追踪星在初期受到子策略匹配度初始化误差的干扰, 本文设计了一个冷启动机制: 在博弈早期阶段 (即 $k \le m$, k 表示当前决策步数, m 为冷启动步长), 采用启动策略进行博弈决策; 待各子策略匹配度间形成显著差异后, 才切换至匹配度最高的子策略进行博弈决策.

该策略融合方法的具体流程如算法 1 所示, 其核心包括匹配度动态更新机制与基于匹配度的策略选择机制. 区别于采用决策树等白盒自解释模型拟合神经网络的传统可解释性方法, 本文所提方法通过计算匹配度实时评估各子策略的适用性, 并选择匹配度最高的子策略进行博弈决策. 在保证神经网络性能的前提下, 该方法为策略生成提供了量化且透明的依据, 使决策过程的内部逻辑更加清晰, 有效提升了策略的可解释性和可靠性.

算法 1. 匹配度驱动的追踪策略融合方法

1) 初始化各航天器的位置和速度
2) 加载各追踪子策略的策略网络 $\pi_{\boldsymbol{\theta}_n}$ 和预测网络 $P_{\boldsymbol{\psi}_n}$
3) 初始化各追踪子策略平均匹配度 $\bar{M}_n(t_0) = \bar{M}_0$
4) **for** $k = 0 : K-1$
5)　各预测网络 $P_{\boldsymbol{\psi}_n}$ 对逃逸星动作量 $\hat{\boldsymbol{a}}_E^n$ 进行预测
6)　**if** $k \le m$
7)　　第 j 颗追踪星依据启动策略生成动作量 $\boldsymbol{a}_j^{\circ}$
8)　**else**
9)　　挑选出平均匹配度 $\bar{M}_n(t_k)$ 最高的子策略 $\mathcal{P}^*$
10)　　第 j 颗追踪星依据子策略 $\mathcal{P}^*$ 生成动作量 $\boldsymbol{a}_j^*$
11)　逃逸星采用人工势场法依据式 (25) 生成动作量 $\boldsymbol{a}_E$
12)　动力学环境根据各航天器动作量进行状态更新
13)　依据式 (14) 计算各子策略的单步平均匹配度 $M_n(t_k)$
14)　依据式 (26) 更新各子策略的平均匹配度 $\bar{M}_n(t_{k+1})$
15) **end**

4　仿真实验与结果分析

4.1　场景参数设计

为增强追踪策略在不同初始位置下的鲁棒性, 各追踪星的初始位置在 X-Z 平面的一个环形区域内随机生成, 其与原点的径向距离范围为 45 km 至 55 km. 仿真实验的场景参数如表 1 所示. 逃逸星的机动能力为追踪星的 1.2 倍.

采用“预测制导 + 人工势场法”对逃逸策略进行建模, 其中引力场分段阈值 d_g 和斥力场影响半径 ρ_0 均设置为 60 km. 通过调节规避强度因子 χ, 构建具有差异化行为特征的策略集合 $\mathcal{E} = \{\pi_E(3), \pi_E(20), \pi_E(\infty)\}$.

表 1　场景参数

Table 1　Parameters of scenario

场景参数	数值
最大博弈时间	$t_{\max} = 250$ min
脉冲控制周期	$T = 600$ s
逃逸星安全距离	$r_c = 3$ km
追踪星单步最大脉冲控制约束	$\Delta V_P = 2.0$ m/s
逃逸星单步最大脉冲控制约束	$\Delta V_E = 2.4$ m/s

4.2　基于 PAC 算法的追踪子策略训练方法验证

基于 PAC 方法, 分别针对逃逸策略集合 $\mathcal{E}$ 中的三类逃逸策略开展博弈训练, 训练参数如表 2 所示. 追踪子策略中的预测网络、策略网络和价值网络均构建为四层全连接神经网络结构, 各层神经元数量依次为 256、256、256 和 128. 激活函数采用泄露线性整流函数 (leaky ReLU), 其定义为 $x = \max\{\alpha x, x\}$. 其中泄露系数 α 设置为 0.01.

表 2　PAC 算法训练参数

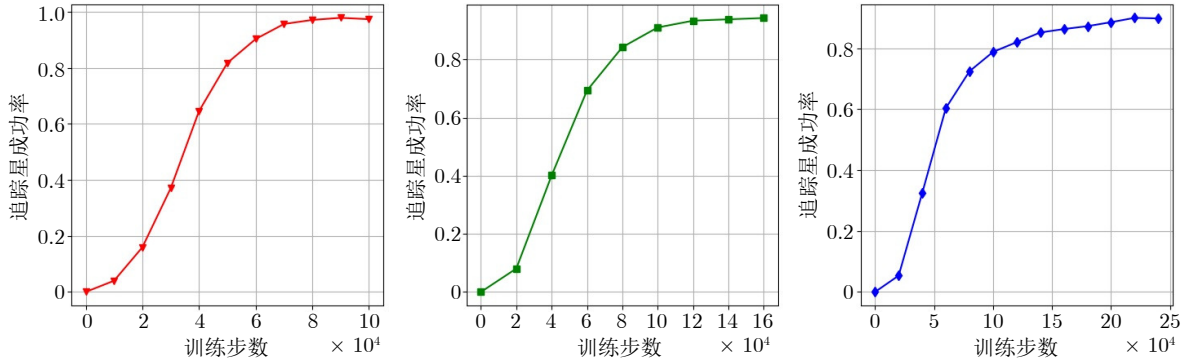
Table 2　Training parameters of PAC algorithm

训练参数	数值
策略网络学习率	$\alpha_a = 0.001$
价值网络学习率	$\alpha_c = 0.001$
预测网络学习率	$\alpha_p = 0.010$
衰减率	$\gamma = 0.97$
软更新速率	$\tau = 0.01$
延迟策略更新频率	$d = 2$
mini-batch 大小	$b = 1\,256$
经验回放池大小	$B = 100\,000$
奖励函数系数	$\alpha_1 = 0.01,\ \alpha_2 = 0.02,\ \alpha_3 = 1.00$
终端奖励	$C = 15$

在训练过程中, 预测网络 $P_{\boldsymbol{\psi}}$ 根据当前状态估计逃逸星的动作量, 策略网络 $\pi_{\boldsymbol{\theta}}$ 和价值网络 $Q_{\boldsymbol{\phi}}$ 采用 MATD3 算法进行优化. 最终获得三个追踪子策略, 分别记作 $\mathcal{P}_1$、$\mathcal{P}_2$ 和 $\mathcal{P}_3$, 其训练曲线如图 2 所示.

为验证预测网络在策略适应性评估中的量化表征效能, 采用子策略 $\mathcal{P}_1$、$\mathcal{P}_2$ 和 $\mathcal{P}_3$ 分别与 8 类逃逸策略 $\{\pi_E(\chi_n)\}_{n=1}^{8}$ 进行蒙特卡洛闭环仿真, 其中逃逸策略的规避强度因子 $\chi \in [3, +\infty)$, 该范围覆盖了从保守规避至激进逃逸的全行为域.

各追踪子策略面对不同逃逸策略时的成功率统计如图 3 所示. 由图中可以观察到, 各追踪子策略在面对规避强度因子 χ 与其陪练策略相近的逃逸策略时具有一定的泛化性能; 而面对与陪练策略行



(a) 子策略 $\mathcal{P}_1$ 训练曲线
(a) Training curve of sub-policy $\mathcal{P}_1$

(b) 子策略 $\mathcal{P}_2$ 训练曲线
(b) Training curve of sub-policy $\mathcal{P}_2$

(c) 子策略 $\mathcal{P}_3$ 训练曲线
(c) Training curve of sub-policy $\mathcal{P}_3$

图 2 基于 PAC 方法的追踪策略训练曲线

Fig. 2 Training curves of pursuit policies based on the PAC method

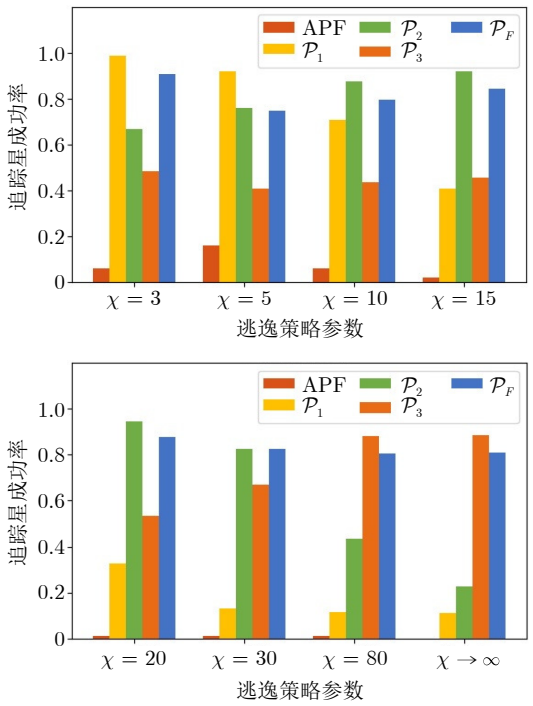


图 3 追踪策略成功率统计

Fig. 3 Success rate statistics of pursuit policies

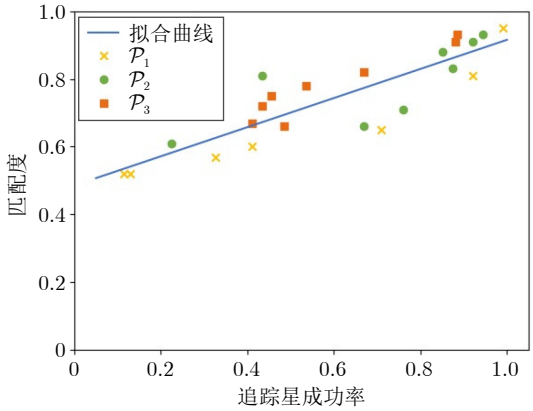


图 4 匹配度和成功率相关性曲线

Fig. 4 Correlation curves between matching degree and success rate

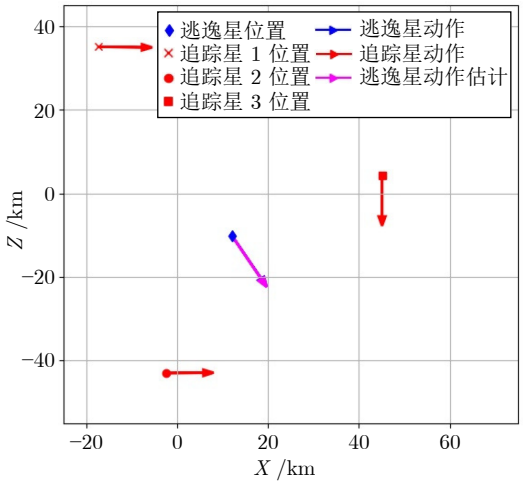


图 5 单步仿真分析: 子策略 $\mathcal{P}_3$ vs $\pi_E(\infty)$

Fig. 5 Single-step simulation analysis: Sub-policy $\mathcal{P}_3$ vs $\pi_E(\infty)$

为差异大的逃逸策略时, 追踪成功率下滑严重.

图 4 展示了匹配度与追踪成功率之间的联合分布特性, 其皮尔逊相关系数达到 0.866, 表明两者呈强正相关关系. 这说明依据预测网络输出所计算的匹配度能够真实反映追踪策略面对未知逃逸策略的适应性, 且匹配度越高策略适应性越强.

为进一步验证预测网络的适应性评估作用, 本文分别采用子策略 $\mathcal{P}_1$、$\mathcal{P}_3$ 与逃逸策略 $\pi_E(\infty)$ 进行数值仿真, 选取关键决策步骤进行详尽分析. 如图 5 所示, 逃逸星向 X-Z 平面第四象限方向进行逃逸, 子策略 $\mathcal{P}_3$ 的预测网络对逃逸星动作进行准确的估计, 其单步匹配度达到 0.995 0. 在此基础上, 追踪星 2 和追踪星 3 分别向 X-Z 平面第四象限方向机

动, 最终成功对逃逸星完成拦截.

相比之下, 如图 6 所示, 子策略 $\mathcal{P}_1$ 的预测网络误判逃逸星将向 $-X$ 方向逃逸, 导致其单步匹配度仅为 0.1338. 由于判断方向偏差, 追踪星 1 与追踪星 2 也错误地向 $-X$ 方向实施拦截, 最终未能完成追踪任务. 上述结果表明, 预测网络动作估计的匹配度确能反映追踪子策略面对当前未知逃逸策略的适应性.

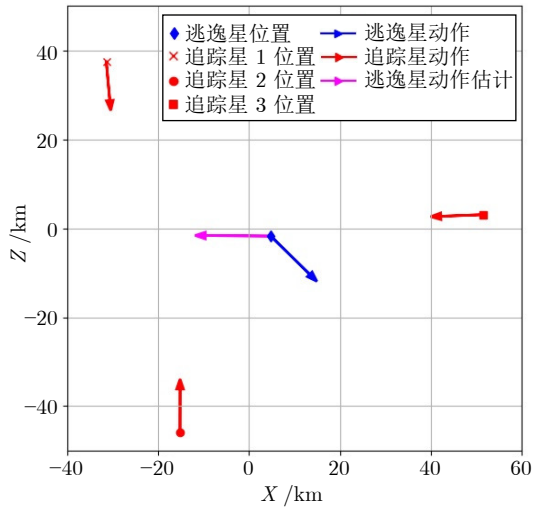


图 6　单步仿真分析: 子策略 $\mathcal{P}_1$ vs $\pi_E(\infty)$

Fig. 6　Single-step simulation analysis: Sub-policy $\mathcal{P}_1$ vs $\pi_E(\infty)$

4.3　匹配度驱动的追踪策略融合方法验证

基于前期训练获得的追踪集 $\Pi = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}$, 构建一个由策略融合器与子策略集合 Π 组成的融合策略 $\mathcal{P}_F$, 该策略依据匹配度动态选择追踪子策略. 初始平均匹配度 $\bar{M}_0$ 设置为 1, 平滑系数 λ 设置为 0.5. 当逃逸星采用相对激进的策略时 (即规避强度因子 χ 较大时), 其会进行快速机动突破追踪星的编队构型. 在这种情况下, 若融合器以 $\mathcal{P}_1$ 或者 $\mathcal{P}_2$ (针对保守逃逸策略训练的子策略) 作为启动策略, 初始阶段可能因响应速度不足而错失最佳拦截时机. 因此, 设置子策略 $\mathcal{P}_3$ 作为初始阶段启动策略, 冷启动步长 $m = 3$.

采用融合策略 $\mathcal{P}_F$ 与 8 类逃逸策略进行蒙特卡洛模拟, 图 3 展示了策略 $\mathcal{P}_F$ 面对各逃逸策略的成功率统计, 表 3 展示了各追踪策略平均与最低成功率统计.

如图 3 和表 3 所示, 策略 $\mathcal{P}_F$ 在整体性能上表现最优, 平均成功率达 82.8%, 最低成功率也保持在 75.0%, 显著优于任一追踪子策略. 上述结果表明, 策略融合器能够通过匹配度指标动态选择最适配当前逃逸策略的追踪子策略, 显著提升了追踪星应对多样化逃逸行为的适应性与泛化能力. 此外, 融合策略 $\mathcal{P}_F$ 在个人计算机上的单步平均决策时间约为 0.015 s, 能够满足实时运行的要求.

表 3　追踪策略平均与最低成功率统计表 (%)

Table 3　Statistical table of average and minimum success rates of pursuit policies (%)

追踪策略	平均追踪成功率	最低追踪成功率
$\mathcal{P}_1$	46.4	11.0
$\mathcal{P}_2$	70.7	22.5
$\mathcal{P}_3$	59.4	41.0
$\mathcal{P}_F$	82.8	75.0

为进一步验证所提方法的优越性, 本文设计了对比实验, 其中追踪星采用基于第 2.3 节所述基础算法改编的 APF 算法. 具体而言, 在第 2.3 节中的 2) 中, 将逃逸星指定为目标点, 且不引入障碍物. 此外, 增加一项逻辑判断: 当目标点 (即逃逸星) 位于追踪星的可达域内时, 直接采用 CW 制导方法计算所需的速度增量. 由于仿真步长远小于参考轨道周期, 追踪星的单脉冲可达域可合理近似为半径由 $\Delta r \approx \Delta V_P T$ 确定的球体[12].

如图 3 所示, 采用 APF 算法的追踪星难以完成追踪任务, 其主要原因归结为以下三点不足: 1) 缺乏博弈意识. 人工势场法仅依据逃逸星下一时刻预测的无控漂移位置确定目标点, 未考虑逃逸星可能的机动行为. 2) 缺乏协同配合. 各追踪星独立生成自身的脉冲控制量, 未融合其他追踪星的状态与意图, 导致整体协作效果欠佳. 3) 缺乏全局优化. 人工势场法仅进行单步优化, 缺少长期优化的能力.

相比之下, 本文所提方法通过以下关键机制克服了上述不足: 首先, 通过在训练环境中引入多样化逃逸策略, 增强了追踪星的博弈能力; 其次, 设计了包含集体奖励的奖励函数, 促进了追踪星间的协同合作; 最后, 借助价值函数估计 $\hat{Q}$, 实现了长期策略优化.

为评估匹配度计算中平滑系数 λ 对融合策略 $\mathcal{P}_F$ 性能的影响, 针对不同平滑系数 λ 开展仿真实验, 其追踪成功率统计如表 4 所示. 实验结果表明, 在不同平滑系数 λ 的设置下, 融合策略 $\mathcal{P}_F$ 在面对各类逃逸策略时仍能保持较高的追踪成功率, 表现出较强的鲁棒性, 说明融合策略 $\mathcal{P}_F$ 的决策性能对平滑系数 λ 的敏感性较低.

为验证所提方法的有效性, 采用策略 $\mathcal{P}_F$ 与逃逸策略 $\pi_E(\infty)$、$\pi_E(15)$ 分别进行打靶仿真, 工况 1 ($\pi_E(\infty)$) 的 X-Z 平面轨迹图和子策略匹配度曲线如图 7 和图 8 所示, 工况 2 ($\pi_E(15)$) 的 X-Z 平面

表 4 不同平滑系数 λ 的追踪成功率统计表 (%)

Table 4 Statistical table of pursuit success rate for different smoothing factors λ (%)

χ	λ		
	0.2	0.5	0.8
$\chi=3$	92	91	92
$\chi=10$	76	80	80
$\chi=30$	78	83	82
$\chi=80$	81	84	79
$\chi\to\infty$	73	81	81

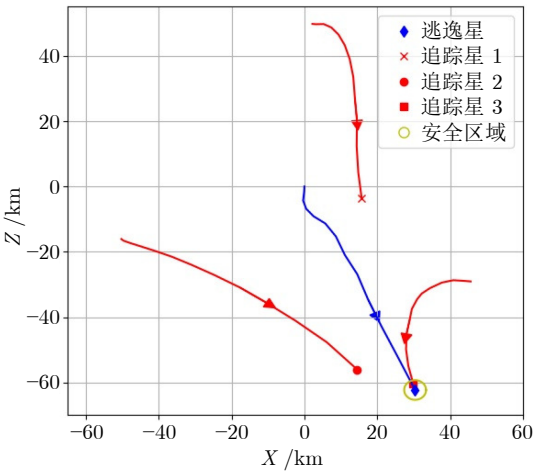


图 7 仿真轨迹图 (工况 1: $\pi_E(\infty)$)

Fig. 7 Simulation trajectory diagram (Case 1: $\pi_E(\infty)$)

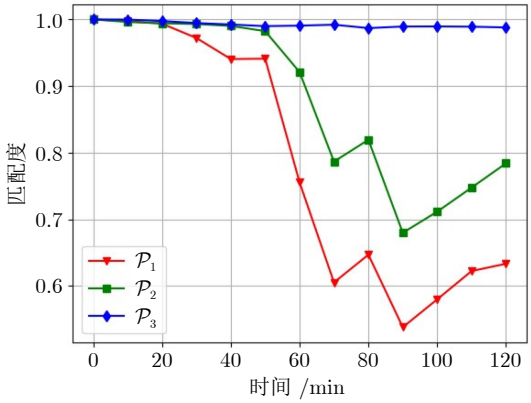


图 8 子策略匹配度曲线 (工况 1: $\pi_E(\infty)$)

Fig. 8 Matching degree curves of sub-policies (Case 1: $\pi_E(\infty)$)

轨迹图和子策略匹配度曲线如图 9 和图 10 所示.

在面对逃逸策略 $\pi_E(\infty)$ 时, 子策略 $\mathcal{P}_3$ 始终保持较高的匹配度, 而子策略 $\mathcal{P}_1$ 和 $\mathcal{P}_2$ 的匹配度则迅速下滑. 因此, 策略融合器优先采纳子策略 $\mathcal{P}_3$ 的决策动作, 最终成功实现对逃逸星的拦截. 值得注意的是, 子策略 $\mathcal{P}_3$ 正是针对 $\pi_E(\infty)$ 所训练的追踪子策略, 该结果表明策略融合器能够根据平均匹配度精准识别并选择与当前逃逸策略最适配的子策略, 进而完成高效决策.

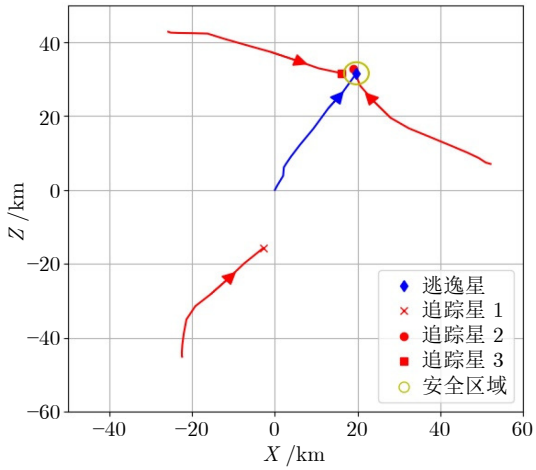


图 9 仿真轨迹图 (工况 2: $\pi_E(15)$)

Fig. 9 Simulation trajectory diagram (Case 2: $\pi_E(15)$)

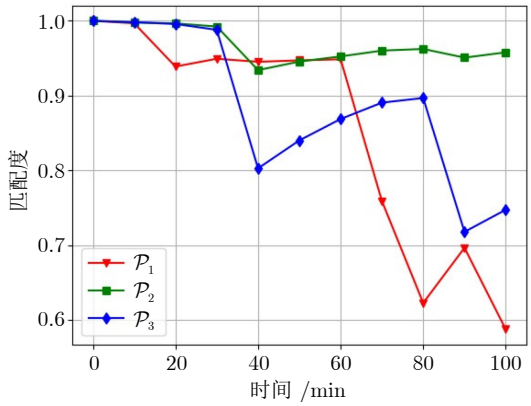


图 10 子策略匹配度曲线 (工况 2: $\pi_E(15)$)

Fig. 10 Matching degree curves of sub-policies (Case 2: $\pi_E(15)$)

类似地, 在面对逃逸策略 $\pi_E(15)$ 时, 子策略 $\mathcal{P}_2$ 的匹配度始终保持在高位, 因此其动作被融合器优先采纳, 最终成功实现对逃逸星的拦截. 值得注意的是, 子策略 $\mathcal{P}_2$ 是针对逃逸策略 $\pi_E(20)$ 训练得到的追踪子策略, 其对应的规避强度因子 χ 与当前策略 $\pi_E(15)$ 最为接近. 因此子策略 $\mathcal{P}_2$ 为最适配的追踪子策略, 并被策略融合器依据匹配度挑选执行.

5 结束语

针对轨道追逃任务中目标行为具有未知性与多样性的挑战, 本文提出一种基于行为预测和策略融合的决策方法 PACF. 在训练阶段, 在传统 AC 框架基础上引入预测网络 (P), 用于估计逃逸星的动

作行为, 并根据预测结果与实际动作之间的匹配度, 评估追踪策略应对未知逃逸策略的适应性. 在执行阶段, 构建策略融合器 (F), 基于各追踪子策略的匹配度动态选择最适配当前逃逸策略的子策略用于博弈决策. 实验结果表明, 各追踪子策略的匹配度与其追踪成功率之间呈显著正相关关系, 皮尔逊相关系数高达 0.866, 验证了预测网络对策略适应性评估的有效性, 有效增强了策略生成的可解释性. 在与多类逃逸策略的博弈测试中, 融合策略平均成功率为 82.8%, 最低成功率为 75.0%, 显著优于任一追踪子策略. 结果表明, 策略融合器能够基于平均匹配度有效筛选最优子策略, 显著提升了追踪星应对多样化逃逸策略时的泛化能力与可靠性.

在本文研究中, 假设追踪星能够实时获取逃逸星的相对位置和速度信息, 并基于 CW 方程反解出其上一时刻真实机动动作. 然而, 实际空间环境中信息获取条件较为苛刻, 测量噪声、观测盲区及通信延迟等因素均可能导致反解精度下降, 从而影响策略性能. 后续工作将进一步引入真实传感器条件下的轨道追逃博弈建模, 以提升方法的实用性和鲁棒性. 同时, 未来研究还将重点提升本文方法在追踪星数量变化下的可扩展性. 目前追踪策略仅适用于"三追一"场景, 若扩展至更多追踪星, 需要调整输入结构、增加状态维度并重新训练. 后续工作将借鉴可变长度输入、注意力机制等多智能体技术, 以提升策略在不同规模追逃任务中的泛化能力.

参考文献

1 Gao Wan-Ying, Wu Jian-Fa, Wei Chun-Ling. Review on spacecraft autonomous decision-making and planning for orbital threat avoidance. *Chinese Space Science and Technology*, 2024, **44**(4): 71−89
(高婉莹, 吴健发, 魏春岭. 航天器威胁规避自主决策规划方法研究综述. 中国空间科学技术 (中英文), 2024, **44**(4): 71−89)

2 Yuan Li, Jiang Tian-Tian. Review on intelligent autonomous control for spacecraft confronting orbital threats. *Acta Automatica Sinica*, 2023, **49**(2): 229−245
(袁利, 姜甜甜. 航天器威胁规避智能自主控制技术研究综述. 自动化学报, 2023, **49**(2): 229−245)

3 Luo Ya-Zhong, Li Zhen-Yu, Zhu Hai. Survey on spacecraft orbital pursuit-evasion differential games. *SCIENTIA SINICA Technologica*, 2020, **50**(12): 1533−1545
(罗亚中, 李振瑜, 祝海. 航天器轨道追逃微分对策研究综述. 中国科学: 技术科学, 2020, **50**(12): 1533−1545)

4 Zhang J R, Zhang K P, Zhang Y, Shi H, Tang L, Li M. Near-optimal interception strategy for orbital pursuit-evasion using deep reinforcement learning. *Acta Astronautica*, 2022, **198**: 9−25

5 Li Z Y, Zhu H, Luo Y Z. An escape strategy in orbital pursuit-evasion games with incomplete information. *Science China Technological Sciences*, 2021, **64**(3): 559−570

6 Chen Q, Qiao D, Shang H B, Liu X F. A new method for solving reachable domain of spacecraft with a single impulse. *Acta Astronautica*, 2018, **145**: 153−164

7 Li Jing-Lin, Jiang Zhong-Ying, Shi Peng, Li Wen-Long. Nash equilibrium solution method of spacecraft game based on the relative motion reachable set. *Flight Dynamics*, 2024, **42**(5): 34−41
(李靖林, 姜中英, 师鹏, 李文龙. 基于相对可达域的航天器博弈均衡求解方法. 飞行力学, 2024, **42**(5): 34−41)

8 Zhang K P, Zhang Y, Shi H, Huang H, Ye J, Wang H B. Escape-zone-based optimal evasion guidance against multiple orbital pursuers. *IEEE Transactions on Aerospace and Electronic Systems*, 2023, **59**(6): 7698−7714

9 Geng Yuan-Zhuo, Yuan Li, Huang Huang, Tang Liang. Terminal-guidance based reinforcement-learning for orbital pursuit-evasion game of the spacecraft. *Acta Automatica Sinica*, 2023, **49**(5): 974−984
(耿远卓, 袁利, 黄煌, 汤亮. 基于终端诱导强化学习的航天器轨道追逃博弈. 自动化学报, 2023, **49**(5): 974−984)

10 Wang H B, Zhang Y. Impulsive maneuver strategy for multi-agent orbital pursuit-evasion game under sparse rewards. *Aerospace Science and Technology*, 2024, **155**: Article No. 109618

11 Zhao L R, Zhang Y L, Dang Z H. PRD-MADDPG: An efficient learning-based algorithm for orbital pursuit-evasion game with impulsive maneuvers. *Advances in Space Research*, 2023, **72**(2): 211−230

12 Geng Y Z, Yuan L, Guo Y N, Tang L, Huang H. Impulsive guidance of optimal pursuit with conical imaging zone for the evader. *Aerospace Science and Technology*, 2023, **142**: Article No. 108604

13 Yang B, Liu P X, Feng J L, Li S. Two-stage pursuit strategy for incomplete-information impulsive space pursuit-evasion mission using reinforcement learning. *Aerospace*, 2021, **8**(10): Article No. 299

14 Wang Ying-Jie, Yuan Li, Tang Liang, Huang Huang, Geng Yuan-Zhuo. Reinforcement learning method for multi-spacecraft orbital game with incomplete information. *Journal of Astronautics*, 2023, **44**(10): 1522−1533
(王英杰, 袁利, 汤亮, 黄煌, 耿远卓. 信息非完备下多航天器轨道博弈强化学习方法. 宇航学报, 2023, **44**(10): 1522−1533)

15 Xu Xu-Sheng, Dang Zhao-Hui, Song Bin, Yuan Qiu-Fan, Xiao Yu-Zhi. Method for cluster satellite orbit pursuit-evasion game based on multi-agent deep deterministic policy gradient algorithm. *Aerospace Shanghai (Chinese & English)*, 2022, **39**(2): 24−31
(许旭升, 党朝辉, 宋斌, 袁秋帆, 肖余之. 基于多智能体强化学习的轨道追逃博弈方法. 上海航天 (中英文), 2022, **39**(2): 24−31)

16 Li Z Y, Chen S, Zhou C, Sun W. Orbital multi-player pursuit-evasion game with deep reinforcement learning. *The Journal of the Astronautical Sciences*, 2025, **72**(1): 1−29

17 Cao Hong-Ye, Liu Xiao, Dong Shao-Kang, Yang Shang-Dong, Huo Jing, Li Wen-Bin, et al. A survey of interpretability research methods for reinforcement learning. *Chinese Journal of Computers*, 2024, **47**(8): 1853−1882
(曹宏业, 刘潇, 董绍康, 杨尚东, 霍静, 李文斌, 等. 面向强化学习的可解释性研究综述. 计算机学报, 2024, **47**(8): 1853−1882)

18 Yang Shu-Heng, Zhang Dong, Xiong Wei, Ren Zhi, Tang Shuo. Decision-making method for air combat maneuver based on explainable reinforcement learning. *Acta Aeronautica et Astronautica Sinica*, 2024, **45**(18): 257−274
(杨书恒, 张栋, 熊威, 任智, 唐硕. 基于可解释性强化学习的空战机动决策方法. 航空学报, 2024, **45**(18): 257−274)

19 Greydanus S, Koul A, Dodge J, Fern A. Visualizing and understanding atari agents. In: Proceedings of the 35th International Conference on Machine Learning. Stockholm, Sweden: PMLR, 2018. 1792−1801

20 Bastani O, Pu Y, Solar-Lezama A. Verifiable reinforcement learning via policy extraction. In: Proceedings of the 32nd Conference on Neural Information Processing Systems. Montréal, Canada: Curran Associates, Inc., 2018.

21 Zhu Y Y, Xiao Y, Chen C L. Extracting decision tree from

trained deep reinforcement learning in traffic signal control. *IEEE Transactions on Computational Social Systems*, 2022, **10**(4): 1997−2007

22 Soares E, Angelov P P, Costa B, Castro M P G, Filev D. Explaining deep learning models through rule-based approximation and visualization. *IEEE Transactions on Fuzzy Systems*, 2020, **29**(8): 2399−2407

23 Danesh M H, Koul A, Fern A, Khorram S. Re-understanding finite-state representations of recurrent policy networks. In: Proceedings of the 38th International Conference on Machine Learning. Virtual Event: PMLR, 2021. 2388−2397

24 Konda V R, Tsitsiklis J N. Actor-critic algorithms. In: Proceedings of the 12th Advances in Neural Information Processing Systems. Denver, USA: 1999.

25 Lillicrap T P, Hunt J J, Pritzel A, Heess N, Erez T, Tassa Y, et al. Continuous control with deep reinforcement learning. In: Proceedings of the International Conference on Learning Representations. San Juan, Puerto Rico: 2016.

26 Fujimoto S, Hoof H, Meger D. Addressing function approximation error in actor-critic methods. In: Proceedings of the 35th International Conference on Machine Learning. Stockholm, Sweden: PMLR, 2018. 1587−1596

27 Zappulla R, Park H, Virgili-Llop J, Romano M. Real-time autonomous spacecraft proximity maneuvers and docking using an adaptive artificial potential field approach. *IEEE Transactions on Control Systems Technology*, 2018, **27**(6): 2598−2605

28 Gao Wan-Ying, Li Ke-Hang. Loose formation control of satellite clusters based on artificial potential field. *Aerospace Control and Application*, 2021, **47**(3): 33−39
(高婉莹, 李克行. 基于人工势场的星群松散编队控制. 空间控制技术与应用, 2021, **47**(3): 33−39)

29 Hwang J, Lee J, Park C. Collision avoidance control for formation flying of multiple spacecraft using artificial potential field. *Advances in Space Research*, 2022, **69**(5): 2197−2209

王英杰 北京控制工程研究所博士研究生. 主要研究方向为轨道追逃博弈, 多智能体强化学习.
E-mail: wangyj980311@163.com
(**WANG Ying-Jie** Ph.D. candidate at Beijing Institute of Control Engineering. His research interests include orbital pursuit-evasion game and multi-agent reinforcement learning.)

袁 利 中国空间技术研究院研究员. 主要研究方向为航天器自主控制, 鲁棒容错控制技术. 本文通信作者.
E-mail: yuanli@spacechina.com
(**YUAN Li** Researcher at China Academy of Space Technology. His research interests include spacecraft autonomous control and robust fault-tolerant control techniques. Corresponding author of this paper.)

黄 煌 北京控制工程研究所研究员. 主要研究方向为航天器智能决策与控制.
E-mail: hhuang33@163.com
(**HUANG Huang** Researcher at Beijing Institute of Control Engineering. Her main research interest is spacecraft intelligent decision-making and control.)

耿远卓 北京控制工程研究所高级工程师. 主要研究方向为航天器轨迹规划与控制.
E-mail: gengyz_hit@163.com
(**GENG Yuan-Zhuo** Senior engineer at Beijing Institute of Control Engineering. His main research interest is spacecraft trajectory planning and control.)