基于强化学习的浓密机底流浓度在线控制算法

袁兆麟 [1] 何润姿 [1] 姚 超 [1] 李 佳 [1] 班晓娟 [1]

摘 要 复杂过程工业控制一直是控制应用领域研究的前沿问题. 浓密机作为一种复杂大型工业设备广泛用于冶金、采矿等领域. 由于其在运行过程中具有多变量、非线性、高时滞等特点, 浓密机的底流浓度控制技术一直是学界、工业界的研究难点与热点. 本文提出了一种基于强化学习技术的浓密机在线控制算法. 该算法在传统启发式动态规划 (Heuristic dynamic programming, HDP) 算法的基础上, 设计融合了评价网络与模型网络的双网结构, 并提出了基于短期经验回放的方法用于增强评价网络的训练准确性, 实现了对浓密机底流浓度的稳定控制, 并保持控制输入稳定在设定范围之内. 最后, 通过浓密机仿真实验的方式验证了算法的有效性, 实验结果表明本文提出的方法在时间消耗、控制精度上优于其他算法.

关键词 自适应动态规划, 强化学习, 最优控制, 浓密机控制, 神经网络

引用格式 袁兆麟, 何润姿, 姚超, 李佳, 班晓娟. 基于强化学习的浓密机底流浓度在线控制算法. 自动化学报, 2021, **47**(7): 1558−1571

DOI 10.16383/j.aas.c190348

Online Reinforcement Learning Control Algorithm for Concentration of Thickener Underflow

YUAN Zhao-Lin[1] HE Run-Zi[1] YAO Chao[1] LI Jia[1] BAN Xiao-Juan[1]

Abstract Complex process industrial control is a widely concerned problem in the field of control application. As a kind of complex huge industrial equipment, thickener has been widely used in metallurgy, mining and other applications. Due to its characteristics of complicated variables, nonlinear and long delay in the operational process, the control strategy of underflow concentration for thickener has always been a hot and difficult issue in the academia and industry. This paper proposes a novel online control algorithm for thickener which is based on reinforcement learning. Inspired by the traditional heuristic dynamic programming (Heuristic dynamic programming, HDP) algorithm. The proposed method designs a double net framework which is composed of the critic network and the model network. To achieve the stabilization of underflow concentration, an optimal method which is based on reviewing the history data in a short term is proposed in the training phase of critic network. Simulation experiments verify efficiency of the proposed method. The results show that the proposed method can maintain the concentration of underflow in a stable horizon and performs better than other algorithms in accuracy and time consuming.

Key words Adaptive dynamic programming, reinforcement learning, optimal control, thickener control, neural networks

Citation Yuan Zhao-Lin, He Run-Zi, Yao Chao, Li Jia, Ban Xiao-Juan. Online reinforcement learning control algorithm for concentration of thickener underflow. *Acta Automatica Sinica*, 2021, **47**(7): 1558−1571

在现代复杂过程工业生产中, 对控制性能指标进行优化是不同控制算法、控制系统的首要任务. 在冶金、采矿领域等复杂过程工业场景下, 浓密机是一种被广泛应用的大型沉降工具, 它通过重力沉降作用可以将低浓度的固液混合物进行浓缩形成高浓度的混合物, 起到减水、浓缩的作用. 在对浓密机进行控制时, 底流浓度是核心控制指标. 该参量与其他过程监控变量如进料流量、进料浓度、出料流量、泥层高度有着复杂的耦合关系. 在大部分的实际生产过程中, 浓密机底流浓度的控制一般是操作员根据个人经验, 通过对底流流量设定值、絮凝剂流量设定值进行调节, 间接地使底流浓度追踪其工艺设定值. 但是由于浓密机运行过程具有非线性、多变量、高时滞等特点, 操作员难以维持底流浓度持续稳定, 浓度存在偏差的底流会导致产品质量退化以及增加工业生产成本.

收稿日期 2019-05-10 录用日期 2019-08-15
Manuscript received May 10, 2019; accepted August 15, 2019
海南省重点研发计划 (ZDYF2019009), 国家重点基础研究发展计划 (2019YFC0605300, 2016YFB0700500),国家自然科学基金 (61572075, 61702036, 61873299) 资助
Supported by Finance Science and Technology Project of Hainan Province (ZDYF2019009), National Key Research and Development Program of China (2019YFC0605300, 2016YFB0700500), National Natural Science Foundation of China (61572075, 61702036, 61873299)
本文责任编委 王占山
Recommended by Associate Editor WANG Zhan-Shan
1. 北京科技大学计算机与通信工程学院 北京 100083
1. School of Computer and Communication Engineering University of Science & Technology Beijing, Beijing 100083

浓密机是一种典型的复杂过程工业设备, 关于过程工业设备优化控制的研究一直是工业界、学术界研究的热点问题. 对于机械结构明确、且能够精确建立动态模型的工业设备, 可以采用基于模型的优化控制方法, 如: 实时优化控制 (Realtime optimization, RTO)[1]、模型预测控制 (Model predictive control, MPC)[2] 等. 但由于浓密机系统机械结构复杂、部分变量难以观测, 因此难以建立准确的数学模型近似其运转机理, 导致基于模型的方法无法适用于此类复杂工业设备的控制. 研究人员提出了基于数据驱动的控制方法来实现对此类无模型工业设备的控制. Dai 等[3] 提出了用于解决赤铁矿研磨系统控制问题的数据驱动优化 (Date driven opimization, DDO) 控制算法. Wang 等[4] 采用基于数据驱动的自适应评价方法解决连续时间未知非线性系统的无穷范围鲁棒最优控制问题.

近年来, 基于强化学习[5−6] 理论的最优控制技术, 也称为自适应动态规划 (Adaptive dynamic programming, ADP)[7−9] 技术, 是控制领域的研究热点话题. 典型的自适应动态规划算法, 如 HDP、双启发式动态规划 (Dual heuristic programming, DHP)、动作依赖启发式动态规划 (Action dependent heuristic dynamic programming, ADHDP)[8] 等均采用多个神经网络分别对被控系统动态模型、控制策略、策略评价模型进行建模. 此类方法可以在模型未知的情况下以数据驱动的方式在线学习控制策略. Liu 等[10] 提出了一种在线自适应动态规划算法用来解决离散时间多输入多输出仿射系统控制问题, 且该方法仅需要训练少量网络参数. Liu 等[11] 采用一种基于强化学习的自适应跟踪控制技术解决多输入多输出系统容错控制问题. Xu 等[12] 采用拉普拉斯特征映射算法提取被控系统全局特征, 并将该全局特征用于 DHP 算法中以增强值函数网络的近似能力.

近年来, 利用自适应动态规划方法解决过程工业控制问题也取得很大研究进展. Wei 等[13] 将煤炭气化过程的最优追踪控制转化为双人零和最优控制问题, 并采用迭代自适应动态规划方法求解最优控制率, 同时给出了收敛稳定性的分析. Jiang 等[14] 利用穿插学习策略迭代 (Interleaved learning policy iteration, ILPL) 实现了对浮选过程操作指标优化的控制, 获得了比传统值函数迭代 (Value iteration, VI)、策略迭代 (Policy iteration, PI) 算法更佳的控制效果. Jiang 等[15] 将强化学习与举升方法结合 (Lifting technology), 实现了对浮选过程设备层与操作层双速率系统的最优控制.

上述算法均使用被控系统实时生成的数据对神经网络进行训练, 该训练方法忽略了系统在短期内产生的历史轨迹数据对模型学习的影响. 同时, 在工业场景下进行设备在线控制对算法实时性要求较高. 上述方法对于控制量的计算均依托于表征控制策略的神经网络, 而对于控制网络或动作网络的训练将产生较大的时间开销. 为了解决上述问题, 本文引入了短期经验回放技术[16−17] 以对短期内的系统运行轨迹数据进行回放训练. 实验证明该技术有效增强了算法收敛稳定性, 且在其他 ADP 类在线控制算法中具有通用性. 同时本文根据浓密机系统特性提出了一种迭代梯度优化算法, 该算法可以在没有动作网络的情况下求解控制输入量. 实验表明该方法能够在提升控制精度的同时, 减少模型学习过程中产生的时间消耗.

本文主要贡献总结如下:

1) 提出了一种基于 ADP 算法架构的启发式评价网络值迭代算法 (Heuristic critic network value iteration, HCNVI). 该算法仅通过评价网络、模型网络和梯度优化算法即可求解系统最优控制输入.

2) 提出了一种适用于评价网络训练的短期经验回放技术. 训练评价网络时, 将短期内系统运行轨迹数据共同用于模型训练, 该方法可以有效增强评价网络收敛速度.

3) 通过浓密机仿真实验验证了 HCNVI 算法的有效性. 实验结果表明本文提出方法在时间消耗、控制精度上均优于其他对比方法.

本文正文部分组织如下: 第 1 节, 对浓密机沉降过程进行形式化描述. 第 2 节, HCNVI 算法介绍以及利用该算法实现浓密机在线控制. 第 3 节, 通过两组仿真实验验证本文提出控制模型的有效性. 第 4 节对本文研究工作进行总结.

1　浓密过程控制问题描述

浓密机在采矿、冶金领域是重要的沉降分离设备, 其运行过程如图 1 所示. 低浓度的料浆源源不断地流入浓密机顶部进料口. 利用沙粒的密度大于水的特性以及絮凝剂的絮凝作用, 料浆中沙粒不断沉降, 并在浓密机底部形成高浓度的底流料浆. 高浓度的底流料浆多以管道输送的形式流至其他工业设备进行后续加工处理.

对于浓密沉降控制过程的性能进行评价, 其核心控制指标为底流浓度 y. 该因素受控制输入、系统状态参量、及其他外部噪音扰动影响. 控制输入包括底流泵转速 $u_1(k)$ 以及絮凝剂泵转速 $u_2(k)$, 系统状态参量为泥层高度 $h(k)$, 外部噪音输入为进料流

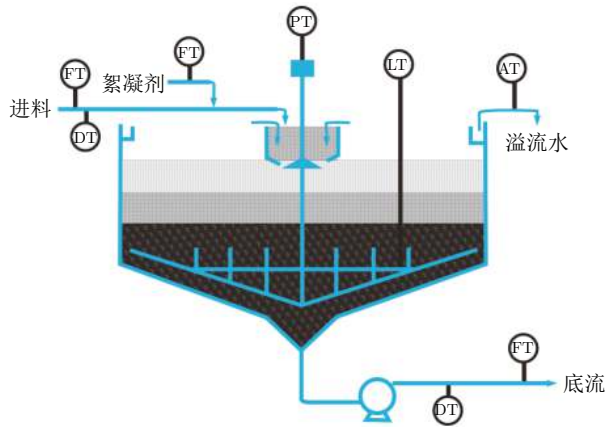


图 1 浓密过程示意图

Fig. 1 Illustration of thickening process.

量 $c_1(k)$、进料浓度 $c_2(k)$. 由于在部分工业场景中, 上游工序产生的物料浓度、物料流量是不可控的. 为了使提出的浓密机控制模型具有通用性, 因此本文将进料状态作为噪音输入量. 浓密机进料颗粒大小, 进料成分都会对浓密机底流浓度产生影响. 不过由于此类变量无法观测且波动较小, 为了简化问题, 本文假定其保持恒定. 根据上述定义, 其中 $\boldsymbol{u}(k)=[u_1(k),u_2(k)]^{\mathrm{T}}\in\mathbf{R}^2$ 为可控制输入量, $\boldsymbol{c}(k)=[c_1(k),c_2(k)]^{\mathrm{T}}\in\mathbf{R}^2$ 为不可控但是可观测的噪音量, $h(k)\in\mathbf{R}$ 为系统状态量, 该参量是表征当前浓密机状态的重要参量, 它可被间接控制但不作为控制目标. 因此, 浓密机系统可表述为式 (1) 形式的非线性系统, 其中 $f(\cdot)$ 为未知非线性函数.

$$[y(k+1),h(k+1)]^{\mathrm{T}}=f(y(k),\boldsymbol{u}(k),\boldsymbol{c}(k),h(k)) \quad (1)$$

本文提出的浓密机底流浓度控制算法, 可以根据当前底流浓度 $y(k)$、泥层高度 $h(k)$、进料流量 $c_1(k)$、进料浓度 $c_2(k)$ 几个状态量, 自动地调节底流泵速 $u_1(k)$ 和絮凝剂泵速 $u_2(k)$, 使底流浓度 $y(\cdot)$ 追踪其设定值 y^*.

2 利用 HCNVI 算法实现浓密机底流浓度在线控制

当前, 工业场景下控制浓密机的方法主要依靠操作员手工控制. 操作员根据生产经验给出絮凝剂添加量的设定值 (m^3/h) 以及底流流量设定值 (m^3/h), 浓密机内相配套的回路控制系统会根据设定值的大小自动调节絮凝剂泵速 (Hz) 与底流泵速 (Hz), 使絮凝剂的实时流量、底流实时流量追踪操作员给出的设定值. 然而, 由于浓密机系统的复杂性, 操作员难以实时、完整地掌握系统运行参数, 因此无法及时、准确地设定目标点位. 这导致在实际生产过程中, 浓密机常常处于非最优工作状态, 底流浓度大范围频繁波动, 偏离理想的底流浓度.

对于浓密过程式 (1), 控制系统的首要目标是使底流浓度 $y(k)$, 追踪其设定值 $y^*(k)$. 另外, 为了保证系统运行安全与仪器寿命, 控制输入必须满足一定的限制条件. 综合上述指标因素, 可以将浓密机控制问题转化为有约束的最优化问题式 (2).

$$\begin{aligned}&\min_{\boldsymbol{u}(k)} \quad J(k)=\sum_{l=k}^{\infty}\gamma^{l-k}U(l)\\&\text{s.t. } [y(k+1),h(k+1)]^{\mathrm{T}}=f(y(k),\boldsymbol{u}(k),\boldsymbol{c}(k),h(k)),\\&\quad u_{i\min}\le u_i(k)\le u_{i\max},i=1,2\end{aligned} \quad (2)$$

$$\begin{aligned}U(k)=&Q\left(y(k)-y^*\right)^2+\\&\left(\boldsymbol{u}(k)-\frac{\boldsymbol{u}_{\mathrm{mid}}}{2}\right)^{\mathrm{T}}R\left(\boldsymbol{u}(k)-\frac{\boldsymbol{u}_{\mathrm{mid}}}{2}\right)\end{aligned} \quad (3)$$

$J(k)$ 为折扣累计评价值函数, 用来评估控制策略的好坏. 式 (3) 是效用函数, 代表在当前状态 $y(k)$ 下, 执行控制输入 $\boldsymbol{u}(k)$ 需要承受的代价. $\gamma\in(0,1]$ 是折扣因子, 代表系统短期控制过程中产生的惩罚值在累计惩罚项所占比重. $Q>0$, R 是对称正定矩阵, $u_{i\min}$, $u_{i\max}$ 分别代表对 $u_i(k)$ 的限制, $\boldsymbol{u}_{\mathrm{mid}}=\dfrac{\boldsymbol{u}_{\max}+\boldsymbol{u}_{\min}}{2}$.

2.1 理论最优控制模型

本节根据对式 (2) 的定义, 求解理想情况下最优控制输入 $\boldsymbol{u}^*(k)$.

式 (2) 可以表示为式 (4) 贝尔曼方程的形式:

$$\begin{aligned}J(k)=&U(k)+\gamma\sum_{l=k+1}^{\infty}\gamma^{l-k-1}U(l)=\\&U(k)+\gamma J(k+1)\end{aligned} \quad (4)$$

根据贝尔曼最优原则, 第 k 时刻的最优评价值函数 $J^*(k)$ 满足离散哈密顿 − 雅可比 − 贝尔曼方程

$$J^*(k)=\min_{\boldsymbol{u}_k}\{U(k)+\gamma J^*(k+1)\} \quad (5)$$

第 k 时刻, 最优的控制输入 $\boldsymbol{u}^*(k)$ 可以表示为

$$\boldsymbol{u}^*(k)=\arg\min_{\boldsymbol{u}_k}\{U(k)+\gamma J^*(k+1)\} \quad (6)$$

由于式 (1) 中 $f(\cdot)$ 是复杂非线性函数, 无法直接对式 (5) 进行求解, 但可以利用算法 1 以值函数迭代的方式求解最优值函数和最优控制律, 其中 $x(k)$ 用于表征系统状态, $\boldsymbol{x}(k)=[y(k),h(k),\boldsymbol{c}(k)^{\mathrm{T}}]^{\mathrm{T}}$. 根据文献 [18], 可以证明当 $i\to\infty$ 时, 值函数 $V_i\to J^*$, 控制律 $\boldsymbol{u}_i\to\boldsymbol{u}^*$.

算法 1. 值迭代算法

初始化: 随机定义 $V_0(\cdot)$

1: 定义控制约束集合 $\Omega_{\boldsymbol{u}}=\{\boldsymbol{u}:\boldsymbol{u}_{\min}\leq\boldsymbol{u}\leq\boldsymbol{u}_{\max}\}$

2: **for** $i=0,1,2,\cdots,\infty$ **do**

3: 策略改进

$$\boldsymbol{u}_i(k)=\arg\min_{\boldsymbol{u}_k\in\Omega_{\boldsymbol{u}}}U(y(k),\boldsymbol{u}(k))+\gamma V_i(\boldsymbol{x}(k+1)) \quad (7)$$

4: 策略评估

$$V_{i+1}(\boldsymbol{x}(k))=U(y(k),\boldsymbol{u}_i(k))+\gamma V_i(\boldsymbol{x}(k+1)) \quad (8)$$

2.2　启发式评价网络值迭代算法

本节将基于算法 1, 提出一种启发式评价网络值迭代算法. 该算法能根据浓密机系统产生的实时监测数据 $\boldsymbol{x}(k)$ 进行在线学习, 并产生满足 $\Omega_{\boldsymbol{u}}$ 约束的控制输入量 $\boldsymbol{u}(k)$, 且最小化 $J(k)$. 算法整体结构如图 2 所示. HCNVI 算法中包含两个神经网络, 分别是模型网络和评价网络. 神经网络均采用单隐层人工神经网络, 其基本结构如图 3 所示. 模型网络的训练全部离线进行, 在控制任务开始后, 将不再对模型网络参数进行调整. 控制动作决策算法根据浓密机实时反馈状态 $\boldsymbol{x}(k)$ 计算控制变量 $\boldsymbol{u}(k)$ 并用于浓密机系统控制, $\boldsymbol{u}(k),\boldsymbol{x}(k)$ 被放入短期经验数据暂存区存储. 模型训练时, 由短期经验暂存区提供训练数据供模型训练. 算法学习过程中, 仅评价网络参数发生改变.

评价网络. HCNVI 采用一个称为评价网络的神经网络来近似算法 1 中的 $V(\cdot)$ 函数. 神经网络选择单隐层人工神经网络, 其基本结构如图 3 所示. 评价网络的具体定义如下:

$$\hat{J}(k)=W_{c2}\tanh(W_{c1}(\boldsymbol{x}(k))) \quad (9)$$

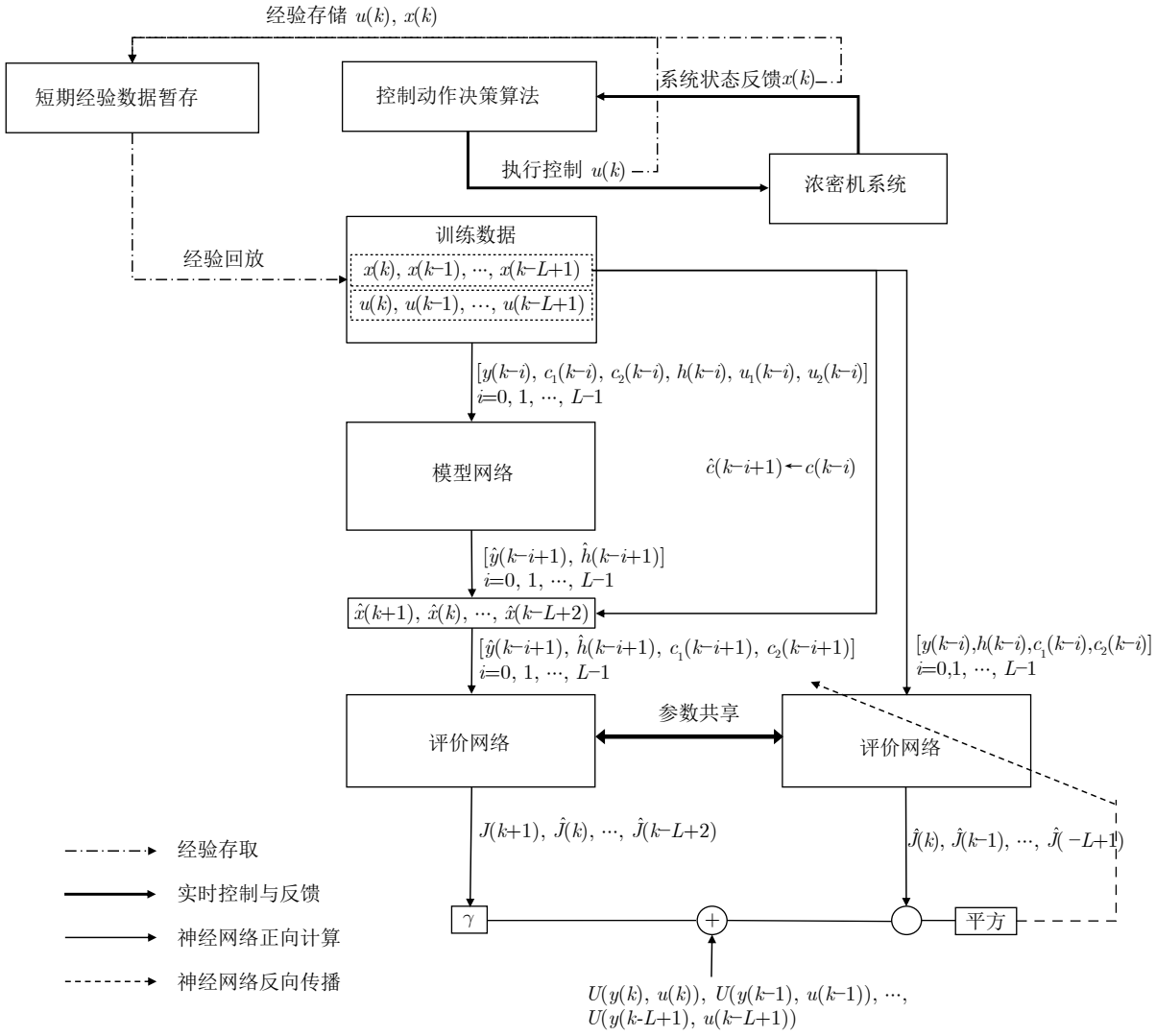


图 2　HCNVI 算法结构示意图

Fig. 2　Structure diagram of algorithm HCNVI

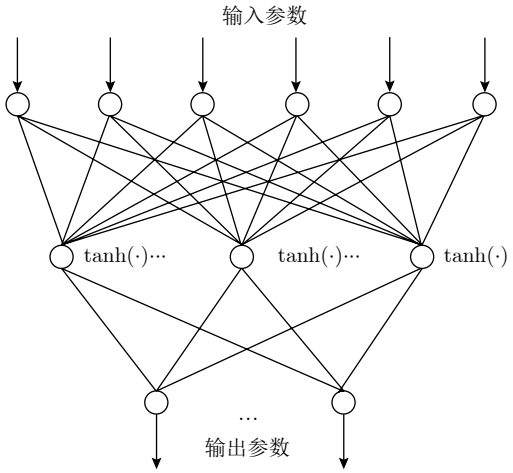


图 3 人工神经网络结构示意图

Fig. 3 Structure diagram of artificial neural network

$\tanh(x)=\dfrac{\mathrm{e}^{x}-\mathrm{e}^{-x}}{\mathrm{e}^{x}+\mathrm{e}^{-x}}$ 是网络的激活函数, 网络输入层包含 4 个节点, 隐层包含 14 个节点, 输出层 1 个节点, W_{c1} 和 W_{c2} 内参数均初始化为 $-1\sim1$ 之间的随机数. 该模型采用由浓密机控制过程中产生的在线数据进行网络训练. 为了保证算法更新的实时性, 本文采用单步时序差分误差 (Temporal difference error, TD error)[5] 计算评价网络估计误差值, 见式 (10).

$$e_c(k)=\hat{J}(k)-(\gamma\hat{J}(k+1)+U(k)) \tag{10}$$

网络损失函数为 $E_c(k)=e_c^2(k)$. 通过极小化该目标函数, 可以使评价网络根据被控系统反馈的状态信号及效用值信号, 增量式地逼近对于当前控制策略的评价函数. 使用链式法则可以计算损失值 $E_c(k)$ 对网络参数的梯度:

$$\begin{aligned}&\frac{\partial e_c^2(k)}{\partial W_{c2}}=2e_c(k)\tanh(W_{c1}\boldsymbol{x}(k))^{\mathrm{T}}\\&\frac{\partial e_c^2(k)}{\partial W_{c1}}=2e_c(k)[W_{c2}^{\mathrm{T}}\odot(1-\tanh^2(W_{c1}\boldsymbol{x}(k)))]\boldsymbol{x}(k)^{\mathrm{T}}\end{aligned} \tag{11}$$

采用梯度下降算法对评价网络进行训练更新:

$$W_{ci}(k)=W_{ci}(k)-l_c\frac{\partial e_c^2(k)}{\partial W_{ci}(k)} \tag{12}$$

l_c 是学习率, 由于浓密机所处环境的外界噪音是不断波动的, 当外界噪音 $\boldsymbol{c}(k)$ 改变时, 网络需要根据训练数据快速收敛, l_c 需设定为固定值以保持学习能力.

由于不同物理量的取值差异很大, 这会导致网络无法有效学习并且造成超参数设定困难. 因此本文采用浓密机系统产生的离线数据中各参量的极值对所有训练数据利用式 (13) 进行归一化放缩.

$$\overline{z}=\frac{2\,(z-z_{\min})}{z_{\max}-z_{\min}}-1 \tag{13}$$

模型网络. 建立模型网络用来对系统动态进行建模, 根据当前系统状态、外部噪音量、控制输入、预测下一时刻底流浓度和泥层高度变化. 网络结构仍采用单隐层神经网络, 如图 3 所示. 模型网络具体定义如下:

$$[\hat{y}(k+1),\hat{h}(k+1)]^{\mathrm{T}}=W_{m2}\tanh\left(W_{m1}(\boldsymbol{\phi}(k))\right) \tag{14}$$

其中, $\boldsymbol{\phi}(k)=[\boldsymbol{x}^{\mathrm{T}}(k),\boldsymbol{u}^{\mathrm{T}}(k)]^{\mathrm{T}}$, 网络输入层包含 6 个节点, 隐层包含 20 个节点, 输出层 2 个节点, W_{m1} 和 W_{m2} 内各个参数均初始化为 $-1\sim1$ 之间的随机数. 通过梯度下降方法训练模型网络:

$$W_{mi}(k)=W_{mi}(k)-l_m\frac{\partial E_m(k)}{\partial W_{mi}(k)} \tag{15}$$

损失函数 $E_m(k)$ 定义为:

$$E_m(k)=\frac{1}{2}\boldsymbol{e}_m^{\mathrm{T}}(k)\boldsymbol{L}_m\boldsymbol{e}_m(k) \tag{16}$$

$$\begin{aligned}e_m(k)=&[\hat{y}(k+1),\hat{h}(k+1)]^{\mathrm{T}}-\\&[y(k+1),h(k+1)]^{\mathrm{T}}\end{aligned} \tag{17}$$

对于模型网络, 同样采用式 (13) 对训练数据进行放缩. 模型网络的训练全部离线进行, 在控制任务开始后, 将不再对模型网络进行调整.

2.3 动作生成

大部分的 ADP 类算法都是通过建立一个动作网络来计算控制输入, 并利用评价网络输出值更新动作网络的参数. HCNVI 方法以 HDP 算法架构为基础, 去掉了动作网络, 直接利用评价网络和模型网络计算控制动作. 该方法可以在环境噪音改变时, 使被控系统更快速地收敛, 并且减少内存占用以及削减训练时间的消耗.

利用评价网络和模型网络计算控制动作 $\boldsymbol{u}(k)$ 的过程如算法 2 所示. 式 (19) 中在估计 $k+1$ 时刻的折扣累计惩罚时, 下一时刻浓密机系统所处外界噪音是未知的. 不过由于真实工业环境下进料噪音都是连续变化的, 很少出现突变, 因此本模型用当前时刻噪音 $\boldsymbol{c}(k)$ 来充当下一时刻噪音 $\boldsymbol{c}(k+1)$.

算法 2. 利用迭代梯度下降算法计算控制动作

输入: 第 k 时刻系统状态 $y(k),h(k),\boldsymbol{c}(k)$

输出: 第 k 时刻的控制动作输出 $\boldsymbol{u}(k)$

1: 随机选取 $\boldsymbol{u}_0=[v_1,v_2]^{\mathrm{T}}$

2: $v_1\sim U(-1,1),v_2\sim U(-1,1)$

3: $i=0$

4: **do**

5: 预测以 $\boldsymbol{u}_i$ 为控制输入情况下, 下一时刻

系统状态

$$[\hat{y}(k+1),\hat{h}(k+1)]=W_{m2}\tanh\left(W_{m1}\left(\boldsymbol{x}(k),\boldsymbol{u}_i\right)\right)\tag{18}$$

6: 令 $\hat{\boldsymbol{x}}(k+1)=[\hat{y}(k+1),\hat{h}(k+1),\boldsymbol{c}(k)^{\mathrm{T}}]^{\mathrm{T}}$, 估计 $k+1$ 时刻评价值

$$\hat{J}(k+1)=W_{c2}\tanh\left(W_{c1}(\hat{\boldsymbol{x}}(k+1))\right)\tag{19}$$

7: 计算第 k 时刻评价值

$$\hat{J}(k)=U\left(y_k,\boldsymbol{u}_i\right)+\gamma\hat{J}(k+1)\tag{20}$$

8: 利用梯度下降算法对 $\boldsymbol{u}_i$ 进行更新

$$\boldsymbol{u}_{i+1}=\boldsymbol{u}_i-l_u\frac{\partial\hat{J}(k)}{\partial\boldsymbol{u}_i}\tag{21}$$

9: 将 $\boldsymbol{u}_{i+1}$ 限定在 $\Omega_{\boldsymbol{u}}$ 的约束内

$$\boldsymbol{u}_{i+1}=\max([-1,-1]^{\mathrm{T}},\min([1,1]^{\mathrm{T}},\boldsymbol{u}_{i+1}))\tag{22}$$

10: $i=i+1$

11: **while** $\|\boldsymbol{u}_{i+1}-\boldsymbol{u}_i\|>\epsilon_a$ and $i<Na$

12: 反归一化 $\boldsymbol{u}(k)$

$$\boldsymbol{u}(k)=\frac{\boldsymbol{u}(i+1)\odot(\boldsymbol{u}_{\max}-\boldsymbol{u}_{\min})}{2}+\boldsymbol{u}_{\mathrm{mid}}\tag{23}$$

13: return $\boldsymbol{u}(k)$

为了验证算法 2 的有效性, 本文对式 (20) 中 $\hat{J}(k)$ 与 $\boldsymbol{u}(k)$ 的关系及迭代求解 $\boldsymbol{u}_i(k)$ 的过程进行了可视化探究. 在第 3.1 节实验 1 介绍的仿真实验中挑选了三个时刻分析了 $\hat{J}(k)$ 与 $\boldsymbol{u}(k)$ 之间的函数关系. 图 4 中的三个子图分别代表训练开始阶段、第一次系统达到稳态时、第二次系统达到稳态时的可视化结果. 横纵坐标代表被归一化后的底流泵速和絮凝剂泵速, 颜色深浅代表 $\hat{J}(k)$ 的大小. 黄色箭头线代表利用算法 2 寻找最优控制输入 $\boldsymbol{u}(k)$ 的梯度下降轨迹. 根据实验结果发现: 在网络训练的三个阶段中, 图中颜色最深的点, 即 $\hat{J}(k)$ 的最小位置是唯一的, 且不存在其他局部最优解. 黄色箭头线能够准确地收敛至全局最优解. 该结果说明由于浓密机运行过程缓慢, 某一时刻的控制输入 $\boldsymbol{u}(k)$ 对下一时刻浓密机状态 $\boldsymbol{x}(k+1)$ 影响相对较小, 且评价网络式 (9) 和效用函数式 (3) 具有连续、可微的性质, 因此 $\hat{J}(k)$ 随 $\boldsymbol{u}(k)$ 变化的分布函数一般情况下为单峰函数. 采用梯度下降算法可以有效地寻找到全局最优的 $\boldsymbol{u}^*(k)$, 而不会收敛到局部最优解, 进而满足式 (7) 的最小化条件,实现最优控制.

2.4　短期经验回放

为了增加评价网络训练的准确性和收敛速度, 本文进一步提出短期经验回放方法优化网络训练损失函数, 并计算优化梯度. 短期经验回放方法将式 (10) 的误差值计算方法修改为

$$e_c(k)=\frac{1}{L}\sum_{i=0}^{L-1}\hat{J}(\boldsymbol{x}(k-i))-(U(k-i)+\gamma\hat{J}(\boldsymbol{x}(k-i+1)))\tag{24}$$

通过存储短期内被控系统的运行轨迹数据, 在训练过程中, 短期轨迹数据可以用来共同计算评价网络的损失值以及优化梯度方向.

HDP、DHP 以及本文提出的 HCNVI 算法都是面向状态值函数进行建模的在线控制算法, 其策略模块的更新都是以模型网络作为媒介, 计算评价网络输出值 $\hat{J}(k)$ 对于控制输入 $\boldsymbol{u}(k)$ 的梯度, 并在此梯度基础上更新动作网络或者利用算法 2 优化 $\boldsymbol{u}(k)$. 因此对于 $\boldsymbol{u}(k)$ 梯度估计的准确性极大地影响了策略模块的更新效果, 进而影响整个控制系统的控制效果与收敛速度. $\boldsymbol{u}(k)$ 的梯度表达式为式 (25)

$$\nabla\boldsymbol{u}(k)=\gamma\frac{\partial\boldsymbol{x}(k+1)}{\partial\boldsymbol{u}(k)}\frac{\partial\hat{J}(k+1)}{\partial\boldsymbol{x}(k+1)}+\frac{\partial U(k)}{\partial\boldsymbol{u}(k)}\tag{25}$$

式中, $\dfrac{\partial\hat{J}(k+1)}{\partial\boldsymbol{x}(k+1)}$ 也称为 $(k+1)$ 时刻的协状态 $\boldsymbol{\lambda}(k+1)$, 代表了评价网络输出值对于系统状态量的梯度. 模型网络可以利用系统离线数据进行训练, 在训练数据量充足时可以达到极高的精度, 可以近似认为 $\dfrac{\partial\boldsymbol{x}(k+1)}{\partial\boldsymbol{u}(k)}$ 的估计是足够精确的. $\boldsymbol{U}(k)$ 作为确定的效用函数, $\dfrac{\partial U(k)}{\partial\boldsymbol{u}(k)}$ 也是确定的. 因此对于 $\nabla\boldsymbol{u}(k)$ 的

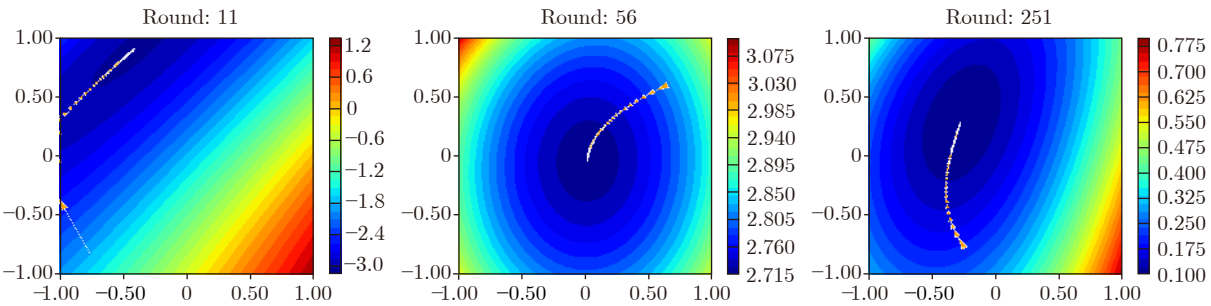


图 4　迭代梯度下降过程可视化

Fig. 4　Visualize the process of iterative gradient decline

估计误差主要来源于对协状态 $\boldsymbol{\lambda}(k+1)$ 的估计误差.

对于浓密机等大型过程工业设备来说, 系统的运行过程缓慢, 短时间内系统状态不会发生剧烈改变, 即 $\boldsymbol{x}(k) \approx \boldsymbol{x}(k+1)$, 且评价网络具有连续可微的性质. 因此可以近似认为 $\boldsymbol{\lambda}(k) \approx \boldsymbol{\lambda}(k+1)$. 同样, 由于系统的运行过程缓慢会导致提供给控制模型学习的训练数据中系统状态参量分布非常集中, 可以近似认为式 (26) 成立.

$$\|\boldsymbol{x}(k-t) - \boldsymbol{x}(t)\| < \delta, \quad \forall 1 \le t < L \tag{26}$$

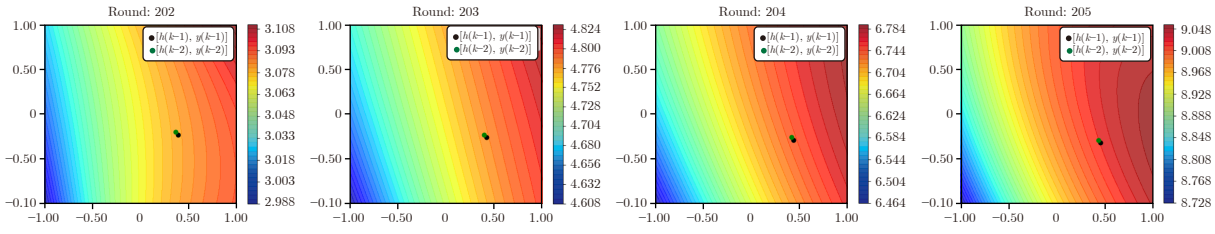
该式表明短期内系统状态点 $\boldsymbol{x}(k-t)$ 都在以 $\boldsymbol{x}(k)$ 为中心, δ 为半径的领域内. 通过式 (24) 将短期 L 条数据共同用于评价网络训练, 可以使评价网络在 $\boldsymbol{x}(k)$ 的邻域内学习地更佳充分, 进而更准确地估计 $\boldsymbol{\lambda}(k)$.

为了更直观地展示增加短期经验回放对评价网络学习过程的影响, 本文对第 3.1 节实验 1 中的评价网络进行了可视化, 实验结果如图 5 所示. 该实验中采用等高线图对评价网络的输出值进行展示, 其中图 5(a) 代表不使用经验回放, 利用式 (10) 训练网络, 图 5(b) 代表使用短期经验回放, 回放数据点数 L 为 2, 利用式 (24) 训练网络. 对于两种算法, 分别绘制了连续四次迭代中, 评价网络在更新后对不同泥层高度 $h(\cdot)$ 和底流浓度 $y(\cdot)$ 的评价值. 图中横纵坐标分别代表被归一化后的泥层高度和底流浓度. 根据实验结果发现. 在图 5(a) 中评价网络的输出值在不同输入下基本趋同. 且在当前时刻系统状态点附近, 网络输出值的梯度很小. 说明单数据点更新会造成评价网络很快地遗忘历史数据, 导致网络输出值整体漂移, 难以稳定地学习到正确的局部梯度. 在图 5(b) 中, 当前系统状态 $(h(k), y(k))$ 所处临域内, 网络输出值具有较大差异, 局部梯度值可以被较好地保持. 准确的梯度 $\boldsymbol{\lambda}(k)$ 可以提高 $\nabla \boldsymbol{u}(k)$ 估计的精确度, 因此对短期数据进行回放训练可以更好地指导控制策略输出更优控制动作, 促使评价网络和被控系统快速收敛. 同时, 当经验回放数据量式 (24) 中 L 的过大, 会导致性能的退化. 其原因在于本文提出的方法是同策略 (On-policy)强化学习方法, 而时间相差较远的历史数据点不能表征由当前控制策略产生的控制轨迹, 因此评价网络会学习到错误的评价值. 另外, L 过大将不再满足性质式 (26), 过多的历史数据回放将不再有助于评价网络学习 $\boldsymbol{x}(k)$ 处的梯度值 $\boldsymbol{\lambda}(k)$, 进而不会提高对 $\nabla \boldsymbol{u}(k)$ 估计的精确度. 通过实验观察, 一般将 L 限定在 5 以内, 本文也将这种经验回放方法称为短期经验回放.

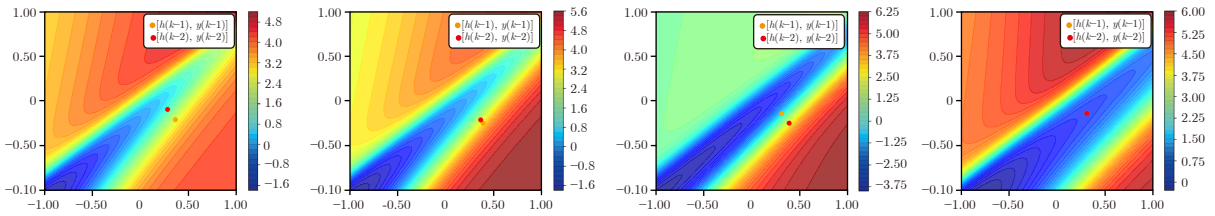
将 HCNVI 算法用于浓密机控制的具体流程如算法 3 所示.

算法 3. 利用 HCNVI 算法实现浓密机在线控制

1: 使用浓密机运行离线数据, 用式 (15) 训练模型网络
2: $k = 0$
3: **while** $k < T$ **do**
4: 根据浓密机系统获得 $y(k), h(k), \boldsymbol{c}(k)$
5: **if** $k \ge 1$ **then**
6: $i = 0$



(a) 无经验回放
(a) No experience replay



(b) 经验回放点数量为2
(b) Experience replay with capacity equal to 2

图 5 短期经验回放对评价网络的输出值的影响

Fig. 5 The effect of short-term experience replay on critic network

7: **do**
8: 令 $L=\min(L_c,k)$, 用式 (24) 求解 $e_c(k)$
9: 利用式 (12) 训练评价网络
10: $i=i+1$
11: **while** $i<N_c$ and $e_c^2(k)>\epsilon_c$
12: 利用算法 2 求解 $\boldsymbol{u}(k)$
13: 将 $\boldsymbol{u}(k)$ 作用于浓密机系统, 并等待 T_d 分钟.
14: $k=k+1$

3　浓密机仿真实验

浓密机仿真模型. 由于在真实工业场景下进行浓密机控制实验成本较高, 本节采用浓密机仿真模型验证本文提出控制算法的有效性, 模型构建方法参考了文献 [19−24]. 该仿真模型建立在如下假设基础上:

1) 进料都是球形颗粒.
2) 絮凝剂在浓密机的静态混合器中作用完全.
3) 流体的扩散以固液混合物形式进行.
4) 忽略颗粒间相互作用、浓密机中把机中轴的影响.

模型推导过程中出现的变量如表 1 ~ 表 3所示

由文献 [23],可得泥层高度与泥层液固质量比之间的关系.

$$h(t)=\frac{W(t)\theta}{A\rho_s}+\frac{W(t)\theta}{A}r(t) \tag{27}$$

根据固体守恒定律, 泥层内固体质量变化量等于由进料导致泥层内固体量增加量与底流导致泥层内固体减少量的差. 因此可以建立泥层内平均单位体积含固量与粒子沉降速度的关系.

$$\frac{\mathrm{d}\left[c_a(t)Ah(t)\right]}{\mathrm{d}t}=c_l(t)\left[u_t(t)+u_r(t)\right]A-c_u(t)u_r(t)A \tag{28}$$

对式 (28) 做变形可得式 (29):

$$c_a(t)\frac{\mathrm{d}h(t)}{\mathrm{d}t}+h(t)p\frac{\mathrm{d}c_u(t)}{\mathrm{d}t}=c_l(t)\left[u_t(t)+u_r(t)\right]A-c_u(t)u_r(t)A \tag{29}$$

联立式 (29), 式 (27), 可得泥层高度 $h(t)$ 与底流浓度 $c_u(t)$ 的一阶变化率

$$\frac{\mathrm{d}h(t)}{\mathrm{d}t}=-\frac{W(t)\theta}{Ac_a^2(t)}\frac{c_l(t)\left[u_t(t)+u_r(t)\right]-c_u(t)u_r(t)}{h(t)-c_a(t)\dfrac{W(t)\theta}{Ac_a^2(t)}} \tag{30}$$

$$\frac{\mathrm{d}c_u(t)}{\mathrm{d}t}=\frac{c_l(t)\left[u_t(t)+u_r(t)\right]-c_u(t)u_r(t)}{p\left(h(t)-c_a(t)\dfrac{W(t)\theta}{Ac_a^2(t)}\right)} \tag{31}$$

表 1　参量定义
Table 1　Variables definition

变量	含义	量纲	初始值	补充说明
$f_i(t)$	进料泵频	Hz	40	扰动量
$f_u(t)$	底流泵频	Hz	85	控制量
$f_f(t)$	絮凝剂泵频	Hz	40	控制量
$c_i(t)$	进料浓度	kg/m^3	73	扰动量
$h(t)$	泥层高度	m	1.48	状态量
$c_u(t)$	底流浓度	kg/m^3	680	目标量

表 2　仿真模型常量
Table 2　Definitions for constant variables

变量	含义	量纲	参考值
ρ_s	干砂密度	kg/m^3	4150
ρ_e	介质表观密度	kg/m^3	1803
μ_e	悬浮体系的表观粘度	Pa · s	1
d_0	进料颗粒直径	m	0.00008
p	平均浓度系数	无	0.5
A	浓密机横截面积	m^2	300.5
k_s	絮凝剂作用系数	s/m^2	0.157
k_i	压缩层浓度系数	m^3/s	0.0005×3600
K_i	进料流量与进料泵频的系数	m^3/r	50/3600
K_u	底流流量与底流泵频的系数	m^3/r	2/3600
K_f	絮凝剂流量与絮凝剂泵频的系数	m^3/r	0.75/3600
θ	压缩时间	s	2300

表 3　部分变量计算方法
Table 3　Definitions for part intermediate variables

变量	含义	公式
$q_i(t)$	进料流量	$q_i(t)=K_if_i(t)$
$q_u(t)$	底流流量	$q_u(t)=K_uf_u(t)$
$q_f(t)$	絮凝剂添加量	$q_f(t)=K_ff_f(t)$
$d(t)$	絮凝作用后的颗粒直径	$d(t)=k_sq_f(t)+d_0$
$u_t(t)$	颗粒的干涉沉降速度	$u_t(t)=\dfrac{d^2(t)\left(\rho_s-\rho_e\right)g}{18\mu_e}$
$u_r(t)$	底流导致的颗粒下沉速度	$u_r(t)=\dfrac{q_u(t)}{A}$
$c_l(t)$	泥层高度处单位体积含固量	$c_l(t)=k_iq_i(t)c_i(t)$
$c_a(t)$	泥层界面内单位体积含固量	$c_a(t)=p\left[c_l(t)+c_u(t)\right]$
$r(t)$	泥层内液固质量比	$r(t)=\rho_l\left(\dfrac{1}{c_a(t)}-\frac{1}{\rho_s}\right)$
$W(t)$	单位时间进入浓密机内的固体质量	$W(t)=c_i(t)q_i(t)$

在该仿真模型中, 絮凝剂泵速 f_f 和底流泵速 f_u 是控制输入 $\boldsymbol{u}=[f_u,f_f]^{\mathrm{T}}$,进料泵速 f_i 和进料浓度 c_i 是外部干扰量 $\boldsymbol{c}=[f_i,c_i]^{\mathrm{T}}$, 底流浓度 c_u 为控制系统追踪变量 $y=c_u$. 理想的控制系统能够在外界干扰量 c 不断波动下, 通过在合理范围内调节 u, 驱使 y 追踪其设定值 y^*. 根据真实生产情况对部分变量做

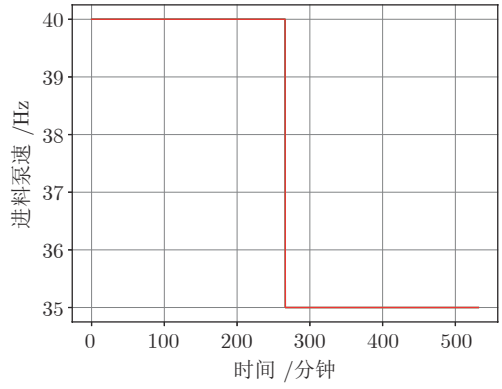
如下定义: $\boldsymbol{u}_{\min}=[40,30]^{\mathrm{T}}$, $\boldsymbol{u}_{\max}=[120,50]^{\mathrm{T}}$, $y_{\min}=280$, $y_{\max}=1200$, $\boldsymbol{c}_{\min}=[40,30]^{\mathrm{T}}$, $\boldsymbol{c}_{\max}=[120,50]^{\mathrm{T}}$, $y^*=680$. 接下来本节将基于浓密机仿真模型式(30)、式(31), 分别进行两组实验验证在两种类型噪音量 $\boldsymbol{c}(k)$ 输入下 HCNVI 模型的控制效果, 并与其他算法进行比较.

3.1 实验 1: 恒定 − 阶跃型噪音输入下浓密机控制仿真实验

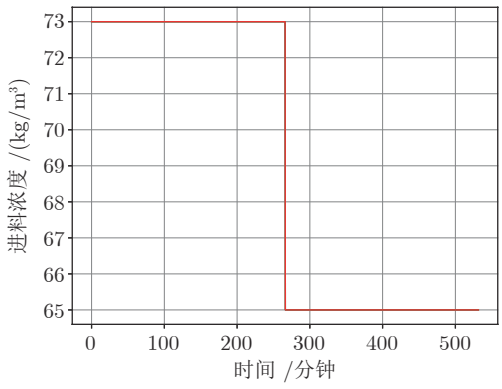
第一组实验中设置干扰量输入 $\boldsymbol{c}$ 为恒定值, 并在某一时刻为其增加阶跃突变, 噪音输入量如图 6 所示. 该实验用来验证控制模型能否在浓密机外在环境发生大幅度变化下, 快速寻找到 $\boldsymbol{u}^*$, 使被控模型达到理想收敛稳态.

使用本文提出的 HCNVI 算法与 HDP、DHP、ILPL 算法进行对比实验. 仿真实验参数如下: 迭代轮次 $T=270$, 仿真步长 $T_d=120$ s, $Q=0.004$, $\gamma=0.6$, $N_a=4\,000$, $N_c=500$, $\epsilon_c=0.001$, $\epsilon_a=0.0001$, $l_m=0.01$, $l_c=0.01$, $l_a=0.009$, $l_u=0.4$, $L_c=2$, $\boldsymbol{L_m}=[0.01,3]$. 其中 HDP、DHP 算法也使用短期经验回放, 回放点数 L 为 2. 实验中 HDP、ILPL、HCNVI 的评价网络结构相同, 且网络参数初始化为相同数值. 实验结果如图 7 所示.

根据实验结果可以发现, 对于不同控制算法, 由于网络参数初始值均为随机设定值, 训练初期底流浓度有较大幅度的波动, 且在设定值两侧持续震荡. 随着各个控制模型的学习, 系统状态与网络参数不断趋于平稳, 直到某一时刻底流浓度开始稳定并与设定值重合且不再产生波动, 此时控制模型参数也不再发生变化, 被控系统和控制模型同时收敛到最优态. 从效用值变化曲线也可以看出, 早期由于底流浓度与其设定值偏差较大, 效用值较高. 但是随着模型与系统趋于稳态, 效用值 $\boldsymbol{u}(k)$ 不断缩减直到接近于 0 的位置. 到达 270 分钟时, 系统进料浓度、进料流量发生突变, 底流浓度无法维持稳态, 开始远离设定值. 控制模型根据噪音量改变后的系统所产生的轨迹数据重新训练, 将底流浓度拉回设定值位置. 由于在第一阶段控制模型已经到达过一次稳态, 在第二阶段仅需要少量迭代就可以使系统重归理想收敛稳态. 通过观察不同控制算法产生的

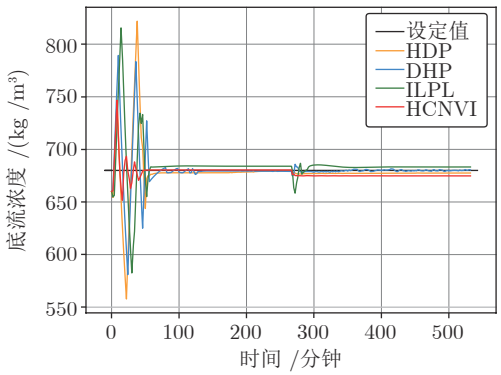


(a) 进料泵速变化
(a) Speed of feed pump changes suddenly

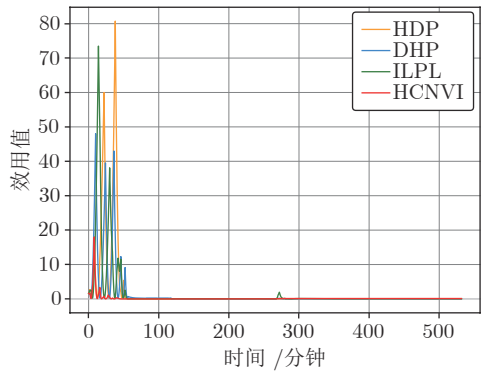


(b) 进料浓度变化
(b) Concentration of feed changes suddenly

图 6 噪音量变化曲线
Fig. 6 Noise input in the simulation experiment



(a) 度流浓度变化
(a) Concentration of underflow



(b) 效用函数值变化
(b) Utility

图 7 HCNVI 与其他 ADP 算法在恒定噪音输入下的对比
Fig. 7 HCNVI versu other ADP algorithms under stable noisy input

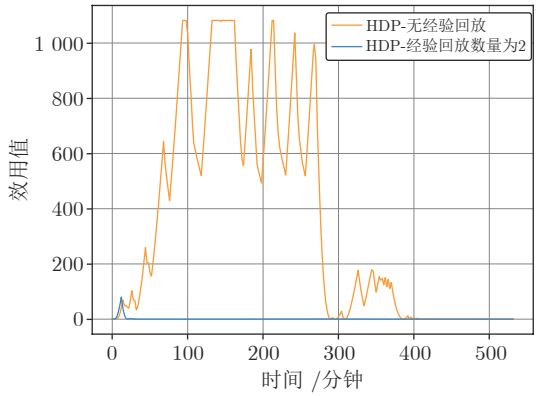
系统轨迹, 可以发现不同控制算法到达最优态所需的时间有较大差别, 且在收敛到最优态的过程中, 底流浓度的波动也有较大差异. 在实验第一阶段, 为使系统达到稳态, HCNVI 算法所需要的迭代次数更少, 训练过程中产生的底流浓度振幅也更小. 并且在噪音量改变后, HCNVI 算法可以迅速地使模型重归最优态, 且底流浓度几乎未发生大幅度波动.

HCNVI 的快速收敛能力主要来源于其采用迭代算法 2 得出的 $\boldsymbol{u}(k)$ 严格满足式 (7) 的最小化条件, 可以使评价网络更快地收敛到最优评价值函数. 而其他 ADP 算法中引入了动作网络, 这会使策略的更新存在一定的滞后性, 进而拖慢评价网络的训练速度.

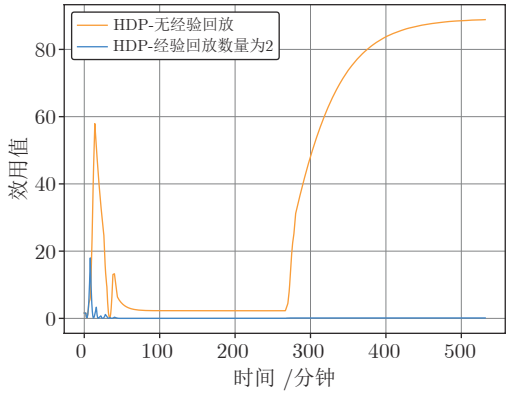
为了验证短期经验回放技术对控制算法性能的影响, 本文分别对比了无经验回放、使用短期经验回放 ($L = 2$) 情况下 HDP、HCNVI 的控制性能. 对比结果如图 8 所示. 在本实验中, 仅比较了两种算法的效用值变化, 效用值越快地收敛到 0 说明算法控制效果越佳. 通过观察图 8(a) 和图 8(b) 中无经验回放情况下的效用值变化曲线, 可以发现曲线波动较大. 相比于使用短期经验回放, 无经验回放情况下控制模型需要更多的迭代轮次才能够使系统达到收敛. 特别是在图 7(a) 的 HCNVI 的实验中, 270 分钟时系统噪音输入量改变, 效用值开始剧增, 底流浓度开始偏离设定值, 评价网络的学习结果如图 5(a) 中的第 4 部分所示. 评价网络对当前状态点 $\boldsymbol{x}(k)$ 的局部梯度估计有较大偏差, 使得利用算法 2 求解的 $\boldsymbol{u}(k)$ 并没有驱使底流浓度向其设定值移动, 被控系统无法收敛. 但在增加了短期经验数据回放后, 无论是本文提出的 HCNVI 算法还是 HDP 算法, 效用函数值可以快速收敛至最低点, 有效实现对被控系统的控制. 该实验结果表明短期经验回放技术对于控制模型的收敛速度改善效果明显, 且对不同 ADP 算法具有通用型.

另外本文进行了十组实验来对比 HCNVI 算法在时间上的优势. 选取 HDP 算法作为参考对象, $T = 270$, 结果如图 9 所示. 由于每次实验中网络初始值不同, 系统运行轨迹以及模型训练过程也不同, 因此每组实验中模型学习以及控制所需的累积时间略有差异. 但是从多次实验结果可以看出, 由于 HCNVI 算法中去掉了动作网络, 仅需要训练评价网络, 所以模型整体训练时间大大缩减, 尽管算法 2 中计算控制输入所需时间相比于 HDP 算法直接利用动作网络前向传播求解控制动作所需时间长, 但是 HCNVI 算法总消耗时间明显少于 HDP 算法.

前人研究表明[25−26], 在启发式动态规划类算法中, 去掉动作网络可以有效减少模型训练时间. 但是在某些复杂系统控制问题中, 去除动作网络会使模型难以拟合复杂策略函数, 最终导致控制效果变差. 在本文的实验中, 由于浓密机系统运行缓慢且



(a) 在HDP算法中引入经验回放对效用值的影响
(a) Short-term experience replay has great influence to minmize the utility in HDP



(b) 在HCVI算法中引入经验回放对效用值的影响
(b) Short-term experience replay has great influence to minmize the utility in HCNVI

图 8　短期经验回放对 HDP 与 HCNVI 的影响

Fig. 8　The influence of short-term experience replay on HDP and HCNVI

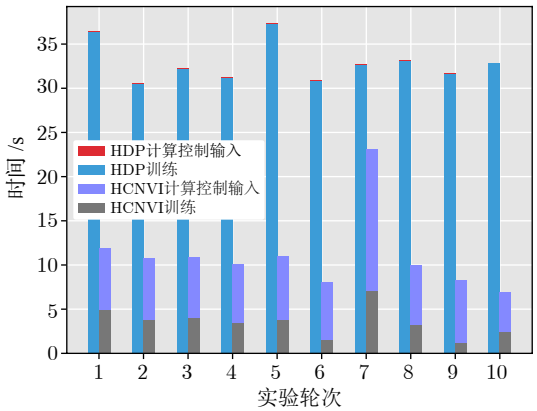


图 9　实验一中 HDP 与 HCNVI 在时间消耗上的对比

Fig. 9　Comparison of time consuming in HDP and HCNVI in Experiment 1

具有较高时滞性, 当前时刻控制输入量 $\boldsymbol{u}(k)$ 对 $\hat{\boldsymbol{x}}(k+1)$ 的影响较小, 即对 $\hat{J}(k)$ 的影响较小. 因此利用算法 2 求解的 $\boldsymbol{u}(k)$ 满足式 (7) 的最小化条件. 而在 HDP、DHP、ILPL 等方法中采用神经网络拟合出的控制策略, 难以输出严格满足式 (7) 的 $\boldsymbol{u}(k)$, 算法 2 的最优性代表 HCNVI 可以最大程度地利用评价网络给出的协状态信息优化当前控制策略, 进而获得更高的控制效果. 但 HCNVI 方法也具有一定的局限性, 当被控系统状态变化速率较快, $\hat{J}(k)$ 随 $\boldsymbol{u}(k)$ 变化的分布函数不再是单峰函数, 算法 2 求解出的 $\boldsymbol{u}(k)$ 极容易陷入到局部最优解, 算法控制效果及收敛速度必然变差. 而此时在 HDP、DHP、ILPL 等方法中采用神经网络拟合的控制策略往往能够给出相对更优、鲁棒性更强的控制动作 $\boldsymbol{u}(k)$, 其控制效果与收敛速率必然优于 HCNVI 算法.

3.2 实验 2: 高斯噪音波动输入下浓密机控制仿真实验

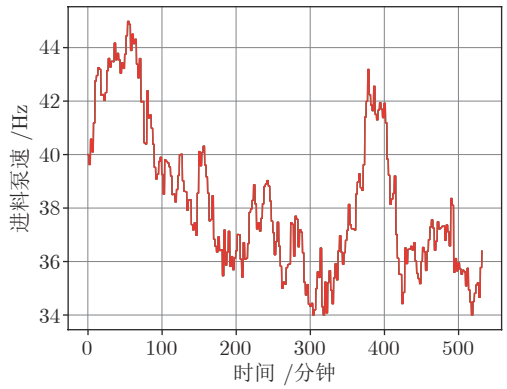
实验 1 中仿真模型的进料状态是恒定的, 只在某一时刻产生突变, 其目的是为了更好地观察不同控制算法的收敛速度. 而真实工业场景下, 浓密机的进料浓度和进料流量是实时波动的. 在本节实验中, 进料流量和进料浓度两个噪音量持续波动, 用来模仿真实工业场景下的浓密机系统环境. 噪音输入的单步变化增量服从高斯分布, 进料波动变化如图 10 所示.

$$\begin{aligned} &c(k+1) = c(k) + \Delta c \\ &\Delta c \sim \mathrm{N}(\mu = 0, \Sigma = \mathrm{diag}\{0.6, 0.6\}) \end{aligned} \tag{32}$$

本实验中 HCNVI 控制器参数与第 3.1 节实验 1 中的算法参数相同, 迭代轮次 $T = 270$, 仿真步长 $T_d = 120$ s. 利用该仿真模型再次对比 HCNVI 与其他算法控制性能的差异, 结果如图 11 所示.

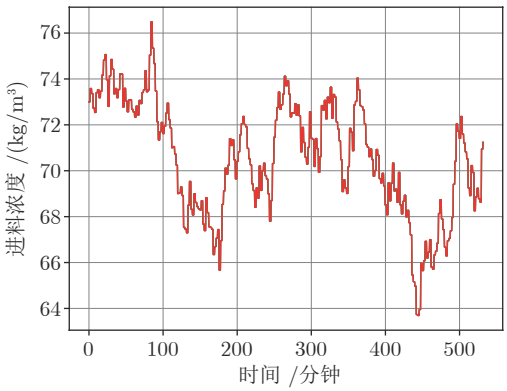
通过观察实验结果发现在环境噪音连续变化条件下, 浓密机底流浓度会发生持续震荡. 随着对模型参数的不断训练, 各个算法的控制性能趋于平稳, 由于进料噪音导致的底流浓度波动稍有减弱. 对比不同控制算法的控制性能, 可以发现 HCNVI 相比于其他 ADP 算法能够更快地将底流浓度锁定在设定值临域范围内, 且浓度振幅小于其他算法. 从效用值变化曲线也可以看出, 相比于其他算法, HCNVI 算法的效用值整体较小, 且在训练后期几乎 0.

该实验结果与第 3.1 节实验 1 中进料噪音突变条件下的实验结果相吻合. HCNVI 算法在外界噪音频繁改变时, 可以更快地响应外部变化, 快速调节评价网络参数, 将底流浓度稳定在目标值附近. 其他算法由于增加了动作网络产生了训练滞后性, 进而导致无法快速适应外部环境的变化, 使其控制性能差于 HCNVI.



(a) 进料泵速变化

(a) The speed of feed pump varies in real time



(b) 进料浓度变化

(b) The feed concentration varies in real time

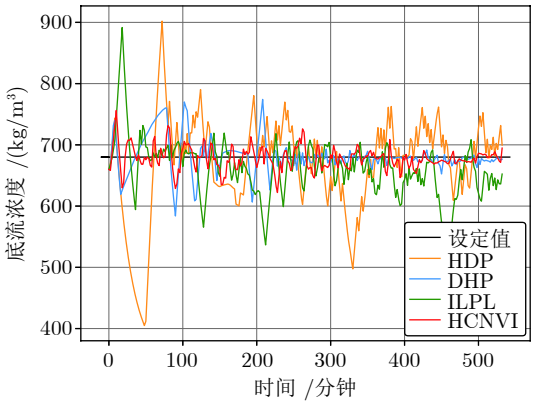
图 10 噪音量变化曲线

Fig. 10 The fluctuation of noisy input

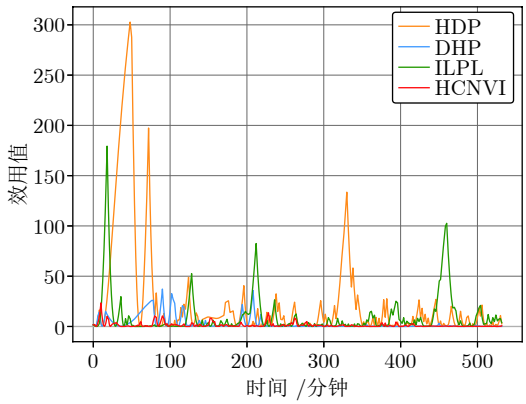
表 4 给出了不同算法在第 3.1 节实验 1 和第 3.2 节实验 2 中底流浓度控制性能指标对比结果. 相比其他算法, HCNVI 算法可以更好地控制底流浓度稳定在其设定值附近, 其控制总体稳定性 (由 MSE、IAE 体现)、控制鲁棒性 (由 MAE 体现) 更佳. 在过程工业控制场景中, 控制系统的 MAE 指标尤为重要, 某一工序的物料性质发生剧烈波动会使下游物料加工工序出现连带波动, 严重影响生产的稳定性和最终产品的质量. HCNVI 算法在 MAE 指标上的优势证实了其在过程工业控制问题中的适用性.

图 12 展示在环境噪音持续变化条件下, 不使用经验回放和使用短期经验回放 ($L = 2$) 两种情况下 HCNVI 算法控制性能. 在无经验回放情况下, 底流浓度稳定性明显较差, 且效用值明显较高, 使用短期经验回放 ($L = 2$) 后模型控制效果较好. 实验结果表明, 短期经验回放技术在环境噪音持续变化下仍对模型控制效果与收敛速度有重要促进作用.

为了展现在噪音持续变化条件下, HCNVI 算



(a) 底流浓度变化
(a) Concentration of underflow



(b) 效用函数值变化
(b) Utility

图 11　HCNVI 与其他 ADP 算法在波动噪声输入下的对比

Fig. 11　HCNVI versu other ADP algorithms under fluctuate noisy input

表 4　不同控制算法之间性能分析

Table 4　Performances analysis of different algorithms

实验组	实验1			实验2		
对比指标	MSE[1]	MAE[2]	IAE[3]	MSE	MAE	IAE
HDP	414.182	141.854	7.246	6 105.619	275.075	54.952
DHP	290.886	109.312	5.392	732.814	96.145	16.560
ILPL	364.397	135.474	8.289	2 473.661	211.615	35.222
HCNVI	**44.445**	**66.604**	**3.867**	**307.618**	**76.176**	**12.998**

法在时间上的优势, 再次重复了 10 次实验对比了 HCNVI 算法与 HDP 算法的时间消耗, $T = 270$. 实验结果如图 13 所示. 在噪音持续变化环境下, HCNVI 算法和 HDP 算法的总时间消耗相比于图 9 中的结果均有增加. 这是由于当外部环境存在持续扰动时, 被控系统和控制模型参数不再如第 3.1 节实验 1 中达到稳定态, 而是始终处于震荡状态, 被控系统轨迹数据不断变化. 每轮学习过程中, 为了满足评价网络的精度 $e_c(k)^2 < \epsilon_c$ 所需要的训练迭代次数增加, 进而导致评价网络训练所需时间及模型总体训练时间增加. 但通过横向对比 HCNVI 算法与 HDP 算法的总时间消耗, HCNVI 算法在训练和执行控制过程中所需的总时间消耗仍明显少于 HDP, 说明利用算法 2 替代动作网络所产生的时间消耗削减在噪音连续波动条件仍十分明显.

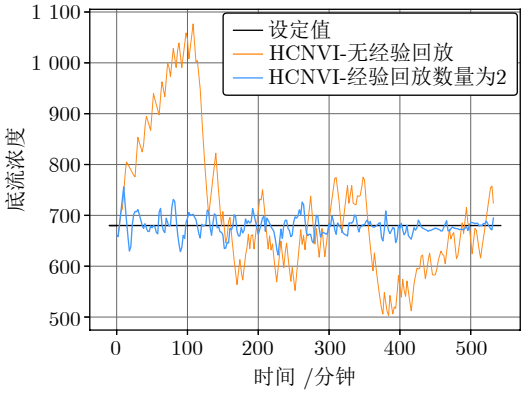
[1](Mean Square Error, MSE)= $\frac{1}{T}\sum_{k=1}^{T}|(y(k)-y^*(k))|^2$

[2](Max Absolute Error, MAE)= $\max_{1\le k\le T}\{|y(k)-y^*(k)|\}$

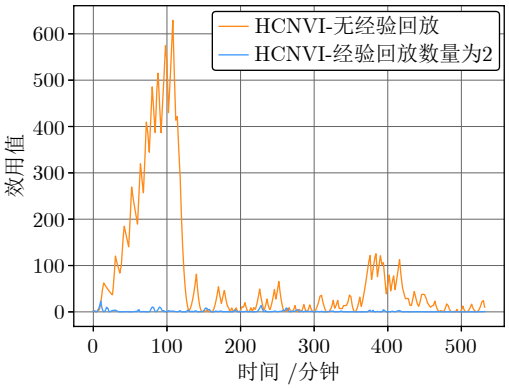
[3](Integral Absolute Error, IAE)= $\frac{1}{T}\sum_{k=1}^{T}|(y(k)-y^*(k))|$

4　结论

本文提出了基于强化学习的自适应控制算法 HCNVI, 该算法通过构建用于识别系统动态方程的模型网络以及用于估计折扣累计代价的评价网络来解决浓密机控制问题. 该方法可以在对浓密机系统未知的情况下, 仅利用浓密机系统输出数据以及历史运行数据即可实现在线学习并获得较好的控制效



(a) 在HCNVI算法中引入经验回放对底流浓度的影响
(a) The influence of short-term experience replay on underflow concentration for Hcnvi



(b) 在HCNVI算法中引入经验回放对效用值的影响
(b) The influence of short-term experience replay on utility for HCNVI

图 12　噪音持续变化下短期经验回放对 HCNVI 的影响

Fig. 12　The influence of short-term experience replay on HCNVI

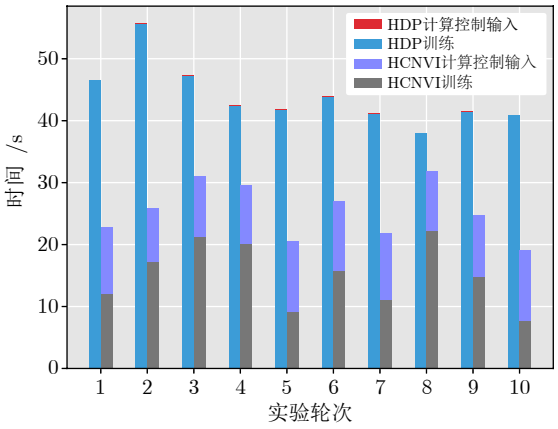


图 13 实验二中 HCNVI 算法与 HDP 算法在时间消耗上的对比

Fig. 13 Comparison of time consuming in HDP and HCNVI in Experiment 2

果. 另外本文提出的短期经验回放技术可以很好地增强评价网络训练的稳定性, 在其他自适应动态规划算法中也具有较好通用性. 根据仿真实验验证结果可以发现, 相比其他在线 ADP 算法, 由于 HCNVI 算法模型结构简单, 且具有较高的学习敏捷性, 因此在浓密机仿真系统控制问题中, HCNVI 算法消耗了更少的训练时间但获得了更优的控制效果. 但是 HCNVI 算法也存在自身的局限性, 其去掉动作网络的可行性是建立浓密机具有运行缓慢、稳定的特性基础之上的. 但是当被控系统相对复杂且不再具有此特性时, 如系统状态量变化过程并不连续或系统运行速度较快, HCNVI 依靠迭代算法求解的控制量难以保持最优性, 控制性能极有可能产生退化. 如何使 HCNVI 算法以及其他无动作网络类自适应动态规划类算法适用于此类复杂被控系统, 在优化训练时间消耗的同时保证其控制性能与收敛速度, 将是未来非常有意义的研究方向.

References

1 Shen Y, Hao L, Ding S X. Real-time implementation of fault tolerant control systems with performance optimization. *IEEE Trans. Ind. Electron*, 2014, **61**(5): 2402−2411

2 Kouro S, Cortes P, Vargas R, Ammann U, Rodriguez J. Model predictive control — A simple and powerful method to control power converters. *IEEE Trans. Ind. Electron*, 2009, **56**(6): 1826−1838

3 Dai W, Chai T, Yang S X. Data-driven optimization control for safety operation of hematite grinding process. *IEEE Trans. Ind. Electron*, 2015, **62**(5): 2930−2941

4 Wang D, Liu D, Zhang Q, Zhao D. Data-based adaptive critic designs for nonlinear robust optimal control with uncertain dynamics. *IEEE Trans. Syst., Man, Cybern., Syst.*, 2016, **46**(11): 1544−1555

5 Sutton S R, Barto G A. *Reinforcement Learning: An Introduction*. Cambridge: MIT Press, 2nd edition, 2018.

6 Lewis F L, Vrabie D, Syrmos V L. *Optimal Control*. New York, USA: John Wiley & Sons, Hoboken, 3rd Edition, 2012.

7 Prokhorov V D, Wunsch C D. Adaptive critic design. *IEEE Transactions on Neural Networks*, 1997, **8**(5): 997−1007

8 Werbos P J. Foreword - ADP: the key direction for future research in intelligent control and understanding brain intelligence. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics).*, 2008, **38**(4): 898−900

9 Duan Yan-Jie, Lv Yi-Sheng, Zhang Jie, Zhao Xue-Liang, Wang Fei-Yue. Deep learning for control: the state of the art and prospects. *Acta Automatica Sinica*, 2016, **42**(5): 643−654
(段艳杰, 吕宜生, 张杰, 赵学亮, 王飞跃. 深度学习在控制领域的研究现状与展望. 自动化学报, 2016, **42**(5): 643−654)

10 Liu Y-J, Tang L, Tong S-C, Chen C L P, Li D-J. Reinforcement learning design-based adaptive tracking control with less learning parameters for nonlinear discrete-time MIMO systems. *IEEE Transactions on Neural Networks and Learning Systems*, 2015, **26**(1): 165−176

11 Liu L, Wang Z, Zhang H. Adaptive fault-tolerant tracking control for MIMO discrete-time systems via reinforcement learning algorithm with less learning parameters. *IEEE Transactions on Automation Science and Engineering*, 2017, **14**(1): 299−313

12 Xu X, Yang H, Lian C, Liu J. Self-learning control using dual heuristic programming with global laplacian eigenmaps. *IEEE Transactions on Industrial Electronics*, 2017, **64**(12): 9517−9526

13 Wei Q-L, Liu D-R. Adaptive dynamic programming for optimal tracking control of unknown nonlinear systems with application to coal gasification. *IEEE Transactions on Automation Science and Engineering*, 2014, **11**(4): 1020−1036

14 Jiang Y, Fan J-L, Chai T-Y, Li J-N, Lewis L F. Data-driven flotation industrial process operational optimal control based on reinforcement learning. *IEEE Transactions on Industrial Informatics*, 2017, **14**(5): 1974−1989

15 Jiang Y, Fan J-L, Chai T-Y, Lewis L F. Dual-rate operational optimal control for flotation industrial process with unknown operational model. *IEEE Transactions on Industrial Electronics*, 2019, **66**(6): 4587−4599

16 Modares H, Lewis F L. Automatica integral reinforcement learning and experience replay for adaptive optimal control of partiallyunknownconstrained-input. *Automatica*, 2014, **50**(1): 193−202

17 Mnih V, Silver D, Riedmiller M. Playing atari with deep reinforcement learning. In: Procedings of the NIPS Deep Learning Workshop 2013, Lake Tahoe, USA: NIPS 2013, 1−9

18 Wang D, Liu D R, Wei Q L, Zhao D B, Jin N. Automatica optimal control of unknown nonaffine nonlinear discrete-time systems basedon adaptive dynamic programming. *Automatica*, 2012, **48**(8): 1825−1832

19 Chai T Y, Jia Y, Li H B, Wang H. An intelligent switching control for a mixed separation thickener process. *Control Engineering Practice*, 2016, **57**: 61−71

20 Kim B H, Klima M S. Development and application of a dynamic model for hindered-settling column separations. *Minerals Engineering*, 2004, **17**(3): 403−410

21 Wang L Y, Jia Y, Chai T Y, Xie W F. Dual rate adaptive control for mixed separationthickening process using compensation signal basedapproach. *IEEE Transactions on Industrial Electronics*, 2017, **PP**: 1−1

22 Wang Meng. Design and development of model software of processes of slurry neutralization, sedimentation and separation. *Northeastern University*, 2011
(王猛. 矿浆中和沉降分离过程模型软件的研发. 东北大学, 2011)

23 Tang Mo-Tang. Hydrometallurgical equipment. *Central South University*, 2009
(唐谟堂. 湿法冶金设备. 中南大学出版社, 2009)

24 Wang Lin-Yan, Li Jian, Jia Yao, Chai Tian-You. Dual-rate intelligent switching control for mixed separation thickening process. *Acta Automatica Sinica*, 2018, **44**(2): 330−343
(王琳岩, 李健, 贾瑶, 柴天佑. 混合选别浓密过程双速率智能切换控制. 自动化学报, 2018, **44**(2): 330−343)

25 Luo B, Liu D R, Huang T W, Wang D. Model-free optimal tracking control via critic-only Q-learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2016, **27**(10):

2134−2144

26 Padhi R, Unnikrishnan N, Wang X H, Balakrishnan S N. A single network adaptive critic (SNAC) architecture for optimal controlsynthesis for a class of nonlinear systems. *Neural Networks*, 2006, **19**(10): 1648−1660

袁兆麟 北京科技大学计算机与通信工程学院博士研究生. 2017 年获得北京科技大学计算机科学与技术系学士学位. 主要研究方向为自适应动态规划和强化学习.
E-mail: b20170324@xs.ustb.edu.cn
(**YUAN Zhao-Lin** Ph.D. candidate at the School of Computer and Communication Engineering, University of Science and Technology Beijing. He received his bachelor degree in computer science from University of Science and Technology Beijing in 2017. His research interest covers adaptive dynamic programming and reinforcement learning.)

何润姿 北京科技大学计算机与通信工程学院硕士研究生. 2017 年获得北京信息科技大学计算机科学与技术系学士学位. 主要研究方向为流体仿真和强化学习.
E-mail: hrz.claire@gmail.com
(**HE Run-Zi** Master student at the School of Computer and Communication Engineering, University of Science and Technology in Beijing. She received her bachelor degree from Beijing Science and Technology University in 2017. Her research interest covers fluid simulation and reinforcement learning.)

姚 超 北京科技大学的助理教授. 2009 年获得北京交通大学计算机科学学士学位, 2016 年获得北京交通大学信息科学研究所博士学位. 2014 年至 2015 年, 他在瑞士洛桑联邦理工学院担任访问博士. 2016 年至 2018 年, 他在北京邮电大学传感技术与商业研究所担任博士后. 主要研究方向为图像和视频处理,计算机视觉.
E-mail: yaochao@ustb.edu.cn
(**YAO Chao** Assistant professor at University of Science Technology, Beijing (USTB), China. He received his bachelor degree in computer science from Beijing Jiaotong University (BJTU), Beijing, China in 2009 and the Ph.D. degree from the Institute of Information Science, BJTU in 2016. From 2014 to 2015, he served as a visiting Ph.D. student at the Ecole Polytechnique Federale de Lausanne, Switzerland. From 2016 to 2018, he served as a post-doctoral at the Institute of Sensing Technology and Business, Beijing University of Posts and Telecommunications, Beijing. His research interest covers image and video processing and computer vision.)

李 佳 北京科技大学计算机与通信工程学院硕士研究生, 主要研究方向为自适应动态规划, 自适应控制, 强化学习.
E-mail: lijia1117@foxmail.com
(**LI Jia** Master student at the School of Computer and Communication Engineering, University of Science and Technology in Beijing. His research interest covers adaptive dynamic programming, adaptive control, and reinforcement learning.)

班晓娟 北京科技大学教授, 中国人工智能学会常务理事. 主要研究方向为人工智能,自然人机交互,三维可视化技术. 本文通信作者.
E-mail: banxj@ustb.edu.cn
(**BAN Xiao-Juan** Professor at University of Science and Technology Beijing and she is an executive council member in Chinese Association for Artificial Intelligence (CAAI). Her research interest covers artificial intelligence, natural human-computer interaction, and 3D visualization. Corresponding author of this paper.)