基于交叉注意力扩散模型的对手建模

何明研，李斯源，刘鹏，黄剑华
（哈尔滨工业大学计算学部，黑龙江 哈尔滨 150001）

摘　要：对手建模作为多智能体博弈对抗的关键技术，其目的为学习对手的行为以减少环境的不确定性并帮助决策。然而，现有的对手建模方法大多采用离线训练加在线适应的结构，在离线训练中采用传统神经动力学模型一步步预测受控智能体的行为，容易形成单步误差进而形成累积误差，且在在线适应中面对未知对手时，亦会导致受控智能体计划状态偏离数据集分布。为解决上述问题，提出基于扩散模型并利用交叉注意力和对手建立关联的方法。该方法利用扩散模型可以同时生成多步规划序列这一特点解决了累积偏差问题，同时提出策略集的概念，通过在线微调的方式不仅解决了计划偏离问题，而且也解决了在线训练初始阶段会破坏离线策略的问题。在开放的密集奖励和稀疏奖励的竞争环境中的实验结果均充分证明了该方法卓越的性能。

关键词：对手建模；多智能体博弈对抗；扩散模型；交叉注意力机制；在线微调
源代码链接：https://github.com/1180300215/CADAO1
中图分类号：TP391.9　　**文献标志码：**A　　**DOI：**10.19678/j.issn.1000-3428.0070737

Opponent Modeling Based on Cross-Attention Diffusion Model

HE Mingyan, LI Siyuan, LIU Peng, HUANG Jianhua
(Faculty of Computing, Harbin Institute of Technology, Harbin 150001, Heilongjiang, China)

【Abstract】As a key technology in multi-agent game confrontation, opponent modeling aims to learn the behavior of the opponent to reduce the uncertainty of the environment and facilitate decision-making. However, most existing opponent modeling methods adopt the structure of offline training and online adaptation. During offline training, traditional neural dynamics model is used to predict the agent step-by-step, which facilitates the formation of single-step and cumulative errors. Additionally, when facing an unknown opponent during online adaptation, the planned states of the controlled agent deviate from the distribution of the dataset. To solve these problems, a method based on a diffusion model and cross-attention is proposed to establish a correlation with the opponent. The cumulative bias problem is solved using the feature that the diffusion model can generate multistep planning sequences simultaneously. The concept of a strategy set is proposed, and the deviation problem is not only solved by online fine-tuning but also prevents the problem that the offline strategy will be destroyed in the initial stages of online training. Experimental results in both open intensive reward and sparse reward competitive environments demonstrate the superior performance of this method.

【Key words】opponent modeling; multi-agent game confrontation; diffusion model; cross-attention mechanism; online fine-tuning

0　引言

对手建模[1-3]是指在合作、竞争的复杂任务场景中，智能体往往需要通过观察其他智能体，对其建立抽象模型，推理其行为、意图等要素，最终用于辅助自身决策时所涉及的方法。在游戏人工智能、军事仿真、自动驾驶、机器人集群控制等应用场景的多智能体系统中，该方法都起到了至关重要的作用。对手建模是多智能体强化学习[4-5]的一个重要分支，核心是帮助受控智能体利用对手策略或根据对手的隐藏信息制定更好的响应策略。在多智能体博弈中，往往因为对手策略的切换而造成环境的不稳定性，所以当采用对手建模方法时，更好地识别对手策略的变换以及较为准确地预测受控智能体的下一步动作显得极为重要。

然而，现有的大多数对手建模方法都是一步步地预测受控智能体的行为，比如 TAO[6]中作者利用 Transformer 自回归地预测受控智能体的下一步动作。但是，在使用神经动力学模型进行预测时，由于数据支持有限和随机环境转换，可能存在单步错误；

基金项目：国家自然科学基金青年项目(62306088)；黑龙江省自然科学基金优秀青年项目(YQ2024007)。
作者简介：何明研，男，硕士研究生，主研方向为多智能体博弈对抗；李斯源，副教授、博士；刘鹏、黄剑华(通信作者)，教授、博士。
收稿日期：2024-12-23　　**修回日期：**2025-02-20　**E-mail：**jhhuang@hit.edu.cn

累积单步误差可能使计划状态偏离数据集分布，可能会导致严重的复合错误。除此之外，大多数现有的工作[7-9]都是在离线数据集上训练后直接和之前没见过的对手进行交互，这都会导致受控智能体的计划状态偏离数据集分布，造成复合误差。还有一部分工作采用在线学习对手模型的设置，这是不切实际和低效的。这是因为在实际场景中，与竞争环境或对手策略进行在线学习并不总是可行的。例如，电子商务平台受到约束[10]，通过被动积累数据离线建模特定用户的行为。

基于此，本文提出一种基于序列化模型的对手建模研究方法——CADAO（Cross-Attention Diffusion Against Opponent）。该方法将对手建模问题分为离线训练和在线微调两个部分。在离线训练部分，本文将其分为编码对手策略和响应对手策略两个阶段，在编码对手策略阶段，使用对手轨迹进行策略学习，为下一阶段提供了良好的对手策略嵌入；在响应对手策略阶段，利用交叉注意力机制在对手策略嵌入和受控智能体之间建立关联。在离线测试阶段，本文提出了对手信息缓冲区的概念，使受控智能体能够获得良好的对手策略嵌入。在在线微调部分，探索中利用基于优先级的轨迹级缓冲区收集优先级高的轨迹，利用其进行微调，从而获得一个新的策略，并将其存到策略集中；当再次与对手进行交互时，从策略集中自适应地选择合适的策略。

本文主要贡献为：1）将扩散模型引入了对手建模领域，解决了使用神经动力学模型会带来的多步误差问题，使实验结果更加稳定，并使用交叉注意力机制使受控智能体和环境中其他智能体信息建立关联；2）提出了策略集的概念，解决了在线学习的初始阶段破坏离线策略的有用行为等潜在问题，同时利用优质缓冲区数据对受控智能体的状态进行了补充，解决了面对未知对手时计划状态偏离数据集分布问题；3）在稀疏奖励和密集奖励的任务中均进行了实验，将本文方法和一系列用于对手建模的离线强化学习算法进行了对比验证，结果表明，本文方法在面对“已知”和“未知”对手时的性能方面均明显优于现有的对手建模方法，尤其是在密集奖励的 MPE 环境中，取得了超越现有前沿方法的实验表现。

1 相关工作

1.1 对手建模相关工作

早期的对手建模研究主要集中在具有固定对手策略的简单环境中。随着非平稳环境的引入，现有的对手建模可以分为显式建模和隐式建模两种方法。

显式对手建模是指在交互过程中对对手策略、对手划分类型和在线检测及响应进行显式建模。ROSMAN 等[11]提出了用于多任务学习的贝叶斯策略重用，而 HERNANDEZ-LEAL 等[12]通过使用马尔可夫决策来建模对手并为未知对手策略添加检测机制，将其扩展到多智能体系统。基于此，在更复杂的环境中，ZHENG 等[13]使用神经网络对对手进行建模，并引入修正的信念模型来提高对手检测精度和速度。在这样的贝叶斯在线对手建模方法的基础上，YANG 等[14]将心智理论引入对手建模，利用高阶决策来对抗同样具备建模能力的对手。

隐式对手建模是指在训练期间提取用于表示学习的对手信息。HE 等[2]提出了一种端到端训练方法，通过使用深度神经网络将对手的观察结果与代理的观察结果合并。HONG 等[15]进一步加入了对手的动作信息，并通过神经网络拟合对手策略。考虑到对手也可能有学习行为，FOERSTER 等[1]利用循环推理来估计对手策略网络的参数，并最大化智能体的奖励。KIM 等[3]和 AL-SHEDIVAT 等[16]利用元学习来适应不断变化的对手。KIM 等[3]引入了一个与塑造其他智能体策略学习动态密切相关的术语，从而考虑智能体当前策略对未来对手策略的影响。为了更好地适应对手的未知策略，JING 等[6]利用了监督预训练 Transformer 的上下文学习能力。

现有的对手建模方法无论是离线训练还是在线建模都会造成复合误差问题，而扩散模型因其可以同时生成多步规划序列而成为一种可行的替代方案。

1.2 扩散模型相关工作

扩散模型[17]已成为一种十分强大的生成模型，在多个领域[18-20]取得了显著进展。在强化学习领域，扩散模型被应用于序列决策任务，特别是在离线强化学习中，用于拟合生成轨迹、规划未来轨迹、替换传统高斯策略、增强经验数据集、提取潜在技能等。

最近，已有一系列将扩散模型应用于顺序决策任务的工作特别关注离线强化学习。作为代表性工作，Diffuser[21]是适合离线数据集上轨迹生成的扩散模型，并通过引导采样规划所需的未来轨迹。在已有的一些工作中，扩散模型在强化学习管道中充

当不同的模块，例如替换传统的高斯策略[22]、增强经验数据集[23]、提取潜在技能[24]等，而且扩散模型促进的规划和决策算法在多任务强化学习[25]、模仿学习[25]和轨迹生成[26]等更广泛的应用中表现良好。

此外，一些工作还探索了条件生成建模在顺序决策制定中的应用，特别是在离线强化学习的环境下，通过使用条件生成模型（如扩散模型），可以绕过传统离线强化学习中的复杂动态规划过程，并解决价值函数估计的不稳定性等问题。Decision Diffuser[27]使用条件扩散模型来进行决策生成。这种方法不仅可以最大化回报，还可以灵活地结合多种约束和技能，以产生新的行为。这种方法在实验中显示出优于传统方法的性能，展示了条件生成模型在决策制定中的潜力。

更重要的是，扩散模型因强大、灵活的分布建模能力，在应对强化学习中长期存在的挑战方面显现出了巨大潜力。本文研究表明在对抗性领域应用扩散模型亦能带来显著优势。

2 问题定义及基本知识

2.1 博弈环境描述

本文使用部分可观察的随机博弈（POSG）[28]来形式化多智能体博弈环境。该环境由七元组$<I,S,\{O^i\}_{i\in I},A,T,\{R^i\}_{i\in I},\{\Omega^i\}_{i\in I}>$定义，其中：$I=\{1,2,\cdots,N\}$是智能体的集合；$S$是状态空间；$O^i$是智能体$i$的观察空间；$A=A^1\times A^2\times\cdots\times A^N$代表联合动作空间；$T:S\times A\times S\to[0,1]$表示转换函数，定义了给定先前状态和联合动作时下一个状态的概率分布；$R^i:S\times A\times S\to R$代表智能体$i$的奖励函数；$\Omega^i:S\times A\times O^i\to[0,1]$代表智能体$i$的观察函数，定义了给定先前状态和联合行动时下一个观察的概率分布。

本文利用上标1来表示受控智能体，上标−1来表示对手智能体，并假设对手策略源于一组固定的由脚本或强化学习算法预训练得到的策略集。

2.2 离线对手建模及在线微调

本文中离线数据集D的轨迹由对手策略$\pi^{-1,p}$和其近似最佳响应策略$\pi^{1,p,*}$相互作用产生，其包括了对手轨迹$T^{-1,p}(\tau^{-1,p}=(o^{-1,p},a^{-1,p},r^{-1,p},\cdots)\cdots)$和受控智能体轨迹$T^{1,p}(\tau^{1,p}=(o^{1,p},a^{1,p},r^{1,p},\cdots)\cdots)$。离线对手建模的目标是使用离线数据集预训练对手感知的自适应控制智能体策略$M_\theta(a^1|o^1;D)$，并将M_θ部署到具有未知测试对手策略的新环境中，使得受控智能体达到最大的预期回报（即累积奖励）：

$$\max_\theta E_{\pi^{-1}\sim\Pi^{\text{test}},D}\left[\sum_{t=0}^{\infty}R_t^1\mid a_t^1\sim M_\theta,\pi^{-1}\right]\quad(1)$$

式中：D是对手的信息数据，在离线预训练期间从离线数据集中采样获得，在在线测试期间收集获得。

在线微调过程即是收集和筛选在线交互数据的微调策略网络。

2.3 扩散概率模型

扩散概率模型是一种用于生成数据的机器学习模型，它模拟了一种从无序状态（如高斯噪声）逐步生成有序数据（如图像或文本）的过程。这个过程类似于物理中的扩散过程。扩散模型通过逐步减少噪声并逐步引入数据特征的方式工作，最终生成高质量的数据样本。这种方法已经被证明在生成高复杂度和高质量的数据方面非常有效，例如在图像和音频合成中。

扩散概率模型中的关键公式包括：

1）噪声添加公式：它描述了如何逐步将噪声添加到数据中。它通常是通过线性插值的形式实现的，如式（2）所示：

$$x_t=\sqrt{\alpha_t}x_{t-1}+\sqrt{1-\alpha_t}\varepsilon\quad(2)$$

式中：ε是高斯噪声；α_t是时间步t的噪声比例。

2）逆过程公式：在逆过程中，模型试图预测并去除噪声。这个过程的一个关键部分是估算给定噪声数据下的原始数据分布，通常表示为$p(x_{t-1}|x_t)$。

3 CADAO 方法

为解决对手建模中环境不稳定问题，本文提出了CADAO方法，总体框架图如图1所示（彩色效果见《计算机工程》官网HTML版，下同）。离线训练的对手建模共分为2个阶段：阶段1是利用Transformer学习表示对手策略的特征，获得有利于下游学习的对手策略嵌入；阶段2是利用扩散模型学习在面对当前对手策略时受控智能体应该采取的行动。对手建模在线微调部分是采取类似[29]的做法，通过与环境交互得到对未知对手策略的评估返回值，再以新的返回值和环境交互，并从中筛选出优质的交互轨迹和离线轨迹一起微调网络。

3.1 离线学习阶段1：基于 Transformer 的对手策略编码器

为了获得有利于下游学习的对手策略表示，本

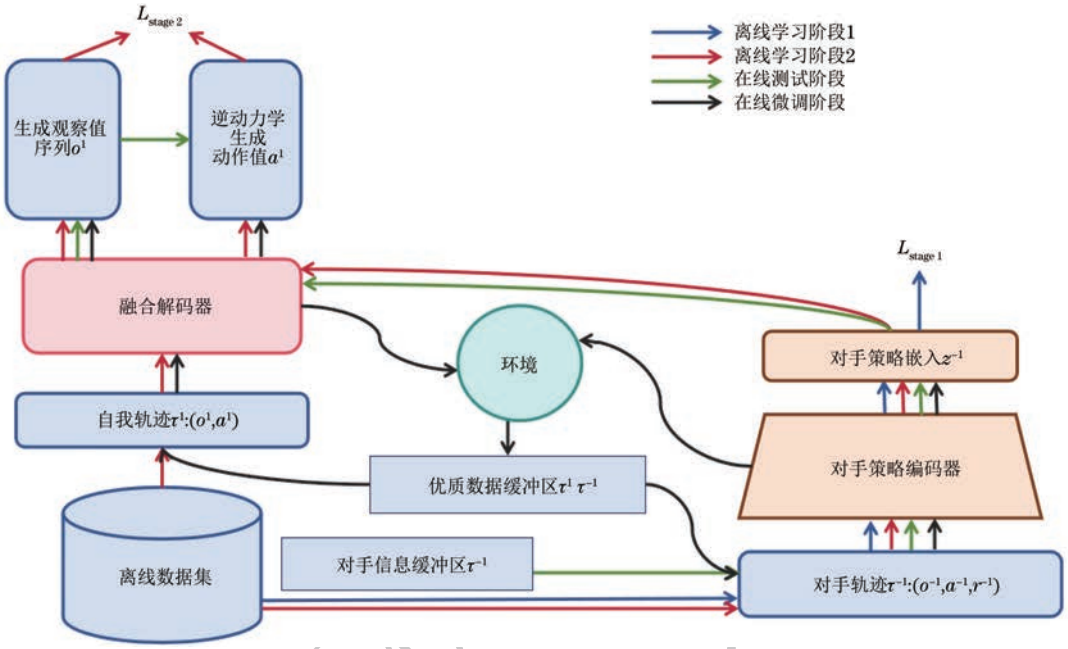

图 1 总体框架图

Fig. 1 General frame diagram

文利用基于 Transformer 的对手策略编码器(借用 TAO[6]的思想),使用离线数据集的对手轨迹进行策略学习(如图 2 所示)。本文在阶段 1 的损失由两部分组成:生成损失和判别损失。其中:生成损失有助于策略嵌入隐式识别对手策略;判别损失有助于策略嵌入区分难以区分的对手策略。

1)生成损失。

使用编码器 M_θ 和辅助解码器 N_φ 进行条件模仿学习[30]。对于给定的对手策略 $\pi^{-1,p}$,根据其观察 $o^{-1,p}$ 以及从其他轨迹中获得的平均轨迹嵌入 $z^{-1,p}$[指经过平均池化(AP)函数获得的轨迹嵌入]来预测其动作 $a^{-1,p}$,生成损失如式(3)所示。

2)判别损失。

使用与轨迹相对应的对手策略类型(即离线对手策略集中的索引)的标签来充分区分它们。本文使用点积来衡量两个不同平均轨迹嵌入之间的相似性。判别损失定义如式(4)所示。

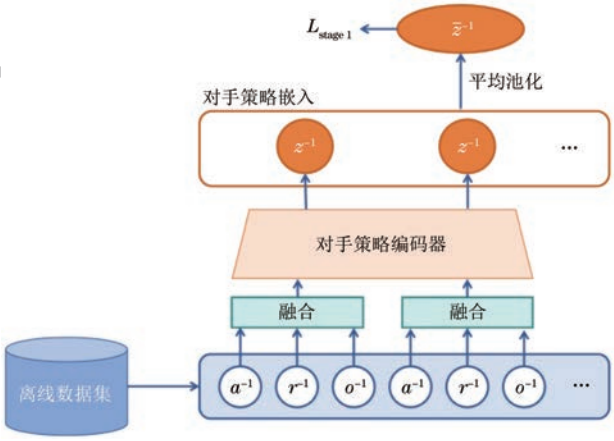

图 2 对手策略编码器

Fig. 2 Opponent policy encoder

$$L_{\text{gen}}=-\frac{1}{P}\sum_{p=1}^{P}E_{\tau_i,\tau_j\sim T^{-1,p}}\left[\sum_{(o,a)\sim\tau_i}\log_a N_\varphi(a\mid o,\text{AP}(M_\theta(\tau_J)))\right] \tag{3}$$

$$L_{\text{dis}}=-E_{T^{-1}}\left[\sum_{i=1}^{B}\frac{1}{F_i-1}\sum_{j\in F_i,j\neq i}\log_a\left(\frac{\exp(\bar{z}_i^{-1}\cdot\bar{z}_j^{-1}/p)}{\sum_{k\in F_i,k\neq i}\exp(\bar{z}_i^{-1}\cdot\bar{z}_k^{-1}/p)}\right)\right] \tag{4}$$

式中:F_i 指和 i 来自同一对手的数据;p 指温度因素;B 指一批数据的数量。

离线学习阶段 1 的损失函数定义如下:

$$L_{\text{stage 1}}=\alpha\cdot L_{\text{gen}}+\beta\cdot L_{\text{dis}} \tag{5}$$

式中:α 和 β 是平衡生成损失和判别损失的加权系数。

3.2 离线学习阶段 2:基于扩散模型的融合解码器

鉴于本文的目标是使用单个模型处理多个对手策略,所以需要一种机制识别当前对手的策略,然后产生适当的响应动作。为此,本文提出将对手策略嵌入信息和受控智能体信息通过交叉注意力[如式(6)所示]来建立联系,使受控智能体能感知到对

手策略的变化(如图 3 所示),具体计算过程如下:

$$\text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\text{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d}}\right)\cdot\boldsymbol{V} \tag{6}$$

$$\boldsymbol{Q}=\boldsymbol{W}_{\boldsymbol{Q}}^{(i)}\cdot\varphi_i(x_k) \tag{7}$$

$$\boldsymbol{K}=\boldsymbol{W}_{\boldsymbol{K}}^{(i)}\cdot z^{-1} \tag{8}$$

$$\boldsymbol{V}=\boldsymbol{W}_{\boldsymbol{V}}^{(i)}\cdot z^{-1} \tag{9}$$

在式(7)中 $\varphi(x)$ 指 U-Net 网络的中间层结果;在式(8)、式(9)中 z^{-1} 指对手策略嵌入。

在扩散过程中,由于智能体的动作本质上是离散的,且为了和历史轨迹建立联系,因此本文选择基于历史部分观察值进行扩散,定义如下:

$$x_k(\tau):=(o_{t-h},o_{t-h+1},\cdots,o_t,o_{t+1},\cdots,o_{t-H+1}) \tag{10}$$

然而,仅从扩散模型中采样观察值并不足以定义控制器,因此本文通过估计动作来推断受控智能体的策略,即给定两个连续的观察值 o_t 到 o_{t+1},利用逆动力学模型[31]生成从 o_t 到 o_{t+1} 的动作,具体定义如下:

$$a_t:=f_\phi(o_t,o_{t+1}) \tag{11}$$

为避免使用传统的分类器或动态规划过程,在融合解码器(如图 3 所示)中,本文使用具有低温采样的无分类器指导提取数据集中的高似然轨迹。为实现无分类器指导,以如下扰动噪声进行采样:

$$\varepsilon:=\varepsilon_\theta(x_k(\tau),\varnothing,k)+\omega(\varepsilon_\theta(x_k(\tau),y(\tau),k)-\varepsilon_\theta(x_k(\tau),\varnothing,k)) \tag{12}$$

式中:ω 作为比例参数试图增强和提取数据集中显示 $y(\tau)$ 的最佳轨迹部分。

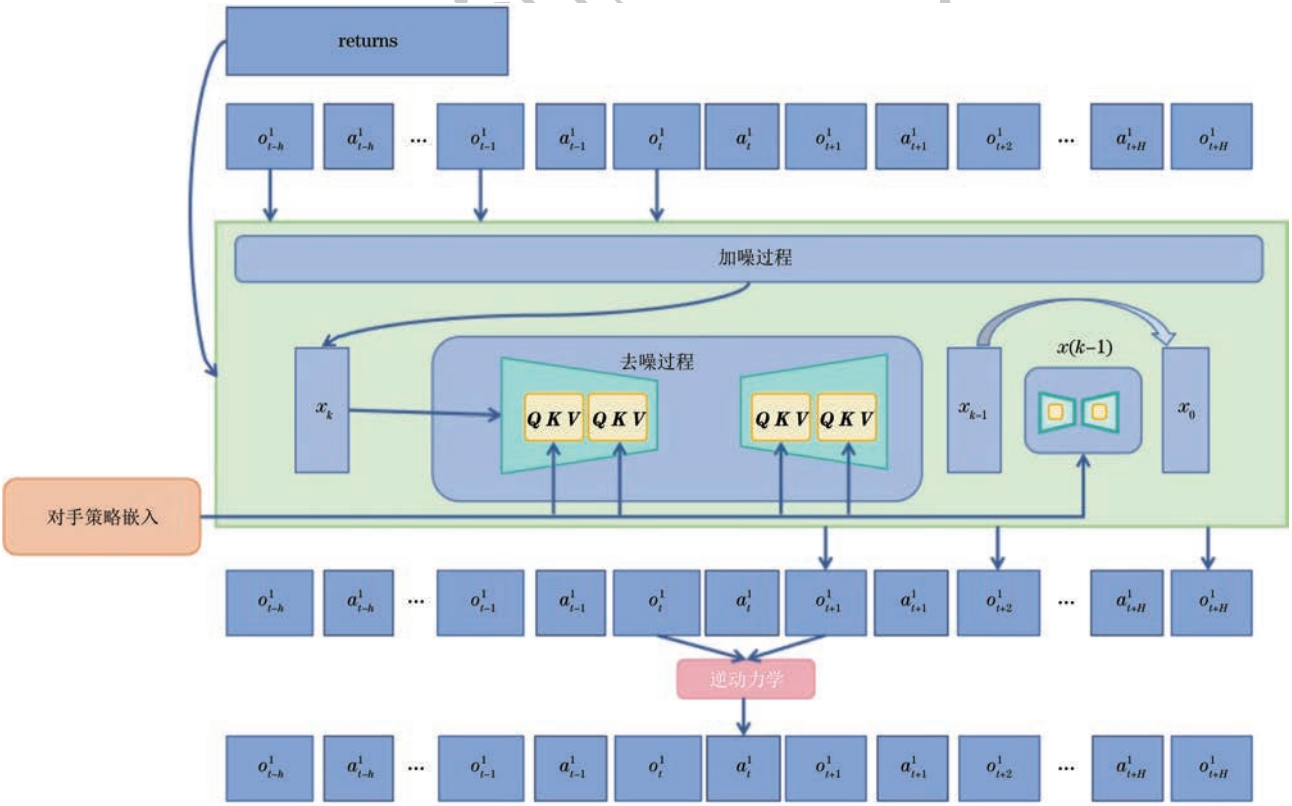

图 3 融合解码器

Fig. 3 Fusion decoder

如图 3 所示,本文以 returns 作为条件,使用扩散过程将观察值采样,并和来自离线数据集中的对手策略嵌入计算交叉注意力。最后,使用逆动力学模型识别应该采取的动作以达到最直接的预测状态。为了生成能最大化回报的轨迹,本文采用以轨迹回报为条件的噪声模型,利用与不同的对手策略交互的最大回报值作为条件,将噪声表示如下:

$$\varepsilon_\theta(x_k(\tau),y(\tau),k,z^{-1}):=\varepsilon_\theta(x_k(\tau),R(\tau),k,z^{-1}) \tag{13}$$

给定对手的数据时,本文通过使用对手策略编码器对其编码以获得良好的对手策略嵌入 z^{-1}。以 z^{-1} 作为对手策略嵌入信息,在去噪阶段和受控智能体信息通过交叉注意力建立关联,融合解码器可以做到识别对手当前的策略并作出相应的行动。其对应的损失如下:

$$L_{\text{diff}}=E_{k,\tau^{1,k}\sim T^{1,k},\beta\in\text{Bern}(p)}\left[\left\|\varepsilon-\varepsilon_\theta(x_k(\tau),(1-\beta)y(\tau)+\beta\cdot 0,k,M_\theta(\text{GetoffD}(T^{-1,k})))\right\|^2\right] \tag{14}$$

$$L_{\text{indyn}}=E_{(o,a,o')\in D}\left[\left\|a-f(o,o')\right\|^2\right] \tag{15}$$

$$L_{\text{stage 2}}=L_{\text{diff}}+L_{\text{indyn}} \tag{16}$$

式中：GetoffD 函数指的是从对手轨迹 $T^{-1,k}$ 中采取 C 个连续 H 大小的片段，这种采样方法是因为当给定对手策略时，其风格在连续时间步上可能更加明显，且这样可以展示不同片段的不同行为；D 指的是受控智能体的轨迹片段。当获得对手策略嵌入之后，以与当前对手策略交互的回报为条件，对噪声 ε 和时间步 k 取样，构造噪声观察数组 $x_k(\tau)$，对噪声进行预测。值得注意的是，β 指本文以 p 概率忽略条件信息，并且逆动力学是通过单个转换而不是轨迹来训练的。

在最终的在线测试阶段，为了应用从离线数据集中获得的知识，本文提出了对手信息缓冲区的概念，用来收集对手在最近 C 次交互中的轨迹信息，从而获得良好的对手策略嵌入。

3.3 在线微调阶段

本文提出一个策略集的概念，通过与未知对手策略进行交互，得到对未知对手策略的表示，经过微调离线阶段 1 之后，采用对扩散模型的微调修改受控智能体策略，如图 4 所示。

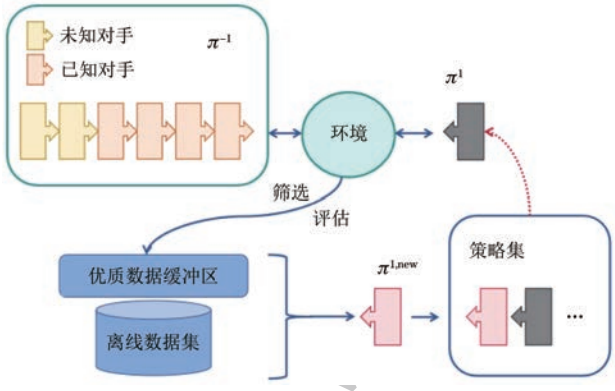

图 4 在线微调阶段

Fig. 4 Online fine-tuning stage

收集与未知对手策略交互的数据进行微调。具体思路为每次遇到新的未知对手策略时，收集与其交互的轨迹微调离线 1 阶段的对手策略编码器，并得到与这个未知对手策略交互的返回评估值，再以新的编码器，返回评估值和环境再次交互，筛选出一条条真实返回值大于返回评估值的轨迹，并将其存到优质轨迹集缓冲区中。然后从优质轨迹缓冲区和离线数据集中抽取数据对对手策略编码器以及融合解码器进行微调，得到新的策略，并将其存放到策略集中。在测试时，每次自适应地从策略集中选择恰当的策略网络和对手进行交互（当和新的对手交互时，会先通过评估阶段选择恰当的策略网络）。这样不仅能够保持离线阶段学习的行为，还可以在在线探索和学习过程中自适应地利用它，同时因为优质轨迹缓冲区中补充了受控智能体的状态集合，进而解决了在面对未知对手策略时，受控智能体的计划状态会偏离数据集分布的问题。

4 实验

4.1 环境及实验评价标准

本文考虑两个标志性的竞争环境基准：1）马尔可夫足球（MS）[32]，一个稀疏奖励的离散足球环境，它举例说明了两人零和游戏；2）粒子世界对手（PA）[33]，一个密集奖励的连续多粒子环境，它举例说明了一个三人非零和游戏。

MS 示意图如图 5 所示，受控智能体和对手的目标都是将球带入另一个目标（受控智能体的目标为蓝色，对手的目标为红色），当这种情况发生时游戏结束。奖励是稀疏的，成功提供 +1，失败提供 −1，在游戏结束时超过最大时间步长限制的实例提供 0 作为一次交互的返回值。

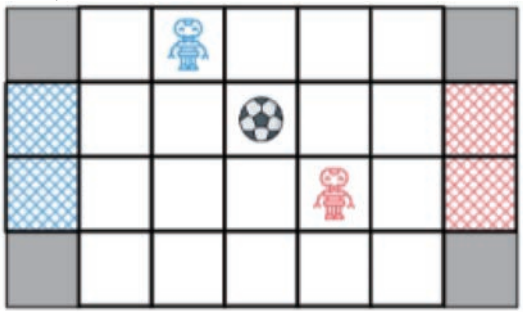

图 5 MS 环境

Fig. 5 MS environment

PA 示意图如图 6 所示，受控智能体的目标是最小化自己和目标地标之间的距离，同时阻止对手接近目标地标。对手的目标是最小化自己与未知的目标地标之间的距离，即对手不知道哪一个地标是目标地标。这就需要受控智能体战略性地分配自己的技能，以确保所有地标的覆盖范围，有效地欺骗对手。两个受控智能体和一个对手被赋予由其目标和状态引起的密集奖励，使用欧几里得距离作为返回值度量。

4.2 基线

为验证本文提出的基于扩散模型的对手建模方法的稳定性和优越性，本文将其与 5 个同样基于嵌入的对手建模基线方法进行比较。

1）DRON-concat[2]：使用辅助网络对对手的特征进行编码，将生成的对手的隐藏状态和受控智能

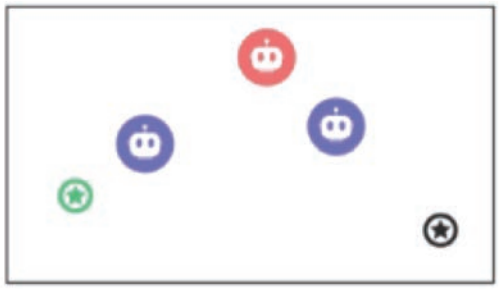

图 6 PA 环境

Fig. 6 PA environment

体观察获得的隐藏状态连接起来。

2)DRON-MoE[2]:将对手手工制作的特征和受控智能体的观察结果获得的隐藏状态视为门控信号,使用混合专家网络。

3)LIAM[34]:使用自动编码器对对手信息进行编码的对手建模方法,利用受控智能体的观察和动作来重建对手的观察和动作,从而将对手策略的语义嵌入潜在空间。

4)MeLIBA[35]:一种使用变分自动编码器(VAE)对对手信息进行编码的对手建模方法,使用 VAE 重建对手的未来动作来推断对手的近似信念。

5)TAO[6]:一种基于 Transformer 的对手建模方法,利用 Transformer 自回归地预测受控智能体的下一步行动。

4.3 实验描述

4.3.1 实验参数设置

在线测试阶段主要针对非平稳对手(非平稳指对手每 E 个回合切换策略)进行测试,并采用这 E 个回合的平均返回值作为实验结果。在测试时采用了 3 种设置:已知对手(等价于离线数据集中见过的对手策略),未知对手(包含以前未见过的对手策略),混合对手(前两种的混合)。对于所有测试实验,参数设置如表 1~表 3 所示。

表 1 离线学习阶段 1 参数

Table 1 Parameters of offline learning stage 1

参数名称	MS 环境	PA 环境
对手观察值维度/维	12	8
对手动作维度/维	5	5
轨迹最大长度	100	100
训练步骤/个	200	500
每批次训练数量/个	128	128
平衡系数 α	1	1
平衡系数 β	1	1
对手策略编码器的隐藏层节点数/个	32	32

表 2 离线学习阶段 2 和在线测试阶段

Table 2 Offline learning stage 2 and online testing stage

参数名称	MS 环境	PA 环境
受控智能体观察值维度/维	12	10
受控智能体动作维度/维	5	5
历史轨迹长度	4	0
扩散轨迹长度	8	20
扩散步骤/个	100	100
融合解码器的隐藏节点数/个	32	32
训练步骤/个	30 000	30 000
对手信息缓冲区大小	5	5
每批次训练数量/个	128	128
每 E 个回合切换策略数量/个	50	50
比例参数 ω	1.2	1.2

表 3 在线微调阶段参数

Table 3 Parameters of online fine-tuning stage

参数名称	MS 环境	PA 环境
收集的 N 个回合数据量	400	400
训练步骤/个	20 000	20 000
评估值	0.5	依具体策略而定

4.3.2 离线数据集描述

在所有环境中本文所描述已知策略均为 5 种,未知策略均为 3 种。在 PA 环境中,已知策略包括 FixOnePolicy、ChaseOnePolicy、MiddlePolicy、BouncePolicy、RLPolicy5 5 种,未知策略包括 FixThreePolicy、ChaseBouncePolicy、RLPolicy6 3 种,其中 RLPolicy5 和 RLPolicy6 是通过使用近端策略优化(PPO)[36]算法进行 5 000 和 10 000 个回合的自我演绎训练获得的。在 MS 环境中,已知策略包括 SnatchAttackPolicy、SnatchEvadePolicy、RLPolicy1、RLPolicy2、RLPolicy3 5 种,未知策略包括 GuardAttackPolicy、GuardEvadePolicy、RLPolicy4 3 种,其中 RLPolicy1 使用 TRCoPO[37]进行 30 000 个回合的自我演绎获得,RLPolicy2 使用 TRGDA[37]进行 10 000 个回合的自我演绎获得,RLPolicy3 和 RLPolicy4 通过使用 PPO 进行 50 000 和 100 000 个回合的自我演绎获得。

本文通过使用 PPO 算法进行预训练来导出对于每个已知对手策略 $\pi^{-1,p}$ 的近似最佳响应策略 $\pi^{1,p,*}$,并收集它们之间的交互数据作为离线数据集。

4.4 实验分析

为了全面分析本文提出的 CADAO 方法的效果,本文在 MS 环境和 PA 环境下进行了验证实验。

4.4.1 与其他前沿对手建模方法的比较

如表 4、表 5 所示(其中,最优指标值用加粗字

体标示，下同），很明显可以发现在 MS 和 PA 的 3 种在线测试设置中 CADAO 与其他基线相比，取得了非常具有竞争力的结果，尤其是在密集奖励任务的 PA 中，CADAO 显著优于其他方法，这表明 CADAO 中引入的扩散模型能够很好地解决之前方法所带来的复合误差问题。在这 3 种设置下，从表中不难发现 MeLIBA 和 LIAM 通常比 DRON-concat 和 DRON-MoE 表示出更好的结果，这很可能是因为重建对手的信息比直接编码能更有效地揭示对手和受控智能体策略之间的潜在关系；DRON-MoE 总体上超过了 DRON-concat，可能是因为 MoE 网络隐含地选择了比连接隐藏状态更合适的响应；TAO 比前几种基线方法性能好很可能是因为 Transformer 强大的上下文学习能力正是对手建模这个领域所需要的；CADAO 取得的结果表明其引入的扩散模型因多步规划序列能力很好地弥补了之前方法的累积误差问题。

表 4　MS 环境 3 种设置下平均返回值

Table 4　Average returns under three settings in MS environment

方法	已知对手	未知对手	混合
TAO	0.32	0.40	0.26
DRON-concat	0	−0.27	−0.16
DRON-MoE	0.05	0.31	0.10
LIAM	0.15	0.25	0.11
MeLIBA	0.12	0.28	0.13
CADAO	**0.45**	**0.53**	**0.37**

表 5　PA 环境 3 种设置下平均返回值

Table 5　Average returns under three settings in PA environment

方法	已知对手	未知对手	混合
TAO	105.6	109.5	101.6
DRON-concat	21.4	−40.0	−12.7
DRON-MoE	94.2	20.5	60.9
LIAM	83.6	60.5	70.5
MeLIBA	72.4	48.5	58.2
CADAO	**164.6**	**179.7**	**166.5**

4.4.2　消融实验

1）扩散模型解决了累积误差问题。

因在 MS 环境中受控智能体往往和对手在十多次交互中就分出胜负，因此本文以 PA 为例（没有终止状态），在每个包含百次交互的回合中查看累积误差造成的影响，如图 7 所示，其中，横轴指回合数，纵轴指当前回合的返回值。基于 Transformer 的 TAO 利用 Transformer 自回归地预测下一步动作，当其中某几步造成偏差后，就会越来越偏离离线数据集中的状态进而形成累积误差，而本文基于扩散模型的 CADAO 方法因其可以同时生成多步规划序列，所以相比之前方法稳定得多，相对来说不会有太大的起伏。

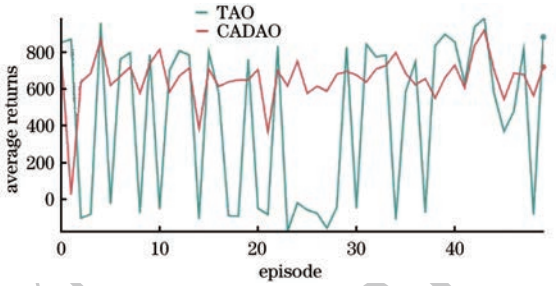

图 7　PA 环境下与 TAO 对比

Fig.7　Comparison with TAO in PA environment

2）使用策略集在线微调解决偏差问题。

如表 6、表 7 所示，不难发现，在两种环境设置下，本文利用缓冲区收集数据并按照优先级进行微调后，面对各种未知策略时的平均返回值比离线训练后直接部署到环境中的方法有了显著提高，并且在混合策略中的实验效果也有了一定提升，这表明利用缓冲区收集数据后解决了计划状态偏离数据集分布的问题。从表 8 中可以看出，当面对离线数据集中的已知策略时，本文提出的策略集通过收集微调后的策略的做法不仅在面对新的对手策略时有较好的性能提升，而且也缓解了在线学习的初始阶段破坏离线策略的有用行为这一问题。

表 6　MS 环境下平均返回值（策略集微调）

Table 6　Average returns in MS environment (strategy set fine-tuning)

策略	离线训练	策略集微调
未知策略 1	0.84	**0.95**
未知策略 2	0.86	**0.98**
未知策略 3	−0.01	**0**
混合策略	0.37	**0.41**

表 7　PA 环境下平均返回值（策略集微调）

Table 7　Average returns in PA environment (strategy set fine-tuning)

策略	离线训练	策略集微调
未知策略 1	36.7	**38.5**
未知策略 2	652.2	**672.3**
未知策略 3	32.7	**34.2**
混合策略	179.7	**191.3**

表 8　面对已知策略的平均返回值

Table 8　Average returns for facing known policies

环境	策略集	仅在线训练
MS	**0.32**	0.24
PA	**154.30**	128.70

3)使用交叉注意力机制建立联系。

如表 9、表 10 所示,除了在对手策略信息和受控智能体信息之间通过交叉注意力建立关联之外,本文还提出了将对手策略信息作为受控智能体观察值的一部分,将其拼接到受控智能体从环境中获取的观察值之后构成新的观察值直接通过神经网络(扩散模型)进行扩散。从结果中明显可以看出,直接拼接固然可以显式地建立联系,但结果远远没有通过交叉注意力建立关联好。这很可能是通过交叉注意力建立关联后可以使神经网络(扩散模型)在面对不同对手策略信息时通过不同的权重有更合适的响应。

表 9 MS 环境下平均返回值(交叉注意力)

Table 9 Average returns in MS environment (cross-attention)

策略	交叉注意力	concat 连接
已知策略	**0.45**	0.10
未知策略	**0.53**	0.20
混合策略	**0.37**	0.14

表 10 PA 环境下平均返回值(交叉注意力)

Table 10 Average returns in PA environment (cross-attention)

策略	交叉注意力	concat 连接
已知策略	**164.6**	56.9
未知策略	**179.7**	33.4
混合策略	**166.5**	45.2

5 结束语

本文提出了一种基于序列化模型的对手建模方法,该方法首先在离线阶段 1 利用对手的轨迹信息获得对手的策略嵌入,之后在离线阶段 2 利用扩散模型对受控智能体的观察值进行扩散,并通过交叉注意力机制,将其与阶段 1 获取的对手策略嵌入进行关联,在之后的在线测试阶段,利用缓冲区收集对手过往轨迹以获得良好的对手策略嵌入,从而识别对手。实验表明,引入扩散模型解决了在使用神经动力学模型对受控智能体预测下一步行动时出现的累积误差问题,在遇到新的对手策略时,利用策略集收集微调过的新策略,自适应地调用策略集中的策略,最终策略集很好地解决了初始在线学习会破坏离线策略这一问题。然而,利用扩散模型生成多步规划序列往往伴随着较高的时间消耗,这在许多需要即时交互的现实场景中会导致显著的延迟问题,也是未来研究中面临的一大挑战。

参考文献

[1] FOERSTER J N, CHEN R Y, AL-SHEDIVAT M, et al. Learning with opponent-learning awareness[EB/OL]. [2024-11-14]. https://arxiv.org/abs/1709.04326.

[2] HE H, BOYD-GRABER J, KWOK K, et al. Opponent modeling in deep reinforcement learning[EB/OL]. [2024-11-14]. https://arxiv.org/abs/1609.05559.

[3] KIM D K, LIU M, RIEMER M D, et al. A policy gradient algorithm for learning to learn in multiagent reinforcement learning[C]// Proceedings of International Conference on Machine Learning. [S. l.]: PMLR, 2021: 5541-5550.

[4] 王腾, 黄俊松, 王乐庭, 等. 基于 MADDPG 的多阵面相控阵雷达引导搜索资源优化算法[J]. 计算机工程, 2024, 50(11): 38-48.
WANG T, HUANG J S, WANG L T, et al. Multi-antenna phased array radar-guided search resource optimization algorithm based on MADDPG[J]. Computer Engineering, 2024, 50(11): 38-48. (in Chinese)

[5] 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[J]. 自动化学报, 2021, 47(7): 1610-1623.
SHI W, FENG Y H, CHENG G Q, et al. Research on multi-aircraft cooperative air combat method based on deep reinforcement learning[J]. Acta Automatica Sinica, 2021, 47(7): 1610-1623. (in Chinese)

[6] JING Y, LI K, LIU B, et al. Towards offline opponent modeling with in-context learning[C]// Proceedings of the 12th International Conference on Learning Representations. New York, USA: ACM Press, 2023: 1-13.

[7] 白天, 吕璐瑶, 李储, 等. 基于深度强化学习的游戏智能引导算法[J]. 吉林大学学报(理学版), 2025, 63(1): 91-98.
BAI T, LÜ L Y, LI C, et al. Game intelligent guidance algorithm based on deep reinforcement learning[J]. Journal of Jilin University (Science Edition), 2025, 63(1): 91-98. (in Chinese)

[8] JIANG H B, JIANG J C, LU Z Q, et al. Model-based opponent modeling[C]// Proceedings of the Advances in Neural Information Processing Systems. New Orleans, USA: Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2022: 28208-28221.

[9] 徐浩添, 秦龙, 曾俊杰, 等. 基于深度强化学习的对手建模方法研究综述[J]. 系统仿真学报, 2023, 35(4): 671-694.
XU H T, QIN L, ZENG J J, et al. Research progress of opponent modeling based on deep reinforcement learning[J]. Journal of System Simulation, 2023, 35(4): 671-694. (in Chinese)

[10] LI S, ZHAO H D. A survey on representation learning for user modeling[C]//Proceedings of the 29th International Joint Conference on Artificial Intelligence. Yokohama, Japan: International Joint Conferences on Artificial Intelligence Organization, 2020: 4997-5003.

[11] ROSMAN B, HAWASLY M, RAMAMOORTHY S. Bayesian policy reuse[J]. Machine Learning, 2016, 104(1): 99-127.

[12] HERNANDEZ-LEAL P, TAYLOR M E, ROSMAN B, et al. Identifying and tracking switching, non-stationary opponents: a Bayesian approach[C]// Proceedings of the AAAI Workshop on Multiagent Interaction without Prior Coordination. Palo Alto, USA: AAAI Press, 2016: 1-15.

[13] ZHENG Y, MENG Z P, HAO J Y, et al. A deep Bayesian policy reuse approach against non-stationary agents[C]// Proceedings of the 32nd International Conference on Neural Information Processing Systems. New York, USA: ACM Press, 2018: 962-997.

[14] YANG T P, MENG Z P, HAO J Y, et al. Towards efficient detection and optimal response against sophisticated

opponents[EB/OL]. [2024-11-14]. https://arxiv.org/abs/1809.04240.
[15] HONG Z W, SU S Y, SHANN T Y, et al. A deep policy inference Q-network for multi-agent systems [EB/OL]. [2024-11-14]. https://arxiv.org/abs/1712.07893.
[16] AL-SHEDIVAT M, BANSAL T, BURDA Y, et al. Continuous adaptation via meta-learning in nonstationary and competitive environments[EB/OL]. [2024-11-14]. https://arxiv.org/abs/1710.03641.
[17] SOHL-DICKSTEIN J, WEISS E A, MAHESWARANATHAN N, et al. Deep unsupervised learning using nonequilibrium thermodynamics[EB/OL]. [2024-11-14]. https://arxiv.org/abs/1503.03585.
[18] ROMBACH R, BLATTMANN A, LORENZ D, et al. High-resolution image synthesis with latent diffusion models [C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Washington D. C., USA: IEEE Press, 2022: 10674-10685.
[19] KONG Z F, PING W, HUANG J J, et al. DiffWave: a versatile diffusion model for audio synthesis [EB/OL]. [2024-11-14]. https://arxiv.org/abs/2009.09761.
[20] 陈子民, 关志涛. 基于条件扩散模型的图像分类对抗样本防御方法[J]. 计算机工程, 2024, 50(12): 296-305.
CHEN Z M, GUAN Z T. Image classification adversarial example defense method based on conditional diffusion model [J]. Computer Engineering, 2024, 50(12): 296-305. (in Chinese)
[21] JANNER M, DU Y L, TENENBAUM J B, et al. Planning with diffusion for flexible behavior synthesis [EB/OL]. [2024-11-14]. https://arxiv.org/abs/2205.09991.
[22] WANG Z D, HUNT J J, ZHOU M Y. Diffusion policies as an expressive policy class for offline reinforcement learning [EB/OL]. [2024-11-14]. https://arxiv.org/abs/2208.06193.
[23] LU C, BALL P, TEH Y W, et al. Synthetic experience replay[EB/OL]. [2024-11-14]. https://arxiv.org/abs/2303.06614.
[24] XIAN Z, GKANATSIOS N, GERVET T, et al. ChainedDiffuser: unifying trajectory diffusion and keypose prediction for robotic manipulation[EB/OL]. [2024-11-14]. https://www.semanticscholar.org/paper/ChainedDiffuser%3A-Unifying-Trajectory-Diffusion-and-Xian-Gkanatsios/c36f3635e090aba84e5e83b904a7697e83730be6.
[25] BAI C J, HE H R, LI X L, et al. Diffusion model is an effective planner and data synthesizer for multi-task reinforcement learning[C]//Proceedings of the Advances in Neural Information Processing Systems. New Orleans, USA: Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2023: 64896-64917.
[26] ZHANG M Y, CAI Z A, PAN L, et al. MotionDiffuse: text-driven human motion generation with diffusion model [EB/OL]. [2024-11-14]. https://arxiv.org/abs/2208.15001.
[27] AJAY A, DU Y L, GUPTA A, et al. Is conditional generative modeling all you need for decision-making? [EB/OL]. [2024-11-14]. https://arxiv.org/abs/2211.15657.
[28] YANG Y, WANG J. An overview of multi-agent reinforcement learning from game theoretical perspective [EB/OL]. [2024-11-14]. https://arxiv.org/pdf/2011.00583v3.
[29] ZHENG Q, ZHANG A, GROVER A. Online decision Transformer[C]//Proceedings of the International Conference on Machine Learning. [S. l.]: PMLR, 2022: 27042-27059.
[30] HUSSEIN A, GABER M M, ELYAN E, et al. Imitation learning: a survey of learning methods[J]. ACM Computing Surveys, 2018, 50(2): 1-35.
[31] AGRAWAL P, NAIR A, ABBEEL P, et al. Learning to poke by poking: experiential learning of intuitive physics [EB/OL]. [2024-11-14]. https://arxiv.org/abs/1606.07419.
[32] LANCTOT M, LOCKHART E, LESPIAU J B, et al. OpenSpiel: a framework for reinforcement learning in games [EB/OL]. [2024-11-14]. https://arxiv.org/abs/1908.09453.
[33] LOWE R, WU Y, TAMAR A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments [EB/OL]. [2024-11-14]. https://arxiv.org/abs/1706.02275.
[34] PAPOUDAKIS G, CHRISTIANOS F, ALBRECHT S V. Agent modelling under partial observability for deep reinforcement learning[C]//Proceedings of the International Conference on Neural Information Processing Systems. New York, USA: ACM Press, 2021: 19210-19222.
[35] ZINTGRAF L, DEVLIN S, CIOSEK K, et al. Deep interactive Bayesian reinforcement learning via meta-learning [EB/OL]. [2024-11-14]. https://arxiv.org/abs/2101.03864.
[36] SCHULMAN J, WOLSKI F, DHARIWAL P, et al. Proximal policy optimization algorithms[EB/OL]. [2024-11-14]. https://arxiv.org/abs/1707.06347.
[37] PRAJAPAT M, AZIZZADENESHELI K, LINIGER A, et al. Competitive policy optimization[EB/OL]. [2024-11-14]. https://arxiv.org/abs/2006.10611.

文字编辑 陆燕菲
栏目编辑 赖玉玲