

文章编号: 1006-3080(2026)03-0399-10

DOI: 10.14135/j.cnki.1006-3080.20251110004

## 基于双行动者双评论家算法的间歇过程优化控制

唐泽楷, 罗娜

(华东理工大学能源化工过程智能制造教育部重点实验室, 上海 200237)

**摘要:** 在使用深度强化学习 (Deep Reinforcement Learning, DRL) 方法对间歇过程进行优化控制时, 存在样本学习效率低、训练时间长和训练不稳定的问题。针对这些问题, 提出了双行动者正则化双评论家 (Double Actors and Regularized Double Critics, DARDC) 算法。算法通过并行双行动者网络增强智能体在复杂状态空间的探索能力, 借助正则化项约束双评论家网络, 有效抑制价值函数估计偏差, 同时引入优先经验回放机制 (Prioritized Experience Replay, PER) 加速训练收敛。为保障智能体输出的控制量合理, 在动作输出中嵌入专家经验规则。在青霉素发酵过程的实验中, 所提算法的性能优于传统深度强化学习算法, 且与比例-积分-微分 (PID) 控制相比, 关键控制指标青霉素平均产量提升 19.1%。结果表明, DARDC 算法具备一定的有效性与鲁棒性, 为间歇过程优化控制提供了新方法。

**关键词:** 深度强化学习; 间歇过程优化控制; 双行动者; 双评论家; 优先经验回放

**中图分类号:** TP183

**文献标志码:** A

间歇过程是一种非连续的工业生产方式, 它将一定量的原料按照既定工艺程序, 在设备中进行加料、反应和出料等操作。该生产方式以批次为单位重复进行, 每一批次生产的成品量是有限的<sup>[1]</sup>。间歇过程作为典型的动态系统, 呈现出时变性、强非线性、组间不确定性以及状态部分可观性等复杂特性<sup>[2]</sup>。在时变性方面, 其动态特性会随时间发生非线性变化, 例如在生物发酵中, 菌体于不同生长阶段的代谢速率差异可达 1~2 个数量级<sup>[3]</sup>。多变量间耦合关系复杂, 致使输入输出呈现高度非线性映射, 例如在化工反应里, 温度的微小波动就可能引发反应速率的指数级改变<sup>[4]</sup>。由于批次间原料存在差异、设备存在磨损等因素, 过程模型参数会出现显著漂移, 导致同一控制策略在不同批次中执行后得到的平均产量相差 20% 以上<sup>[5]</sup>。这些特性相互交织, 让间歇过程的控制成为工业自动化领域极具挑战性的难题。优化间歇过程的控制轨迹对于提升生产效率、保证产品质量、降低成本及增强过程可控性具有重要意义<sup>[6]</sup>。

针对间歇过程控制这一问题, 传统的控制方法是模型预测控制 (Model Predictive Control, MPC)<sup>[7]</sup> 和非线性模型预测控制 (Nonlinear Model Predictive Control, NMPC)<sup>[8]</sup>。Lopez-Montero 等<sup>[9]</sup> 提出了一种基于间歇性测量和移动窗口估计的批处理产品质量轨迹跟踪控制方法。该方法通过投影到潜在结构模型, 建立易于测量的过程变量与产品质量之间的关系, 并将其转化为递归形式以融入 MPC 框架。通过移动窗口估计策略, 该方法能够利用间歇性测量数据预测未来的产品质量轨迹, 并在控制过程中实时调整操作变量, 以应对未建模的干扰和模型与实际系统的不匹配问题。Jang 等<sup>[10]</sup> 提出了一种用于发酵反应器的鲁棒 NMPC 方案, 以应对多种不确定性, 包括参数误差、外部干扰和测量噪声。该方案结合了多阶段 NMPC、批量最小二乘估计和后退约束, 通过优化控制输入和估计不确定参数, 显著减少了约束违反, 确保了发酵过程安全和产品质量。但上述方法均需要建立精确的过程模型来描述过程的动态特性, 否则

收稿日期: 2025-11-10

作者简介: 唐泽楷(2000—), 男, 福建莆田人, 硕士生, 主要研究方向为人工智能、强化学习。E-mail: 3199836319@qq.com

通信联系人: 罗娜, E-mail: naluo@ecust.edu.cn

引用本文: 唐泽楷, 罗娜. 基于双行动者双评论家算法的间歇过程优化控制[J]. 华东理工大学学报(自然科学版), 2026, 52(3): 399-408.

Citation: TANG Zekai, LUO Na. Optimal Control of Batch Process Based on Double Actors and Double Critics Algorithm[J]. Journal of East China University of Science and Technology, 2026, 52(3): 399-408.

将严重影响控制性能,当非线性较强且优化频率较高时,NMPC中非线性优化问题的求解也具有挑战性<sup>[11]</sup>,并且对计算资源要求较高,这就导致了实时性能下降<sup>[12]</sup>。

与基于模型的方法相比,基于深度强化学习(Deep Reinforcement Learning, DRL)<sup>[13]</sup>结合了深度学习强大的特征提取能力与强化学习的动态决策机制,在复杂高维状态空间的序列决策任务中展现出显著优势<sup>[14]</sup>。该控制方法不需要精确的过程模型,它将间歇过程作为马尔可夫决策过程(Markov Decision Process, MDP)任务中的环境,并与智能体直接交互,通过学习过往交互数据的方式来得到最优控制策略。Kim等<sup>[15]</sup>提出了一种基于强化学习和预测控制的两阶段最优控制框架,用于解决发酵罐中的动态优化问题。该框架旨在提高发酵罐的生产效率和产量,同时在模型与实际系统不匹配以及实时干扰的情况下,保持系统的鲁棒性和稳定性。邓绍斌等<sup>[16]</sup>提出了一种基于局部策略交互探索的深度确定性策略梯度的控制方法,将青霉素发酵过程中的温度作为控制对象,结果表明,深度强化学习算法可以有效地应用到青霉素发酵过程中,并可以提高青霉素的产量。Li等<sup>[17]</sup>在DRL的训练框架中融入了专家经验,以确保模型在探寻更优策略的过程中,能够实现更为稳定的训练过程。在Fujimoto等<sup>[18]</sup>提出的Twin Delayed Deep Deterministic Policy Gradient(TD3)算法的启发下,Joshi等<sup>[19]</sup>针对间歇过程控制中传统深度强化学习方法存在的训练不稳定和策略过估计等问题,在TD3框架的基础上引入双行动者网络与延迟更新机制,有效提升了策略搜索的多样性与稳定性,并在典型间歇过程案例中验证了其优越性。随后,马军伟等<sup>[20]</sup>将双行动者深度确定性策略梯度算法应用于间歇过程优化控制。尽管基于深度强化学习的方法在间歇过程控制优化中取得了一定成效,但受限于间歇过程的强非线性、非平稳和高维特性,现有方法普遍存在样本效率低、训练不稳定的问题,且在面对操作条件变化或外部扰动时,鲁棒性与泛化能力不足,难以满足实际工业应用需求。

鉴于TD3算法在连续过程控制中的优异表现,本文在其基础上提出了一种双行动者正则化双评论家(Double Actors and Regularized Double Critics, DARD)算法,以更有效地解决间歇过程的优化控制问题。该方法采用双行动者-双评论家网络架构,其中两个并行行动者网络能够增强智能体在策略空间的探索能力,有效避免陷入局部最优;双评论家网络则通过引入正则化约束抑制因最大化操作导致的

值函数高估问题,并利用正则化参数限制评论家网络间的差异幅度,从而提升训练过程的稳定性。此外,为行动者网络配置独立的优先级经验回放池,基于样本重要性加权采样,加速关键状态-动作对的学习效率。在动作输出层嵌入专家经验规则,对控制量的物理边界和变化速率进行约束,确保决策结果符合工业过程实际需求。最后,基于Indepensim仿真平台开展实验验证,通过对比不同方法的网络训练收敛速度、青霉素发酵过程中的关键控制变量的动态曲线、状态变量的演化趋势以及青霉素平均产量等指标,综合评估所提算法与其他方法的性能差异。

## 1 预备知识

从强化学习的角度来看,间歇过程优化控制问题可以被建模为一个典型的MDP<sup>[21]</sup>。对于间歇生产过程,其状态 $s_t$ 通常由关键工艺变量组成,而动作 $a_t$ 则对应可调控控制输入。目标是在有限的生产周期内,通过策略 $\pi(a_t|s_t)$ 优化最终性能指标,即最大化累积奖励 $r(s_t, a_t)$ :

$$J(\pi) = \mathbb{E}_{\pi} \left[ \sum_{t=0}^T r(s_t, a_t) \right] \quad (1)$$

其中, $J(\pi)$ 是策略 $\pi$ 的期望回报(Expected Return),表示在整个间歇过程周期内累积奖励的期望值; $\pi(a_t|s_t)$ 是在给定状态 $s_t$ 下采取动作 $a_t$ 的条件概率分布;在确定性策略中,简写为 $a_t = \mu(s_t)$ 。 $s_t \in \mathcal{S}$ 是在时刻 $t$ 的系统状态(State),例如底物质量浓度、产物质量浓度、pH等关键工艺变量; $a_t \in \mathcal{A}$ 是在时刻 $t$ 采取的动作(Action),通常为连续变量,如补料流速、搅拌速率、温度设定值等; $r(s_t, a_t) \in \mathbb{R}$ ,用于衡量状态-动作对在工艺优化目标上的表现,例如青霉素质量浓度提升、能耗降低等; $T$ 表示间歇过程的有限时域长度,即一个生产周期内的最大决策步数; $\mathbb{E}_{\pi}[\cdot]$ 是在策略 $\pi$ 下对所有可能轨迹进行的数学期望。

针对连续动作空间的优化问题,经典的Actor-Critic(AC)方法提供了有效的解决方案。AC方法同时训练一个策略网络(Actor)和一个价值网络(Critic),Actor模块负责生成控制策略,Critic模块负责评估策略执行的价值,能够在高维连续空间中稳定学习控制策略。在具体实现上,DDPG(Deep Deterministic Policy Gradient)<sup>[22]</sup>将AC方法扩展到深度神经网络,并采用经验回放和目标网络机制,解决了高维连续控制问题中样本效率低和训练不稳定的问题。其核心是使用价值网络计算 $Q$ 值,并通过确定性策略梯度更新策略网络:

$$\nabla_{\theta} J \approx \mathbb{E}_{s_t \sim \mathcal{D}} [\nabla_a Q(s_t, a | \omega) |_{a=\mu(s_t)} \nabla_{\theta} \mu(s_t | \theta)] \quad (2)$$

其中,  $\nabla_{\theta} J$  是关于策略参数  $\theta$  的性能目标梯度;  $\mathcal{D}$  是经验回放缓冲区 (Replay Buffer), 用于存储状态转移样本  $(s_t, a_t, r_t, s_{t+1})$ , 打破样本间的相关性;  $Q(s_t, a | \omega)$  是动作价值函数 (Action-Value Function), 由价值网络近似拟合, 参数为  $\omega$ ;  $\mu(s_t | \theta)$  是策略网络输出的确定性策略, 参数为  $\theta$ ;  $\nabla_a Q(s_t, a | \omega) |_{a=\mu(s_t)}$  是价值网络对动作的梯度, 反映动作变化对价值函数的影响;  $\nabla_{\theta} \mu(s_t | \theta)$  是策略网络对参数的梯度, 决定策略如何更新。

针对 DDPG 在实际间歇过程应用中可能出现策略过估计 (Overestimation) 和探索不足的问题, 引入了双价值网络和延迟更新机制, 通过选择两个价值网络中的最小  $Q$  值来降低过估计偏差, 并延迟策略网络更新以增强训练稳定性:

$$y = r_t + \gamma \min_{i=1,2} Q_i(s_{t+1}, \mu'(s_{t+1})) \quad (3)$$

其中,  $y$  是 TD3 的目标  $Q$  值 (Target  $Q$  Value), 用于价值网络的训练;  $r_t$  是在时刻  $t$  的即时奖励;  $\gamma \in (0, 1)$  表示折扣因子 (Discount Factor), 权衡即时奖励与未来奖励的重要性;  $Q_i(s, a)$  是第  $i$  个价值网络的动作价值函数 (TD3 采用双 Critic:  $i = 1, 2$ );  $\min_{i=1,2} Q_i(\cdot)$  取两个价值网络估计值中的较小值, 以减少过估计偏差;  $s_{t+1}$  是在执行动作  $a_t$  后的下一时刻状态;  $\mu'(s_{t+1})$  是目标策略网络的输出策略, 用于生成下一步的候选动作。

## 2 双行动者正则化双评论家算法

本文在 AC 框架下设计 DARDC 算法, 如图 1 所

示, 图中  $\theta$  和  $w$  分别表示策略网络参数与价值网络参数, 右上标“'”表示相对应的目标网络参数;  $s$  与  $s'$  表示当前状态与下一个状态;  $dt$  表示该发酵批次是否结束。该算法采用双行动者网络与双评论家网络的结构, 并使用优先经验采样策略来对经验回放池进行采样。

在 DARDC 算法中, 具体包括以下几个部分:

(1) 双行动者模块 (Double Actor Module)

在提出的算法中, 引入两个并行的行动者网络 (Actor1 和 Actor2), 分别输出候选动作:

$$a_t^1 = \mu_1(s_t | \theta_1), \quad a_t^2 = \mu_2(s_t | \theta_2) \quad (4)$$

针对确定性策略可能导致的探索不足问题, 在策略输出的动作上叠加噪声, 以保证训练初期的充分探索:

$$a \leftarrow \mu(s | \theta) + \varepsilon, \varepsilon \sim N(0, \sigma) \quad (5)$$

其中,  $a$  为输出动作,  $\varepsilon$  为高斯噪声。

两个策略网络均通过策略梯度进行更新, 其更新目标为最大化对应的价值网络评估值:

$$\max_{\theta_i} \mathbb{E}[Q(s_t, \mu_i(s_t | \theta_i))] \quad (6)$$

其中,  $i = 1, 2$ 。

每个策略网络都配备一个对应的目标网络 (Target Actor), 采用软更新机制确保训练过程的稳定性:

$$\theta'_i \leftarrow \tau \theta_i + (1 - \tau) \theta'_i, i = 1, 2 \quad (7)$$

使用单个行动者网络可能会由于缺乏动作空间的探索使智能体陷入局部最优, 而不是全局最优。而采用双行动者网络将允许智能体评估不同方向的

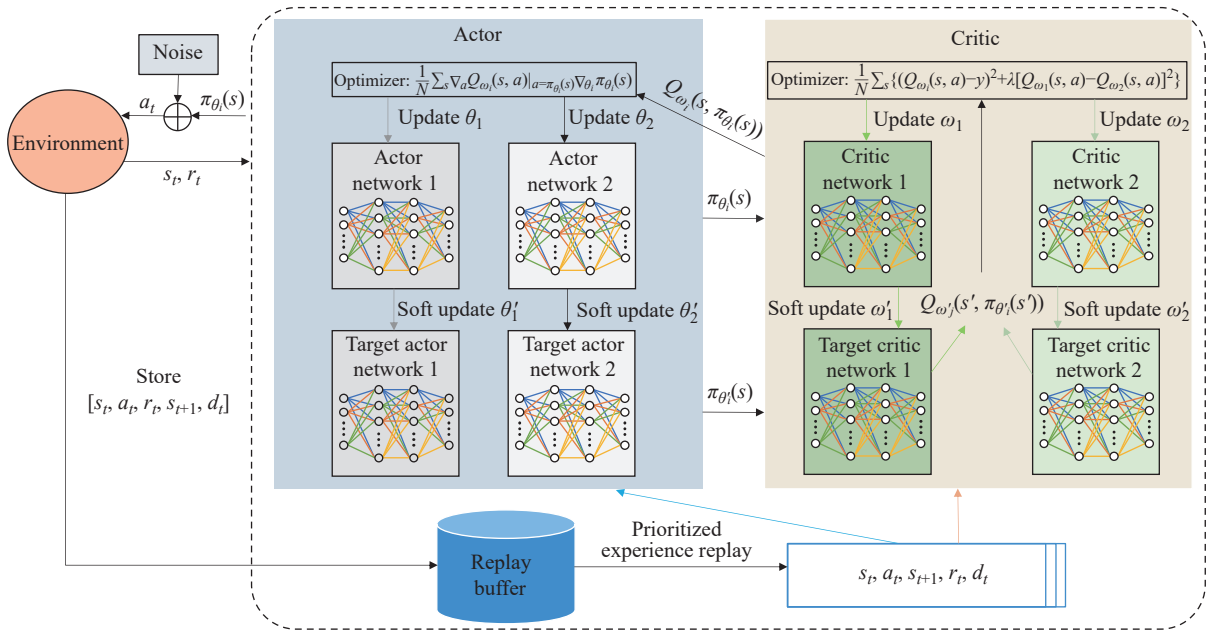


图1 DARDC 算法结构图

Fig. 1 Algorithm structure diagram of DARDC

策略,而不是受单一策略路径的限制<sup>[23]</sup>。双行动者网络可以通过遵循收益较高的策略来增强智能体的勘探能力,即比较评论家模块对其输出的价值函数大小,来判断遵循哪个行动者网络的策略动作值,有效缓解单一策略可能陷入局部最优的问题。

## (2) 双评论家模块 (Regularized Double Critic Module)

由于单一评论家网络存在因为最大化偏差导致对价值函数的过高估计的问题,算法中采用双价值结构,由两个价值网络 (Critic1 和 Critic2) 并行估计动作价值  $Q_{\omega_1}(s_t, a_t | \omega_1)$  和  $Q_{\omega_2}(s_t, a_t | \omega_2)$ 。

通过最小化双价值估计值,减小单一价值带来的过估计偏差<sup>[24]</sup>:

$$y = r_t + \gamma \min_{i=1,2} Q_{\omega_i}(s_{t+1}, \mu'(s_{t+1})) \quad (8)$$

在每个价值网络中,网络参数独立初始化并异步更新,一般通过均方误差 (MSE) 损失函数更新。为减少策略更新时的剧烈波动,避免因  $Q$  值震荡导致网络训练崩溃,通过增加惩罚函数项的方式来降低两个评论家网络价值估计过程中的不确定性<sup>[25]</sup>:

$$L(\omega_i) = \frac{1}{N} \sum_s \{ (Q(s, a | \omega_i) - y)^2 + \lambda [Q(s, a | \omega_1) - Q(s, a | \omega_2)]^2 \} \quad (9)$$

同样地,每个 Critic 都配备目标网络 (Target Network),采用软更新:

$$\omega'_i \leftarrow \tau \omega_i + (1 - \tau) \omega'_i, i = 1, 2 \quad (10)$$

## (3) 经验回放池 (Replay Buffer)

为打破样本之间的时间相关性,提高数据利用率,采用经验回放池存储与环境交互得到的转移样本  $(s_t, a_t, r_t, s_{t+1})$ 。一般基于 AC 框架的 DRL 算法在每次更新神经网络参数时都采用随机均匀采样策略从经验回放池中采样样本,这种方法对所有经验样本赋予了相同的优先级,而忽略了不同样本对网络学习效果的重要性差异,会导致算法需要更多训练步数才能够收敛,降低了有效样本的利用效率,浪费了计算资源。为了提高有效样本的利用效率, DARDC 算法采用优先经验采样策略<sup>[26]</sup>,根据样本的 TD 误差赋予不同的采样概率,使得训练更关注高价值或学习难度较大的样本,样本的 TD 误差计算如式 (11) 所示:

$$\delta = |Q(s_t, a_t) - y| \quad (11)$$

其中,目标  $y$  取两个目标评论家网络  $Q$  值的最小值计算得到,公式如式 (12) 所示:

$$y = r + \gamma(1 - d) \min(Q_{\omega'_1}(s', a') Q_{\omega'_2}(s', a')) \quad (12)$$

## (4) 环境交互 (Environment Interaction)

环境根据当前动作  $a_t$  返回新的状态  $s_{t+1}$  及奖励  $r_t$ 。经验样本被存入经验回放池,并用于后续批量训练。

DARDC 算法的伪代码如表 1 所示。

表 1 DARDC 算法流程

Table 1 Algorithm process of DARDC

Algorithm1 DARDC

Initialize actor networks  $\pi_{\theta_1}, \pi_{\theta_2}$  and critic networks  $Q_{\omega_1}, Q_{\omega_2}$  with random parameters;

Initialize:  $\theta'_i \leftarrow \theta_i, \theta'_2 \leftarrow \theta_2, \omega'_1 \leftarrow \omega_1, \omega'_2 \leftarrow \omega_2$ ;

Initialize replay buffer: B;

For Episode=1→E do:

Initialize the random process  $N$ , obtain the initial state of the environment  $s$ ;

For  $t=1 \rightarrow T$  do:

Select action with  $\arg \max(Q_{\omega_1}(s, a), Q_{\omega_2}(s, a))$  added  $\varepsilon \sim N(0, \sigma)$

Execute action  $a$ , observe reward  $r$ , new state  $s'$  and done flag  $d$

Store transitions  $(s, a, r, s', d)$  in the replay buffer B

For  $i=1, 2$  do:

Sample  $N$  transitions  $(s, a, r, s', d)$  in the replay buffer B according to the prior experience sampling strategy

$a'_i \leftarrow \mu_i(s_i | \theta_i), a'_2 \leftarrow \mu_2(s_i | \theta_2)$

$\varepsilon \sim \text{clip}(N(0, \bar{\sigma}), -c, c)$

$\hat{V}(s'; v) = (1 - v) \max_{i=1,2} \min_{j=1,2} Q_{\omega_j}(s', \pi_{\theta'_i}(s')) + v \min_{i=1,2} \min_{j=1,2} Q_{\omega'_j}(s', \pi_{\theta'_i}(s'))$

$y_i \leftarrow r + \gamma(1 - d) \hat{V}(s')$

Update critical network parameters

$\omega_i \leftarrow \arg \min_{\omega} \frac{1}{N} \sum_s \{ (Q_{\omega}(s, a) - y)^2 + \lambda [Q_{\omega_1}(s, a) - Q_{\omega_2}(s, a)]^2 \}$

Update actor network parameters according to policy gradient

$\nabla_{\theta_i} J(\theta_i) = \frac{1}{N} \sum_s \nabla_a Q_{\omega_i}(s, a) \bigg|_{a=\mu_{\theta_i}(s)} \nabla_{\theta_i} \mu_{\theta_i}(s)$

Soft update target actor network and target critical network parameters

$\theta'_i \leftarrow \tau \theta_i + (1 - \tau) \theta'_i$

$\omega'_i \leftarrow \tau \omega_i + (1 - \tau) \omega'_i$

End For

End For

End For

## 3 实验分析

### 3.1 青霉素发酵过程优化控制

为规避实际发酵过程不可逆操作的风险,本文采用 Indpensim 开源仿真平台构建数字仿真训练环境。该平台基于 Goldrick 等<sup>[27]</sup>建立的产黄青霉深层发酵动力学模型,通过耦合菌体生长、形态分化、底物代谢与产物合成的非线性常微分方程组,实现对

发酵过程动态特性的高保真模拟。系统呈现典型的部分可观测多输入多输出特性, 单批次发酵时长为 230 h, 采样间隔为 0.2 h, 形成包含 1150 个控制步骤的决策序列。

在青霉素发酵过程中, 要将对生物质生长和代谢产生抑制作用较强的关键工艺状态控制在合理范围内, 如表 2 所示。否则, 生物质的生长将受到不可逆转的损害, 发酵过程将失败。此外, 反应器的体积和质量不应超过其最大容量, 这主要是通过定时排放操作来实现。

青霉素发酵过程的目标是使获得的青霉素产量最大化, 包括批内倾倒的青霉素和批末反应容器内剩余的青霉素, 如式 (13) 所示:

$$Y_i = \rho_T V_T + \sum_1^T \rho_i F_{\text{dis}} \Delta t \quad (13)$$

其中,  $Y_i$  为青霉素产量;  $\rho_T$  和  $V_T$  分别是批末青霉素质量浓度和批末青霉素体积;  $\rho_i$  是批内青霉素质量浓度;  $F_{\text{dis}}$  是溶液倾倒速率;  $\Delta t$  为时间差, 间隔 0.02 h 进行一次取样。

同时, 在保证青霉素产量最大化情况下, 由于批间不确定性, 另一个目标是 minimized 批次间的产量变化, 通过计算批次平均产量来衡量:

$$Y = \frac{1}{N} \sum_1^N Y_i \quad (14)$$

针对青霉素发酵过程的控制问题, 本文构建了基于深度强化学习的交互式优化框架, 如图 2 所示。该框架遵循马尔科夫决策过程范式, 智能体在每个决策时刻  $t$  接收发酵环境的部分可观测状态  $s_t \in S$ , 通过策略网络生成多维连续控制动作  $a_t \in A$  (对应 7 项操作变量: 基质流量( $F_s$ )、油流量( $F_{\text{oil}}$ )、水流量( $F_w$ )、苯乙酸流量( $F_{\text{PAA}}$ )、曝气速率( $F_g$ )、容器压力( $p$ )及  $F_{\text{dis}}$ ), 执行后获得发酵环境反馈的下一状态  $s_{t+1}$  和即时奖励  $r_t$ 。将五元组  $(s_t, a_t, s_{t+1}, r_t, d_t)$  存入优先经验回放池, 通过时序差分误差驱动的优先经验采样机制进行行动者网络和评论家网络的参数更新。

在仿真环境的非稳态批次处理操作中, 若允许控制输入在有界范围内自由变化, 可能会使智能体因数值变化过于剧烈, 导致仿真在发酵结束前提前中断。因此, 我们将专家知识嵌入到这个框架中, 作为智能体学习的起点。

将动作空间对应于需手动控制的 7 个输入变量 ( $F_s, F_{\text{oil}}, F_g, p, F_w, F_{\text{dis}}, F_{\text{PAA}}$ ), 智能体动作输出设计如式 (15) 所示, 其中动作映射到配方的  $(1-\beta)$  和  $(1+\beta)$  之间, 通过策略网络输出的动作参数  $a_t^i$  来调整。在 Indpensim 中执行修改后的操作点, 生成奖励和下一个状态。 $\beta$  是调节控制输入变量浮动范围的超参数, 通过网格搜索法确认最佳参数值, 本文设置为 0.4。

$$\begin{cases} u_t^i = u_i(1 + \beta a_t^i) \\ u_t = [F_s, F_{\text{oil}}, F_g, p, F_{\text{dis}}, F_w, F_{\text{PAA}}] \end{cases} \quad (15)$$

其中,  $a_t^i \in [-1, 1]$ 。

基于工艺特点, 设置青霉素发酵过程状态变量为一个 16 维向量, 如式 (16) 所示, 包括 10 个在线测量状态变量, 5 个离线测量状态变量和发酵时间。

$$\begin{cases} s_t = [x_t^{\text{online}}, x_t^{\text{offline}}, t] \\ x_t^{\text{online}} = [\rho_{\text{DO}_2}, \rho_{\text{DCO}_2}, V, m_t, T', \text{pH}, \\ \varphi_{\text{CO}_2}, \varphi_{\text{O}_2}, v_{\text{OUR}}, v_{\text{CER}}] \\ x_t^{\text{offline}} = [\rho_{\text{PAA}}, \rho_{\text{N}}, \rho_{\text{P}}, \rho_{\text{X}}, \mu] \end{cases} \quad (16)$$

其中,  $\rho_t, \rho_X$  分别为青霉素、菌体的质量浓度;  $T'$  为温度;  $\varphi_{\text{CO}_2}, \varphi_{\text{O}_2}$  分别为尾气中  $\text{CO}_2, \text{O}_2$  的体积分数;  $v_{\text{OUR}}$  为摄氧速率;  $v_{\text{CER}}$  为  $\text{CO}_2$  释放速率。

奖励函数设计遵循最优控制的目标, 即最大限度地提高批次产量, 并降低批次间产量的不稳定性。因此, 奖励函数设计为两部分的加权和: 一是每批次反应器中增加的青霉素产量和倾倒的溶液中青霉素产量之和, 二是累积批次平均值, 如式 (17) 所示。

$$r = w_1 r_{\text{yield}} + w_2 r_{\text{sta}} \quad (17)$$

其中,  $r_{\text{yield}}$  为单批次产量奖励项, 如式 (18) 所示;  $r_{\text{sta}}$  为批次间产量稳定性奖励项, 如式 (19) 所示。权重系数通过网格搜索法确定, 本文设置  $w_1 = 0.7, w_2 = 0.3$ 。

表 2 关键状态变量的合理数值范围

Table 2 Reasonable numerical range of key state variables

| Key state   | $\rho_{\text{PAA}}/(\text{mg} \cdot \text{L}^{-1})$ | $\rho_{\text{N}}/(\text{mg} \cdot \text{L}^{-1})$ | $\mu/(\text{mPa} \cdot \text{s})$ | $\rho_{\text{DO}_2}/(\text{mg} \cdot \text{L}^{-1})$ | $\rho_{\text{DCO}_2}/(\text{mg} \cdot \text{L}^{-1})$ | $V/\text{L}$ | $m_t/\text{t}$ |
|-------------|-----------------------------------------------------|---------------------------------------------------|-----------------------------------|------------------------------------------------------|-------------------------------------------------------|--------------|----------------|
| Lower bound | 200                                                 | 150                                               | —                                 | 5.1                                                  | —                                                     | 0            | 0              |
| Upper bound | 2400                                                | —                                                 | 100                               | —                                                    | 7.57                                                  | 100 000      | 140            |

$\rho_{\text{PAA}}$ —Phenylacetic acid mass concentration;  $\rho_{\text{N}}$ —Ammonia mass concentration;  $\mu$ —Viscosity;  $\rho_{\text{DO}_2}$ —Dissolved oxygen mass concentration;  $\rho_{\text{DCO}_2}$ —Dissolved carbon dioxide mass concentration;  $V$ —Vessel volume;  $m_t$ —Vessel mass

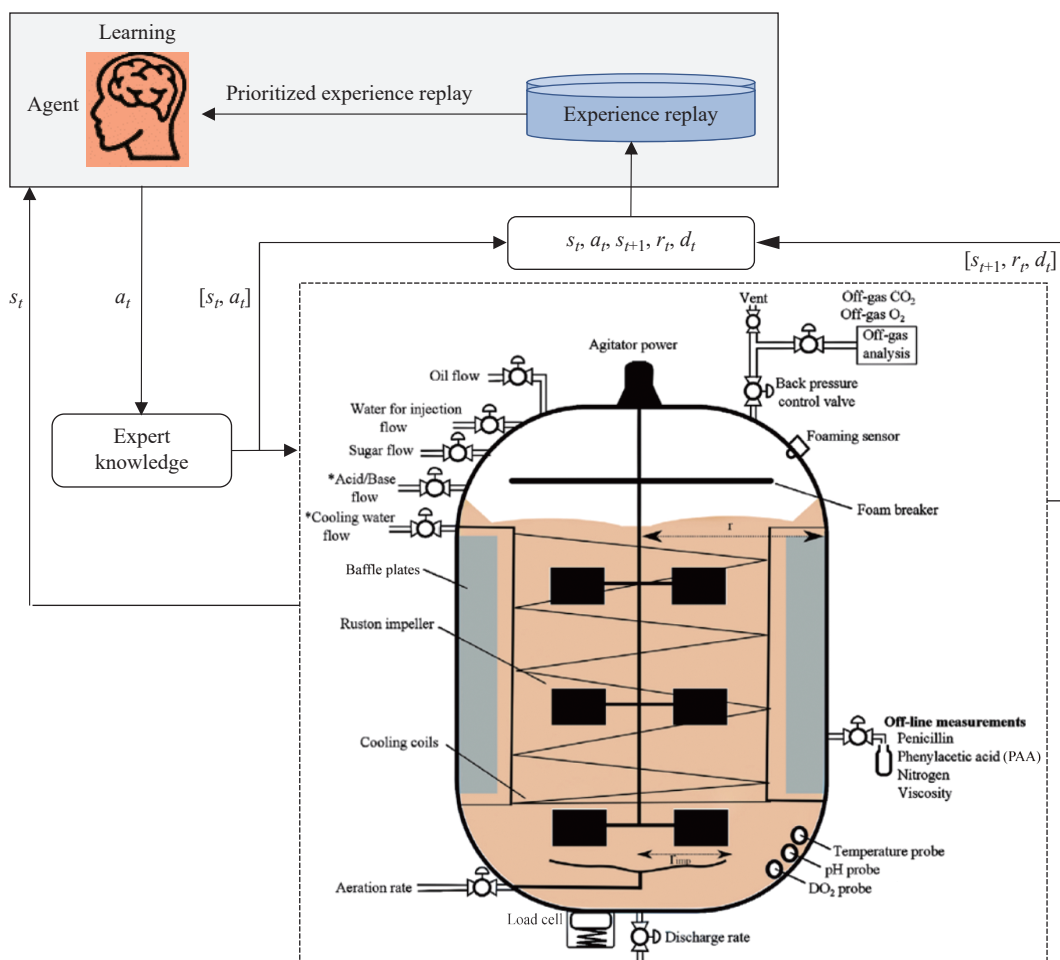


图 2 基于 DRL 的青霉素发酵控制框架

Fig. 2 Control framework of penicillin fermentation based on DRL

$$r_{\text{yield}} = \rho_t V_t - \rho_{t-1} V_{t-1} + \rho_t F_{\text{dis}} \Delta t \quad (18)$$

$$r_{\text{sta}} = \bar{r}_n \quad (19)$$

其中,  $\bar{r}_n$  表示随机初始参数批次得到的产量平均值。

### 3.2 实验设置

实验基于 Python 语言, 显卡配置为 NVIDIA GeForce RTX 3080。为了验证基于 DARDC 算法的青霉素发酵过程控制方法的效果, 以 Indpensim 发酵仿真环境为实验对象进行仿真实验, 并与 RECIPE、PID、NMPC、SAC<sup>[28]</sup> 和 DDPG 方法的实验效果进行

对比。RECIPE 和 PID 方法可在 Indpensim 库中直接调用<sup>[29]</sup>。

首先使用随机初始状态的 Indpensim 环境对 DARDC 模型进行训练, 训练轮次为 500 轮, 算法训练 10 次, 取均值作为最终的模型训练实验结果。训练完成后, 将得到的基于 DRL 的控制方法与 RECIPE、PID、NMPC、SAC 和 DDPG 控制方法一起在无随机初始状态和随机初始状态的 Indpensim 环境下进行测试, 观察动作空间, 状态空间和产量的变化。DARDC 模型的实验参数设置如表 3 所示。

表 3 实验参数设置

Table 3 Experimental parameter setting

| Actor network learning rate | Critic network learning rate | Replay buffer length | Priority intensity | Importance sampling | Stability | Batch size | Discount factor | Regularization parameter | Soft update parameter |
|-----------------------------|------------------------------|----------------------|--------------------|---------------------|-----------|------------|-----------------|--------------------------|-----------------------|
| 0.0003                      | 0.001                        | 30 000               | 0.6                | 0.4                 | $10^{-6}$ | 128        | 0.99            | 0.005                    | 0.001                 |

### 3.3 模型训练效果

DARDC 模型的训练过程如下: DARDC 模型与随机初始状态下的 Indpensim 环境进行交互, 在每个

批次中, 模型网络参数每一步都会被更新。训练结束后, 使用存储的最佳模型用于过程控制的测试。

在对比 DARDC 算法与 DDPG、SAC 算法的过

程中, 三者的基础参数设置相同, 并采取同样的专家规则和优先经验回放机制。图3示出了DDPG、SAC与DARDC方法在训练过程中奖励趋势变化的对比图。从图中可以看出, 虽然DDPG方法最终的奖励曲线不断地上升并收敛, 但是其训练过程不稳定, 存在多次波动。SAC方法的奖励曲线效果略好于DDPG方法, 训练过程更加平稳。而DARDC方法收敛速度快于DDPG和SAC, 训练过程稳定性强于DDPG和SAC, 且最终的奖励值高于另外两者。

### 3.4 单批次与多批次过程控制效果

训练完成后, 首先在确定初始状态的单批次发酵过程上对6个控制方法进行测试。6个控制方法生成的控制变量输入轨迹曲线如图4所示。观测到的控制轨迹呈现出显著的非稳态运行模式, 伴随着时间推移, 控制输入变量的波动显著。这一动态特性与反应器内微生物在不同生长阶段的行为相吻合。微生物发酵过程的起始阶段以生物量的迅速累积为特征, 即为快速生长期。在此期间, 为促进生物量的增长, 需适时补充糖类、油脂及氮源等营养物质。随后, 系统逐渐过渡至一个相对稳定的青霉素生产阶段。当生物量累积至一定水平时, 青霉素的

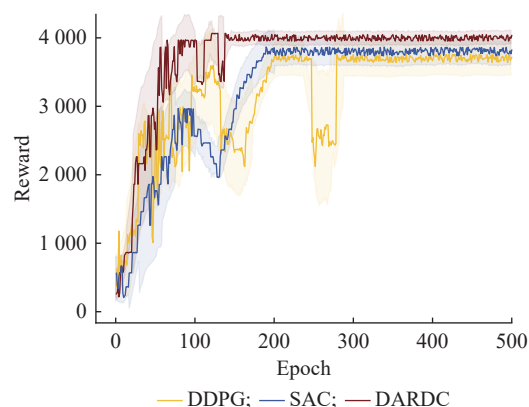


图3 训练曲线对比图

Fig. 3 Comparison of training curves

合成活动开始。此时, PAA作为青霉素合成路径中不可或缺的侧链前体, 其添加量需要大幅增加。然而, 需要注意的是, 过高的PAA质量浓度在培养体系中对微生物生长及青霉素的产生具有抑制作用。因此, 在青霉素发酵初期, 生物量快速增长, 青霉素的合成效率相对较低, 应避免引入过量PAA溶液。这些动态变化特征在图4所示的控制轨迹中得到了直观的体现。

对比NMPC控制方法和其他5种控制方法的控

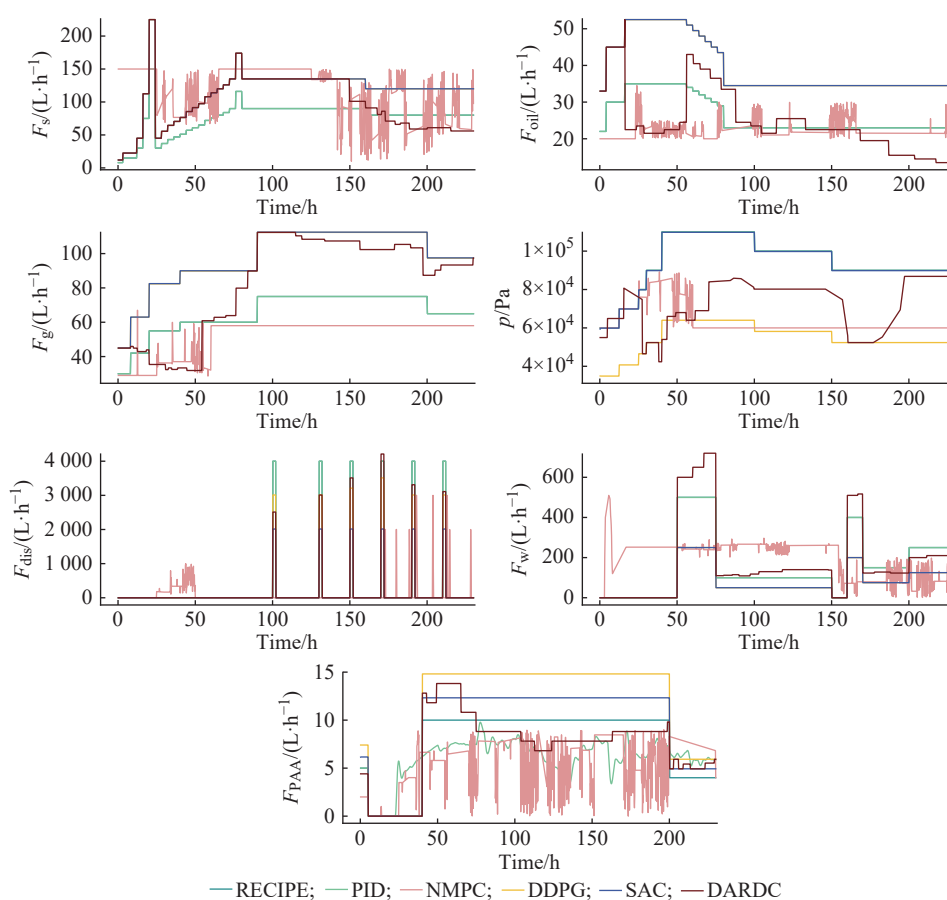


图4 控制变量在单批次发酵过程中的控制轨迹

Fig. 4 Control trajectory of control variables in single batch fermentation process

制轨迹,可以观察到 NMPC 控制方法的控制信号变化率明显更高,并且出现振荡现象。这表明控制信号不处于理想的稳定状态。相比之下,RECIPE 和 PID 控制方法显示出更一致的控制轨迹。对于 DRL 控制方法,由于在动作空间上施加了约束,生成的轨迹是 RECIPE 控制方法的修改版本。

图 5 描述了在青霉素发酵期间关键过程状态变量的变化轨迹。灰色虚线和红色虚线代表一些状态变量的约束范围。有效控制 PAA、N、DO<sub>2</sub> 和 DCO<sub>2</sub> 的状态对于青霉素发酵过程至关重要,因为一旦这些参数超出适宜范围,将会对生物量的积累以及青霉素的生产产生显著的抑制效应。从青霉素质量浓度变化曲线中可以看出,DARDC 在整个批次中都处于领先地位。DDPG 在 140 h 前的表现良好,但在 140 h 后,由于未能将 PAA 质量浓度控制在合理范围内,生物量和青霉素质量浓度上升趋势开始下降。

在对单批次发酵过程进行测试后,对 6 个控制方法进行 30 次随机初始状态下多批次发酵过程控制测试,用以检验扰动下各控制方法的性能和稳定性。如图 6 所示,DARDC 的批次平均产量最高,达到 4054.58 kg。NMPC 的批次平均产量为 3819.99 kg,

DDPG 的批次平均产量为 3678.21 kg,SAC 的批次平均产量为 3827.87 kg。这 4 种控制方法超过了现有的 RECIPE 和 PID 控制方法(平均产量分别为 2838.25 kg 和 3403.44 kg)。同时,DARDC 控制方法也保持着较好的稳定性也较好,它对所有测试批次都保持较小的方差,标准差为 577.66 kg。而 DDPG 的最高批次产量达到 5640.01 kg,最差批次产量仅为 914.23 kg,其产率的标准差在所有控制方法中最大,说明其性能受过程干扰影响较大,非常不稳定。与 NMPC 相比,DARDC 方法无需建立青霉素发酵过程模型,只需要更少的测量作为模型输入状态,且实现了更好的产品产量。对于缺少状态测量的工业场景,DARDC 方法不需要状态估计,从与环境的交互中就可以隐式学习控制策略。这意味着在过程优化任务期间不需要额外的工作,例如在设备中安装额外的传感器或开发状态估计模型。除此之外,DARDC 方法的另一个优点是计算速度快。DARDC 方法将计算时间从在线预测阶段转移到离线训练阶段。也就是说,在训练阶段之后,DARDC 方法在线预测阶段具有更优的时间效率。

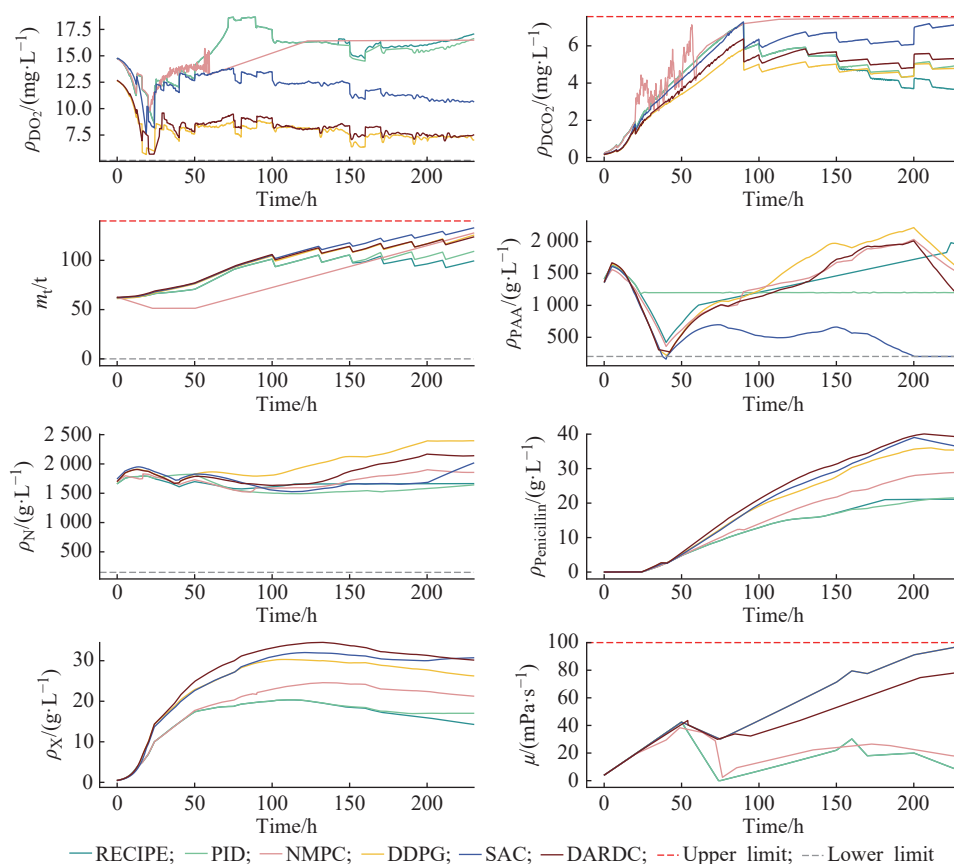


图 5 关键状态变量在单批次发酵过程中的输出轨迹

Fig. 5 Output trajectory of key state variables in single batch fermentation process

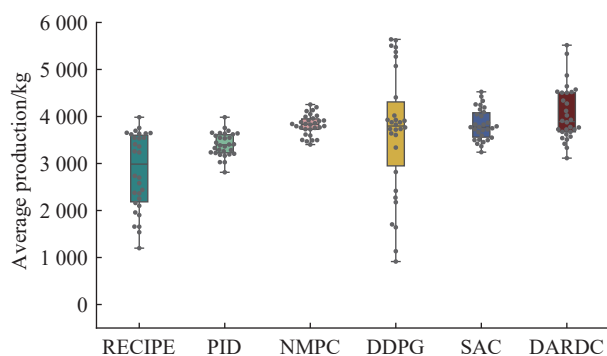


图6 不同控制策略在多批次发酵过程下的青霉素产量箱线图

Fig. 6 Box plot of penicillin yield under different control strategies in multi batch fermentation process

## 4 结 论

本文提出了一种基于深度强化学习的间歇过程优化控制方法 DARDC, 通过采用典型 AC 架构构建深度强化学习算法, 实现无需依赖过程机理模型的间歇过程智能控制。具体而言, 设计了双行动者网络并行训练机制以增强智能体的环境探索能力, 同时引入双评论家网络结构并设置正则化约束项, 有效缓解价值函数高估问题并维持网络间参数差异的稳定边界。为进一步提升算法训练性能, 采用了优先经验回放策略优化样本利用率, 显著缩短了网络收敛时间。实验结果表明, 模型训练性能优于 DDPG 和 SAC 算法。在随机批次发酵场景下, DARDC 方法相较于其他 DRL 控制方法展现出较大的训练效率提升, 青霉素平均产量相比较 PID 方法从 3403.44 kg 提升至 4054.58 kg, 且具有更优的鲁棒性与稳定性。该方法未来可通过融合离线强化学习框架, 利用历史发酵数据进行预训练, 有望推进工业场景的部署进程。

### 参考文献:

- [1] SITTIG W. The present state of fermentation reactors[J]. Journal of Chemical Technology and Biotechnology, 1982, 32(1): 47-58.
- [2] KUMAR A, UDUGAMA I A, GARGALO C L, *et al.* Why is batch processing still dominating the biologics landscape? Towards an integrated continuous bioprocessing alternative[J]. Processes, 2020, 8(12): 1641.
- [3] GOLDRICK S, DURAN-VILLALOBOS C A, JANKAUSKAS K, *et al.* Modern day monitoring and control challenges outlined on an industrial-scale benchmark fermentation process[J]. Computers & Chemical Engineering, 2019, 130: 106471.
- [4] DAEMMRICH A. Synthesis by microbes or chemists? Pharmaceutical research and manufacturing in the antibiotic era[J]. History and Technology, 2009, 25(3): 237-256.
- [5] LI D, ZHU F, WANG X, *et al.* Multi-objective reinforcement learning for fed-batch fermentation process control[J]. Journal of Process Control, 2022, 115: 89-99.
- [6] OH T H, PARK H M, KIM J W, *et al.* Integration of reinforcement learning and model predictive control to optimize semi-batch bioreactor[J]. AIChE Journal, 2022, 68(6): e17658.
- [7] MORARI M, LEE J H. Model predictive control: Past, present and future[J]. Computers & Chemical Engineering, 1999, 23(4/5): 667-682.
- [8] HENSON M A. Nonlinear model predictive control: Current status and future directions[J]. Computers & Chemical Engineering, 1998, 23: 187-202.
- [9] LOPEZ-MONTERO E B, WAN J, MARJANOVIC O. Trajectory tracking of batch product quality using intermittent measurements and moving window estimation[J]. Journal of Process Control, 2015, 25: 115-128.
- [10] JANG H, LEE J H, BIEGLER L T. A robust NMPC scheme for semi-batch polymerization reactors[J]. IFAC-PapersOnLine, 2016, 49(7): 37-42.
- [11] HONG M S, BRAATZ R D. Mechanistic modeling and parameter-adaptive nonlinear model predictive control of a microbioreactor[J]. Computers & Chemical Engineering, 2021, 147: 107255.
- [12] SHAKYA A K, PILLAI G, CHAKRABARTY S. Reinforcement learning algorithms: A brief survey[J]. Expert Systems with Applications, 2023, 231: 120495.
- [13] DOGRU O, XIE J, PRAKASH O, *et al.* Reinforcement learning in process industries: Review and perspective[J]. IEEE/CAA Journal of Automatica Sinica, 2024, 11(2): 283-300.
- [14] NIAN R, LIU J, HUANG B. A review on reinforcement learning: Introduction and applications in industrial process control[J]. Computers & Chemical Engineering, 2020, 139: 106886.
- [15] KIM J W, PARK B J, OH T H, *et al.* Model-based reinforcement learning and predictive control for two-stage optimal control of fed-batch bioreactor[J]. Computers & Chemical Engineering, 2021, 154: 107465.
- [16] 邓绍斌, 朱军, 周晓峰, 等. 基于局部策略交互探索的深度确定性策略梯度的工业过程控制方法 [J]. 计算机应用, 2022, 42(5): 1642-1648.
- [17] LI H, QIU T, YOU F. AI-based optimal control of fed-batch biopharmaceutical process leveraging deep reinforcement learning[J]. Chemical Engineering Science, 2024, 292: 119990.

- [18] FUJIMOTO S, HOOFF H, MEGER D. Addressing function approximation error in actor-critic methods[C]//Proceedings of the 35th International Conference on Machine Learning. [s.l.]: PMLR, 2018: 1587-1596.
- [19] JOSHI T, MAKKER S, KODAMANA H, *et al.* Twin actor twin delayed deep deterministic policy gradient (TATD3) learning for batch process control[J]. *Computers & Chemical Engineering*, 2021, 155: 107527.
- [20] 马军伟, 徐琛, 陶洪峰, 等. 基于双行动者深度确定性策略梯度算法的间歇过程控制[J]. *信息与控制*, 2023, 52(6): 773-783.
- [21] ZHANG Y, ZHANG K, HU T, *et al.* Research on intelligent monitoring and concentration prediction for penicillin fermentation process[J]. *Biotechnology and Bioengineering*, 2025, 122(3): 570-578.
- [22] LILLICRAP T P, HUNT J J, PRITZEL A, *et al.* Continuous control with deep reinforcement learning[EB/OL]. (2015-09-09). <https://arxiv.org/abs/1509.02971>.
- [23] KANG C, RONG C, REN W, *et al.* Deep deterministic policy gradient based on double network prioritized experience replay[J]. *IEEE Access*, 2021, 9: 60296-60308.
- [24] LYU J, MA X, YAN J, *et al.* Efficient continuous control with double actors and regularized critics[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, 36(7): 7655-7663.
- [25] BONEY R, KANNALA J, ILIN A. Regularizing model-based planning with energy-based models[C]//Proceedings of the Conference on Robot Learning. [s.l.]: PMLR, 2020: 182-191.
- [26] SCHAU T, QUAN J, ANTONOGLOU I, *et al.* Prioritized experience replay[EB/OL]. (2015-11-18). <https://arxiv.org/abs/1511.05952>.
- [27] GOLDRICK S, ȘTEFAN A, LOVETT D, *et al.* The development of an industrial-scale fed-batch fermentation simulation[J]. *Journal of Biotechnology*, 2015, 193: 70-82.
- [28] HAARNOJA T, ZHOU A, ABBEEL P, *et al.* Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor[C]//Proceedings of the 35th International Conference on Machine Learning. [s.l.]: PMLR, 2018: 1861-1870.
- [29] ZHANG M, WANG X, DECARDI-NELSON B, *et al.* Smpl: Simulated industrial manufacturing and process control learning environments[J]. *Advances in Neural Information Processing Systems*, 2022, 35: 26631-26644.

## Optimal Control of Batch Process Based on Double Actors and Double Critics Algorithm

TANG Zekai, LUO Na

(Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, East China University of Science and Technology, Shanghai 200237, China)

**Abstract:** When using deep reinforcement learning methods to optimize the control of batch processes, there are problems such as low sample learning efficiency, long training time and unstable training. To solve these problems, a Double Actors and Regularized Double Critics (DARDC) algorithm is proposed. The algorithm enhances the agent's exploration ability in the complex state space through the parallel double actor network, effectively suppresses the value function estimation deviation by virtue of the regularization term constraint double critic network, and introduces the prioritized experience replay mechanism to accelerate the training convergence. Further, in order to ensure the rationality of control quantities output by the agent, expert experience rules are embedded into the action output. In the experiment of penicillin fermentation process, the performance of the proposed algorithm is better than the traditional deep reinforcement learning algorithm, and compared with PID control, the average penicillin yield of the key control index is increased by 19.1%. The results show that the DARDC algorithm has certain effectiveness and robustness, and provides a new method for the optimal control of batch process.

**Key words:** deep reinforcement learning; optimal control of batch process; double actors; double critics; prioritized experience replay

(责任编辑: 李娟)