控制与决策
Control and Decision

基于事后经验回放和策略延迟更新的深度强化学习充电枪装配策略

王福杰, 彭永岗, 李醒, 郭芳, 秦毅, 戚远航

引用本文:
王福杰, 彭永岗, 李醒, 等. 基于事后经验回放和策略延迟更新的深度强化学习充电枪装配策略[J]. 控制与决策, 2026, 41(5): 1439-1448.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2025.0386

基于事后经验回放和策略延迟更新的深度强化学习充电枪装配策略

王福杰[1]，彭永岗[1]，李 醒[2,3†]，郭 芳[1]，秦 毅[1]，戚远航[4]

(1. 东莞理工学院 计算机科学与技术学院，广东 东莞 523808；2. 东北大学 信息科学与工程学院，沈阳 110819；3. 东北大学 流程工业综合自动化国家重点实验室，沈阳 110819；4. 电子科技大学中山学院 计算机学院，广东 中山 528402)

摘 要: 为解决充电枪装配过程中传统强化学习算法训练样本效率低、策略不稳定和对硬件资源利用不充分的问题, 提出融合事后经验回放 (HER) 和策略延迟更新 (DPU)的软演员-评论家 (SAC) 算法 (SAC with HER-DPU). 首先, 建立充电枪装配模型, 通过在经验回放池中引入 HER, 重新定义目标以生成“伪成功”经验; 然后, 在算法的梯度更新部分加入 DPU, 通过多次更新价值网络后再更新策略网络, 确保策略更新基于更稳定的价值估计; 最后, 在使用 SAC with HER-DPU 算法进行充电枪装配训练时采用双线程训练架构, 将数据收集和神经网络训练解耦. 实验结果表明, 所提算法的收敛时间为 33.2 h, 平均装配步数为 75 步, 相较于 SAC 算法, 收敛时间减少 21.4 h, 平均装配步数少 17 步, 可有效提高训练的样本效率、策略稳定性和训练速度.

关键词: 深度强化学习；充电枪装配；软演员-评论家算法；事后经验回放；策略延迟更新

中图分类号: TP242 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2025.0386

引用格式: 王福杰, 彭永岗, 李醒, 等. 基于事后经验回放和策略延迟更新的深度强化学习充电枪装配策略 [J]. 控制与决策, 2026, 41(5): 1439-1448.

Deep reinforcement learning-based charging gun assembly strategy with hindsight experience replay and delayed policy updates

WANG Fu-jie[1]，*PENG Yong-gang*[1]，*LI Xing*[2,3†]，*GUO Fang*[1]，*QIN Yi*[1]，*QI Yuan-hang*[4]

(1. School of Computer Science and Technology， Dongguan University of Technology， Dongguan 523808，China； 2. School of Information Science and Engineering, Northeastern University， Shenyang 110819， China； 3. State Key Laboratory of Integrated Automation of Process Industries， Northeastern University， Shenyang 110819， China； 4. School of Computer Science， University of Electronic Science and Technology of China，Zhongshan Institute，Zhongshan 528402，China)

Abstract: To address the challenges of low sample efficiency, unstable policy updates, and insufficient hardware utilization in traditional reinforcement learning methods for charging gun assembly, we propose an enhanced soft actor-critic (SAC) algorithm that integrates hindsight experience replay (HER) and delayed policy updates (DPU). First, a charging gun assembly model is established. The HER is integrated into the replay buffer to redefine goals and generate "pseudo-success" experiences. Then, the DPU is applied during the gradient update phase, where the value network is updated multiple times before each policy update to ensure more stable value estimation. Finally, during training with the SAC-HER-DPU algorithm, a dual-thread architecture is adopted to decouple data collection from neural network training, improving overall training efficiency. Experimental results show that the proposed algorithm achieves convergence in 33.2 hours, with an average of 75 assembly steps. Compared to the baseline SAC algorithm, it reduces convergence time by 21.4 hours and decreases the average number of assembly steps by 17. Moreover, it effectively improves sample efficiency, policy stability, and training speed.

Keywords: deep reinforcement learning；charging gun assembly；SAC；HER；DPU

收稿日期: 2025-04-15；**录用日期:** 2025-07-25.

基金项目: 国家自然科学基金项目 (62203116, 62273095)；国家重点研发计划课题 (2024YFB3312403)；广东省基础与应用基础研究基金面上项目 (2024A1515010222, 2022A1515240058)；广东省普通高校重点领域专项 (2023ZDZX1040, 2022ZDZX1045).

责任编委: 张丹.

†通信作者. E-mail: lixing8245@163.com.

0 引 言

装配是工业机器人领域的研究热点[1-2], 对电动汽车的充电枪装配可以抽象成类圆形轴孔装配[3], 国内外学者对机器人轴孔装配方法的研究主要集中在两个方向: 基于轴孔模型和基于自主学习的装配研究. 但是, 基于模型的装配方法难以被广泛应用于轴孔模型的装配任务, 原因在于轴孔之间摩擦力模型难以精确建立, 不同的轴孔形状有不同的接触模型, 尽管是简单的轴孔形状也有几十种接触状态. 因此需要无需对物理接触进行建模分析、不依赖先验知识的控制算法, 在非结构化环境下完成装配任务. 受到人类利用所学经验完成轴孔装配任务[4]的启发, 机器人可通过基于学习的方法自主学习装配技能[5]. 近年来, 强化学习 (RL) 在解决装配这类连续控制的问题上展现出巨大潜力, 强化学习算法在机器人中的应用是构建一个智能体, 以获得当前任务的最大奖励为目的学习机器人的控制策略. 然而, 强化学习无法学习环境的深层特征, 不能有效应对未出现过的状态, 故研究者开始尝试使用深度强化学习 (DRL)[6-7], 其优势在于能够有效处理高维状态空间和连续动作空间的序列决策问题, 同时减少对大量人工标注数据的依赖, 为机器人自主装配提供新途径.

研究人员在将 DRL 应用于装配领域做出许多探索, 深度确定性策略梯度 (DDPG) 算法[8]、近端策略优化 (PPO) 算法[9]、深度Q网络 (DQN) 算法[10] 和软演员-评论家 (SAC) 算法[11] 等均被用于求解装配策略问题. 例如, 文献 [12] 提出了针对不同机器人插孔装配的知识迁移框架, 以增强装配模型的泛化能力和数据利用效率, 在该框架中, 源域模型通过 PPO 算法进行训练, 模型能够准确预测环境信息, 并根据预测动态调整机器人的运动; 文献 [13] 使用 DQN 算法训练一个智能体来组装工件, 建立装配线的数字孪生模型, 进行一系列轴孔装配实验, 实验结果表明, 该模型具有非常好的装配成功率.

上述文献主要集中于传统轴孔装配的研究, 这些装配对象通常具有结构对称且单一的特点, 插入路径直接, 与充电枪装配存在显著差异. 充电枪接口包含多个功能部件 (如导向槽、电气触点), 需要分阶段精准对位[14]. 同时, 充电枪与充电座的配合公差仅为 0.25 mm[3]. 因此, 许多 DRL 算法在进行充电枪装配时会存在如下问题:

1) 文献 [12-13,15] 均只使用单一 DRL 算法进行装配训练, 但 DRL 算法依赖奖励信号优化策略, 在充电枪装配任务中, 充电枪与充电座的配合公差小, 成功插入的奖励信号极稀疏, 许多低质量数据 (未达目标的轨迹) 会污染训练样本池, 导致数据利用率低.

2) 在 Actor-Critic 结构的强化学习算法中[8-9,11], 如果 Actor 网络和 Critic 同时更新, 则环境的动态变化、传感器的反馈等因素会导致策略的过快更新, 最终高误差状态的策略更新导致行为发散[16], 从而引发控制策略不稳定.

3) 文献 [13,15,17] 在 DRL 算法训练时, 将数据收集和学习放在同一个线程中, 会导致学习过程受限于数据收集的速度, 无法充分利用多核中央处理器 (CPU) 和图形处理器 (GPU) 的计算能力, 造成资源闲置.

基于以上分析, 本文以 UR5 机器人为控制对象, 充电枪与充电座装配为目标, 设计融合事后经验回放和策略延迟更新的软演员-评论家算法 (SAC with HER-DPU), 用于充电枪装配任务. 该算法能够有效提升训练样本效率, 增强策略稳定性, 并可优化硬件资源利用率. 本文的主要贡献如下:

1) 针对充电枪装配时复杂动作空间导致的样本效率低下问题, 在经验回放池引入 HER, 通过重构失败轨迹的虚拟目标, 实现装配经验的高效复用.

2) 针对充电枪装配场景下策略更新震荡与Q值高估问题, 在神经网络更新部分引入 DPU, 即多次更新价值网络后更新一次策略网络, 从而独立控制策略网络和价值网络的更新频率, 有效平衡环境探索与数据利用的矛盾, 避免因反馈过快而引发的控制策略不稳定导致模型收敛至次优解问题.

3) 本文融合 SAC 算法、HER 和 DPU 的优势, 提出 SAC with HER-DPU 用于完整的充电枪装配任务. 该方法采用双线程架构, 将智能体数据收集和神经网络训练解耦, 有效避免传统单线程模式下数据采集和模型更新的串行等待问题. 最终通过实验验证所提出算法的有效性.

1 充电枪装配问题描述

本文的充电枪装配任务主要可分为寻孔、对齐、插入这 3 个阶段. 为描述装配过程中的位姿关系, 分别建立充电枪坐标系$\{T\}$和充电座坐标系$\{G\}$, 如图 1 所示. 具体充电枪姿态偏差定义如下:

$$\begin{cases} \Delta l = \|P_T - P_G\|, \\ \Delta\omega = |\theta_T^x - \theta_G^x|, \\ \Delta\varepsilon = |\theta_T^y - \theta_G^y|, \\ \Delta\mu = |\theta_T^z - \theta_G^z|. \end{cases} \tag{1}$$

其中: P_T是坐标系$\{T\}$原点坐标; P_G是坐标系$\{G\}$

原点坐标, 单位为 m; θ_T^x, θ_G^x, θ_T^y, θ_G^y, θ_T^z, θ_G^z分别是充电枪和充电座绕XYZ轴的旋转角, 单位均为 rad; Δl是充电枪接触面中心点与充电座中心线之间的距离偏差; $\Delta\omega$、$\Delta\varepsilon$、$\Delta\mu$分别是充电枪和充电座在XYZ轴上的姿态偏差.

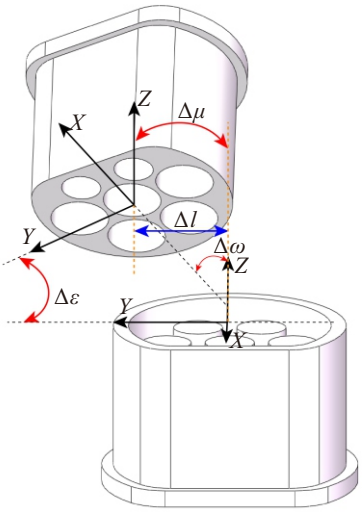


图1　充电枪装配

在充电枪装配的寻孔和对准阶段, 需解决充电枪与充电座之间的位姿匹配问题, 即确定两者在空间中的位置与姿态对应关系. 如式 (1) 所示, 通过对充电枪与充电座的位姿关系进行分析可知, 仅当$\Delta\omega$, $\Delta\varepsilon$, $\Delta\mu$以及Δl均趋近于零时, 才能实现精确的姿态匹配. 因此, 在后续的状态空间建模和奖励函数设计过程中, 需要充分考虑充电枪和充电座的位姿约束, 以确保智能体能够正确完成装配任务.

综上, 本文的装配任务是控制机器人末端执行器上的充电枪, 在Δl, $\Delta\omega$, $\Delta\varepsilon$, $\Delta\mu$接近 0 时, 将充电枪插入充电座的指定深度.

2　结合事后经验回放和策略延迟更新的深度强化学习算法设计

DRL 的核心包括状态、动作、奖励、策略和值函数, 其目标是通过试错和反馈找到最优策略. 智能体通过累积奖惩来度量一次轨迹执行, 即

$$G_t=\sum_{k=0}^{\infty}\gamma^k r_{t+k+1}. \tag{2}$$

其中: G_t是从时间步 t 开始的累积奖励; r_{t+k+1}是在时刻$t+k+1$时的即时奖励; $\gamma\in[0,1)$是折扣因子, 表示未来奖励的重要性.

2.1　SAC 算法

本文选择 SAC 算法作为基底算法, 由于充电枪插入过程中复杂的六维接触力/力矩分布和非线性的接触状态, SAC 算法通过引入熵项$\mathcal{H}$, 激励策略在训练中保持探索性, 进而避免陷入局部最优. 此外, SAC 算法的经验池充分利用历史经验提升样本效率, 特别适用高采样成本的充电枪装配任务.

SAC 算法的目标是最大化累积奖励和最大化策略的熵来促进更全面的探索, 其核心在于优化如下目标函数[11]:

$$J(\pi)=\sum_{t=0}^{T}\mathbb{E}_{(s_t,a_t)\sim\rho_\pi}[r(s_t,a_t)+\alpha\mathcal{H}(\pi(\cdot|s_t))]. \tag{3}$$

其中: $r_t=r(s_t,a_t)$是智能体在状态$s_t\in\mathcal{S}$下执行动作$a_t\in\mathcal{A}$时获得的奖励, $\mathcal{A}$和$\mathcal{S}$分别是动作空间和状态空间; ρ_π是策略π诱导的状态-动作联合分布; $\mathbb{E}_{(s_t,a_t)\sim\rho_\pi}$是在策略$\pi$下, s_t和a_t的期望值; α是熵项$\mathcal{H}$的熵系数; $\pi(\cdot|s_t)$是在状态s_t下, 策略π对所有可能动作$a\in\mathcal{A}$给出的概率分布. 同时, 熵项定义为

$$\mathcal{H}(\pi(\cdot|s_t))=-\mathbb{E}_{a_t}[\log\pi(a_t|s_t)]=-\int_{\mathcal{A}}\pi(a_t|s_t)\log\pi(a_t|s_t)\mathrm{d}a_t. \tag{4}$$

其中: $\mathbb{E}_{a_t}$是对动作a_t的期望值, $\pi(a_t|s_t)$是在状态s_t下选择动作a_t的概率. SAC 的目标在于寻找最优策略π^*, 以最大化目标函数$J(\pi)$中熵增加的奖励.

2.2　事后经验回放 (HER)

充电枪装配是一个目标导向的深度强化学习任务, 机器人需通过大量试验找到正确的插入位置以获得成功奖励. SAC 通过经验回放池提高训练稳定性和样本效率, 但回放数据多为失败经验, 影响训练效果. 因此, 本文引入 HER 算法重用失败经验, 使智能体从“失败的尝试”中学习, 加速策略优化.

HER 通过替换目标状态生成“伪成功”经验, 并重新计算奖励$r'_t=R(s_t,a_t,s_{t+1},\mathrm{done},g')$. 目标选取方式包括 Future、Episode 和 Random, 其中本文采用 Future 方式, 引导智能体朝指定装配位置探索, 避免陷入局部最优[18]. 为充分利用失败经验, 使用额外的奖励计算方式R, 如下所示:

$$\begin{cases}R=R_{\mathrm{pre}}+0.5, & d<0.001 \text{ or } \Delta\theta<0.005;\\ R=0, & \text{otherwise.}\end{cases} \tag{5}$$

其中: d是充电枪与充电座之间的欧氏距离, 单位为 m; $\Delta\theta$是充电枪与充电座之间的欧拉角误差和, 单位为 rad; R_{pre}是没有进行事后经验回放前的奖励值.

SAC with HER 算法通过结合 SAC 方法与 HER 机制, 提升强化学习在稀疏奖励环境中的样本效率和泛化能力. 在每个 episode 开始时, 首先采样初始状s_0, 然后基于策略$\pi_\theta(a_t|s_t,g)$生成状态序列$\zeta=\{s_0,s_1,\ldots,s_T\}$. 为了提高数据利用率, SAC with HER 采用 HER 策略进行数据存储.

在每个环境交互步骤中, 记录并存储状态转移元组$(s_t, a_t, r_t, s_{t+1}, \text{done}, g)$, 其中奖励信号由环境提供, 即$r_t = r(s_t, a_t, g)$. 为增强目标学习能力, 从状态序列$\zeta$采样额外目标集合$\varphi = \{g'_1, g'_2, \dots, g'_m\}$, 并对每个替代目标$g'$重新计算奖励$r'_t = r(s_t, a_t, g')$, 随后存储修正后的经验$(s_t, a_t, r'_t, s_{t+1}, \text{done}, g')$以扩充训练数据集.

在策略优化阶段, SAC with HER 依据随机梯度下降 (SGD) 更新Q网络、策略网络和熵系数α, 以优化决策过程. 通过最小化均方贝尔曼误差优化Q网络参数ϕ_1, ϕ_2.

2.3 策略延迟更新 (DPU)

尽管 SAC with HER 已具备较高样本利用率和探索能力, 但是初期训练阶段, 价值网络尚处于未充分收敛状态, 其对状态-动作对(s, a)的价值估计$Q(s, a)$往往存在较大偏差. 若此时同时更新策略和价值函数, 则策略更新时所依据的梯度计算为

$$\nabla_\theta J_\pi(\theta) \approx \mathbb{E}_{s\sim D}[\nabla_\theta \log \pi_\theta(a|s)(\alpha \log \pi_\theta(a|s) - Q(s,a))]. \tag{6}$$

其中: $\mathbb{E}_{s\sim D}[\cdot]$是对经验回放缓冲区$\mathcal{D}$中采样的状态$s$求平均; $Q(s, a)$是输出的值函数, 评估状态s下采取动作a所能获得的预期收益. 但是$Q(s, a)$的误差可能使梯度方向发生偏移, 导致策略参数θ朝向次优解更新, 进而影响探索效率和最终任务表现.

为此, 本文通过设定延迟参数d, 在每进行d次价值网络更新后, 仅执行一次策略网络更新. 具体而言, 价值网络采用下述均方误差损失函数进行更新:

$$L(\phi) = \mathbb{E}_{(s,a,r,s')\sim D}[(Q_\phi(s,a) - (r + \gamma \min_{i=1,2} Q_{\phi'_i}(s', \mu_{\theta'}(s'))))^2], \tag{7}$$

其中双Q网络的引入及其目标网络$Q_{\phi'_i}$能在一定程度上减缓过估计问题. 当 Critic 网络经过充分更新后, 策略的更新则基于更为稳定且准确的$Q(s, a)$值. 此时, Actor 更新的策略梯度可表示为式 (6), 并按如下规则更新策略参数:

$$\theta \leftarrow \theta + \lambda_\pi \nabla_\theta J_\pi(\theta) \text{ if } t \bmod d = 0. \tag{8}$$

其中: λ_π是策略学习率, t是更新步骤计数. 如此设计能够确保在每次策略更新时, Critic 网络已通过多次更新获得较为准确的价值评估, 进而使策略调整依据更为可靠的反馈, 降低了误导性梯度带来的风险.

2.4 结合事后经验回放和策略延迟更新的 SAC 算法

为解决在充电枪装配时, 传统 DRL 算法对于交互数据的利用率差和价值网络的不稳定导致所训练的策略较差的问题. 本文提出融合事后经验回放和策略延迟更新的 SAC 算法, 如图 2 所示.

基于仿真环境进行充电枪装配, 使用 SAC with HER-DPU 算法进行充电枪插入充电座的装配任务训练. 伪代码如算法 1 和算法 2 所示.

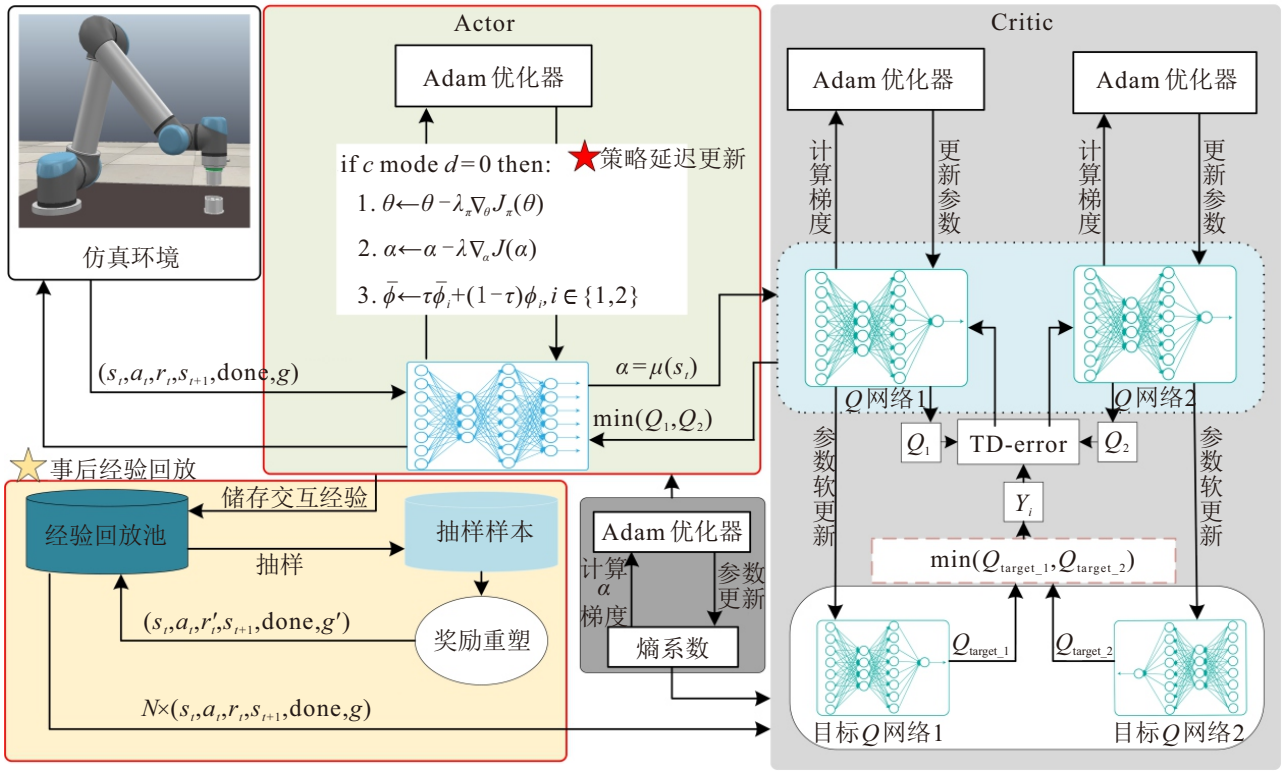


图2 结合事后经验回放和策略延迟更新的 SAC 算法框架

算法 1　学习更新线程.

1: 输入: Q 函数参数ϕ_1, ϕ_2, 策略参数θ, 熵系数α
2: 初始化: 目标Q 函数$\overline{\phi}_1 \leftarrow \phi_1, \overline{\phi}_2 \leftarrow \phi_2$
3: 初始化: 经验回放缓冲区$\mathcal{D}$, 策略延迟更新计数器$c \leftarrow 0$
4: while learn and $\mathcal{D}_{\text{num}} > \text{batch\_size}$ do
5:　从 $\mathcal{D}$采样一个小批量数据
6:　for 每个梯度更新步 do
7:　　更新Q函数参数:
8:　　$\phi_i \leftarrow \phi_i - \lambda_Q \nabla_{\phi_i} J_Q(\phi_i),\ i \in \{1, 2\}$
9:　　$c \leftarrow c + 1$　▷下列代码为策略延迟更新
10:　　if $c \bmod d = 0$ then
11:　　　更新策略参数$\theta \leftarrow \theta - \lambda_\pi \nabla_\theta J_\pi(\theta)$
12:　　　调整熵系数$\alpha \leftarrow \alpha - \lambda \nabla_\alpha J(\alpha)$
13:　　　软更新目标Q函数:
14:　　　$\overline{\phi}_i \leftarrow \tau\overline{\phi}_i + (1-\tau)\phi_i,\ i \in \{1, 2\}$
15:　　end if
16:　end for
17: end while

算法 2　数据收集线程.

1: 输入: 策略网络π_θ, 目标空间$\mathcal{G}$, 经验回放缓冲区$\mathcal{D}$
2: while true do
3:　采样初始状态s_0, 从目标空间$\mathcal{G}$采样目标g
4:　while not done do
5:　　选择动作$a_t \sim \pi_\theta(a_t|s_t, g)$
6:　　执行动作a_t, 获得新状态s_{t+1}
7:　　计算奖励$r_t = r(s_t, a_t, g)$
8:　　存入$(s_t, a_t, r_t, s_{t+1}, \text{done}, g)$到$\mathcal{D}$
9:　　从$\{s_0, s_1, \ldots, s_T\}$采样额外目标集$\varphi = \{g'_1, g'_2, \ldots, g'_m\}$
10:　　for 每个$g' \in \varphi$ do
11:　　　计算奖励$r'_t = r(s_t, a_t, g')$
12:　　　存入$(s_t, a_t, r'_t, s_{t+1}, \text{done}, g')$到$\mathcal{D}$
13:　　end for
14:　end while
15: end while
16: 发送 learn = false 给学习线程

2.5　双线程训练架构

在使用 SAC with HER-DPU 算法框架的训练过程中, 数据收集需要与环境进行交互, 生成状态、动作、奖励等数据. 学习则是基于这些数据进行模型的优化和策略更新. 将数据收集和学习放在同一个线程中进行, 会导致学习过程受限于数据收集的速度, 无法充分利用计算资源, 进而影响训练效率和收敛速度. 为克服这个问题, 本文采用双线程训练架构, 解耦智能体数据收集和学习过程, 从而避免因等待数据收集而造成的学习过程延迟, 显著提高训练效率, 伪代码如算法 1 和算法 2 所示.

综上所述, 针对 SAC 算法在充电枪装配任务中存在的不足, 本文提出相应的改进策略. 为提升样本利用率, 通过修改经验存储方式, 将整条轨迹记录于经验池中, 并从中使用 HER 算法生成伪成功样本, 从而显著提高样本利用率. 此外, 为缓解因噪声或不精确反馈导致的控制策略不稳定问题, 本文引入 DPU 机制, 延迟策略网络的更新. 同时, 采用双线程训练架构, 将数据采集与模型更新过程解耦, 以进一步提升训练效率.

3　充电枪装配策略设计

充电枪装配这个任务涉及精确的机械装配, 包含狭窄公差匹配、环境不确定性、动力学接触约束等挑战[3], 导致传统基于轨迹规划的方法难以应对未知偏差和接触力的不稳定性. 针对这一问题, 本文提出 SAC with HER-DPU 算法框架, 并且为了将上述算法应用于充电枪装配, 需要构建合理的状态空间、动作空间和奖励函数, 以保证智能体能够高效学习到可行的装配策略.

3.1　机械臂状态动作设计

3.1.1　状态空间

充电枪装配可以看作马尔可夫决策过程, 而虚拟环境中的观测对应完全可观察的马尔科夫决策过程, 状态空间旨在全面描述机器人当前状态以及与目标位置的关系. 本文的任务是控制机器人进行充电枪插入充电座, 需要考虑到装配过程中的力、力矩、充电枪的位置和姿态、目标位置和姿态、装配深度, 所以状态空间定义为

$$S = [F_x, F_y, F_z, M_x, M_y, M_z, T_x, T_y, T_z, O_x^T, O_y^T, O_z^T, G_x, G_y, G_z, O_x^G, O_y^G, O_z^G, H]. \quad (9)$$

其中: F_x, F_y, F_z, M_x, M_y, M_z分别是六维力传感器XYZ轴上的力和力矩, 单位为 N, N·m; T_x, T_y, T_z, G_x, G_y, G_z分别是XYZ轴上的充电枪位置坐标和充电座位置坐标; O_x^T, O_y^T, O_z^T, O_x^G, O_y^G, O_z^G分别是绕XYZ轴旋转的角度, 单位为 rad; H是装配深度, 单位为 m.

3.1.2　动作空间

本文的动作空间由以下六维向量组成:

$$A = [T_x, T_y, T_z, O_x^T, O_y^T, O_z^T]. \quad (10)$$

其中: T_x, T_y, T_z是机械臂末端执行器在XYZ方向上的平移距离, 范围设置为 [−0.001, 0.001], 单位为 m; O_x^T, O_y^T, O_z^T是机械臂末端执行器绕XYZ轴的旋转度数, 范围设置为 [−0.001, 0.001], 单位为 rad.

3.2 奖励函数设计

为引导机械臂完成充电枪装配任务, 设计基于位移误差、姿态误差的奖励函数以及成功装配时的奖励.

3.2.1 姿态奖励

本文的姿态奖励根据当前姿态与目标姿态之间的欧拉角误差作为奖励的依据, 具体函数定义如下:

$$R_o = \frac{-\|\Delta\theta_{\text{euler}}\|}{\max\_\text{error}}. \tag{11}$$

其中: $\Delta\theta_{\text{euler}}$是当前欧拉角与目标欧拉角的误差之和; $\max_\text{error}$是理论上的最大误差, 在本文中为π.

3.2.2 位置奖励

位置奖励是通过计算充电枪与充电插座之间在每一个回合中的欧氏距离得到的, 具体函数为

$$R_d = -\frac{\Delta d}{\Delta t} = -\frac{d_{\text{after}} - d_{\text{before}}}{\Delta t}. \tag{12}$$

其中: d_{after}和d_{before}分别是动作执行后和执行前充电枪与充电座之间的距离; Δt是时间间隔, 文中Δt为 0.001.

3.2.3 插入奖励

为引导机器人完成充电枪插入充电座的操作任务, 本文定义插入深度奖励, 具体函数定义如下:

$$\begin{cases} R_{\text{in}} = \mathrm{e}^{1-\frac{l_{\text{in}}}{l_h}}, \ d < 0.001; \\ 0, \ \text{otherwise}. \end{cases} \tag{13}$$

其中: l_{in}是充电枪插入充电座的深度, l_h是完成装配充电枪需要插入的最大深度.

3.2.4 成功奖励

为提升装配任务完成效率, 本文设计了成功装配奖励, 具体函数定义如下:

$$\begin{cases} R_s = 100, \ \text{success}; \\ 0, \ \text{otherwise}. \end{cases} \tag{14}$$

在本文中, 装配成功的条件是: 充电枪的位置与充电座位置之间的位置误差小于 0.25 mm, 并且它们之间的姿态误差小于 0.005 rad. 综上所述, 总的奖励定义为

$$R_t = R_o + R_d + R_{\text{in}} + R_s. \tag{15}$$

3.3 终止条件

为确保充电枪插入任务的安全性并提高训练效率, 除成功装配外, 还设置以下终止条件: 1) 超出最大步数; 2) 接触力/力矩过大: 末端执行器沿任意轴的接触力或力矩超过设定阈值 (10 N, 5 N·m); 3) 超出装配范围: 末端执行器偏离目标位置超过 0.2 m, 任务失败, 确保机械臂始终处于合理操作范围; 4) 姿态误差过大: 训练过程中姿态误差超限, 表明探索方向错误, 终止当前回合训练. 综上, 结合塑形奖励、HER、DPU 及双线程架构, 可有效克服文献 [11,17-18] 所述稀疏奖励问题, 提高交互数据利用率, 加速训练收敛.

4 实验与结果分析

4.1 实验参数设置

在 Coppeliasim 里搭建仿真训练装配模型, 选用 UR5 作为装配任务的控制对象. 实验装配对象采用的是符合国标 GB/T 20234.2 的充电枪和充电枪座. 操作系统使用的是 Ubuntu 20.04, Python 版本为 3.8. 训练硬件使用的是 Intel i5-13600KF, RTX4060 8 G 显存, 内存为 32 GB, 通讯架构如图 3 所示, 算法具体超参数如表 1 所示.

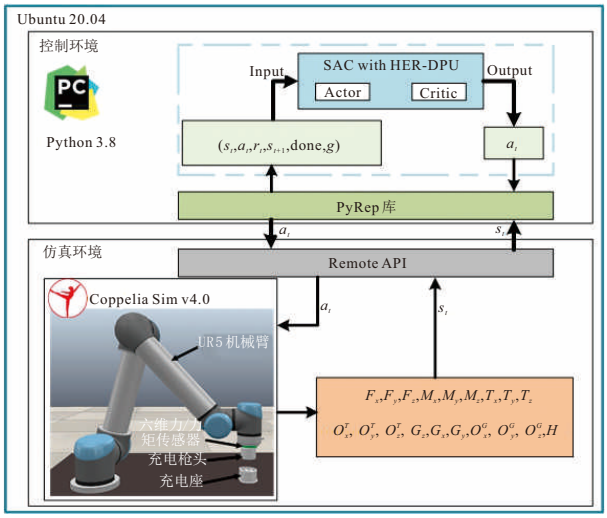


图3 Coppeliasim 通讯架构

表1 模型超参数

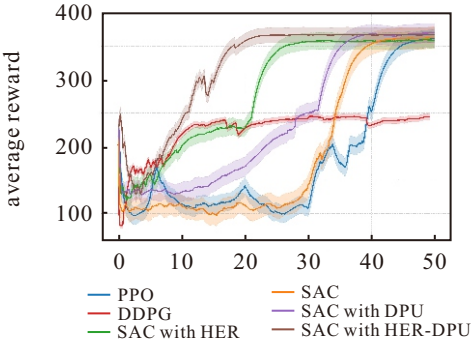
参数名称	值	说明
actor_lr	1×10^{-4}	Actor 网络的学习率
critic_lr	1×10^{-4}	Critic 网络的学习率
alpha_lr	1×10^{-3}	温度系数的学习率
hidden_dim	256	网络隐藏层的维度
gamma	0.99	折扣因子
tau	0.005	软更新参数
buffer_size	1×10^{6}	经验回放缓冲区的大小
batch_size	256	每次训练的批量大小
target_entropy	-8	目标熵值

在本文中, 为了学习神经网络的权重和偏置, 采用 Adam 优化器. Actor 网络的输入为当前状态, 输出为机器人应执行的动作, 网络结构包含两个隐藏层, 每层有 256 个神经元, 激活函数采用 ReLU, 以增加网络的非线性表达能力, 输出层使用 tanh 激活函数, 将动作输出限制在适当的范围内. Critic 网络的输入为状态与动作的组合, 输出为对应的Q值, 评估该状态-动作对的价值. Critic 网络同样包含两个隐藏层, 每层也有 256 个神经元, 激活函数使用 ReLU,

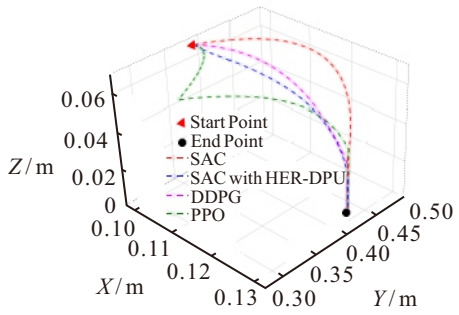
输出层为一个标量Q值, 不使用激活函数.

4.2　算法性能对比实验

本实验小节旨在验证 SAC with HER-DPU 算法在充电枪装配任务中的性能. 在相同的性能参数和装配环境下, 对比 SAC、SAC with DPU、SAC with HER、SAC with HER-DPU、PPO 和 DDPG 六种算法的训练效果, 并评估其收敛速度、累积奖励及装配轨迹, 总体累积奖励如图 4(a) 所示.



(a)　算法学习



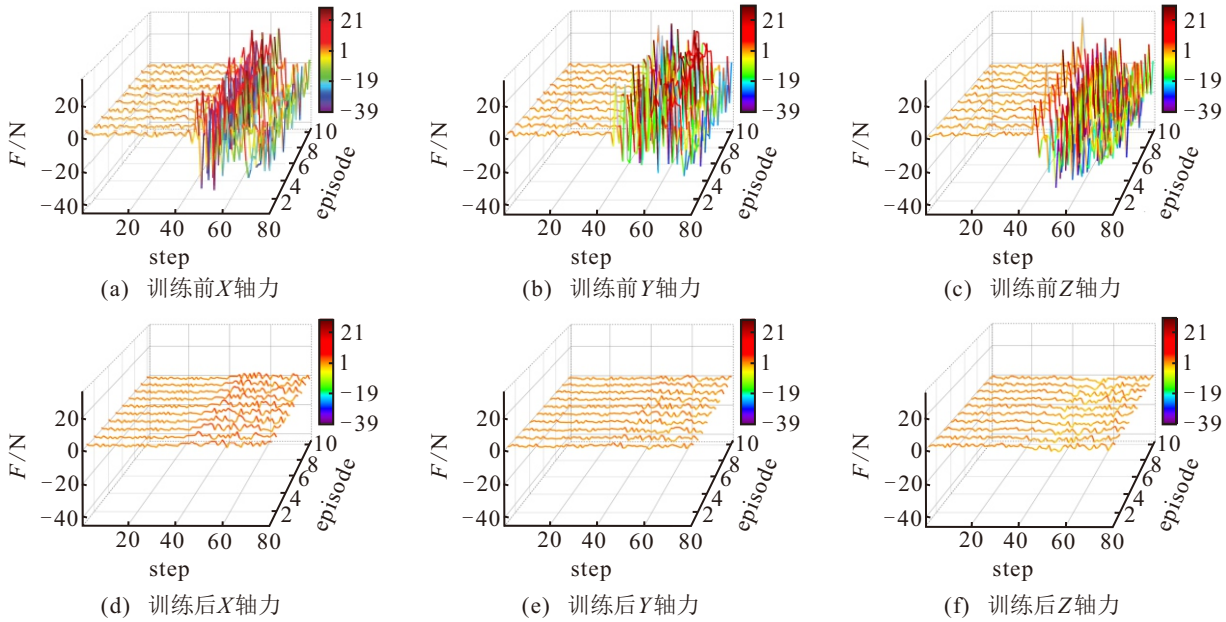
(b)　固定位置装配轨迹

图4　装配轨迹对比

实验结果表明, 6 种算法均能实现稳定收敛, 但在收敛速度和装配精度方面存在显著差异. SAC with HER-DPU 算法在约 19 000 回合时即达到收敛状态, 是所有算法中收敛速度最快的. 此外, 尽管 DDPG 算法在 11 000 回合左右开始收敛, 但其未能成功完成装配任务, 而 SAC、SAC with DPU、SAC with HER 和 SAC with HER-DPU 这 4 种算法均能完成装配任务. SAC with HER-DPU 算法的轨迹最优, 表现出更高的路径规划精度和装配稳定性, 如图 4(b) 所示. 相比之下, SAC 算法和 PPO 算法的装配轨迹较长, 表明其路径规划效率较低, 而 DDPG 算法的轨迹波动较大, 容易导致充电枪与充电座发生碰撞, 进而产生过大装配力, 导致装配失败. 综合来看, SAC with HER-DPU 算法在收敛速度、轨迹优化以及装配任务完成率方面均优于其他算法, 验证了其在充电枪装配任务中的有效性.

4.3　算法策略柔顺性测试实验

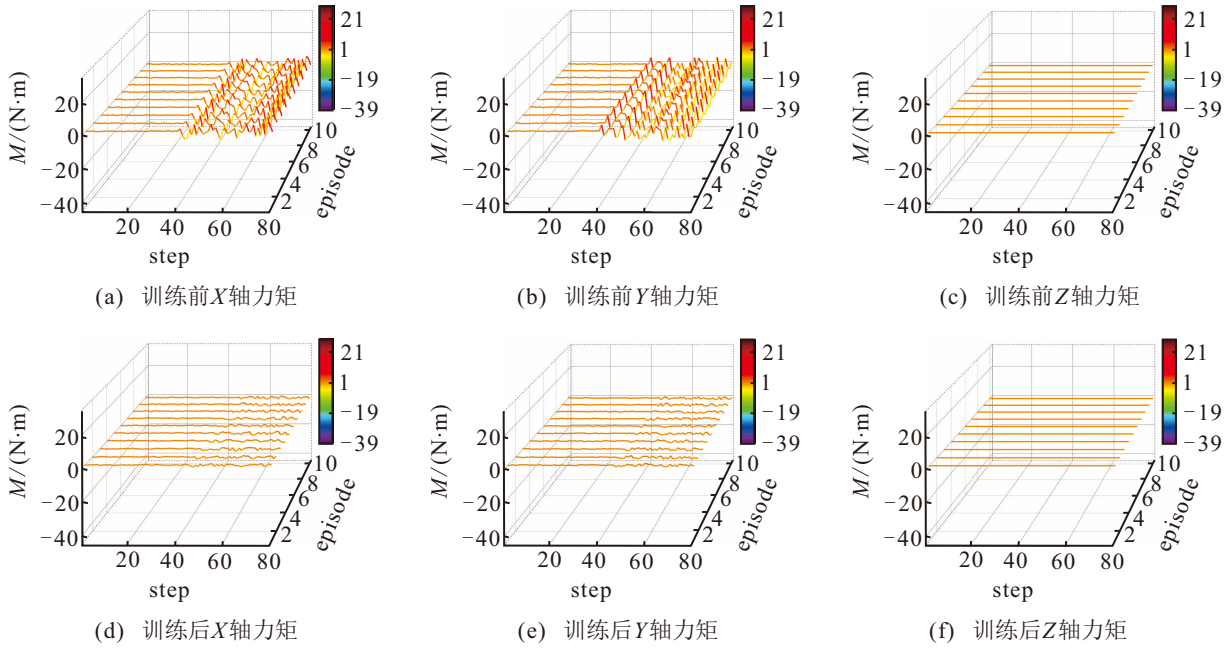
为验证 SAC with HER-DPU 算法在装配过程中的柔顺性, 设计 10 组实验, 分别对比该算法在训练初期和完全收敛后其装配过程中接触力和力矩的变化情况. 在训练之前, 装配策略表现为较大的力波动, 力的绝对值最大可达 35 N, 力矩的绝对值最大可达 5 N·m. 而训练后的装配策略, 力和力矩的变化趋于平稳, 力的绝对值最大只有 1.5 N, 力矩的绝对值最大只有 0.5 N·m, 尤其是在接触点附近, 力的控制更加精确, 减少了不必要的震动和过大的冲击力. 通过对比 10 组数据, 可以清晰地看到在训练后, 接触力和力矩的值显著降低, 且符合预期的柔顺装配要求, 从而验证了训练后的策略在装配任务中的柔顺性. 策略训练前后装配力和力矩如图 5 和图 6 所示.



(a)　训练前X轴力　(b)　训练前Y轴力　(c)　训练前Z轴力

(d)　训练后X轴力　(e)　训练后Y轴力　(f)　训练后Z轴力

图5　训练前后的装配力



(a) 训练前X轴力矩 (b) 训练前Y轴力矩 (c) 训练前Z轴力矩

(d) 训练后X轴力矩 (e) 训练后Y轴力矩 (f) 训练后Z轴力矩

图6 训练前后的装配力矩

4.4 算法策略稳定性和收敛速度测试实验

为测试 SAC with HER-DPU 算法训练所得策略的稳定性和收敛速度, 在相同实验环境下, 对比 SAC、SAC with HER、SAC with DPU、SAC with HER-DPU、DDPG、DDPG with HER、DDPG with DPU、DDPG with HER-DPU 以及 PPO 共 9 种算法在收敛后的策略. 实验共进行 100 次装配测试, 采用固定充电座方式以消除随机初始化对实验结果的影响.

评估指标包括收敛时间、装配成功率、平均装配步数、最大接触力和最大接触力矩, 实验结果详见表 2.

表2 算法测试结果

算法	收敛时间/h	装配成功率/%	平均装配步数	最大接触力/N	最大接触力矩/(N·m)
SAC	33.2	98	92	9	1.2
SAC with HER	19.5	95	84	6.2	0.8
SAC with DPU	30.8	94	80	3.5	0.7
SAC with HER-DPU	11.8	99	75	2.1	0.5
DDPG	13.4	0	82	121.5	9.3
DDPG with HER	18.5	21	79	60.1	5.8
DDPG with DPU	12.8	1	85	110.9	8.4
DDPG with HER-DPU	19.5	26	81	100.5	10.5
PPO	39.1	96	93	10	1.8

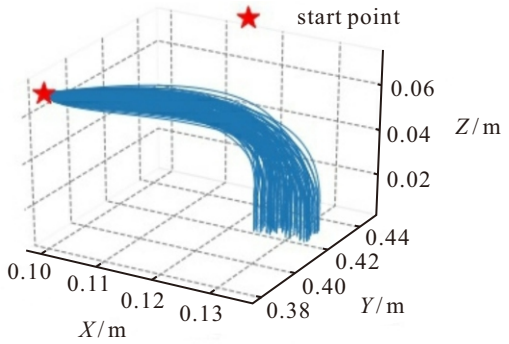
SAC with HER-DPU 结合了 SAC 的高效学习能力与 HER 的目标重设机制, 增强了智能体在目标导向任务中的适应性. DPU 机制平滑策略更新, 减少波动, 提高目标追踪稳定性, 使智能体能够在极小的力和力矩作用下完成装配任务.

实验结果表明: SAC、SAC with HER 和 SAC with DPU 这 3 种算法均能实现柔顺装配. 但 SAC 算法由于交互数据利用率低, 收敛速度最慢; SAC with HER 算法收敛更快, 但策略波动较大, 可能导致训练失败; SAC with DPU 改善了稳定性, 但仍受限于样本效率. 相比之下, DDPG 算法及其变体由于采用确定性策略, 容易陷入局部最优, 导致装配力矩过大, 最终装配失败. PPO 算法通过策略优化提高训练稳定性, 但受 on-policy 机制限制, 样本效率低, 收敛时间长. 综合表 2 的算法测试结果来看, SAC with HER-DPU 算法在稳定性、收敛速度和装配成功率方面均表现最佳, 验证了 SAC with HER-DPU 算法训练所得策略的稳定性和较快的收敛速度.

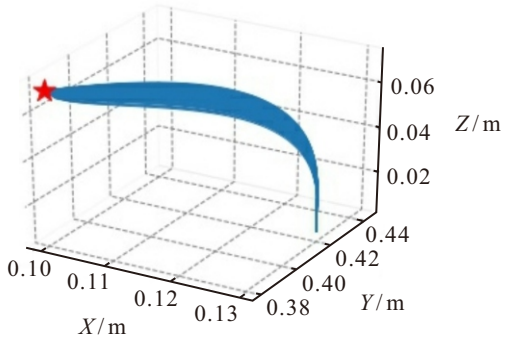
4.5 算法策略泛化性测试实验

为验证 SAC with HER-DPU 算法在不同初始条件下的策略泛化能力, 设计两组对比实验, 固定充电插座的初始位置并在原固定位置基础上引入 ± 0.03 m 的随机扰动. 图 7 展示了两组条件下的装配轨迹. 每组均进行 100 次装配实验, 记录成功率、平均装配步

数及最大接触力/力矩, 如表 3 所示.



(a) 初始化位置随机的轨迹



(b) 初始化位置固定的轨迹

图7　装配轨迹对比

表3　泛化性测试结果

条件	成功率/%	平均装配步数	最大接触力/N	最大接触力矩/(N·m)
初始位置固定	99	76	2.3	0.7
初始位置随机	96	79	2.7	0.9

实验结果表明, 当充电插座位置固定时, 装配成功率达 99%. 在引入位置随机扰动后, 成功率仍保持在 96%, 且策略轨迹平稳, 验证了本文算法在充电枪装配任务中的良好泛化性能.

4.6　算法策略通用性测试实验

为了验证本文提出的 SAC with HER-DPU 算法在不同硬件条件下的通用性, 在保持相同的性能参数和装配环境的前提下, 除了使用前文提到的 UR5 机械臂外, 还引入 Emika Panda 机械臂以及 LBR iiwa 7 R800 机械臂进行实验. 在实验中, 使用两种标准接口 GB/T 20234.2 和 SAE J1772 作为实验装配对象, 进行相应的充电枪装配训练. 最后, 对训练好的模型均进行 100 次装配实验, 记录成功率. 训练时的奖励曲线如图 8 所示, 训练结果如表 4 所示.

实验结果表明, SAC with HER-DPU 算法在不同机械臂和装配对象下均能收敛和完成充电枪装配任务. 虽然不同机械臂的收敛时间和装配精度存在一定差异, 但整体装配成功率均保持在 97% 以上. 这

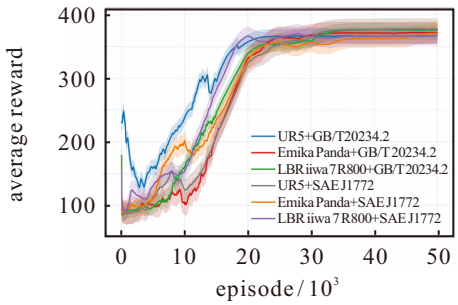


图8　通用性测试奖励

表4　通用性测试结果

机械臂类型	充电枪接口标准	装配成功率/%
UR5	GB/T 20234.2	99
Emika Panda	GB/T 20234.2	100
LBR iiwa 7 R800	GB/T 20234.2	98
UR5	SAE J1772	98
Emika Panda	SAE J1772	99
LBR iiwa 7 R800	SAE J1772	97

说明该算法具有良好的通用性, 能够应对不同硬件条件下的装配任务.

5　结　论

针对传统 DRL 在充电枪装配时, 数据利用率低、控制策略不稳定、硬件资源利用不充分的问题, 提出 SAC with HER-DPU 算法框架以改善这些问题. 通过在经验回放池中引入 HER 生成“伪成功”经验, 在梯度更新阶段加入 DPU, 优化策略更新, 最后结合双线程架构并行训练. 实验结果表明, 引入 HER 算法、DPU 和双线程架构的 SAC 算法, 进行充电枪装配时, 具有较快的收敛速度、更好的稳定性、柔顺性和泛化性. 未来工作将模仿学习与 DRL 进行结合, 让训练更加高效.

参考文献 (References)

[1] Wu K, Chen R K, Chen Q, et al. Robotic assembly of deformable linear objects via curriculum reinforcement learning[J]. IEEE Robotics and Automation Letters, 2025, 10(5): 4770-4777.

[2] Ge M D, Jin H Z, Xu T, et al. Attitude alignment based on hole attitude estimation in robot assembly[J]. IEEE Transactions on Instrumentation and Measurement, 2025, 74: 1-11.

[3] 徐建明, 胡松达, 董建伟, 等. 基于接触状态识别的机器人操作充电枪寻孔策略[J]. 控制与决策, 2022, 37(7): 1794-1802.
(Xu J M, Hu S D, Dong J W, et al. A strategy for robot-operated charging gun hole search based on contact state recognition[J]. Control and Decision, 2022, 37(7): 1794-1802.)

[4] Liang W X, Liu Y L, Wang J K, et al. Trajectory progress-based prioritizing and intrinsic reward mechanism for robust training of robotic

manipulations[J]. IEEE Transactions on Automation Science and Engineering, 2025, 22: 9829-9842.

[5] Stevšić S, Christen S, Hilliges O. Learning to assemble: Estimating 6D poses for robotic object-object manipulation[J]. IEEE Robotics and Automation Letters, 2020, 5(2): 1159-1166.

[6] 彭自然, 贺振宇, 肖伸平, 等. 基于深度强化学习模型TD3优化和改进的电动汽车制动能量回收策略[J]. 控制与决策, 2025, 40(8): 2361-2372.
(Peng Z R, He Z Y, Xiao S P, et al. Regenerative braking strategy of electric vehicles optimized and improved by deep reinforcement learning model TD3[J]. Control and Decision, 2025, 40(8): 2361-2372.)

[7] 户高铭, 蔡克卫, 王芳, 等. 基于深度强化学习的无地图移动机器人导航[J]. 控制与决策, 2024, 39(3): 985-993.
(Hu G M, Cai K W, Wang F, et al. Mapless navigation of mobile robots based on deep reinforcement learning[J]. Control and Decision, 2024, 39(3): 985-993.)

[8] Li L Y, Xu H B, Ma J, et al. Joint EH time and transmit power optimization based on DDPG for EH communications[J]. IEEE Communications Letters, 2020, 24(9): 2043-2046.

[9] Cheng Y H, Huang L Y, Wang X S. Authentic boundary proximal policy optimization[J]. IEEE Transactions on Cybernetics, 2022, 52(9): 9428-9438.

[10] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540): 529-533.

[11] Haarnoja T, Zhou A, Abbeel P, et al. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor[C]. International Conference on Machine Learning. Stockholm, 2018: 1861-1870.

[12] Men Y, Jin L G, Cui T, et al. Policy fusion transfer: The knowledge transfer for different robot peg-in-hole insertion assemblies[J]. IEEE Transactions on Instrumentation and Measurement, 2023, 72: 1-10.

[13] Li J Z, Pang D, Zheng Y, et al. A flexible manufacturing assembly system with deep reinforcement learning[J]. Control Engineering Practice, 2022, 118: 104957.

[14] Shareef H, Islam M M, Mohamed A. A review of the state-of-the-art charging technologies, placement methodologies, and impacts of electric vehicles[J]. Renewable and Sustainable Energy Reviews, 2016, 64: 403-420.

[15] Zhu Z L, Liu Y K, Zhang L, et al. Simulation of robotic peg-in-hole assembly strategy based on DRL[J]. Journal of System Simulation, 2024, 36(6): 1414-1424.

[16] Wang S, Zhang B, Liang Q, et al. Research on decision making of intelligent vehicle based on composite priority experience replay[J]. Intelligent Decision Technologies, 2024, 18(1): 599-612.

[17] Li M F, Liu H B, Xie F, et al. Point cloud-based end-to-end formation control using a two stage SAC algorithm[J]. IEEE Robotics and Automation Letters, 2025, 10(3): 2319-2326.

[18] Zuo G, Zhao Q, Lu J, et al. Efficient hindsight reinforcement learning using demonstrations for robotic tasks with sparse rewards[J]. International Journal of Advanced Robotic Systems, 2020, 17(1): 2319-2326.

作者简介

王福杰 (1991−), 男, 副教授, 博士, 主要研究方向为机器人控制、人工智能驱动的机器人优化控制及应用, E-mail: fjwang@dgut.edu.cn;

彭永岗 (1998−), 男, 硕士生, 主要研究方向为基于人工智能的机器人控制与应用, E-mail: 1779740717@qq.com;

李醒 (1982−), 女, 特聘教授, 博士, 主要研究方向为机器人建模及智能控制, E-mail: lixing8245@163.com;

郭芳 (1992−), 女, 特聘副研究员, 博士, 主要研究方向为柔性机器人建模、机器人非线性控制, E-mail: 2019091@dgut.edu.cn;

秦毅 (1987−), 男, 特聘副教授, 博士, 主要研究方向为微机电系统建模、非线性控制、智能制造, E-mail: qinyidee@163.com;

戚远航 (1993−), 男, 副教授, 博士, 主要研究方向为复杂系统建模与优化、人工智能及智能优化, E-mail: qiyuanhang@zsc.edu.cn.