软件学报 ISSN 1000-9825, CODEN RUXUEW
2026,37(9):3557−3576 [doi: 10.13328/j.cnki.jos.007608] [CSTR: 32375.14.jos.007608]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

模糊映射熵驱动的强化学习系统安全监控方法*

杨 敏[1], 周子渊[1], 李晓锋[2], 刘关俊[1]

[1](同济大学 计算机科学与技术学院, 上海 201804)

[2](北京控制工程研究所, 北京 100190)

通信作者: 刘关俊, E-mail: liuguanjun@tongji.edu.cn

摘 要: 深度强化学习虽已在多种复杂任务中取得卓越成果, 但其策略在动态高维环境下仍缺乏实时安全保障, 因而亟需在部署阶段引入能够实时评估并纠正智能体决策的安全监控机制. 现有数据驱动的黑盒监控方法侧重离散或二元决策, 难以直接迁移到连续动作空间. 针对上述问题, 提出了模糊映射熵驱动的安全监控框架, 仅依赖状态、动作和成本数据即可构建, 无需任何环境模型. 该方法首先利用高斯混合模型 (Gaussian mixture model, GMM) 对离线收集的安全轨迹进行状态簇硬划分和动作簇软隶属, 并提出模糊映射熵在兼顾均衡性与模型复杂度的前提下自适应确定最优动作簇数. 随后在 Mamdani 框架下构建模糊逻辑规则, 并通过残差网络与对抗判别器联合微调簇中心, 使生成动作更贴近真实的安全分布. 在线阶段, 监控器基于 GMM 后验概率计算每条待执行状态-动作对的簇一致性度量. 一旦该度量低于阈值, 即通过模糊推理生成平滑的安全替换动作, 从而在风险发生之前完成修正. 在 Safety-Gymnasium 的 3 个导航任务上, 对 PPO-Lag、TRPO-Lag 与 CPPO-PID 策略进行了监控评估. 结果显示, 该框架在几乎不降低乃至略微提升任务回报的前提下, 显著降低累计安全成本, 并保持较高的预警覆盖率, 验证了该监控框架在连续动作场景中的有效性和实用性.

关键词: 模糊映射熵; 安全监控; Mamdani 型模糊规则; 残差网络

中图法分类号: TP311

中文引用格式: 杨敏, 周子渊, 李晓锋, 刘关俊. 模糊映射熵驱动的强化学习系统安全监控方法. 软件学报, 2026, 37(9): 3557–3576. http://www.jos.org.cn/1000-9825/7608.htm

英文引用格式: Yang M, Zhou ZY, Li XF, Liu GJ. Fuzzy-mapping-entropy-driven Safety Monitoring Method for Reinforcement Learning Systems. Ruan Jian Xue Bao/Journal of Software, 2026, 37(9): 3557–3576 (in Chinese). http://www.jos.org.cn/1000-9825/7608.htm

Fuzzy-mapping-entropy-driven Safety Monitoring Method for Reinforcement Learning Systems

YANG Min[1], ZHOU Zi-Yuan[1], LI Xiao-Feng[2], LIU Guan-Jun[1]

[1](School of Computer Science and Technology, Tongji University, Shanghai 201804, China)

[2](Beijing Institute of Control Engineering, Beijing 100190, China)

Abstract: Deep reinforcement learning has achieved excellent results in many complex tasks, but its policies still lack real-time safety guarantees in dynamic, high-dimensional environments. It is therefore urgent to introduce a safety monitoring mechanism during deployment that can evaluate and correct agent decisions in real time. Existing data-driven black-box monitoring methods focus on discrete or binary decisions and are hard to transfer directly to continuous action spaces. To address the above issue, this study proposes a fuzzy-mapping-entropy-driven safety monitoring framework, which can be constructed solely from state, action, and cost data without requiring any environment model. Specifically, the method first uses a Gaussian mixture model (GMM) to perform hard partitioning of states and soft membership of actions on the offline collected safe trajectories, and proposes fuzzy mapping entropy to adaptively determine the optimal number of action clusters under the premise of balancing uniformity and model complexity. Next, fuzzy logic rules are built in the

* 基金项目: 国家自然科学基金 (92582104)
本文由“形式化方法与应用”专题特约编辑安杰副研究员、顾斌研究员、李建文教授推荐.
收稿时间: 2025-09-08; 修改时间: 2025-10-28; 采用时间: 2025-12-08; jos 在线出版时间: 2025-12-24
CNKI 网络首发时间: 2026-03-26

Mamdani framework, and cluster centers are jointly fine-tuned with a residual network and an adversarial discriminator to make the generated actions closer to the real safe distribution. In the online phase, the monitor computes a cluster consistency measure for each pending state-action pair based on GMM posterior probabilities. If this measure falls below a threshold, fuzzy inference is used to generate a smooth safe replacement action, thus correcting the action before a risk occurs. PPO-Lag, TRPO-Lag, and CPPO-PID policies are evaluated under the proposed monitoring framework on three navigation tasks in Safety-Gymnasium. The results show that the framework significantly reduces cumulative safety costs while keeping almost the same or slightly higher task returns and maintains a high warning coverage rate, which confirms its effectiveness and practicality in continuous action scenarios.

Key words: fuzzy mapping entropy; safety monitoring; Mamdani fuzzy rule; residual network (ResNet)

深度强化学习 (deep reinforcement learning, DRL) 通过深度神经网络 (deep neural network, DNN) 对环境进行感知与特征提取, 并结合强化学习 (reinforcement learning, RL) 算法对策略进行迭代优化, 已在雅达利游戏和星际争霸 II 等仿真平台上取得显著成果[1]. 近年来, DRL 已在自动驾驶、机器人控制和智能制造系统等实际场景中得到了广泛应用[2,3], 但由于策略在动态复杂的环境下仍难以保证完全鲁棒, 一旦出现未预见的决策偏差, 就可能导致设备损坏甚至人员伤亡. 为了在正式部署前提升系统的安全性, 研究者通常在训练阶段引入对抗样本方法, 例如采用 FGSM (fast gradient sign method) 或 PGD (projected gradient descent) 对环境观测添加微小扰动, 并结合最小-最大优化 (min-max optimization) 等鲁棒训练策略来增强模型的抗干扰能力, 使 DRL 策略在面对扰动时仍能保持正确决策[4,5]. 另一些研究者则通过在训练过程中对危险行为施加惩罚来开展安全强化学习, 包括约束 MDP、拉格朗日方法[6]、广义李雅普诺夫函数[7]和后向值函数[8]等方法. 然而, 这些方法在离线训练阶段难以覆盖复杂场景下的所有状态-动作组合, 无法全面保障系统运行时的安全性. 因此, 越来越多的研究开始关注 DRL 系统的安全监控, 通过实时评估智能体的动作与环境反馈, 对高风险行为即时纠正, 避免潜在故障造成灾难性后果. 如图 1 所示, 强化学习安全监控框架由观测模块、RL 智能体、环境模块和监控/干预模块这 4 部分组成. 系统运行时, 观测模块获取并更新环境状态, RL 智能体根据策略输出动作并作用于环境, 环境返回新的状态与奖励, 监控/干预模块对每次观测与策略决定的动作进行风险评估, 一旦检测到高风险, 即对 RL 智能体的动作进行干预, 确保系统在动态或未知环境中的安全可控运行.

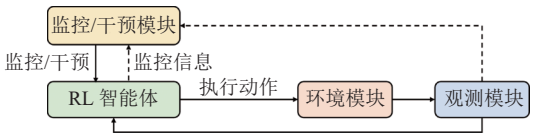


图 1 强化学习系统安全监控框架

现有的 DRL 系统的安全监控方法主要分为两类, 一类是基于形式化验证的监控器[9−15], 主要依赖于环境或策略的可验证抽象, 并以线性时序逻辑 (linear temporal logic, LTL) 和概率计算树逻辑 (probabilistic computation tree logic, PCTL) 等形式化语言描述期望安全规范, 通过模型检查工具或概率模型验证来提供可证明的安全性, 这些技术在理论上能严格保证安全, 具有良好的可解释性和可靠性, 但通常依赖于硬编码的安全属性, 难以在无模型或高维黑箱环境中推广. 另一类是基于数据驱动的黑盒监控器[16−22], 它们不需要访问智能体的内部网络结构或环境动力学, 通过采集模型运行的轨迹信息和反馈数据构建监控器, 并在检测到潜在危险时进行预警或动作干预. 然而, 现有基于数据驱动的黑盒监控器的研究多聚焦于离散动作或二元安全判断, 在无人车导航等高维连续动作场景中, 难以实现风险度量与针对性干预.

为了在连续动作空间构建基于数据驱动方式的黑盒监控器, 本文引入了模糊信息度量, 提出了一种基于模糊映射熵的闭环强化学习安全监控框架, 旨在提升预训练强化学习的安全性. 本文的贡献如下.

(1) 提出一种闭环的强化学习系统安全监控框架, 该框架通过离线轨迹数据构建状态簇-动作簇安全模糊规则库, 在线阶段采用轻量级一致性度量对原策略动作进行实时评估, 若度量结果低于安全阈值时平滑替换为规则库对应的最优安全动作.

(2) 定义模糊映射熵来量化聚类后各状态簇-动作簇映射的不确定性, 并结合熵值归一化与簇数惩罚自动选择

最优簇数, 同时设计了残差网络与对抗判别器的联合优化机制, 对离线聚类得到的动作簇中心进行微调, 使其在保持全局安全属性的同时更贴近真实轨迹分布, 提高模糊规则库的准确性.

(3) 在 Safety-Gymnasium 的 3 个典型导航任务中, 与 PPO-Lag、TRPO-Lag 和 CPPO-PID 基线策略相比, 在保持原有回报水平的前提下, 应用本文监控方法进行动作干预后平均成本均有不同程度的下降, 相较于无监控基线具备更优的安全表现.

本文第 1 节回顾强化学习监控方法及模糊熵的相关工作. 第 2 节阐述模糊逻辑系统和模糊熵, 为后续监控框架提供理论基础. 第 3 节详细介绍本文提出的模糊熵驱动的强化学习系统监控框架, 包括关键实现细节与算法流程. 第 4 节是实验分析, 通过在几种仿真环境中对所提监控框架的预测准确性和干预效果进行对比实验, 并分析结果. 第 5 节总结全文并讨论监控框架的局限与改进方向.

1 相关工作

1.1 强化学习监控方法的相关工作

随着 DRL 在自动驾驶、智能机器人和工业控制等对风险因素高敏感的场景中的深入应用, 对部署后智能体行为进行实时监测与干预的需求愈发迫切. 目前, 针对 DRL 系统的安全监控方法主要可分为两大范式: 一是形式化驱动监控, 其核心在于通过模型抽象与形式化验证技术, 在策略训练或预部署阶段为系统提供可证明的安全性保证; 二是数据驱动黑盒监控, 其核心是基于轨迹、动作与环境反馈数据构建监控器, 在 DRL 智能体运行时对其进行监测, 并在检测到潜在危险时及时采取预警和动作干预措施.

在形式化驱动监控方法中, 现有的研究主要通过基于抽象与模型检查来为 DRL 策略提供可证明的安全保证. 一方面, 一些工作将形式化验证融入策略训练闭环. 例如, Jin 等人[9]将 CEGAR (counterexample guided abstraction refinement) 技术引入 DRL 训练闭环框架中, 首先在粗粒度抽象的状态空间上进行策略学习, 随后运用模型检查工具验证 LTL 安全规范, 若验证未通过, 则通过反例自动细化抽象并重新训练, 有效提升了验证效率与覆盖度. Marchesini 等人[10]在策略学习中嵌入任务级形式化属性校验, 将违规度量作为奖励函数的惩罚项, 无需额外学习值函数即可引导策略趋向安全. 另一方面, 部分研究聚焦于训练后策略的后验审查. 例如, Kwon 等人[11]针对非确定性环境, 将训练后策略转化为仅包含可达状态的 MDP (Markov decision process) 模型, 并通过 Storm 模型检查器依据 PCTL 安全规范对策略的成功与失败概率进行了量化分析. 在此基础上, Gross 等人[12]提出了一种无需依赖网络层数或 DRL 算法细节的方法, 通过按需构建可达 MDP 子模型并结合 Storm 进行形式化验证, 显著降低了状态空间复杂度. 另外, Yang 等人[13]则通过将连续状态抽象为决策单元并训练 DNN 策略, 基于抽象状态构建 MDP 并进行概率模型检验, 有效缓解了状态爆炸并获得了更严格的不安全概率上限. 此外, 为了提升策略的透明度, 部分工作将可解释性分析与形式化验证相结合. 例如, Yang 等人[14]提出了 Reintrainer (reinforcement trainer) 框架, 该框架结合形式化验证与可解释性分析, 并通过内置的 DRLP (deep reinforcement learning property language) 约束编码语言根据验证结果动态调整策略, 以确保预定义约束得到满足. Ernest 等人[15]则基于 GFT (general fuzzy tree) 构建并训练了一种面向星际争霸 II 特定控制场景的模糊逻辑强化学习智能体, 使其决策过程透明且能满足形式化安全规范. 上述方法依赖环境模型的精确描述, 并通过预设安全规范为策略提供严格的安全保证.

与形式化驱动监控方法不同, 数据驱动黑盒监控不需要依赖环境模型, 而是利用智能体的轨迹、动作与环境反馈数据构建安全监控器, 并在 DRL 部署后进行实时监控. 其中, 部分工作专注于在运行时动态收集安全属性或通过测试构造安全临界场景. 例如, Marzari 等人[16]将 CEGAR 思想延伸至 DRL 系统运行时, 通过在导航过程中动态收集安全属性, 并依据实时反例驱动精化, 使监控器能够适应未知地图与突发障碍. Tappler 等人[17]提出一种结合搜索与模糊测试的综合测试框架, 通过搜索生成解决任务的参考轨迹及其回溯边界状态构建安全测试套件, 利用模糊测试生成多样化轨迹评估强化学习智能体的鲁棒性. 另一类工作借助机器学习技术进行安全二分类或预测监控, 例如, Zolfagharian 等人[18]利用 Q 值与状态抽象构建安全监控黑盒模型, 通过随机森林预测并在执行前发出安全违规预警. Cai 等人[19]将 DRL 系统的在线监控建模为多变量时序分类问题, 并借助时空动态图神经网络捕捉

轨迹依赖, 实现了对 DRL 系统故障的高效检测与预警. Gong 等人[20]则将离线安全学习转化为轨迹级二分类问题, 通过将预采集轨迹划分为安全和不安全两类, 以对比损失训练策略网络, 使其既能按照高回报安全轨迹进行模仿, 又可避免低回报或违规轨迹, 实验表明该方法实现了更高累积回报和更低安全成本. 此外, 部分工作在策略执行时进行在线动作替换, Selim 等人[21]在策略执行前引入数据驱动的可达性分析层, 对不安全动作进行检测并替换为最近似的安全动作. Yang 等人[22]则将 DRL 策略抽象为概率自动机, 并利用反向广度优先搜索识别关键状态-动作对, 在关键状态下进一步搜索最优替代动作. 上述方法大多聚焦离散或二元决策, 不适用于对连续动作空间的风险量化. 为解决上述难题, 本文设计了一个基于风险量化与最小干预的监控机制, 该机制无需环境模型, 仅依赖环境运行的状态-动作和成本信息, 能够在线对连续动作进行风险量化与平滑修正, 满足 RL 系统的安全监控需求.

1.2 模糊熵的相关工作

为了量化动作在安全与风险两个模糊子集之间隶属度分布的离散程度, 本文以模糊熵作为核心度量. 模糊熵最早源于模糊集理论, 用于度量隶属度分布的不确定性. Cheng[23]在模糊动力系统框架下首次引入熵概念, 并将其推广为条件映射熵, 用于评估映射前后隶属度的变化. Yan 等人[24]将条件映射熵应用于序列映射, 以覆盖更复杂的时序场景. Wu[25]通过多维投影与噪声抑制技术, 有效缓解了高维数据映射中的噪声敏感性, 从而提升了映射熵在高维环境下的稳定性和鲁棒性. Hu 等人[26]则利用节点映射关系的熵差, 对复杂网络中链路预测性能进行评估, 显著改善了预测准确度. 为体现不同侧互信息的重要性, 后续研究提出了加权映射熵. 例如, Yang 等人[27]基于 Bowen 覆盖构造了 $\propto$ 加权拓扑熵. Tsukamoto[28]将 Bowen 型定理推广至加权情形, 提出了加权拓扑压强与加权测度熵的等价关系. Li 等人[29]在信息检索中利用词项语义相似度为熵赋权, 以增强查询结果的适应性. 受现有映射熵研究启发, 本文设计了量化状态簇-动作簇映射的不确定性的模糊映射熵, 用于 DRL 系统的在线安全监控框架中.

2 基础知识

本文所提方法主要基于模糊逻辑系统和模糊熵, 其中模糊逻辑系统生成隶属度来描述模糊关系, 而模糊熵对这些隶属度分布进行不确定性量化. 下面对相关概念和知识进行介绍.

2.1 模糊逻辑系统

传统的逻辑体系 (如二值逻辑、多值逻辑等) 要求对概念进行严格的二分判断, 而在实际工程与决策场景中, 许多概念并不具备明确的“是/否”边界, 例如“高矮”“温度”“安全”等. 为了处理此类“介于真/假之间”的信息, Zadeh[30]提出了模糊集合理论与模糊逻辑, 通过引入连续隶属度函数, 将传统集合的硬性边界推广为可在 $[0,1]$ 区间内平滑变化的归属性描述, 实现从模糊语言到精确数值的映射过程.

在数学上, 给定某个实数域区间 $X \subseteq \mathbb{R}$, 一个模糊集合 A 就是指一个映射:

$$\mu_A : X \to [0,1] \tag{1}$$

对任意 $x \in X$, 隶属函数 $\mu_A(x)$ 用于定量刻画元素 x 属于模糊集合 A 的程度. 常见的隶属函数形式包括三角形、梯形及高斯型等. 考虑到本文所研究的状态-动作空间均为高维连续型, 因此选用高斯型隶属函数, 其优势是平滑可导且参数容易通过极大似然或 EM 算法从数据中学习. 尽管隶属函数能够精确刻画输入变量在各模糊集上的归属性程度, 但要将这些隶属度信息转化为具体的控制或决策输出, 还需借助系统化的推理流程. Mamdani 型模糊推理系统最早由 Mamdani 等人[31]提出, 是广泛应用于工业控制与决策领域的模糊推理架构之一, 旨在通过一组“若…则…”形式的模糊规则来刻画输入与输出之间的模糊映射关系. 设输入空间为 $X \subseteq \mathbb{R}$, 输出空间为 $Y \subseteq \mathbb{R}$, 系统的规则库通常由如下形式的模糊规则构成:

$$\text{Rule } i: \text{ If } x \text{ is } X_i, \text{ then } y \text{ is } Y_i,\ i = 1,\ldots,n \tag{2}$$

其中, $X_i : X \to [0,1]$ 与 $Y_i : Y \to [0,1]$ 分别表示输入与输出的隶属度函数. 对于任意精确输入 $x^* \in X$, 首先需要在第 i 个输入模糊集中通过隶属度函数计算其隶属度, 即:

$$\alpha_i = \mu_{X_i}(x^*) = X_i(x^*),\ i = 1,\ldots,n \tag{3}$$

其中, $\alpha_i \in [0,1]$ 为该规则的前件激活度, 度量了“X^*属于模糊集 X_i”的程度. 若 $\alpha_i = 0$, 表明输入 x^* 与模糊集 X_i 无重叠, 该规则在后续推理过程中不会被激活; 若 $\alpha_i = 1$, 则表明 x^* 完全匹配该模糊集的中心. 在输出空间 Y 上根据激活度与该规则结论隶属度 $Y_i(y)$ 计算局部输出隶属度:

$$\mu_i^{\text{out}}(y) = \alpha_i \wedge Y_i(y) = \min(\alpha_i, Y_i(y)) \tag{4}$$

随后对所有 n 条规则的局部输出隶属度聚合为综合隶属度:

$$\mu_{Y'}(y) = \max_{1\leqslant i\leqslant n} \mu_i^{\text{out}}(y) = \max_{1\leqslant i\leqslant n} \min(\alpha_i, Y_i(y)) \tag{5}$$

在获得综合隶属度 $\mu_{Y'}(y)$ 后, 需要将该模糊分布转换为一个具体的数值输出, 来用于控制或决策执行. Mamdani 型方法通常采用重心法进行去模糊化:

$$\hat{y} = \frac{\int_Y y\mu_{Y'}(y)\mathrm{d}y}{\int_Y \mu_{Y'}(y)\mathrm{d}y} \tag{6}$$

重心法能够在多峰或复合隶属度分布之间实现平滑折中. 若综合隶属度曲线高度集中于某一区域, 则 $\hat{y}$ 反映该区域的中心点; 若隶属度曲线呈多峰分布, 重心法会给出加权平均结果, 平衡不同峰值对最终输出的贡献.

2.2 模糊熵

在模糊逻辑系统中, 隶属函数 $\mu(x) \in [0,1]$ 刻画了元素在模糊集合中“部分归属”的程度, 体现了系统对输入信息的模糊不确定性. 然而, 仅靠隶属度无法度量整体模糊系统的复杂度和不确定性程度, 还需要一种能够将隶属度分布映射到实数度量的不确定性指标, 即“模糊熵 (fuzzy entropy)”. 考虑一个有限全集 $X = \{x_1, x_2, \ldots, x_n\}$ 上的模糊集合 A, 其隶属度向量为:

$$\mu = (\mu_1, \mu_2, \ldots, \mu_n),\ \mu_i = \mu_A(x_i) \in [0,1] \tag{7}$$

在此基础上, de Luca 等人[32]定义了模糊熵 $H_{\text{fmap}}(A)$, 该函数必须满足以下 3 条核心公理.

(a) 退化零值公理

$$H_{\text{fmap}}(A) = 0 \Leftrightarrow \forall i \in \{1,\ldots,n\}:\ \mu_i \in \{0,1\} \tag{8}$$

当且仅当 A 退化为经典硬集 (隶属度只取 0 或 1) 时, 熵值 $H_{\text{fmap}}(A)$ 为 0; 若存在某个 i 使得 $0 < \mu_i < 1$, 则必须有 $H_{\text{fmap}}(A) > 0$.

(b) 最大熵边界公理

$$H_{\text{fmap}}(A)\text{取得最大值} \Leftrightarrow \forall i:\ \mu_i = \frac{1}{2} \tag{9}$$

当且仅当整个模糊集 A 在全集 X 上的隶属度均为 $\frac{1}{2}$ 时, 熵 $H_{\text{fmap}}(A)$ 达到最大; 对于任意非均匀隶属度分布, $H_{\text{fmap}}(A)$ 均严格小于该最大值.

(c) 锐化单调公理

设 A^* 为通过将 A 中各 μ_i 向 0 或 1 边缘锐化得到的锐化版本, 即对所有 i 均满足:

$$\begin{cases} 0 < \mu_i < \dfrac{1}{2} \Rightarrow \mu_i^* < \mu_i \\ \dfrac{1}{2} < \mu_i < 1 \Rightarrow \mu_i^* > \mu_i \\ \mu_i = \dfrac{1}{2} \Rightarrow \mu_i^* = \dfrac{1}{2} \end{cases} \tag{10}$$

则须满足 $H_{\text{fmap}}(A^*) \leqslant H_{\text{fmap}}(A)$. 如果将 A 的隶属度向更确定的 0 或 1 端点移动后得到 A^*, 则熵值不会增大.

满足上述 3 条公理的函数 $H_{\text{fmap}}(A)$ 即为合法的模糊熵, 则模糊熵 $H_{\text{fmap}}(A)$ 具体形式为:

$$H_{\text{fmap}}(A) = \sum_{i=1}^{n} \left[-\mu_i \ln(\mu_i) - (1-\mu_i)\ln(1-\mu_i)\right] \tag{11}$$

该形式与 de Luca 等人[32]提出的经典模糊熵在结构上保持一致, 并在形式上严格满足退化零值、最大熵边界和锐化单调这 3 条公理. 当模糊集退化为经典硬集 (即所有隶属度 $\mu_i \in \{0,1\}$) 时, 熵值 $H_{\text{fmap}}(A) = 0$, 满足退化零值

公理; 当所有隶属度均为 $\mu_i = 0.5$ 时, 系统达到最大不确定性, 熵值取得最大值; 若某一隶属度向 0 或 1 方向锐化, 则熵值减小或保持不变, 从而满足锐化单调公理. 需要说明的是, 满足 3 条公理的模糊熵函数并不唯一, 公式 (11) 仅是该公理体系下的一种典型实现形式.

3 基于模糊映射熵的强化学习安全监控框架

3.1 框架概述

本文提出了一种基于模糊映射熵的强化学习闭环安全监控框架, 旨在为预训练强化学习智能体提供实时的安全保障, 如图 2 所示, 该框架分为离线构建与在线监控两个阶段. 在离线阶段, 首先对零成本安全样本采用高斯混合模型 (Gaussian mixture model, GMM) 进行软聚类, 得到样本在状态簇与动作簇上的后验隶属度. 进一步地, 将同一状态簇内样本的动作隶属度累积为概率向量, 从而在动作簇层面形成条件映射分布. 随后, 引入归一化模糊映射熵与簇数惩罚项, 以在均衡性与模型复杂度之间自适应选择最优动作簇数. 考虑到仅基于聚类的结果难以刻画局部状态差异, 进而难以生成与真实数据高度一致的细节动作, 本文在此基础上引入残差网络与对抗判别器对动作簇中心进行局部修正, 最终构建 Mamdani 型模糊规则库. 在在线监控阶段, 框架首先计算待执行状态-动作对的聚类一致性度量, 并将其与预设安全阈值进行比较. 当度量值低于阈值时, 系统通过 Mamdani 模糊推理及反模糊化生成安全替换动作; 否则, 保留原始策略输出. 通过这一机制, 该闭环框架将实时风险量化与连续动作的平滑修正相结合, 实现了对连续动作空间下强化学习系统的在线安全监控. 接下来将对该框架的核心环节进行详细阐述.

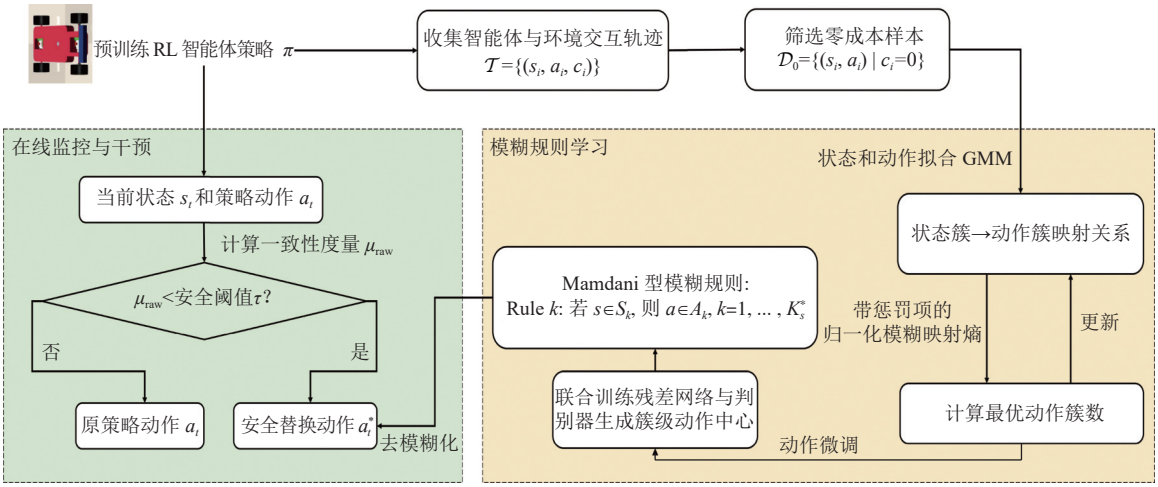


图 2 基于模糊映射熵的强化学习闭环安全监控框架图

3.2 高斯混合模型 (GMM) 聚类

在多数基于聚类的映射方法中, 通常将每个状态簇硬性对应至单一动作簇, 但这一处理忽略了状态簇在多个动作簇之间可能存在的部分隶属关系, 难以反映样本间映射的不确定性. 当状态簇与动作簇数量较大时, 此类二元分配机制往往导致连续隶属信息的丢失. 为克服该问题, 本节引入 GMM 对安全样本进行软聚类, 使得每个样本能够同时获得其在各状态簇与动作簇上的后验隶属度. 进一步地, 通过在同一状态簇内累积所有样本在不同动作簇上的隶属度, 可将该状态簇在动作簇空间中的映射表示为一个概率向量, 从而在动作簇层面形成能够刻画全局隶属强度的分布. 具体描述如下.

本文首先在实验环境中运行训练有素的智能体, 并仅保留其产生零成本的安全轨迹, 以确保聚类与规则构建所依赖的数据不包含风险因素. 在此基础上构建原始数据集 $\mathcal{T}=\{(s_i,a_i,c_i)\}_{i=1}^{N}$, 其中 N 表示轨迹数量, $s_i\in\mathbb{R}^{d_s}$ 表示

第 i 个时间步的状态向量, $a_i \in \mathbb{R}^{d_a}$ 表示对应的动作向量, $c_i \in \mathbb{R}_{\geqslant 0}$ 为该状态-动作对的成本, 成本越高表示该行为风险越大, 而 $c_i = 0$ 表示该行为是安全的. 安全数据集可形式化表示为:

$$\mathcal{D}_0 = \{(s_i, a_i) \mid c_i = 0\}_{i=1}^{N_{\text{safe}}},\ N_{\text{safe}} \leqslant N \tag{12}$$

在 GMM 聚类中, 本文将聚类自由度集中于动作簇数的选择. 由于动作直接决定策略的安全边界, 且同一状态域内往往存在多种可行动作, 因此需在候选集合上自适应确定最优动作簇数. 相对而言, 状态簇数被固定来保证状态空间划分的稳定性. 具体地, 状态簇数固定为 K_s^*, 动作簇数候选集合设定为 $\mathcal{K}_a = \{K_a^{(1)}, K_a^{(2)}, \ldots\}$, 最优动作簇数记为 K_a^*. 随后, 分别在状态空间与动作空间拟合 GMM:

$$p(s) = \sum_{k=1}^{K_s^*} \pi_k^s \mathcal{N}(s \mid \mu_k^s, \Sigma_k^s),\ p(a) = \sum_{j=1}^{K_a} \pi_j^a \mathcal{N}\left(a \mid \mu_j^a, \Sigma_j^a\right) \tag{13}$$

其中, $\mathcal{N}(s \mid \mu_k^s, \Sigma_k^s)$ 表示以均值向量 $\mu_k^s \in \mathbb{R}^{d_s}$、协方差矩阵 $\Sigma_k^s \in \mathbb{R}^{d_s \times d_s}$ 为参数的多元高斯密度函数, $\mathcal{N}\left(a \mid \mu_j^a, \Sigma_j^a\right)$ 表示以均值向量 $\mu_j^a \in \mathbb{R}^{d_a}$、协方差矩阵 $\Sigma_j^a \in \mathbb{R}^{d_a \times d_a}$ 为参数的多元高斯密度函数. 本文在所有动作候选簇数下均进行 GMM 拟合并保留所得簇中心, 在确定最优簇数 K_a^* 后, 使用对应的簇中心集合 $\{\mu_j^a\}_{j=1}^{K_a^*}$ 作为状态簇到动作簇映射的初始估计.

对于任意 $\mathcal{D}_0$ 中的样本 (s_i, a_i), 基于 GMM 计算其在状态空间中属于第 k 个状态簇 S_k 的后验隶属度:

$$\tau_{i,k}^s = p(S_k \mid s_i) = \frac{\pi_k^s \mathcal{N}(s_i \mid \mu_k^s, \Sigma_k^s)}{\sum_{k'=1}^{K_s^*} \pi_{k'}^s \mathcal{N}(s_i \mid \mu_{k'}^s, \Sigma_{k'}^s)},\ \sum_{k=1}^{K_s^*} \tau_{i,k}^s = 1 \tag{14}$$

以及在动作空间中属于第 j 个动作簇 A_j 的后验隶属度:

$$\tau_{i,j}^a = p(A_j \mid a_i) = \frac{\pi_j^a \mathcal{N}\left(a_i \mid \mu_j^a, \Sigma_j^a\right)}{\sum_{j'=1}^{K_a} \pi_{j'}^a \mathcal{N}\left(a_i \mid \mu_{j'}^a, \Sigma_{j'}^a\right)},\ \sum_{j=1}^{K_a} \tau_{i,j}^a = 1 \tag{15}$$

依据最大后验原则, 将样本 (s_i, a_i) 划分至最可能的状态簇, 从而得到状态子域 $S_k = \left\{i \mid k = \arg\max_{k'} \tau_{i,k'}^s\right\}$, 其中 $k = 1, \ldots, K_s^*$. 该划分方式将高维连续状态空间划分为若干状态簇, 每个状态子域都可以看作是相似环境状态的集合. 在动作划分方面则保留 GMM 的软隶属度, 以反映同一状态下的多模态可行动作, 避免将复杂行为模式压缩为单一簇. 在第 k 个状态簇 S_k 内, 将该簇中所有样本的动作隶属度累积并归一化为:

$$\tilde{\mu}_{kj} = \frac{\sum_{i \in S_k} \tau_{i,j}^a}{\sum_{j'=1}^{K_a} \sum_{i \in S_k} \tau_{i,j'}^a},\ \text{s.t.}\ \sum_{j=1}^{K_a} \tilde{\mu}_{kj} = 1 \tag{16}$$

其中, $\sum_{i \in S_k} \tau_{i,j}^a$ 表示在状态簇 S_k 内所有样本对动作簇 j 的隶属度总和, $\tilde{\mu}_{kj}$ 可视为在状态簇 k 条件下, 动作属于簇 j 的经验条件概率. 向量 $\tilde{\mu}_k = (\tilde{\mu}_{k1}, \ldots, \tilde{\mu}_{kK_a})$ 能够刻画状态簇与动作簇之间的概率映射, 从而避免硬分配导致的信息丢失与方差增大问题.

3.3 模糊映射熵

传统模糊熵的度量对象是单个模糊集合的隶属度分布, 而在本工作中, 我们的重点是由多个状态簇共同指向动作簇的映射整体, 即跨状态簇汇总后的动作簇层面分布. 而第 3.2 节得到的局部分布 $\tilde{\mu}_{kj}$ 只能反映某一状态簇下的局部映射特征, 无法揭示整体的均衡性与覆盖情况.

为此, 本文提出“模糊映射熵”的概念, 以刻画状态簇到动作簇映射的全局不确定性. 当熵值较高时, 说明各动作簇在全局范围内对应均衡, 映射关系稳定且样本利用充分; 当熵值较低时, 说明映射集中于少数动作簇, 存在模式覆盖不足与方差增大的风险. 在此基础上, 进一步结合簇数惩罚, 以兼顾映射均衡与模型复杂度. 映射均衡确保

各动作簇在不同状态子域中得到充分对应, 避免少数簇长期占优; 而控制模型复杂度能够抑制随动作簇数增加带来的复杂度升高, 从而实现均衡性与复杂度的权衡.

基于第 3.2 节得到的各状态簇对动作簇的 $\tilde{\mu}_{kj}$, 可在动作簇层面构建全局分布. 令:

$$p_j^{(f)} = \frac{1}{K_s^*}\sum_{k=1}^{K_s^*}\tilde{\mu}_{kj},\ \sum_{j=1}^{K_a} p_j^{(f)} = 1,\ p_j^{(f)} \geqslant 0 \tag{17}$$

其中, $p_j^{(f)}$ 反映跨所有状态簇后, 动作簇 j 的综合隶属度水平. 若 $\tilde{\mu}_{kj}$ 趋于均匀, 则 $p_j^{(f)}$ 更接近均匀, 反之则更为集中. 基于该全局分布即可度量状态簇到动作簇映射的整体不确定性. 接下来给出模糊映射熵的定义.

定义 1. 模糊映射熵. 令 $\{p_j^{(f)}\}_{j=1}^{K_a}$ 为公式 (17) 中定义的动作簇层面概率分布, 则在动作簇数 K_a 下的模糊映射熵定义为:

$$H_{\text{fmap}}(K_a) = -\sum_{j=1}^{K_a} p_j^{(f)} \ln p_j^{(f)} \tag{18}$$

该熵值满足 $0 \leqslant H_{\text{fmap}}(K_a) \leqslant \ln K_a$. 当 $p_1^{(f)} = \ldots = p_{K_a}^{(f)} = 1/K_a$ 时, $H_{\text{fmap}}(K_a) = \ln K_a$; 当某一 $p_j^{(f)} = 1$ 且其余为 0 时, $H_{\text{fmap}}(K_a) = 0$. 遵循文献 [32] 对模糊熵的公理化要求, 为了保证模糊映射熵 $H_{\text{fmap}}(K_a)$ 作为“状态簇→动作簇映射不确定性度量”具有合理性与一致性, 其定义满足以下公理化要求.

(a) 退化零值公理

若映射分布 $p_j^{(f)}$ 退化为硬映射计数, 即存在某个 j^* 使得 $p_{j^*}^{(f)} = 1$ 且对所有 $j \neq j^*$ 有 $p_j^{(f)} = 0$, 则此时映射不含任何模糊不确定性, 应满足 $H_{\text{fmap}}(K_a) = -\sum_{j=1}^{K_a} p_j^{(f)} \ln p_j^{(f)} = 0$. 反之, 若至少有两个 $p_j^{(f)} \in (0,1)$, 则必须有 $H_{\text{fmap}}(K_a) > 0$. 该公理保证当簇间映射完全硬分配时, 熵值为 0.

(b) 最大熵边界公理

当每个状态簇对所有动作簇的累积隶属度相同时, 映射分布 $p_j^{(f)}$ 完全均匀, 即 $p_1^{(f)} = p_2^{(f)} = \ldots = p_{K_a}^{(f)} = \dfrac{1}{K_a}$, 此时模糊映射熵达到最大值 $H_{\text{fmap}}(K_a) = \ln K_a$. 对于任意非均匀分布, 有 $H_{\text{fmap}}(K_a) < \ln K_a$. 该公理体现出最均匀的映射具有最大不确定性.

(c) 锐化单调公理

若对某一映射分布 $p_j^{(f)}$ 进行锐化调整, 将部分权重从较小 p_j 向较大 $p_{j'}$ 移动, 使得新的分布 $p_j^{(f)*}$ 对任何 $j \in \{1,\ldots,K_a\}$ 均满足:

$$\begin{cases} 0 < p_j^{(f)} < \dfrac{1}{K_a} \Rightarrow p_j^{(f)*} < p_j^{(f)} \\ p_j^{(f)} > \dfrac{1}{K_a} \Rightarrow p_j^{(f)*} > p_j^{(f)} \\ p_j^{(f)} = \dfrac{1}{K_a} \Rightarrow p_j^{(f)*} = \dfrac{1}{K_a} \end{cases} \tag{19}$$

则必须保证 $H_{\text{fmap}}\left(p_j^{(f)*}\right) \leqslant H_{\text{fmap}}\left(p_j^{(f)}\right)$, 即任何让分布向边缘值 $\{0,1\}$ 靠拢的锐化操作不应增加熵值, 保证映射越确定, 则熵越低.

推论 1. 当各状态簇对动作簇的累积隶属度 $\tilde{\mu}_{kj}$ 均取 $\{0,1\}$ 时, 映射分布 $p_j^{(f)}$ 退化为硬分配计数分布:

$$p_j^{(f)} = \frac{1}{K_s^*}\sum_{k=1}^{K_s^*}\tilde{\mu}_{kj} = \frac{|\{k\colon \text{map}(k) = j\}|}{K_s^*} \tag{20}$$

其中, $|\{k\colon \text{map}(k) = j\}|$ 表示被硬映射到动作簇 j 的状态簇数量. 此时, $H_{\text{fmap}}(K_a)$ 等价于离散熵 $-\sum_j p_j^{(f)} \ln p_j^{(f)}$, 与传统映射熵保持一致.

由于模糊映射熵 $H_{\text{fmap}}(K_a)$ 会随着动作簇数的增大而单调增加, 会偏向较大簇数. 为在映射均衡性与模型复杂

度之间取得平衡, 本文引入对簇数的显式惩罚项, 如下定义带惩罚的归一化模糊映射熵.

定义 2. 带惩罚项的归一化模糊映射熵. 设 $\Phi\colon \mathbb{N}^+ \to [0,1]$ 为作用于簇数 K_a 的惩罚函数, 带惩罚项的归一化模糊映射熵定义为:

$$J(K_a) = \frac{H_{\text{fmap}}(K_a)}{\ln K_a} - \alpha\Phi(K_a) \tag{21}$$

其中, $\alpha > 0$ 为权衡系数, 通过求解 $K_a^* = \arg\max_{K_a \in \mathcal{K}_a} J(K_a)$, 即可在保证动作簇分布均衡的同时有效抑制复杂度的过度增加, 从而确定最优动作簇数.

为更直观地说明上述过程, 图 3 给出了一个简化示例. 首先, 原始状态-动作对数据集 $\mathcal{D}_0$ 经 GMM 聚类划分为 3 个状态簇 S_1、S_2、S_3. 在候选动作簇数 $K_a = \{2,3\}$ 下, 再对动作集合进行 GMM 聚类, 得到各动作样本的软隶属度 $\tau_{i,j}^a$. 随后, 在每个状态簇内对样本的动作隶属度进行累积与归一化, 得到条件分布矩阵 $\tilde{\mu}$. 为了刻画状态簇到动作簇映射的全局不确定性, 在状态簇维度上进一步进行均值汇聚, 得到全局动作分布 p^f, 并据此计算模糊映射熵 H_{fmap}, 结合惩罚项 $\Phi(K_a)$ 得到目标函数值 $J(K_a)$. 对比 $K_a = 2$ 与 $K_a = 3$ 的结果可见, 尽管 $H_{\text{fmap}}(3) > H_{\text{fmap}}(2)$, 但在引入惩罚后 $J(2) > J(3)$, 最终得到最优动作簇数为 $K_a^* = 2$.

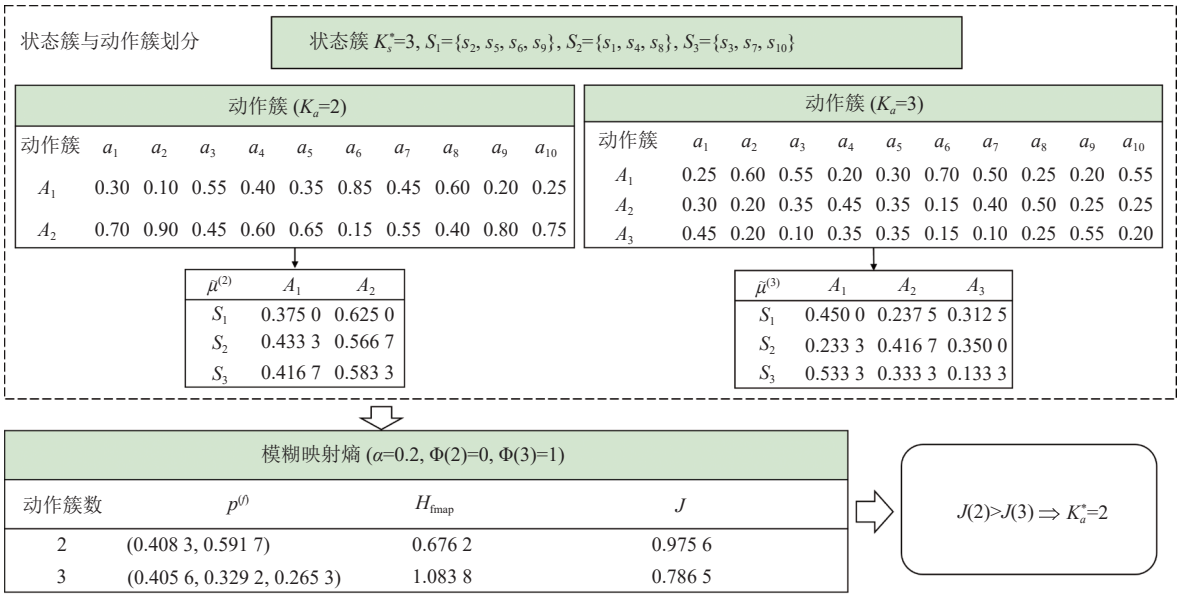


图 3 基于模糊映射熵的最优动作簇数选择示例

在获得最优动作簇参数后, 对每个样本 (s_i, a_i) 计算其在状态和动作 GMM 中的后验簇标签:

$$k_i = \arg\max_{1 \leqslant k \leqslant K_s^*} p(S_k \mid s_i),\ j_i = \arg\max_{1 \leqslant j \leqslant K_a^*} p(A_j \mid a_i) \tag{22}$$

并据此构建状态簇到动作簇的映射字典:

$$\text{map}(k) = \arg\max_{1 \leqslant j \leqslant K_a^*} \sum_{i:\, k_i = k} \mathbb{I}[j_i = j],\ k = 1, \ldots, K_s^* \tag{23}$$

其中, $\mathbb{I}[j_i = j]$ 是指示函数, 当且仅当样本 i 的动作簇标签 j_i 恰好等于候选动作簇 j 时, $\mathbb{I}[j_i = j] = 1$; 否则为 0.

3.4 基于残差的动作对抗分布对齐

在获得状态簇到动作簇的映射字典 $\text{map}(k)$ 之后, 对应的动作簇中心记为 $\mu_{\text{map}(k)}^a$, 表示状态簇 S_k 的初始动作估计. 由于这些簇中心由安全样本聚类得到, 能够在全局上提供基本的安全参考. 然而, 簇中心仅为统计意义下的均值, 缺乏对局部状态差异的刻画, 难以生成与真实数据高度一致的细节动作. 为此, 本文设计残差网络在簇中心基

础上进行细粒度修正, 从而提升局部拟合能力. 但若仅依赖残差修正, 生成动作的整体分布可能偏离原始策略. 为保证全局分布一致性, 本文进一步引入对抗判别器, 使修正后的动作在统计分布上对齐于真实数据.

具体来说, 残差预测器定义为轻量级多层感知机 (MLP): $r_\theta\colon \mathbb{R}^{d_s+d_a}\to\mathbb{R}^{d_a}$, 其中 θ 为网络参数, 输入为状态向量与对应动作簇中心的拼接, 输出为对该动作簇中心的修正残差. 对于每个安全样本 (s_i,a_i), 先从其对应的状态簇标签 k 中读取初始动作簇中心 $\mu^a_{\mathrm{map}(k)}$, 将状态 s_i 与动作簇中心 $\mu^a_{\mathrm{map}(k)}$ 级联后输入残差网络, 得到修正后的动作预测:

$$\hat{a}_i=\mu^a_{\mathrm{map}(k)}+r_\theta([\,s_i,\mu^a_{\mathrm{map}(k)}\,]) \tag{24}$$

其中, $\hat{a}_i$ 表示通过残差网络对原动作簇中心的偏移, 为了让残差网络重点拟合真实动作并适度抑制动作簇中心的偏差, 采用均方误差损失:

$$\mathcal{L}_{\mathrm{res}}(\theta)=\sum_{i=1}^{N_{\mathrm{safe}}}\|a_i-\hat{a}_i\|^2 \tag{25}$$

进一步地, 为了在更新残差网络时保证生成动作在全局分布上与原始策略动作保持一致, 在联合空间 (s,a) 上引入二分类判别器 $D_\phi\colon \mathbb{R}^{d_s+d_a}\to(0,1)$, 其中判别器参数 ϕ 用于区分真实样本对 (s_i,a_i) 与生成样本对 $(s_i,\hat{a}_i)$. 判别器的目标是最小化如下交叉熵损失:

$$\mathcal{L}_D(\phi)=-\sum_{i=1}^{N_{\mathrm{safe}}}[\ln D_\phi(s_i,a_i)+\ln(1-D_\phi(s_i,\hat{a}_i))] \tag{26}$$

同时, 为了使残差网络生成的 $\hat{a}_i$ 能欺骗判别器, 从而使得它们在联合分布空间内与真实样本尽可能难以区分, 定义如下对抗训练损失:

$$\mathcal{L}_{\mathrm{adv}}(\theta)=-\sum_{i=1}^{N_{\mathrm{safe}}}\ln D_\phi(s_i,\hat{a}_i) \tag{27}$$

通过以下交替最小化过程来联合训练残差网络与判别器:

$$\min_\theta(\mathcal{L}_{\mathrm{res}}(\theta)+\alpha\,\mathcal{L}_{\mathrm{adv}}(\theta)),\ \min_\phi\mathcal{L}_D(\phi) \tag{28}$$

其中, $\alpha>0$ 为平衡系数, 用于控制拟合真实动作与对抗分布对齐两者之间的权重, 实现对动作簇中心的微调与全局分布对齐. 在联合训练完成后, 对残差输出进行聚合, 得到每个状态簇对应的最终动作中心. 具体地, 对状态簇 S_k 内所有样本的残差 r_i 做平均, 得到簇内残差均值 $\bar{r}_k$:

$$\bar{r}_k=\frac{1}{|\{i:k_i=k\}|}\sum_{i:k_i=k}r_\theta\left(\left[s_i,\mu^a_{\mathrm{map}(k)}\right]\right),\ k=1,\ldots,K^*_s \tag{29}$$

其中, $|\{i\colon k_i=k\}|$ 表示状态簇 S_k 中所有的样本数量, 簇内残差均值可以使极端的单个样本残差被平均掉, 并代表簇内所有样本中最典型的修正方向. 通过将簇内残差均值与对应的原始动作簇中心 $\mu^a_{\mathrm{map}(k)}$ 结合, 得到微调后的簇级输出中心:

$$c_k=\mu^a_{\mathrm{map}(k)}+\bar{r}_k,\ k=1,\ldots,K^*_s \tag{30}$$

基于上述可构建出初步的 Mamdani 型安全模糊规则:

$$\mathrm{Rule}\ k:\text{若}\ s\in S_k,\ \text{则}\ a\in A_k,\ k=1,\ldots,K^*_s \tag{31}$$

每个前件模糊集 S_k 的隶属度函数采用 GMM 中第 k 个高斯分布所对应的密度函数, 即:

$$\mu_{S_k}(s)=\mathcal{N}(s\,|\,\mu^s_k,\Sigma^s_k) \tag{32}$$

其中, μ^s_k 和 Σ^s_k 是用 GMM 在状态空间拟合时得到的第 k 个簇的均值与协方差. 每个模糊规则的后件模糊集 A_k 的隶属函数采用以 c_k 为均值、标准差 σ_k 的高斯型隶属度函数:

$$\mu_{A_k}(a)=\exp\left(-\frac{\|a-c_k\|^2}{2\sigma_k^2}\right),\ \sigma_k=\sqrt{\frac{1}{|\{i\colon k_i=k\}|}\sum_{i\colon k_i=k}\left\|a_i-\mu^{\mathrm{a}}_{\mathrm{map}(k)}\right\|_2^2} \tag{33}$$

3.5 实时安全监控

基于构建的 Mamdani 型模糊规则, 本节在运行阶段对每个状态-动作对进行实时安全监控, 以提升系统在部署过程中的安全性. 具体而言, 首先根据当前状态在各状态簇中的隶属概率及其对应动作在映射动作簇中的隶属概率, 计算整体一致性度量, 用于刻画状态-动作对与已学习安全簇分布的符合程度, 并作为风险判定的量化依据. 随后, 将一致性度量与预设的安全阈值进行比较, 为不同风险水平设定一个可操作的判定边界: 若一致性高于阈值, 则认为该状态-动作对处于安全分布内, 可以直接执行; 若低于阈值, 则视为偏离安全簇分布, 从而触发 Mamdani 型模糊推理, 通过各规则前件的激活度与后件隶属度截断值生成局部输出, 经动作空间最大并集聚合并最终通过重心法去模糊化, 得到新的安全替换动作, 以避免潜在风险.

借鉴 Zhang 等人[33]将状态与动作隶属度进行组合计算以度量策略一致性的思路, 本文通过 GMM 后验概率分别获得状态与动作对各簇的隶属度, 然后取二者的较小值并在所有簇中选最大值, 得到整体一致性度量. 与文献 [33] 基于预定义理想动作隶属度不同, 本研究无需专家参考动作, 而是直接评估当前状态-动作对是否符合已学习的安全簇分布.

对于每条要执行的状态-动作对 (s_t, a_t), 首先计算状态和动作的一致性度量:

$$\mu_{\text{raw}}(s_t, a_t) = \max_{1 \leqslant k \leqslant K_s^*} \max_{j \in \mathcal{A}(k)} \min\left(\mathcal{N}(s_t \mid \mu_k^{s_t}, \Sigma_k^{s_t}), \mu_A(a_t; k, j)\right) \tag{34}$$

其中, $\mathcal{N}(s_t \mid \mu_k^{s_t}, \Sigma_k^{s_t})$ 表示状态 s_t 属于簇 S_k 的后验概率, $\mathcal{A}(k)$ 表示状态簇 S_k 所映射的一组允许动作簇索引, $\mu_A(a_t; k, j)$ 表示动作 a_t 在对应动作簇 A_j 上的隶属度. 该一致性度量本质上对应于模糊逻辑中的状态与动作联合隶属机制, $\min(\cdot,\cdot)$ 用于刻画状态与动作在同一映射簇内同时具有较高隶属度的情况, 而 $\max_k$ 则在所有候选簇中选择最优匹配, 以增强判定的鲁棒性. 其中动作隶属度 $\mu_A(a_t; k, j)$ 定义为基于后验概率的平滑形式:

$$\mu_A(a_t; k, j) = \exp[-\alpha_{\text{action}}(1 - p_{k,j}(s_t, a_t))] \tag{35}$$

其中, $p_{k,j}(s_t, a_t) = \dfrac{\mu_A[j]}{\max_\ell \mu_A[\ell]}$, $\mu_A[j]$ 为动作 GMM 模型输出的动作属于第 j 个动作簇的后验概率, $\alpha_{\text{action}} > 0$ 为平滑系数 (本文设置 α_{action}=0.1), 用于控制一致性随相对概率下降的衰减速度. 当当前动作在某个允许动作簇上的后验概率接近所有动作簇中的最大值时, $\mu_A(a_t; k, j)$ 接近 1, 表示该动作与安全簇分布一致. 由此得到的一致性度量描述了状态-动作对与已学习安全簇分布之间的贴合程度, 数值越高表示越接近历史安全模式、风险水平越低; 反之则意味着该行为偏离安全分布, 更可能引发潜在风险. 最后, 将该度量与阈值 τ 进行比较, 可完成对当前决策的安全性判定. 安全判断公式如下:

$$\begin{cases} \mu_{\text{raw}}(s_t, a_t) \geqslant \tau, & \text{安全} \\ \mu_{\text{raw}}(s_t, a_t) < \tau, & \text{不安全} \end{cases} \tag{36}$$

若判定不安全, 则基于 Mamdani 型模糊规则对 $(s_t, \cdot)$ 在动作空间上进行模糊推理, 令:

$$u_k(a) = \min\{\mu_{S_k}(s_t), \mu_{A_k}(a)\} \tag{37}$$

并在动作空间求最大并集, 得到替换动作的综合隶属度函数:

$$B'(a) = \max_{1 \leqslant k \leqslant K_s^*} u_k(a) = \max_{1 \leqslant k \leqslant K_s^*} \min\{\mu_{S_k}(s_t), \mu_{A_k}(a)\} \tag{38}$$

最后通过重心法对 $B'(a)$ 去模糊化, 得到具体的安全替换动作:

$$a_t^* = \frac{\int_{\mathbb{R}^{d_a}} a B'(a)\, \mathrm{d}a}{\int_{\mathbb{R}^{d_a}} B'(a)\, \mathrm{d}a} \tag{39}$$

本文构建的强化学习安全监控闭环执行流程如算法 1 所示, 在模糊规则构建阶段, 先收集并筛选成本为零的安全样本, 分别在状态和动作空间中用 GMM 聚类, 通过模糊映射熵与惩罚项自动选择最优簇数, 进而构建初步的状态簇到动作簇的映射字典, 并联合训练残差网络与判别器生成簇级动作中心, 构建出 Mamdani 型模糊规则. 在线闭环监控阶段, 智能体每次执行原策略输出动作后都要计算该状态-动作对的一致性度量并与阈值比较, 当低于阈值时立即触发模糊推理生成模糊输出, 并通过去模糊化后得到安全替换动作, 否则直接执行原策略动作, 从而实现对不安全行为的实时监控并显著提升系统安全性.

算法 1. 强化学习安全监控闭环执行流程.

输入: 预训练强化学习智能体策略 π, 状态簇数 K_s^*, 动作簇数候选集合 $\mathcal{K}_a=\{K_a^{(1)},K_a^{(2)},\ldots,K_a^{(n)}\}$, 惩罚系数 α, 安全阈值 τ;

输出: 一致性度量阈值 τ, 实时安全替换.

//模糊规则构建阶段

1. 收集交互轨迹 $\mathcal{T}=\{(s_i,a_i,c_i)\}$, 并筛选零成本样本 $\mathcal{D}_0=\{(s_i,a_i)\mid c_i=0\}$;
2. 对于给定的 K_s 和 K_a, 先利用公式 (13) 在状态与动作空间进行 GMM 拟合, 再依据公式 (14) 和 (15) 将样本硬分配到状态簇并保留动作软隶属度, 并按公式 (16) 在每个状态簇内聚合得到映射分布 $\tilde{\mu}_{kj}$;
3. 通过公式 (18) 计算模糊映射熵 $H_{\text{fmap}}(K_a)$, 通过公式 (21) 求解最优动作簇数 K_a^*;
4. 通过公式 (26) 构建初步的状态簇到动作簇的映射字典;
5. 定义残差网络 r_θ 与判别器 D_ϕ, 通过公式 (27)–(29) 联合训练并更新每簇中心 μ_k^{new};
6. 通过公式 (31)–(33) 构建 Mamdani 型模糊规则;

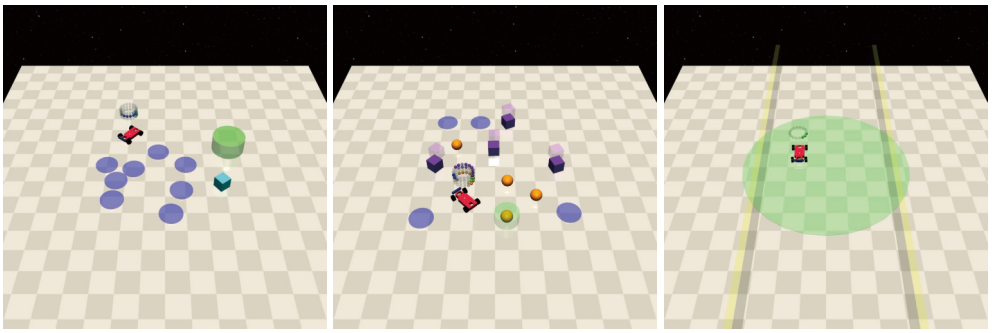
//在线闭环监控阶段

7. while 智能体执行动作 do
8. 　　按原策略执行动作 $a_t\sim\pi(s_t)$;
9. 　　通过公式 (34) 计算并输出一致性度量 μ_{raw};
10. 　　if $\mu_{\text{raw}}<\tau$ then
11. 　　　　通过公式 (38) 聚合生成模糊输出 $B'(a)$;
12. 　　　　通过公式 (39) 去模糊化获得 a_t^* 并执行该动作;
13. 　　else
14. 　　　　执行原策略动作 a_t;

4 实验分析

4.1 实验环境介绍

为了评估所提出监控框架在实际任务中的有效性, 本文选取开源强化学习环境库 Safety-Gymnasium[34]中的 3 个小车导航任务作为实验环境, 分别是 SafetyCarGoal1-v0 (目标导向导航任务)、SafetyCarButton1-v0 (目标激活任务) 和 SafetyCarCircle1-v0 (路径保持任务), 图 4 为这 3 个环境的示意图. 下面对 3 个环境进行介绍.



(a) SafetyCarGoal1-v0 环境　(b) SafetyCarButton1-v0 环境　(c) SafetyCarCircle1-v0 环境

图 4　实验环境示意图

• SafetyCarGoal1-v0: 状态向量维度 72, 包含当前位置、速度分量及与绿色目标柱的相对坐标; 动作为空间二维连续加速度, 回合上限 1 000 步. 任务要求在避开 8 个紫色陷阱和 1 个青色花瓶的同时到达目标, 奖励根据距离

变化给予正/负向反馈, 碰撞触发即时成本.

• SafetyCarButton1-v0: 状态维度 88, 比 SafetyCarGoal1 多了 4 个按钮的相对位置, 动作与步长设置相同. 智能体需避障并激活正确的黄色按钮, 错误激活或碰撞均会产生额外成本, 奖励规则与 SafetyCarGoal1-v0 任务一致.

• SafetyCarCircle1-v0: 状态维度 40, 包含位置、速度及与预设圆心半径的偏差; 动作同为二维连续加速度, 回合上限 500 步. 要求智能体沿圆形跑道行驶, 结合半径偏差惩罚和切向速度激励保持稳定轨迹, 碰撞或越界均触发即时成本.

4.2 基线算法

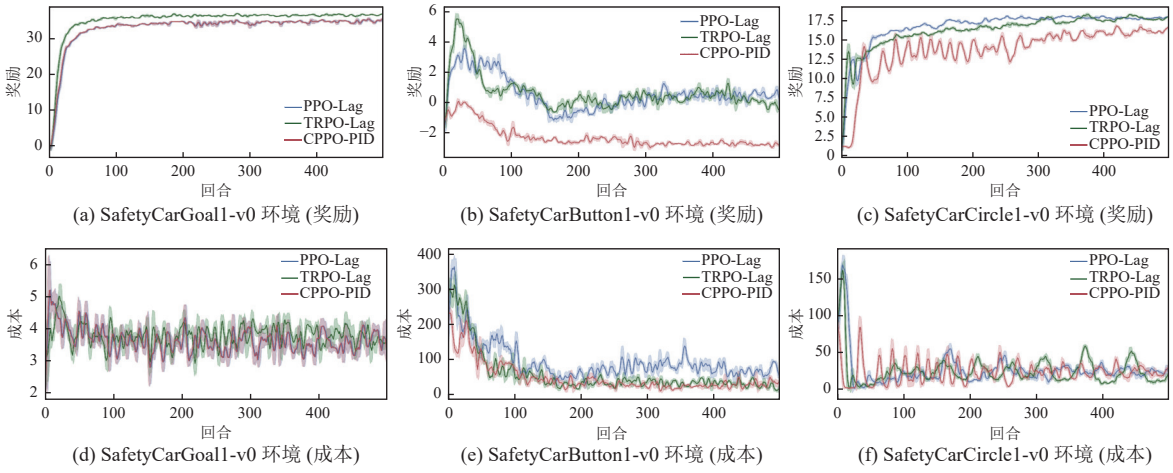
为了验证所提监控方法的效果, 本文以 3 种主流强化学习算法 PPO-Lag、TRPO-Lag 和 CPPO-PID 训练得到的 DRL 系统作为对比基线, 3 种算法简单介绍如下.

• PPO-Lag 是在经典的 PPO (proximal policy optimization) 算法基础上引入拉格朗日乘子机制, 通过在优化目标中添加对成本约束的软惩罚项, 实现了在最大化期望收益的同时约束累计成本不超上限. 该方法利用拉格朗日乘子动态调节成本惩罚权重, 在收益与安全约束之间寻求平衡.

• TRPO-Lag 则基于 TRPO (trust region policy optimization) 算法, 结合拉格朗日乘子框架, 采用二阶策略更新保证单调改进, 并在此基础上同时满足成本约束条件. 其核心优势在于利用信赖域方法稳定策略更新, 确保训练过程中的安全性与收敛性.

• CPPO-PID 是对 CPPO (constrained PPO) 的改进, 集成了经典的比例-积分-微分 (PID) 控制器以调整拉格朗日乘子的更新. 通过引入比例 (P)、积分 (I)、微分 (D) 这 3 项反馈, 该方法能够更精细地控制成本约束的动态变化, 缓解训练过程中成本波动过大的问题, 从而提升算法的稳定性和收敛速度.

在实验中, 针对 3 个导航环境, 分别以文献 [34] 设置的超参数训练了 PPO-Lag、TRPO-Lag 和 CPPO-PID, 其收敛曲线如图 5 所示, 均已稳定收敛. 值得注意的是, 在 SafetyCarGoal1-v0 环境中, PPO-Lag 与 CPPO-PID 曲线几乎重合, 这是因为该环境安全成本极低, 拉格朗日乘子权重波动不足以区分两者的优化目标.



(a) SafetyCarGoal1-v0 环境 (奖励) (b) SafetyCarButton1-v0 环境 (奖励) (c) SafetyCarCircle1-v0 环境 (奖励)

(d) SafetyCarGoal1-v0 环境 (成本) (e) SafetyCarButton1-v0 环境 (成本) (f) SafetyCarCircle1-v0 环境 (成本)

图 5 不同实验环境中 3 个算法的训练收敛过程

基于训练完成的 PPO-Lag 策略, 在每个环境中进行了 300 回合测试, 采集状态、动作和成本信息, 用来构建 Mamdani 型模糊规则库作为安全监控器, 监控器构建过程中的超参数如表 1 所示. 状态簇数固定为 30, 该设定综合考虑了 Safety-Gymnasium 环境中状态维度较高的特点, 通常在 40–88 维之间, 并兼顾了高维聚类过程的稳定性. 当簇数过大时, 状态空间划分过细会导致训练不稳定, 簇数过小则难以有效区分不同的状态分布. 动作簇数在 8–12 的范围内进行自动选择, 该范围的设定基于对动作空间特征的分析. Safety-Gymnasium 环境的动作维度较低, 一般只有 2–3 个连续控制量. 若簇数设置过多, 样本分布会过于稀疏, GMM 拟合容易退化, 若簇数设置过少, 则会限制替代动作的表达能力, 难以覆盖不同状态下的策略差异. 其中, 在最优动作簇的选择上, 计算了不同动作簇对

应的模糊映射熵, 表 2 展示了在 SafetyCarGoal1-v0、SafetyCarButton1-v0 和 SafetyCarCircle1-v0 这 3 种环境中, 不同的动作簇数对应的带惩罚项的归一化模糊映射熵值. 其中加粗数值表示模糊映射熵的最大值及其对应的最优动作簇数. 映射熵越大表明状态簇到动作簇的映射分布越均衡, 在 SafetyCarGoal1-v0 环境中, 动作簇设置为 11 对应的熵值最高, 在 SafetyCarButton1-v0 和 SafetyCarCircle1-v0 环境中, 动作簇设置为 8 对应的熵值最高, 因此, 将上述簇数分别作为各环境的最优动作簇数. 在监控器评估方面, 通过将所提闭环安全监控框架嵌入 3 种基线算法的测试阶段, 进行 500 回合的测试.

表 1 监控器超参数设置

参数	设置	参数	设置
采集轨迹回合数	300	判别器学习率	1×10^{-4}
监控回合数	500	Adam优化器参数 β_1, β_2	0.9, 0.99
状态簇数	30	对抗损失权重 α_{adv}	0.1
动作簇数候选集合	{8, 9, 10, 11, 12}	批量大小	64
GMM最大迭代次数	100	归一化模糊映射熵的惩罚函数 $\Phi(K_a)$	$K_a - K_{a,\min}$
GMM收敛阈值	1×10^{-5}	归一化模糊映射熵的惩罚权重 α	0.4
GMM初始均值选取采用方法	K-means	一致性度量的安全阈值 τ	{0.1, 0.3, 0.5, 0.7, 0.9}
残差网络学习率	1×10^{-4}	—	—

表 2 不同动作簇数对应的模糊映射熵值

环境	动作簇数	模糊映射熵
SafetyCarGoal1-v0	{8, 9, 10, **11**, 12}	{0.971, 0.966, 0.960, **0.972**, 0.961}
SafetyCarButton1-v0	{**8**, 9, 10, 11, 12}	{**0.950**, 0.943, 0.946, 0.944, 0.947}
SafetyCarCircle1-v0	{**8**, 9, 10, 11, 12}	{**0.952**, 0.943, 0.933, 0.929, 0.927}

4.3 监控预警性能对比

为了量化所提出在线监控模块对真实风险的覆盖能力, 本文采用预警覆盖率作为评价指标. 在在线监控阶段, 每当智能体在时间步 t 的真实成本 $c_t > 0$ 时, 即发生了真实风险, 将该时刻标记为真实风险时刻 ($R_t = 1$); 同时, 当一致性度量 $\mu_{\text{raw}}(s_t, a_t)$ 低于安全阈值 ϵ 时, 视为监控模块成功发出预警 ($\hat{R}_t = 1$). 通过对所有时间步进行上述标记, 可分别统计出真实风险时刻与成功预警时刻的次数. 预警覆盖率 (*Recall*) 定义为: 在所有真实风险时刻中, 被监控模块成功预警的比例, 其计算公式如下:

$$Recall = \frac{\sum_{t=1}^{T} \mathbf{1}(c_t > 0 \wedge \mu_{\text{raw}}(s_t, a_t) < \epsilon)}{\sum_{t=1}^{T} \mathbf{1}(c_t > 0)},$$

其中, $\mathbf{1}(\cdot)$ 表示指示函数, 条件成立取 1, 否则取 0. 预警覆盖率 *Recall* 值指标越高, 说明监控模块对真实高风险动作的检测能力越强, 能够在大多数风险发生前及时发出预警. 表 3 展示了本文所提监控框架在 3 个环境中对 3 种预训练算法 PPO-Lag、TRPO-Lag 和 CPPO-PID 测试 500 回合的预警覆盖率. 总体来看, 所提监控方法在不影响原始策略执行的情况下, 均能为 3 个算法提供较高的风险预警覆盖率.

4.4 安全监控框架性能

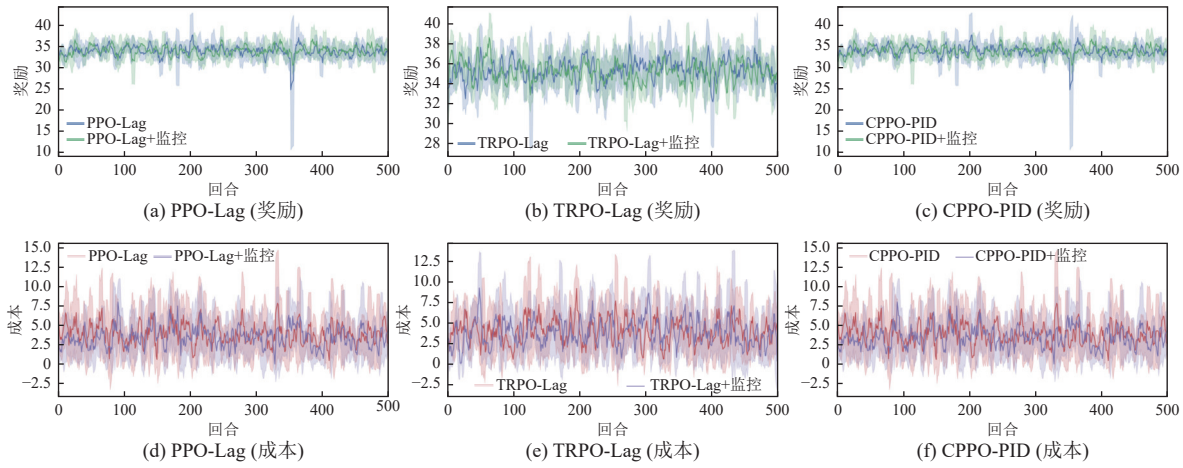
本文进一步分析了监控框架进行动作干预前后智能体的奖励与成本变化情况, 如表 4 所示, 可以观察到, 在线监控干预后各算法在 3 个环境中的奖励变化较小或有所提升, 而成本均不同程度下降. 在 SafetyCarGoal1-v0 中, 3 种算法的平均奖励基本稳定, 成本出现小幅下降; 在 SafetyCarButton1-v0 和 SafetyCarCircle1-v0 中, 所有算法的奖励都有一定提升, 成本显著减少. 总体来看, 在线监控与动作修正能够在维持或改善任务表现的同时, 有效降低安全风险成本. 图 6–图 8 分别展示了 3 个环境中各算法干预前后的奖励和成本对比情况.

表 3 在不同环境中对不同算法的预警性能对比 (%)

环境	算法	预警覆盖率
SafetyCarGoal1-v0	PPO-Lag	83.19
	TRPO-Lag	87.57
	CPPO-PID	83.46
SafetyCarButton1-v0	PPO-Lag	86.69
	TRPO-Lag	81.23
	CPPO-PID	85.68
SafetyCarCircle1-v0	PPO-Lag	96.37
	TRPO-Lag	97.80
	CPPO-PID	92.07

表 4 在不同环境下监控干预前后的奖励和成本对比

环境	算法	未监控		监控后	
		奖励	成本	奖励	成本
SafetyCarGoal1-v0	PPO-Lag	33.77	3.90	**33.88**	**3.48**
	TRPO-Lag	**35.33**	3.98	35.20	**3.85**
	CPPO-PID	33.77	3.90	**33.88**	**3.48**
SafetyCarButton1-v0	PPO-Lag	0.35	358.83	**0.57**	**238.42**
	TRPO-Lag	−1.36	68.54	**2.51**	**65.46**
	CPPO-PID	−2.62	76.35	**−0.02**	**62.12**
SafetyCarCircle1-v0	PPO-Lag	12.67	289.85	**13.23**	**273.1**
	TRPO-Lag	13.51	279.00	**13.99**	**266.09**
	CPPO-PID	14.54	79.00	**14.57**	**75.41**



(a) PPO-Lag (奖励) (b) TRPO-Lag (奖励) (c) CPPO-PID (奖励)

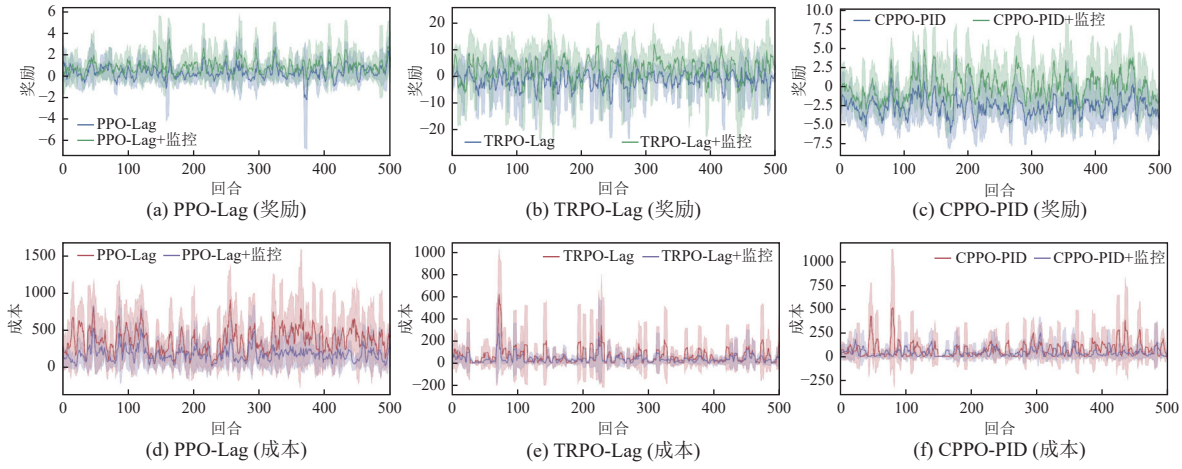
(d) PPO-Lag (成本) (e) TRPO-Lag (成本) (f) CPPO-PID (成本)

图 6 SafetyCarGoal1-v0 环境中不同算法监控前后的奖励和成本对比

4.5 在线监控对成本分布的影响

为了直观展示在线监控模块对成本分布的影响, 本文将每回合的预测成本划分为低风险、中风险和高风险这 3 个区间, 不同环境的风险阈值分别为: SafetyCarGoal1-v0 环境中, 设置低风险预测成本 $\hat{c} \leqslant 1$, 中风险 $1 < \hat{c} \leqslant 3$, 高风险 $\hat{c} > 3$; SafetyCarButton1-v0 环境中, 设置低风险 $\hat{c} \leqslant 20$, 中风险 $20 < \hat{c} \leqslant 300$, 高风险 $\hat{c} > 300$; SafetyCarCircle1-v0 环境中, 设置低风险 $\hat{c} \leqslant 80$, 中风险 $80 < \hat{c} \leqslant 200$, 高风险 $\hat{c} > 200$. 图 9 中对比了 3 种环境下各算法在实施监控前后的成本分布情况, 在 SafetyCarGoal1-v0 中, 未监控时高风险回合数最多, 监控后高风险回合显著减少, 低风险和中

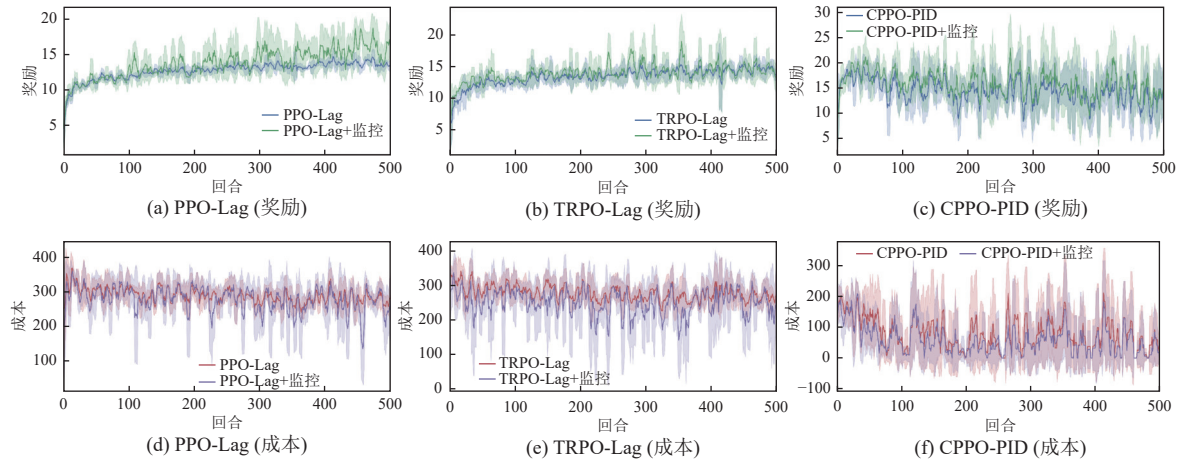
风险回合同步上升; 在 SafetyCarButton1-v0 中, 监控干预同样将大量高风险回合转移至低风险和中风险区间, 且 TRPO-Lag 与 CPPO-PID 的高风险回合下降尤为明显; 在 SafetyCarCircle1-v0 中, 监控后各算法的低风险和中风险回合均有增长, 高风险回合数普遍下降. 以上结果表明, 在线监控模块能够通过实时预警与动作修正, 有效抑制高成本动作的发生, 将更多回合保持在安全范围内, 从而优化整体成本分布.



(a) PPO-Lag (奖励) (b) TRPO-Lag (奖励) (c) CPPO-PID (奖励)

(d) PPO-Lag (成本) (e) TRPO-Lag (成本) (f) CPPO-PID (成本)

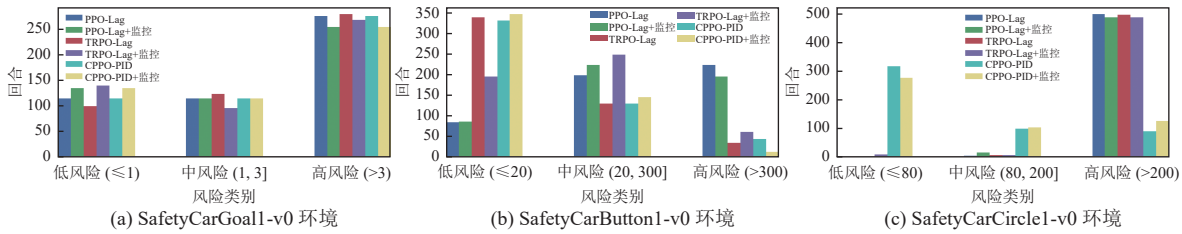
图 7 SafetyCarButton1-v0 环境中不同算法监控前后的奖励和成本对比



(a) PPO-Lag (奖励) (b) TRPO-Lag (奖励) (c) CPPO-PID (奖励)

(d) PPO-Lag (成本) (e) TRPO-Lag (成本) (f) CPPO-PID (成本)

图 8 SafetyCarCircle1-v0 环境中不同算法监控前后的奖励和成本对比



(a) SafetyCarGoal1-v0 环境 (b) SafetyCarButton1-v0 环境 (c) SafetyCarCircle1-v0 环境

图 9 不同算法监控前后的成本分布比较

4.6 安全阈值参数分析

为了进一步评估所提方法对安全约束强度的敏感性, 本节在 SafetyCarGoal1-v0 环境下, 基于 PPO-Lag 算法考

察不同安全阈值参数下的性能表现. 实验中将安全阈值分别设定为{0.1, 0.3, 0.5, 0.7, 0.9}, 并记录在干预前后智能体的平均回报与累计成本. 结果如表 5 所示.

表 5　在 SafetyCarGoal1-v0 环境下针对 PPO-Lag 算法不同安全阈值的回报与成本比较

安全阈值	未监控		监控后		奖励变化率 (%)	成本变化率 (%)
	奖励	成本	奖励	成本		
0.1			32.12	4.05	−4.89	+3.85
0.3			33.45	4.26	−0.95	+9.23
0.5	33.77	3.90	**33.88**	**3.48**	**+0.33**	**−10.77**
0.7			33.58	3.65	−0.56	−6.41
0.9			33.71	3.93	−0.18	+0.77

从表 5 可以看出, 安全阈值的设置对智能体的性能与安全性具有显著影响. 当阈值过小时, 监控器可能会频繁触发干预, 不仅未能有效降低成本, 反而引入额外代价, 同时造成回报下降, 这可能是由于过度严格的约束削弱了策略的执行效果. 随着阈值的适度增大, 监控效果逐渐显现, 智能体在保持回报基本稳定的同时实现了成本的下降, 在安全与性能之间取得了较好的平衡. 当阈值进一步增大时, 虽然回报基本保持不变, 但成本削减效果可能逐渐减弱, 甚至出现约束过宽而导致监控效果下降的情况. 由此可见, 安全阈值在性能与安全之间起到了关键的调节作用, 适中的取值能够更可能实现两者的折中效果.

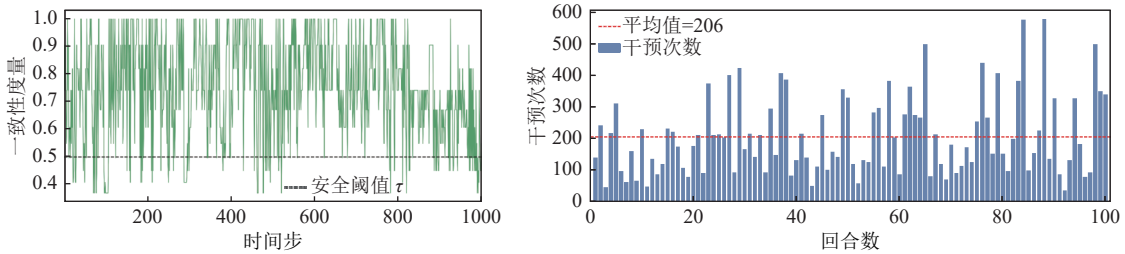
4.7　监控干预比例分析

本文进一步对所提监控框架最小干预特性进行了分析, 通过引入干预比例 (intervention rate, *IR*) 指标, 来量化监控器在测试过程中对原策略输出的调整频率. *IR* 定义如下:

$$IR = \frac{N_{\text{intervene}}}{N_{\text{steps}}} \times 100\%,$$

其中, $N_{\text{intervene}}$ 表示在在线执行阶段, 一致性度量 $\gamma(s_t, a_t)$ 低于安全阈值 τ 并触发模糊推理替换动作的次数, N_{steps} 为智能体在测试过程中执行的总步数. *IR* 值反映监控器在多大程度上对原策略进行了修正, 数值越低表示监控器干预越少, 即仅在必要时才对潜在不安全的动作进行调整.

图 10 展示了 SafetyCarButton1-v0 环境中 PPO-Lag 算法在监控阶段的表现. 其中, 图 10(a) 为单回合一致性度量随时间步的变化曲线, 可以看到 μ_{raw} 在安全阈值上下波动, 当其低于阈值时触发监控干预; 图 10(b) 为 100 个回合的干预次数统计结果, 每个回合包含 1 000 个时间步, 干预次数多集中在 200 次左右, 表明监控器在该场景下需较为频繁地修正潜在风险动作, 以确保执行阶段的安全性. 表 6 总结了不同环境与算法下的监控干预比例. 结果表明, 监控器在各环境中均能保持相对稳定的干预水平. 其中, PPO-Lag 算法在各类任务中表现出更高的策略稳定性和一致性, 干预频率相对较低; 而 TRPO-Lag 与 CPPO-PID 在部分任务中的干预比例更高, 表明其策略分布波动较大, 监控器需要更频繁地介入以维持系统安全. 综合来看, 监控器在运行过程中能有效识别风险并在必要时进行干预, 体现了所提框架的最小干预特性.



(a) 单回合监控一致性度量　　(b) 各回合监控干预次数分布 (τ=0.5)

图 10　SafetyCarButton1-v0 环境中 PPO-Lag 算法监控结果

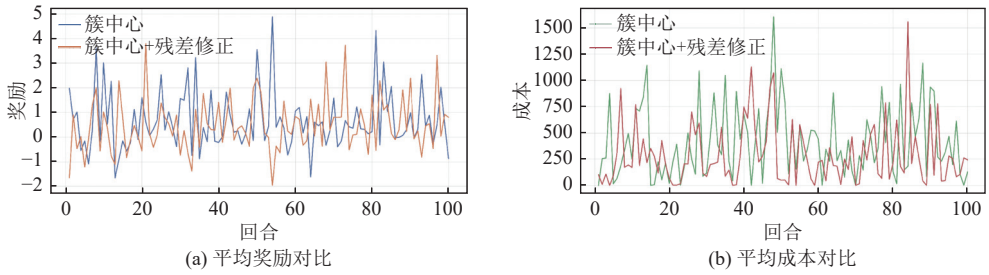
表 6 各环境下不同算法的监控干预比例 (%)

环境	算法	监控干预比例
SafetyCarGoal1-v0	PPO-Lag	26.4
	TRPO-Lag	41.3
	CPPO-PID	26.4
SafetyCarButton1-v0	PPO-Lag	20.6
	TRPO-Lag	61.8
	CPPO-PID	59.1
SafetyCarCircle1-v0	PPO-Lag	20.7
	TRPO-Lag	19.8
	CPPO-PID	15.3

4.8 基于残差的动作对抗分布对齐消融实验

为验证所提出的基于残差的动作对抗分布对齐机制的有效性, 本文在 SafetyCarButton1-v0 环境下, 以 PPO-Lag 策略为基础进行了消融实验. 该机制旨在通过残差网络实现动作的细粒度修正, 以弥补由安全样本聚类得到的动作簇中心在刻画局部状态差异方面的不足. 同时, 为防止残差修正导致生成动作偏离原始策略分布, 设计了对抗判别器以在统计层面约束修正后动作的分布, 实现局部拟合精度与全局一致性的动态平衡.

实验对比了两种设置: (1) 仅使用安全样本簇中心生成动作的基线方法; (2) 在簇中心基础上引入残差修正与对抗分布对齐的完整机制. 图 11(a) 展示了两种方法在 SafetyCarButton1-v0 环境下 100 回合测试的平均奖励对比结果, 图 11(b) 展示了对应的平均成本变化. 从结果可见, 加入残差修正与对抗对齐后, 智能体的平均奖励显著提高、平均成本明显下降, 说明该机制能够在保持策略分布一致性的同时增强局部动作表达能力与安全鲁棒性, 为模糊安全监控提供更精确的动作分布支撑.



(a) 平均奖励对比 (b) 平均成本对比

图 11 SafetyCarButton1-v0 环境下 PPO-Lag 算法的动作对抗分布对齐消融结果

5 总 结

本文提出了一种基于模糊映射熵的闭环强化学习安全监控框架, 通过离线规则学习和在线轻量干预相结合的方式, 在无需环境动力学模型的前提下, 为连续动作空间的深度强化学习系统提供了实时的安全保障. 离线阶段利用 GMM 软聚类与归一化模糊映射熵自动选簇, 并通过残差网络-对抗判别器微调簇中心, 构建安全的 Mamdani 模糊规则库; 在线阶段采用一致性度量触发去模糊化替换, 实现最小的动作修正. 实验证明, 在 Safety-Gymnasium 3 个导航任务中, 该框架在保持原有回报基本不变甚至略有提升的同时显著降低累计成本. 成本分布分析进一步说明, 监控器能够有效抑制高风险回合并提升低风险回合占比, 从而改善 DRL 系统的整体安全性.

尽管所提框架在实时性方面具有明显优势, 但其规则学习主要依赖于被判定为安全的轨迹样本, 而这些样本并不能真实反映未来风险, 一些当前看似安全的状态在后续仍可能演化为高风险状态. 后续工作可在规则构建阶段显式纳入高风险样本, 并以可行控制输入为依据学习更准确的安全边界; 此外, 通过引入基于在线反馈的规则微调机制, 使监控器能够随运行数据不断修正规则, 从而在复杂环境中获得更强的适应性与鲁棒性.

References

[1] Zhou ZY, Liu GJ, Tang Y. Multiagent reinforcement learning: Methods, trustworthiness, applications in intelligent vehicles, and challenges. IEEE Trans. on Intelligent Vehicles, 2024, 9(12): 8190–8211. [doi: 10.1109/TIV.2024.3408257]

[2] Zhang ZQ, Li HJ, Chen TT, Sze NN, Yang WZ, Zhang YH, Ren G. Decision-making of autonomous vehicles in interactions with jaywalkers: A risk-aware deep reinforcement learning approach. Accident Analysis & Prevention, 2025, 210: 107843. [doi: 10.1016/j.aap.2024.107843]

[3] Tang C, Abbatematteo B, Hu JH, Chandra R, Martín-Martín R, Stone P. Deep reinforcement learning for robotics: A survey of real-world successes. In: Proc. of the 39th AAAI Conf. on Artificial Intelligence. Philadelphia: AAAI, 2025. 28694–28698. [doi: 10.1609/aaai.v39i27.35095]

[4] Guo WR, Liu GJ, Zhou ZY, Wang JC, Tang Y, Wang MM. Robust training in multiagent deep reinforcement learning against optimal adversary. IEEE Trans. on Systems, Man, and Cybernetics: Systems, 2025, 55(7): 4957–4968. [doi: 10.1109/TSMC.2025.3561276]

[5] Zhou ZY, Liu GJ, Guo WR, Zhou MC. Adversarial attacks on multiagent deep reinforcement learning models in continuous action space. IEEE Trans. on Systems, Man, and Cybernetics: Systems, 2024, 54(12): 7633–7646. [doi: 10.1109/TSMC.2024.3454118]

[6] Chow Y, Nachum O, Duenez-Guzman E, Ghavamzadeh M. A Lyapunov-based approach to safe reinforcement learning. In: Proc. of the 32nd Int'l Conf. on Neural Information Processing Systems. Montréal: Curran Associates Inc., 2018. 8103–8112.

[7] Zhao LQ, Gatsis K, Papachristodoulou A. Stable and safe reinforcement learning via a barrier-Lyapunov actor-critic approach. In: Proc. of the 62nd IEEE Conf. on Decision and Control (CDC). Singapore: IEEE, 2023. 1320–1325. [doi: 10.1109/CDC49753.2023.10383742]

[8] Pankayaraj P, Varakantham P. Constrained reinforcement learning in hard exploration problems. In: Proc. of the 37th AAAI Conf. on Artificial Intelligence. Washington: AAAI, 2023. 15055–15063. [doi: 10.1609/aaai.v37i12.26757]

[9] Jin P, Tian JX, Zhi DP, Wen XJ, Zhang M. TRAINIFY: A CEGAR-driven training and verification framework for safe deep reinforcement learning. In: Proc. of the 34th Int'l Conf. on Computer Aided Verification. Haifa: Springer, 2022. 193–218. [doi: 10.1007/978-3-031-13185-1_10]

[10] Marchesini E, Marzari L, Farinelli A, Amato C. Safe deep reinforcement learning by verifying task-level properties. arXiv:2302.10030, 2023.

[11] Kwon R, Kwon G. Safety verification of non-deterministic policies in reinforcement learning. IEEE Access, 2024, 12: 185768–185788. [doi: 10.1109/ACCESS.2024.3513459]

[12] Gross D, Spieker H. Probabilistic model checking of stochastic reinforcement learning policies. arXiv:2403.18725, 2024.

[13] Yang JF, Zhang M, Chen X, Li Q. Formal verification of probabilistic deep reinforcement learning policies with abstract training. In: Proc. of the 26th Int'l Conf. on Verification, Model Checking, and Abstract Interpretation. Denver: Springer, 2025. 125–147. [doi: 10.1007/978-3-031-82700-6_6]

[14] Yang ZX, Zheng JQ, Chen GH. Reinfier and reintrainer: Verification and interpretation-driven safe deep reinforcement learning frameworks. arXiv:2410.15127, 2025.

[15] Ernest N, Arnett T, Phillips Z. Formal verification of fuzzy-based XAI for strategic combat game. Complex Engineering Systems, 2023, 3(1): 4. [doi: 10.20517/ces.2022.54]

[16] Marzari L, Marchesini E, Farinelli A. Online safety property collection and refinement for safe deep reinforcement learning in mapless navigation. arXiv:2302.06695, 2023.

[17] Tappler M, Córdoba FC, Aichernig BK, Könighofer B. Search-based testing of reinforcement learning. arXiv:2205.04887, 2022.

[18] Zolfagharian A, Abdellatif M, Briand LC, Ramesh S. SMARLA: A safety monitoring approach for deep reinforcement learning agents. IEEE Trans. on Software Engineering, 2025, 51(1): 82–105. [doi: 10.1109/TSE.2024.3491496]

[19] Cai Y, Wan XH, Liu ZH, Zheng Z. DRLFailureMonitor: A dynamic failure monitoring approach for deep reinforcement learning system. In: Proc. of the 35th IEEE Int'l Symp. on Software Reliability Engineering (ISSRE). Tsukuba: IEEE, 2024. 487–498. [doi: 10.1109/ISSRE62328.2024.00053]

[20] Gong Z, Kumar A, Varakantham P. Offline safe reinforcement learning using trajectory classification. In: Proc. of the 39th AAAI Conf. on Artificial Intelligence. Philadelphia: AAAI, 2025. 16880–16887. [doi: 10.1609/aaai.v39i16.33855]

[21] Selim M, Alanwar A, El-Kharashi MW, Abbas HM, Johansson KH. Safe reinforcement learning using data-driven predictive control. In: Proc. of the 5th Int'l Conf. on Communications, Signal Processing, and Their Applications (ICCSPA). Cairo: IEEE, 2022. 1–6. [doi: 10.1109/ICCSPA55860.2022.10018994]

[22] Yang M, Liu GJ, Zhou ZY, Wang JC. Probabilistic automata-based method for enhancing performance of deep reinforcement learning systems. IEEE/CAA Journal of Automatica Sinica, 2024, 11(11): 2327–2339. [doi: 10.1109/JAS.2024.124818]

[23] Cheng WC. Conditional fuzzy entropy of maps in fuzzy systems. Theory of Computing Systems, 2011, 48(4): 767–780. [doi: 10.1007/

s00224-010-9268-5]

[24] Yan KS, Zeng FP. Conditional fuzzy entropy of fuzzy dynamical systems. Fuzzy Sets and Systems, 2018, 342: 138–152. [doi: 10.1016/j.fss.2017.12.011]

[25] Wu J. Similarity-weighted entropy for quantifying genetic diversity in viral quasispecies. Virus Evolution, 2025, 11(1): veaf029. [doi: 10.1093/ve/veaf029]

[26] Hu HF, Wang YN, Li Z, Tian Y, Ren YM. Link prediction based on the derivation of mapping entropy. Complexity, 2021, 2021(1): 4156832. [doi: 10.1155/2021/4156832]

[27] Yang KX, Chen EC, Lin ZJ, Zhou XY. Weighted topological entropy of random dynamical systems. arXiv:2207.09719, 2022.

[28] Tsukamoto M. New approach to weighted topological entropy and pressure. Ergodic Theory and Dynamical Systems, 2023, 43(3): 1004–1034. [doi: 10.1017/etds.2021.173]

[29] Li XM, Lipp J, Shakir A, Huang R, Li J. BMX: Entropy-weighted similarity and semantic-enhanced lexical search. arXiv:2408.06643, 2024.

[30] Zadeh LA. Probability measures of fuzzy events. Journal of Mathematical Analysis and Applications, 1968, 23(2): 421–427. [doi: 10.1016/0022-247X(68)90078-4]

[31] Mamdani EH, Assilian S. An experiment in linguistic synthesis with a fuzzy logic controller. Int'l Journal of Man-machine Studies, 1975, 7(1): 1–13. [doi: 10.1016/S0020-7373(75)80002-2]

[32] de Luca A, Termini S. A definition of a nonprobabilistic entropy in the setting of fuzzy sets theory. Information and Control, 1972, 20(4): 301–312. [doi: 10.1016/S0019-9958(72)90199-4]

[33] Zhang SY, Song HY, Wang QX, Pei Y. Fuzzy logic guided reward function variation: An oracle for testing reinforcement learning programs. arXiv:2406.19812, 2024.

[34] Ji JM, Zhang BR, Zhou JY, Pan XH, Huang WD, Sun RY, Geng YR, Zhong YF, Dai JT, Yang YD. Safety-Gymnasium: A unified safe reinforcemei learning benchmark. In: Proc. of the 37th Int'l Conf. on Neural Information Processing Systems. New Orleans: Curran Associates Inc., 2023. 18964–18993.

作者简介

杨敏, 博士生, 主要研究领域为强化学习安全性.

周子渊, 博士生, 主要研究领域为强化学习, 多智能体系统, 对抗性攻击.

李晓锋, 研究员, CCF 专业会员, 主要研究领域为可信软件, 软件自适应, 智能化软件工程.

刘关俊, 博士, 教授, 博士生导师, CCF 高级会员, 主要研究领域为 Petri 网, 模型检测, 形式化方法, 机器学习, 人机物系统, 工作流系统, 无人机协同系统.