不确定工业过程运行指标异步更新强化学习决策算法

李金娜 [1] 袁 林 [1] 丁进良 [2]

摘 要 运行指标决策问题是实现工业过程运行安全和生产指标优化的关键. 考虑到多运行指标决策问题求解的复杂性和工业过程生产条件动态波动引发生产指标状态的不确定性, 提出了一种策略异步更新强化学习算法自学习决策运行指标, 并给出算法收敛性的理论证明. 该算法在随机自适应动态规划框架下, 利用样本均值代替计算生产指标状态转移概率矩阵, 因此无需要求生产指标状态转移概率矩阵已知. 并且通过引入时钟和定义其阈值, 采用集中式策略评估、多策略异步更新方式用以简化求解多运行指标决策问题, 提高强化学习的学习效率. 利用可测量数据, 自学习得到的运行指标能够保证生产指标优化, 并且限制在规定范围之内. 最后, 采用中国西部某大型选矿厂的实际数据进行仿真验证, 表明该方法的有效性.

关键词 运行优化控制, 强化学习, 数据驱动控制, 自适应动态规划, 安全运行

引用格式 李金娜, 袁林, 丁进良. 不确定工业过程运行指标异步更新强化学习决策算法. 自动化学报, 2023, **49**(2): 461−472

DOI 10.16383/j.aas.c210983

Asynchronous Updating Reinforcement Learning Algorithm for Decision-making Operational Indices of Uncertain Industrial Processes

LI Jin-Na[1] YUAN Lin[1] DING Jin-Liang[2]

Abstract The decision-making operational index has been a key issue for achieving safe and optimal operation of industrial processes. Considering the complexity of decision making of multiple operational indices and the uncertainty of production indices caused by changes of working condition in industrial processes, this paper proposes a reinforcement learning algorithm with policy asynchronous updating for the first time aiming at self-learning operational indices, followed by the theoretical proof of convergence of the proposed algorithm. To this end, under the framework of stochastic adaptive dynamic programming, the sample mean is utilized rather than calculating the state transition probability matrix of production indices, with the outcome that the state transition probability matrix of production indices is not required to be known a priori. Distinctly from traditional synchronized policy updating, the centralized policy evaluation and asynchronous updating of multiple policies are implemented in the proposed algorithm based on the introduction of a time clock with its threshold, such that solving the concerned decision-making problem of multiple operational indices becomes easier and the learning efficiency of reinforcement learning is improved. Thus, the self-learned operational indices using measured data can ensure the optimality of production indices and limit them within the prescribed range. Experiments are conducted using the real date collected from a large-scale mineral processing plant in west China in order to illustrate the effectiveness of the approach.

Key words Optimal operational control, reinforcement learning, data-driven control, adaptive dynamic programming, safe operation

Citation Li Jin-Na, Yuan Lin, Ding Jin-Liang. Asynchronous updating reinforcement learning algorithm for decision-making operational indices of uncertain industrial processes. *Acta Automatica Sinica*, 2023, **49**(2): 461−472

工业过程运行指标决策的内涵是以工业过程生产指标优化为目标的运行指标决策问题 (如图 1 所示). 生产指标是指反映企业或者生产线最终产品的质量、产量、成本和能量消耗等相关的指标, 运行指

收稿日期 2021-10-18 录用日期 2022-04-28
Manuscript received October 18, 2021; accepted April 28, 2022
国家重点研发计划项目 (2018YFB1701104), 国家自然科学基金 (62073158, 61673280, 61525302, 61833004), 辽宁省兴辽计划 (XLYC1808001), 辽宁省科技计划项目 (2020JH2/10500001), 辽宁省自然基金重点领域联合开放基金 (2019-KF-03-06), 辽宁省教育厅基本科研项目 (LJKZ0401) 资助
Supported by National Key Research and Development Plan Project (2018YFB1701104), National Natural Science Foundation of China (62073158, 61673280, 61525302, 61833004), Project of Liaoning Province Prosperity Plan (XLYC1808001), Science and Technology Planning Project of Liaoning Province (2020 JH2/10500001), Open Project of Key Field Alliance of Liaoning Province (2019-KF-03-06), and Basic Research Project of Education Department of Liaoning Province (LJKZ0401)
本文责任编委 胡清华
Recommended by Associate Editor HU Qing-Hua

1. 辽宁石油化工大学信息与控制工程学院 抚顺 113000 2. 东北大学流程工业综合自动化国家重点实验室 沈阳 110819
1. School of Information and Control Engineering, Liaoning Petrochemical University, Fushun 113000 2. State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819

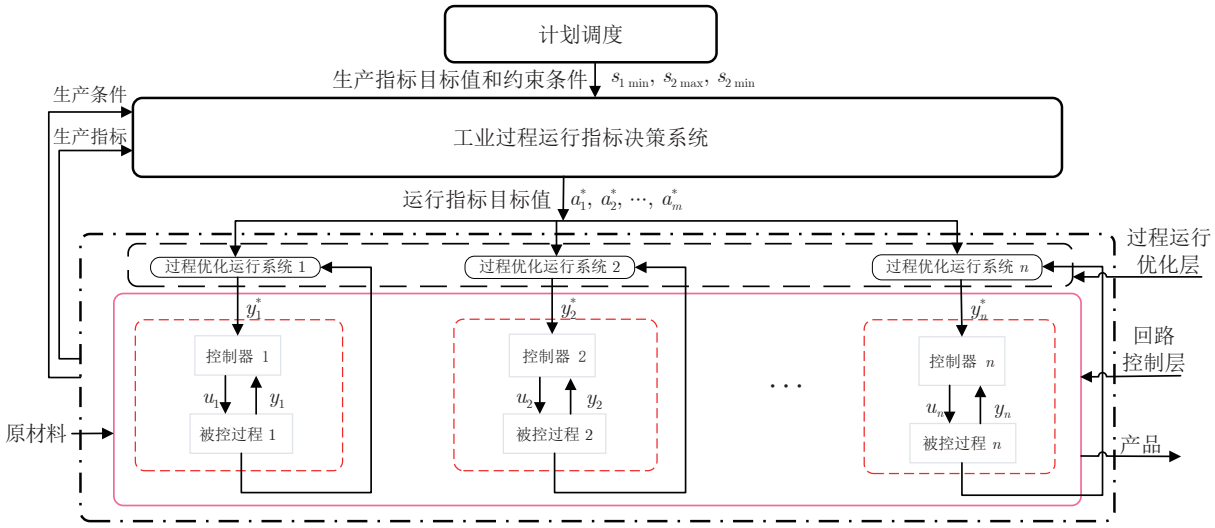

图 1 工业过程运行指标决策问题

Fig. 1 Decision-making problem of operational indices in industrial processes

标是指反映控制单元的产品在运行周期内的质量、效率、能耗和物耗等相关的指标[1−2]. 面对激烈的国内外市场竞争, 能量节约和安全生产的民生需求和政策导向, 以及原材料和运行工况 (生产条件) 的动态波动, 研究工业过程运行指标决策问题, 提高产品的质量、产量和能量使用效率等生产指标, 保证安全运行, 这是增强企业竞争力和可持续发展的必然选择.

关于工业过程运行指标决策问题的研究一直是工业界、学术界研究的热点. 工业过程运行指标决策是一个复杂的多目标优化问题, 其复杂性包括: 1) 工业过程通常由多个控制单元构成, 每个控制单元有各自的运行指标需求, 目标是协同优化整个工业过程的多个生产指标; 2) 生产指标和运行指标之间的动态关系呈现非线性和不确定性特征. 因此, 传统的利用操作人员现场经验协调各运行指标的方式无法保证工业过程生产指标的优化[1−3]. 那么, 如何简化求解此多目标优化问题, 设计一种减少计算耗时并优化生产指标的方法, 是本文研究的根本动机.

相比于集中式运行指标决策方法[4−5], 分布式运行指标决策方法[6−11] 有利于简化求解的复杂性. 文献 [6, 8−9] 针对多个生产指标优化问题, 融合性能预测与反馈控制, 提出了运行指标动态校正方法. 在此基础上, 文献 [7] 引入强化学习思想, 基于案例推理策略, 给出了数据驱动的运行指标动态修正方法. 但上述方法仍需要利用操作人员的经验调整运行指标, 很难保证生产指标的优化. 文献 [10] 采用强化学习技术, 基于博弈理论, 给出工业过程运行指标自学习方法, 保证生产指标以近似最优的方式跟踪理想值. 注意到, 文献 [10] 没有考虑生产条件波动对生产指标性能的影响. 但实际工业过程原料成分、运行工况、设备状态等多种不确定因素导致生产条件动态波动. 文献 [11] 以最大化产品产量为目标, 利用历史数据, 提出了一种多执行网络集成强化学习算法, 自学习决策运行指标. 但该研究成果忽略了实际工业过程运行指标需要满足的约束条件, 并且性能指标为单次采样时刻奖赏值, 无法保证累积生产指标的优化.

综合分析上述分布式运行指标决策方法, 在生产条件动态波动、生产指标和运行指标存在静态约束的情况下, 如何以数据驱动的方式分布式自学习决策工业过程运行指标仍是一个挑战性难题. 这是本文研究的第二个动机.

自适应动态规划技术是智能最优控制领域研究的热点. 该方法的本质是采用强化学习技术求解哈密顿−雅可比−贝尔曼 (Hamilton-Jacobi-Bellman, HJB) 方程, 以迭代方式求解最优控制策略[12−16]. 文献 [10, 15, 17–19] 等针对复杂大系统, 提出了一系列自适应动态规划方法用来分布式自学习最优控制策略, 优化控制系统性能. 但现有分布式最优控制策略自学习方法, 往往忽略了系统不确定性 (如环境动态波动等) 导致的状态不确定性 (随机性), 无法在随机变化的环境下保证系统性能的优化, 甚至无法保证系统的稳定性. 针对随机最优控制问题的自适应动态规划方法还鲜见报道. 文献 [20] 针对离散随机过程, 提出了一种自适应动态规划方法, 自学习最优控制策略, 但解决的是单变量控制问题, 并且要求系统状态转移概率矩阵已知. 然而, 实际工业过程生产指标状态转移概率矩阵无法准确计算. 此外, 现有的分布式强化学习技术中策略更新

为多个控制变量同步更新, 多个执行网络同步训练将产生较大的时间开销和计算负载. 因此, 现有的自适应动态规划技术仍无法直接用于解决本文研究的两个动机问题.

文献 [21–23] 利用惩罚函数和 Barrier 函数能解决系统状态变量和控制输入约束问题. 受其启发, 本文在效用函数中引入 Barrier 函数和惩罚函数, 用以解决生产指标和运行指标静态约束问题. 利用样本均值代替计算生产指标状态转移概率矩阵, 首次提出了一种策略异步更新强化学习算法, 并给出了算法收敛性的理论证明. 研究中面临的挑战性难题是在保证算法收敛性的前提下, 如何实现策略异步更新和如何证明算法的收敛性. 为此, 本文引入时钟并定义其阈值, 执行集中式性能评估, 多策略异步更新, 并且基于随机最优控制理论, 采用数学归纳法证明了所提算法的收敛性. 所提出的方法不要求生产指标状态转移概率矩阵已知, 多策略异步更新方式提高了学习效率, 同时有效地解决了生产指标和运行指标的静态约束问题, 实现了不确定工业过程生产指标优化, 并且保证系统安全运行. 实验验证了所提方法的有效性和可行性.

本文主要的创新点如下:

1) 首次提出了一种策略异步更新强化学习算法, 采用集中式性能评估, 多策略异步更新, 可以减少计算成本和时间, 提高学习效率. 并且, 本文给出了所提算法收敛性的理论证明.

2) 本文不要求系统状态转移概率矩阵已知, 在随机自适应动态规划框架下, 利用样本均值代替计算生产指标状态转移概率矩阵, 提出的强化学习算法利用可测量数据, 在生产条件动态波动的情况下, 自学习得到的运行指标能够优化生产指标.

1　工业过程运行指标决策问题描述

工业过程的运行指标和生产指标之间的动态具有强耦合性、非线性、受生产条件变化影响等特征[1−2, 10−11]. 本文研究的目标是在充分考虑上述特征的情况下, 给出一种快速地自学习决策运行指标的方法, 优化生产指标, 并且保证生产指标和运行指标满足静态约束条件. 本文不研究过程运行优化层和回路控制层如何设计设定值和控制输入, 实现运行指标跟踪理想运行指标 (如图 1 所示). 为此, 本文首先假设生产指标与运行指标之间的动态关系如下:

$$s_{k+1} = f(s_k,\, a_{1k},\, a_{2k},\, \cdots,\, a_{mk},\, d_k) \tag{1}$$

式中, $s_k = s(k) \in \mathbf{R}^n$ 和 $a_{ik} = a_i(k) \in \mathbf{R}^{q_i}$ $(i = 1,\, 2,\, \cdots,\, m)$ 分别表示工业过程生产指标和运行指标, $d_k = d\,(k) \in \mathbf{R}^{\kappa}$ 表示生产条件, $f(\cdot)$ 为未知非线性连续函数. 运行指标决策问题可以建模为一个部分可观察马尔科夫决策过程, 并用一个六元组 $\mathcal{G} =< S,\, A,\, P,\, r,\, O,\, \gamma >$ 表示, 其中 S、A、O 分别表示状态空间, 动作空间和由可观察数据构成的观察空间, r 表示立即奖赏, γ $(0 < \gamma \leq 1)$ 表示折扣因子. $P(s_{k+1}|s_k,\, a_k)$ 表示在当前状态 s_k 下, 采取动作行为 a_k 产生下一刻状态 s_{k+1} 的状态转移概率. 在实际的工业过程中, 生产指标采样周期通常为天和小时等. 本文中采样时刻 $k = 0,\, 1,\, \cdots$ 表示第 k 天或小时. 具体的状态空间、观察空间和动作空间说明如下:

1) 状态空间 S: 对于系统 (1), 生产指标 s_k 为状态变量, $s_k \in S$, S 是连续空间.

2) 观察空间 O: 在实际生产中, 生产条件 d_k 可以表示单位时间吞吐量、原料质量和运行工况等. 本文假设状态变量和生产条件是可测量的, 那么 $\{s_k,\, d_k\} \in O$. 工业生产中生产条件不可能保持不变, 通常具有统计特性. 受生产条件波动的影响, 未来的生产指标 s_{k+1} 是不确定的.

3) 动作空间 A: m 个运行指标 a_{ik} $(i = 1,\, 2,\, \cdots,\, m)$ 构成决策变量, 因此 $a_{ik} \in A$. 本文假设决策是确定的, 即 $P(a_k|s_k) = 1$, 决策运行指标, 优化工业过程的生产指标. 实际工业过程运行指标是连续的且需要满足一定的约束条件, 因此 A 为连续动作空间.

现有的自适应动态规划方法为解决连续动作空间的最优控制问题提供了可借鉴的理论和方法. 因此, 本文在自适应动态规划框架下, 拟提出一种策略异步更新强化学习算法, 自学习运行指标. 为实现本文研究目标, 本文定义如下性能指标:

$$J = \mathrm{E}\left\{\sum_{k=0}^{\infty} \gamma^k c\,(a_{1k},\, \cdots,\, a_{mk})\right\} \tag{2}$$

式中, $c\,(\cdot)$ 是效用函数, 算子 $\mathrm{E}\,\{\cdot\}$ 表示数学期望.

注 1. 本文目标是优化不确定工业过程生产指标, 因此效用函数要刻画生产指标. 为具体明确, 本文取生产指标为产品产量 s_{1k} 和产品质量 s_{2k}. 目标是最大化产品产量, 控制产品质量在规定范围之内. 因此, 改写式 (2) 得到如下优化问题:

问题 1.

$$\min_{a_{1k},\, a_{2k},\, \cdots,\, a_{mk}} \sum_{k=0}^{\infty} \frac{1}{s_{1k}} \tag{3}$$

s. t.

$$\begin{cases} s_{1(k+1)} = f_1(s_{1k},\, s_{2k},\, a_{1k},\, a_{2k},\, \cdots,\, a_{mk},\, d_k) \\ s_{2(k+1)} = f_2(s_{1k},\, s_{2k},\, a_{1k},\, a_{2k},\, \cdots,\, a_{mk},\, d_k) \end{cases} \tag{4}$$

$$\begin{cases} s_{1k} \geq s_{1\min} \\ s_{2\min} \leq s_{2k} \leq s_{2\max} \\ a_{i\min} \leq a_{ik} \leq a_{i\max} \end{cases} \tag{5}$$

式中, $s_{1\min}$、$s_{2\min}$、$s_{2\max}$ 为正实数.

为满足生产指标和运行指标的静态约束条件, 类似文献 [22−23], 本文引入一个 Barrier 函数:

$$\begin{aligned} B_r(s_{1k}, s_{2k}) = & -\ln\left(\frac{\gamma_1(s_{1k}-s_{1\min})}{\gamma_1(s_{1k}-s_{1\min})+1}\right) - \\ & \ln\left(\frac{\gamma_2(s_{2\max}-s_{2k})}{\gamma_2(s_{2\max}-s_{2k})+1}\right) - \\ & \ln\left(\frac{\gamma_3(s_{2k}-s_{2\min})}{\gamma_3(s_{2k}-s_{2\min})+1}\right) \end{aligned} \tag{6}$$

和一个惩罚函数[21]:

$$\varphi(a_{ik}) = 2\sum_{i=1}^{m}\int_{\bar{a}_i}^{a_{ik}} \tanh^{-\mathrm{T}}\left(U_i^{-1}(s-\bar{a}_i)\right)U_iP_i\mathrm{d}s \tag{7}$$

式中, $a_{i\min}=[\theta_{i1}, \theta_{i2}, \cdots, \theta_{in_i}]^{\mathrm{T}}$, $a_{i\max}=[\rho_{i1}, \rho_{i2}, \cdots, \rho_{in_i}]^{\mathrm{T}}$, $\bar{a}_i=(a_{i\max}+a_{i\min})/2$. γ_1、γ_2、γ_3 均为正实数, $U_i=\mathrm{diag}\{(\rho_{i1}-\theta_{i1})/2, (\rho_{i2}-\theta_{i2})/2, \cdots, (\rho_{in_i}-\theta_{in_i})/2\}$, P_i 为适维正定矩阵, $\tanh^{-\mathrm{T}}=(\tanh^{-1})^{\mathrm{T}}$, $\tanh^{-1}$为标准的通用的反双曲函数. 由式 (5) 和式(6) 可知, $B_r(\cdot)>0$; 当 $s_k\in S$ 时, $B_r(\cdot)$是单调减函数; 当 s_k 接近 S 的边界时, $B_r(\cdot)\to\infty$. Barrier 函数用来确保生产指标满足静态约束条件 (5)[22−23].

由此, 问题 1 转化为如下优化问题 2.

问题 2.

$$\min_{a_{1k}, a_{2k}, \cdots, a_{mk}} \mathrm{E}\left\{\sum_{i=0}^{\infty}\gamma^i c\left(a_{1(k+i)}, \cdots, a_{m(k+i)}\right)\right\} \tag{8}$$

s. t. 式 (4)

式中

$$c(a_{1k}, \cdots, a_{mk}) = \frac{1}{s_{1k}} + \varphi(a_{ik}) + B_r(s_{1k}, s_{2k})$$

注 2. 效用函数 $c(a_{1k}, \cdots, a_{mk})$ 中 $1/s_{1k}$ 表示产品产量的倒数, 如果想要产品产量最大化, 那么需要 $1/s_{1k}$ 最小化. 不同于文献 [10−11], 性能指标 (8) 包含了通过折扣因子 γ 衰减作用后累积性能的期望值. 折扣因子使得邻近 k 时刻的产品产量比未来的值更重要. 并且, 性能指标中还包含了运行指标和生产指标的静态约束信息.

注 3. 通过引入 Barrier 函数和惩罚函数, 本文将静态约束转化为性能指标函数. 问题 2 本质上是一个最优控制问题, 运行指标成为动态系统 (4) 的控制输入, 那么最大化产品产量和控制产品质量在规定范围的多目标多约束优化问题 1 被转化为单目标无静态约束的最优控制问题 2.

相比于集中式控制或者变量决策, 分布式控制具有减轻计算负载的优势. 下面将针对优化问题 2 给出具体的求解算法.

2 最优运行指标决策

本节针对优化问题 2, 基于强化学习技术和随机优化控制理论, 提出了一种策略异步更新强化学习算法, 并且证明了算法的收敛性.

2.1 随机最优控制方法

由于生产条件 d_k 的随机性特征, k 时刻生产指标取值具有多种可能性. 因此, 利用贝叶斯法则, 性能指标 (8) 可以改写为:

$$j_k = \mathrm{E}_{s_k}\left(\mathrm{E}\left[V(s_k)\,|s_k\right]\right) \tag{9}$$

式中, E_{s_k} 为关于工业过程生产指标 s_k 的数学期望, 值函数 $V(s_k)=\mathrm{E}(\sum_{i=k}^{\infty}\gamma^{i-k}c(a_{1i}, \cdots, a_{mi}))$.

定义 $\bar{V}(s_k)=\mathrm{E}\left[V(s_k)\,|s_k\right]$, 其满足:

$$j_k = \mathrm{E}_{s_k}(\bar{V}(s_k)) \tag{10}$$

则最优性能函数满足如下[24]:

$$\begin{aligned} {j_k}^* = & \min_{a_{1k}, \cdots, a_{mk}}(j_k) = \\ & \mathrm{E}_{s_k}(\min_{a_{1k}, \cdots, a_{mk}}\bar{V}(s_k)) = \\ & \mathrm{E}_{s_k}(\bar{V}^*(s_k)) \end{aligned} \tag{11}$$

上式表明最小化 j_k 可以等价地设计最小化 $\bar{V}(s_k)$ 的决策规则. 如果固定 $k+1$ 时刻的生产指标 s_{k+1}, 则有:

$$\begin{aligned} \mathrm{E}\left[V(s_k)\,|s_{k+1}\right] = & \mathrm{E}\left[c_k(a_{1k}, \cdots, a_{mk})\,|s_{k+1}\right] + \\ & \gamma\mathrm{E}\left[V(s_{k+1})\,|s_{k+1}\right] \end{aligned} \tag{12}$$

基于随机最优控制理论和动态规划理论[10, 20, 24], 当所有运行指标取最优策略 $a_i^*(s_{k+1})$ $(i=1, 2, \cdots, m)$ 时, k 时刻性能的数学期望为:

$$\mathrm{E}\left[V(s_k)\,|s_{k+1}\right] = c_k + \gamma\bar{V}^*(s_{k+1}) \tag{13}$$

由于 $k+1$ 时刻工业过程生产指标 s_{k+1} 具有不确定性, 则有:

$$\bar{V}^*(s_k) = \min_{a_{1k}, \cdots, a_{mk}}\left\{c_k + \gamma\mathrm{E}_{s_{k+1}}\left[\bar{V}^*(s_{k+1})\right]\right\} \tag{14}$$

利用最优性的必要条件, 最优的运行指标为:

$$\begin{aligned} a_{ik}^* = & \bar{a}_i + U_i\tanh\left[-\frac{\gamma}{2}(\bar{a}_iP_i)^{-\mathrm{T}}\frac{\partial s_{k+1}}{\partial a_{ik}}\cdot\right. \\ & \left.\frac{\partial\mathrm{E}_{s_{k+1}}(\bar{V}^*(s_{k+1}))}{\partial s_{k+1}}\right], \quad i=1, 2, \cdots, m \end{aligned} \tag{15}$$

将式 (15) 代入式 (14), 得到离散时间 HJB 方程:

$$\bar{V}^*(s_k) = c_k(a_{1k}^*, \cdots, a_{mk}^*) + \gamma\mathrm{E}_{s_{k+1}}[\bar{V}^*(s_{k+1})] \tag{16}$$

式中, $s_{k+1}=f(s_k, a_{1k}^*, \cdots, a_{mk}^*, d_k)$.

注 4. 由式 (15) 可知, 本文采用分布式状态反馈优化控制的方式, 分布式设计运行指标. 与构成运行指标增广向量, 采用集中式方法设计运行指标相比, 减少计算负载.

注 5. 根据随机最优控制理论和动态规划理论, 满足式 (16) 的运行指标式 (15) 能够最小化性能指标式 (8). 由式 (15), 有 $a_{i\min}\le a_{ik}^*\le a_{i\max}$, 即运行指标满足静态约束条件.

由于生产指标动态 (4) 未知, 并且计算 $\mathrm{E}_{s_{k+1}}[\bar{V}^*(s_{k+1})]$ 需要状态转移概率矩阵 $P(s_{k+1}|s_k, a_k)$ 已知, 使得求解 HJB 方程 (16), 进一步得到形如式 (15) 的最优运行指标变得非常困难. 下面将给出具体的求解算法.

2.2　运行指标自学习决策方法

本节将拓展现有的自适应动态规划方法, 提出一种新的运行指标自学习决策算法, 在优化性能式 (8) 的意义下, 实现: 1) 产品产量最大化; 2) 控制产品质量在规定范围之内; 3) 运行指标限制在规定范围之内, 实现工业过程生产指标优化并且保证安全运行.

定义 1[20, 22, 25]**.** 如果运行指标 a_{ik} ($i=1, 2, \cdots, m$) 满足: 1) 镇定系统式 (4); 2) 当生产指标 s_k 满足约束条件式 (5) 时, j_k 是有界的, 那么运行指标 a_{ik} 称为是可允许的.

为了用数值方法求解离散 HJB 方程 (16), 本文提出了策略异步更新强化学习算法 1, 图 2 给出了算法 1 的执行机制.

算法 1. 策略异步更新强化学习算法

1) 初始化. 给定可允许的运行指标初始值, 令迭代指标 $j=0$.

2) 集中式性能评估. 根据式 (17) 求解 $\bar{V}^{(j)}(s_k)$:

$$\bar{V}^{(j)}(s_k)=c_k+\gamma\mathrm{E}_{s_{k+1}}[\bar{V}^{(j)}(s_{k+1})] \tag{17}$$

式中, $c_k(a_{1k}^{(\ell_1)}, a_{2k}^{(\ell_2)}, \cdots, a_{mk}^{(\ell_m)})$, $s_{k+1}=f(s_k, a_{1k}^{(\ell_1)}, a_{2k}^{(\ell_2)}, \cdots, a_{mk}^{(\ell_m)}, d_k)$, 并且 $\ell_i\le j$ ($i=1, 2, \cdots, m$).

3) 运行指标异步更新. 设置时钟 t_{clock} 和阈值 $t_{threshold}$, 如果 $t_{clock}\le t_{threshold}$, 有:

$$a_{ik}^{(j+1)}=\bar{a}_i+U_i\tanh\left[-\frac{\gamma}{2}(\bar{a}_iP_i)^{-\mathrm{T}}\cdot\frac{\partial s_{k+1}}{\partial a_{ik}}\frac{\partial\mathrm{E}_{s_{k+1}}(\bar{V}^{(j)}(s_{k+1}))}{\partial s_{k+1}}\right] \tag{18}$$

否则, $a_{ik}^{(j+1)}=a_{ik}^{(j)}$.

4) 如果 $\|\bar{V}^{(j+1)}(s_k)-\bar{V}^{(j)}(s_k)\|\le\varepsilon$ (ε 为一个很小的正数), 则停止迭代计算; 否则, $j\leftarrow j+1$ 并返回步骤 2).

注 6. 在算法 1 步骤 3) 中, 本文利用步骤 2) 得到的 $\bar{V}^{(j)}$ 更新运行指标 $a_{ik}^{(j+1)}$ 时, 需要计算 $\mathrm{E}_{s_{k+1}}(\bar{V}^{(j)}(s_{k+1}))$, 并且通常需要利用神经网络估计运行指标 $a_{ik}^{(j+1)}$ [26−27], 即涉及复杂的计算. 通过引入时钟 t_{clock}, 如果部分运行指标的计算时间超过阈值 $t_{threshold}$, 则保持上一次迭代值, 即保持 $a_{ik}^{(j)}$ 不变. 因此称为策略异步更新. 定理 1 给出了算法 1 的收敛性证明.

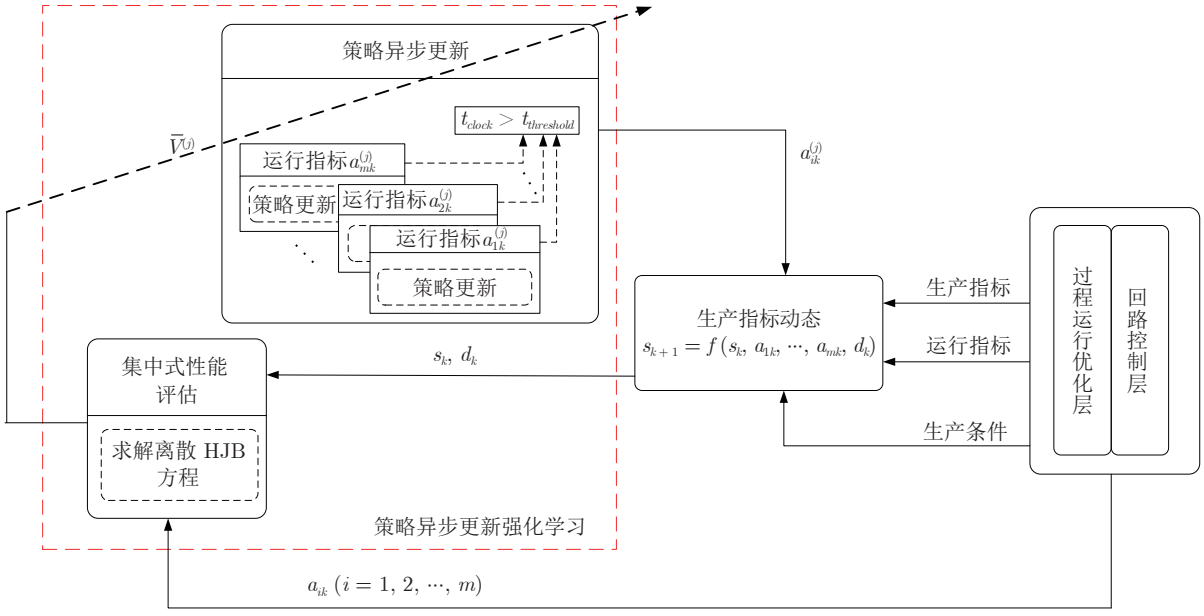

图 2　运行指标自学习机制

Fig. 2　Self-learning mechanism of operational indices

定理 1. 假设 $\bar{V}^{(j)}(s_k)$ 和 $a_i^{(j+1)}(s_k)$ $(i=1, 2, \cdots, m)$ 可以由式 (17) 和式 (18) 得到, 则对于所有的 $s_k \in S$ 和任意迭代指标 j 如下结论成立

$$\bar{V}^{(j+1)}(s_k) \le \bar{V}^{(j)}(s_k) \tag{19}$$

$$\bar{V}^{(j)}(s_k) < \infty \tag{20}$$

$$\begin{cases} \lim\limits_{j\to\infty} \bar{V}^{(j+1)}(s_k) = \bar{V}^{*}(s_k) \\ \lim\limits_{j\to\infty} a_{ik}^{(j+1)} = a_{ik}^{*} \end{cases} \tag{21}$$

证明. 1) 采用数学归纳法. 当迭代指标 $j=1$ 时, 由式 (17) 定义:

$$\bar{V}_{q+1}^{(1)}(s_k) = c_k\left(a_{1k}^{(\ell_1)}, \cdots, a_{mk}^{(\ell_m)}\right) + \gamma \mathrm{E}_{s_{k+1}}\left[\bar{V}_q^{(1)}(s_{k+1})\right] \tag{22}$$

式中, $q=0, 1, \cdots$; $\bar{V}_0^{(1)}(s_k) = \bar{V}^{(0)}(s_k)$; $a_{ik}^{(1)} = \bar{a}_i + U_i \tanh[-\gamma/2 \cdot (\bar{a}_i P_i)^{-\mathrm{T}} \partial s_{k+1}/\partial a_{ik} \times \partial \mathrm{E}_{s_{k+1}}(\bar{V}^{(0)}(s_{k+1}))/\partial s_{k+1}]$.

欲证明:

$$\bar{V}_{q+1}^{(1)}(s_k) \le \bar{V}^{(0)}(s_k) \tag{23}$$

需要先证明 $\bar{V}_1^{(1)}(s_k) \le \bar{V}^{(0)}(s_k)$. 根据式 (22), 有:

$$\bar{V}_1^{(1)}(s_k) = c_k\left(a_{1k}^{(\ell_1)}, \cdots, a_{mk}^{(\ell_m)}\right) + \gamma \mathrm{E}_{s_{k+1}}\left[\bar{V}_0^{(1)}(s_{k+1})\right] \tag{24}$$

式中, $s_{k+1} = f\left(s_k, a_{1k}^{(\ell_1)}, a_{2k}^{(\ell_2)}, \cdots, a_{mk}^{(\ell_m)}\right)$, $\ell_i \le 1$. 由式 (18) 可知:

$$\{a_{1k}^{(1)}, \cdots, a_{mk}^{(1)}\} = \arg\min_{a_{ik}} \{c_k(a_{1k}, a_{2k}, \cdots, a_{mk}) + \gamma \mathrm{E}_{s_{k+1}}[\bar{V}^{(0)}(s_{k+1})]\} \tag{25}$$

则有如下不等式成立:

$$c_k(a_{1k}^{(1)}, \cdots, a_{mk}^{(1)}) + \gamma \mathrm{E}_{s_{k+1}}\left[\bar{V}_0^{(1)}(s_{k+1})\right] \le c_k\left(a_{1k}^{(0)}, a_{2k}^{(0)}, \cdots, a_{mk}^{(0)}\right) + \gamma \mathrm{E}_{s_{k+1}}[\bar{V}^{(0)}(s_{k+1})] = \bar{V}^{(0)}(s_k) \tag{26}$$

如果部分运行指标由于计算超时没有变化, 如仅 a_{1k} 更新为 $a_{1k}^{(1)}$, 其他为 $a_{ik}^{(0)}(i=2, \cdots, m)$, 那么类似式 (25), 有 $a_{1k}^{(1)} = \arg\min_{a_{1k}} c_k(a_{1k}, a_{2k}^{(0)}, \cdots, a_{mk}^{(0)}) + \gamma \mathrm{E}_{s_{k+1}}[\bar{V}^{(0)}(s_{k+1})]$, 有:

$$c_k(a_{1k}^{(1)}, \cdots, a_{ik}^{(0)}, \cdots, a_{mk}^{(0)}) + \gamma \mathrm{E}_{s_{k+1}}\left[\bar{V}_0^{(1)}(s_{k+1})\right] \le c_k(a_{1k}^{(0)}, \cdots, a_{mk}^{(0)}) + \gamma \mathrm{E}_{s_{k+1}}\left[\bar{V}^{(0)}(s_{k+1})\right] = \bar{V}^{(0)}(s_k) \tag{27}$$

其他情况类似. 因此, 可得出结论: $\bar{V}_1^{(1)}(s_k) \le \bar{V}^{(0)}(s_k)$.

假设 $\bar{V}_q^{(1)}(s_k) \le \bar{V}^{(0)}(s_k)$, 根据式 (22)、式 (26) 和式 (27), 有:

$$\begin{aligned} \bar{V}_{q+1}^{(1)}(s_k) &= c_k\left(a_{1k}^{(\ell_1)}, a_{2k}^{(\ell_2)}, \cdots, a_{mk}^{(\ell_m)}\right) + \\ &\quad \gamma \mathrm{E}_{s_{k+1}}\left[\bar{V}_q^{(1)}(s_{k+1})\right] \le \\ &\quad c_k\left(a_{1k}^{(\ell_1)}, a_{2k}^{(\ell_2)}, \cdots, a_{mk}^{(\ell_m)}\right) + \\ &\quad \gamma \mathrm{E}_{s_{k+1}}\left[\bar{V}^{(0)}(s_{k+1})\right] \le \\ &\quad c_k\left(a_{1k}^{(0)}, a_{2k}^{(0)}, \cdots, a_{mk}^{(0)}\right) + \\ &\quad \gamma \mathrm{E}_{s_{k+1}}\left[\bar{V}^{(0)}(s_{k+1})\right] = \bar{V}^{(0)}(s_k) \end{aligned} \tag{28}$$

由此可得:

$$\bar{V}_{q+1}^{(1)}(s_k) \le \bar{V}^{(0)}(s_k) \tag{29}$$

受文献 [20] 启发, 根据式 (22), 定义如下:

$$L\left(\bar{V}(s_k)\right) = c_k\left(a_{1k}^{(1)}, a_{2k}^{(1)}, \cdots, a_{mk}^{(1)}\right) + \gamma \mathrm{E}_{s_{k+1}}[\bar{V}(s_{k+1})] \tag{30}$$

初始化 $\bar{V}_0^{(1)}(s_k) = \bar{V}^{(0)}(s_k)$, 由于算子 L 是一个收缩映射, 那么由式 (30) 生成的序列 $\{\bar{V}_{q+1}^{(1)}(s_k)\}$ 收敛到 $\bar{V}^{(1)}(s_k)$, 即:

$$\lim_{q\to\infty} \bar{V}_{q+1}^{(1)}(s_k) = \bar{V}^{(1)}(s_k) \tag{31}$$

由式 (29) 和式 (31), 可得:

$$\bar{V}^{(1)}(s_k) \le \bar{V}^{(0)}(s_k) \tag{32}$$

假设 $\bar{V}^{(j)}(s_k) \le \bar{V}^{(j-1)}(s_k)$, 类似式 (28) ~ 式 (32), 有:

$$\bar{V}^{(j+1)}(s_k) \le \bar{V}^{(j)}(s_k) \tag{33}$$

2) 根据现有自适应动态规划理论, 给定可允许的运行指标 $\tilde{a}_{ik}$ $(i=1, 2, \cdots, m)$, 则存在一个有界函数 $\bar{V}^{(j)}(s_k)$ 满足式 (17)[25−26]. 因此 $\bar{V}^{(j)} < \infty$.

3) 根据 1) 和 2), 有:

$$\lim_{j\to\infty} \bar{V}^{(j+1)}(s_k) = \bar{V}^{*}(s_k) \tag{34}$$

由式 (15)、式 (18) 和式 (34), 有 $\lim_{j\to\infty} a_{ik}^{(j+1)} = a_{ik}^{*}$. □

注 7. 通过引入时钟和定义其阈值, 执行策略异步更新, 运行指标最终收敛到问题 2 的最优解. 由于算法 1 本质上是强化学习方法, 因此称为策略异步更新强化学习算法.

注 8. 不同于现有的多控制策略同步更新强化学习算法[10−11, 15, 17−18], 本文不仅给出多个控制策略 (即运行指标) 异步更新算法, 并且基于随机最优控制

理论, 采用数学归纳法给出了算法收敛性的理论证明. 各运行指标分布地、异步地更新策略, 而不是集中[12–14, 25, 28]、同步更新方式[10–11, 15, 17–18], 其优势在于提高学习效率.

由式 (17) 和式 (18) 可知, 要实现运行指标自学习决策, 求解 $\bar{V}^{(j)}$ 是需要解决的关键问题. 但是在工业过程生产指标和运行指标动态未知、生产条件存在频繁波动的情况下, 如何求解 $\bar{V}^{(j)}$ 是一个难题. 下面将基于提出的算法 1, 在多执行−评判网络结构下提出数据驱动的运行指标自学习决策算法.

2.3　多执行−评判网络结构

采用评判网络和多执行网络结构估计值函数 $\bar{V}^{(j)}(s_k)$ 和运行指标 $a_{ik}^{(j+1)}$. 即基于神经网络估计方法[10, 20, 26–27], 给出神经网络结构的评判网络:

$$\hat{\bar{V}}^{(j)}(s_k) = (w_c^{(j)})^{\mathrm{T}}\sigma((v_c^{(j)})^{\mathrm{T}}s_k) \tag{35}$$

和多执行网络:

$$\hat{a}_i^{(j+1)}(k) = (w_i^{(j+1)})^{\mathrm{T}}\sigma((v_i^{(j+1)})^{\mathrm{T}}s_k),\quad i = 1, 2, \cdots, m \tag{36}$$

式中, $v_c^{(j)}$ 和 $w_c^{(j)}$ 分别是评判网络输入层到隐层的权值和隐层到输出层的权值, $v_i^{(j+1)}$ 和 $w_i^{(j+1)}$ 分别表示每个执行网络输入层到隐层的权值和隐层到输出层的权值. 由于

$$\begin{aligned}&\mathrm{E}_{s_{k+1}}[\bar{V}^{(j)}(s_{k+1})] \approx \\ &\quad\sum_{l=1}^{M} P(s_{l(k+1)}|s_k, a_k)\cdot\hat{\bar{V}}^{(j)}(s_{l(k+1)}) = \\ &\quad\sum_{l=1}^{M}\hat{\bar{V}}^{(j)}(s_{l(k+1)})\cdot\frac{n(s = s_{l(k+1)})}{N} = \\ &\quad\frac{1}{N}\sum\hat{\bar{V}}^{(j)}(s_{k+1})\end{aligned} \tag{37}$$

式中, $s_{l(k+1)}\,(l = 1, 2, \cdots, M)$ 表示在 $k+1$ 时刻随机变量 s 的可能取值, N 为样本数, $n(s = s_{l(k+1)})$ 表示 $s_{l(k+1)}$ 出现的次数. 对于所有 $s_k \in S_j$, 利用梯度下降方法, 有:

$$w_c^{(j)}(k+1) = w_c^{(j)}(k) - \eta_c\frac{\partial \mathrm{E}_c^{(j)}}{\partial w_c^{(j)}} \tag{38}$$

$$v_c^{(j)}(k+1) = v_c^{(j)}(k) - \eta_c\frac{\partial \mathrm{E}_c^{(j)}}{\partial v_c^{(j)}} \tag{39}$$

$$w_i^{(j+1)}(k+1) = w_i^{(j+1)}(k) - \eta_i\frac{\partial \mathrm{E}_i^{(j+1)}}{\partial w_i^{(j+1)}} \tag{40}$$

$$v_i^{(j+1)}(k+1) = v_i^{(j+1)}(k) - \eta_i\frac{\partial \mathrm{E}_i^{(j+1)}}{\partial v_i^{(j+1)}} \tag{41}$$

其中

$$\begin{aligned}&\mathrm{E}_c^{(j)} = \frac{1}{2}\left(e_c^{(j)}\right)^{\mathrm{T}}e_c^{(j)}\\ &e_c^{(j)} = \hat{\bar{V}}^{(j)}(s_k) - c_k\left(\hat{a}_1^{(j)}, \cdots, \hat{a}_m^{(j)}\right) - \\ &\qquad\gamma\frac{1}{N}\sum_{l=0}^{N}\hat{\bar{V}}^{(j)}\left(s_{l(k+1)}\right)\end{aligned} \tag{42}$$

$$\begin{aligned}&\mathrm{E}_i^{(j+1)} = \frac{1}{2}\left(e_i^{(j+1)}\right)^{\mathrm{T}}e_i^{(j+1)}\\ &e_i^{(j+1)} = \hat{a}_i^{(j+1)}(k) - a_i^{(j+1)}(k) = \\ &\qquad\hat{a}_i^{(j+1)}(k) - U_i\tanh\left[-\frac{\gamma}{2}(\bar{a}_iP_i)^{-\mathrm{T}}\cdot\right.\\ &\qquad\left.\frac{\partial s_{k+1}}{\partial a_i(k)}\cdot\frac{\partial\left(\frac{1}{N}\sum\limits_{l=0}^{N}\hat{\bar{V}}^{(j)}\left(s_{l(k+1)}\right)\right)}{\partial s_{k+1}}\right] - \bar{a}_i\end{aligned} \tag{43}$$

算法 2. 多执行−评判网络架构下的运行指标自学习决策算法

1) 初始化. 给出初始可允许运行指标 $a_{ik}^{(0)}$, 初始神经网络权值 $w_c^{(0)}$, $v_c^{(0)}$, $w_i^{(0)}$, $v_i^{(0)}$ $(i = 1, 2, \cdots, m)$, 令 $j = 0$.

2) 采集由 $a_{ik}^{(j)}$ 产生的包括运行指标、生产指标和生产条件的 N 组数据.

3) 评判神经网络训练: 执行式 (35)、式 (38)、式 (39) 和式 (42) 估计 $\hat{\bar{V}}^{(j)}(s_k)$.

4) 多执行网络异步更新: 运行时钟 t_{clock}, 当 $t_{clock} \le t_{threshold}$ 时, 执行式 (36)、式 (40)、式 (41) 和式 (43) 更新运行指标 $a_{ik}^{(j+1)}$; 当 $t_{clock} > t_{threshold}$ 停止更新. 如果某些执行网络的权值没有在 $t_{threshold}$ 时间内收敛到满意的误差范围内, 那么相应的运行指标 $\hat{a}_i^{(j+1)} = \hat{a}_i^{(j)}$.

5) 如果 $|\hat{\bar{V}}^{(j)}(s_k) - \hat{\bar{V}}^{(j-1)}(s_k)| \le \varepsilon$, 结束; 否则, $j = j+1$, 返回步骤 2).

算法 2 给出了具体的决策运行指标的程序. 为更清楚理解算法 2, 图 3 给出了算法 2 执行流程图.

注 9. S_j 表示 $a_{ik}^{(j)}$ 作用系统 (4) 后收集的系统状态构成的状态空间. 如果 $\bigcup_j S_j = S$, 那么由文献 [13–14], 在神经网络估计值函数 $\bar{V}^{(j)}$ 和运行指标 $a_{ik}^{(j+1)}$ 精确的情况下, 算法 2 能得到 $\bar{V}^*$ 和 a_{ik}^*, 即 $\lim_{j\to\infty}\bar{V}^{(j)} = \bar{V}^*$ 和 $\lim_{j\to\infty}a_{ik}^{(j+1)} = a_{ik}^*$.

注 10. 为保证 $\bigcup_j S_j = S$, 并且提高数学期望 $\mathrm{E}_{s_{k+1}}[\bar{V}^{(j+1)}]$ 的估计精度, 可采取的方法包括: 1) 在 $a_{ik}^{(j)}$ 中加入探测噪声, 以便满足系统持续激励条

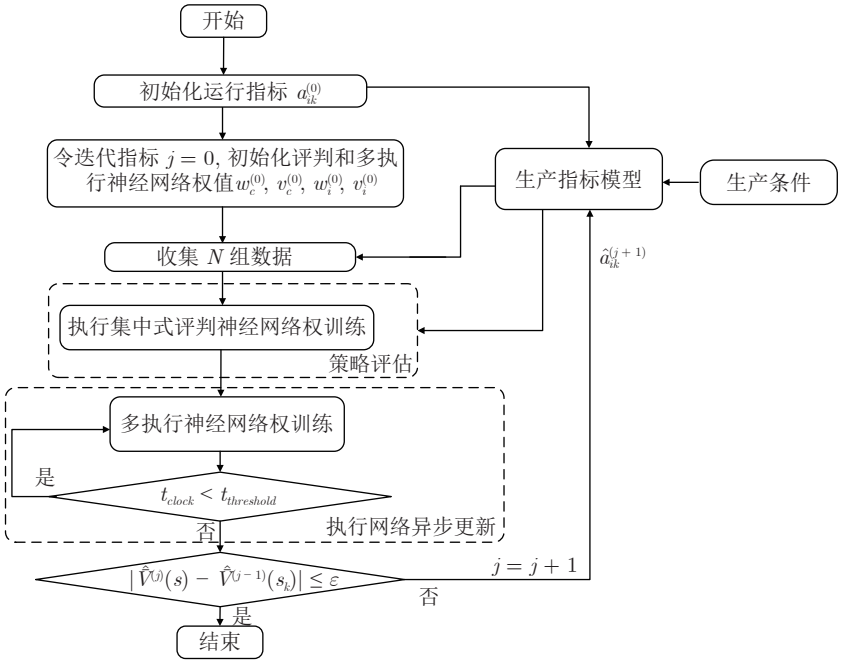

图 3 多执行−评判结构下运行指标自学习决策流程图

Fig. 3 Flowchart of self-learning decision making of operational indices with multiple actors-critic structure

件[10, 20, 29−30]; 2) 权衡计算负载和估计精度, 设定足够大的 N; 3) 采用蒙特卡洛方法产生随机生产条件. 此外, 选择适当的神经网络层数和神经元的个数, 或者采用经验回放方法利用历史数据分别计算误差 $e_c^{(j)}$ 和 $e_i^{(j+1)}$ 的均值, 提高评判网络和执行网络估计的精度和收敛速度[10, 20, 23, 31].

注 11. 现有的自适应动态规划方法, 通常忽视系统不确定性引发的状态不确定性. 文献 [20] 针对离散随机过程, 提出了强化学习方法, 用以学习最优控制策略, 但要求状态转移概率矩阵已知. 本文提出的方法无需计算状态转移概率矩阵, 通过计算样本均值可以计算 $\mathrm{E}_{s_{k+1}}\left[\bar{V}^{(j)}(s_{k+1})\right]$. 并且提出了策略异步更新强化学习方法, 用以提高学习效率. 此外, 本文提出的方法应用到工业过程生产指标优化问题, 给出了优化生产指标并控制运行指标在规定范围之内的运行指标自学习决策方法.

注 12. 为计算式 (43) 中 $\partial s_{k+1}/\partial a_i(k)$, 可以采用类似式 (35) 和式 (36) 的神经网络估计方法, 先估计生产指标动态 s_k, 然后再计算导数值.

注 13. 与经典的深度 Q 网络 (Deep Q network, DQN) 算法以及融合 DQN、执行−评判网络结构和策略梯度方法的多智能体深度确定性策略梯度 (Multi-agent deep deterministic policy gradient, MADDPG) 算法[32] 相比, 本文所提算法的不同之处在于: 1) 算法 2 中本文利用神经网络拟合的是值函数 $\bar{V}$, 而不是代替 Q 表的 Q 函数; 2) 算法 2 中多个执行网络异步更新, 而经典的 DQN 算法通常是根据估计的 Q 函数决定一个智能体的动作, MADDPG 算法往往是多执行网络同步更新. 本文多个控制策略异步更新避免了部分智能体神经网络估计控制策略用时过长, 提高学习效率, 并且给出了算法收敛性证明. 如何将所提方法扩展到 MADDPG 算法是未来拟研究的方向.

3 铁矿选矿生产指标优化试验

本节利用从中国西部某大型铁矿选矿厂获得的实际数据, 包括生产指标 (精矿产量和精矿品位)、7 个运行指标变量和 5 个生产条件变量, 开展本文提出的运行指标自学习决策算法的验证, 具体包括: 1) 实现生产指标优化, 即最大化精矿产量, 控制精矿品位在理想范围内, 并且运行指标限制在规定范围之内; 2) 学习效率和生产指标对比分析.

3.1 选矿过程描述及实验设置

如图 4 所示, 铁矿选矿由大量工序/设备组成, 包括筛分、竖炉焙烧、磨矿、低强度 (弱) 和高强度 (强) 磁选以及两个脱水单元[7, 10]. 本文主要关注两个生产指标, 即精矿产量 s_1 和精矿品位 s_2. 表 1 分别给出 7 个运行指标 a_1、a_2、a_3、a_4、a_5、a_6、a_7 的含义和需要满足的约束条件. 生产条件由 5 个变量组成, 可以增广为一个随机向量.

在本实验中, 采样周期为天和小时, 表示生产

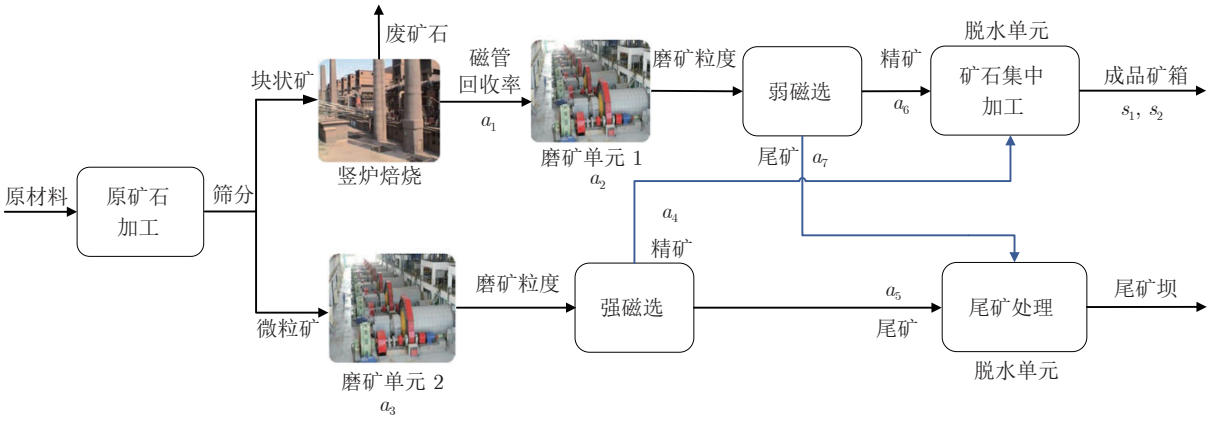

图 4　选矿过程流程图

Fig. 4　Flow chart of mineral separation process

表 1　运行指标

Table 1　Operational indices

单元	运行指标	取值范围 (%)
竖炉	a_1 : 磁管回收率	$a_{1\max} = 84.8$ $a_{1\min} = 81.3$
磨矿单元 1	a_2 : 磨矿粒度	$a_{2\max} = 84.0$ $a_{2\min} = 48.6$
磨矿单元 2	a_3 : 磨矿粒度	$a_{3\max} = 88.8$ $a_{3\min} = 63.3$
强磁选	a_4 : 精矿品位	$a_{4\max} = 53.4$ $a_{4\min} = 45.9$
	a_5 : 尾矿品位	$a_{5\max} = 23.2$ $a_{5\min} = 17.9$
弱磁选	a_6 : 精矿品位	$a_{6\max} = 57.8$ $a_{6\min} = 53.5$
	a_7 : 尾矿品位	$a_{7\max} = 20.2$ $a_{7\min} = 15.9$

指标和运行指标按天或小时来测量. 取精矿产量下限 $s_{1\min} = 260$ 吨/小时和 $s_{1\min} = 6\,000$ 吨/天, 精矿品位下限 $s_{2\min} = 53.5\%$, 精矿品位上限 $s_{2\max} = 54.5\%$. 本文通过 Matlab 软件实现算法. 假设收集到的铁矿石加工历史数据有足够的代表性, 可以用来反映真实生产过程. 现场收集的 532 个数据被分为两组, 分别用于生产指标动态神经网络的训练和验证. 精矿品位和精矿产量的动态模型均采用 16-16-1 的神经网络结构来估计, 损失函数定义为:

$$\frac{1}{m}\sum_{k=1}^{m}(s_{ik}-\hat{s}_{ik})^2, \qquad i=1,\,2 \tag{44}$$

式中, s_{ik} 为实际数据, $\hat{s}_{ik}$ 为神经网络估计值, m 为正整数. 图 5 给出了精矿产量和精矿品位的训练集与验证集的损失函数变化图. 由图 5 可以看出, 模型在验证集上的误差是随着训练集的误差下降而

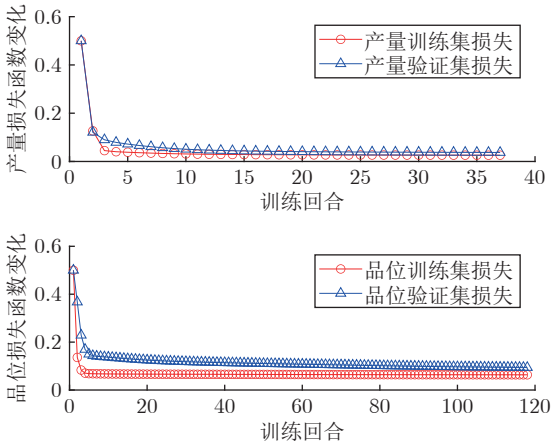

图 5　精矿产量和精矿品位损失函数

Fig. 5　Loss functions of the concentrate yield and concentrate grade

下降的, 表明本文训练得到的神经网络模型不存在过拟合或者欠拟合的现象.

对生产条件历史数据做统计分析, 生产条件向量近似服从高斯分布 $\mathrm{N}(\mu,\,\sigma^2)$, 均值 $\mu = [31.70\ 43.38\ 13.75\ 71.14\ 58.56]$, 方差 $\sigma^2 = [0.92\ 0.44\ 0.57\ 3.62\ 2.75]$.

3.2　算法验证和结果比较

用蒙特卡洛方法产生与历史数据同分布的生产条件, 评判神经网络采用 2-10-1 的结构, 7 个执行神经网络均采用 2-14-1 的结构. 神经网络的学习率为 0.05, 训练误差为 0.001, 取折扣因子 $\gamma = 0.8$. 执行算法 2, 图 6 和图 7 分别为执行网络和评判网络神经网络权学习过程. 图 8 为 200 天 7 个运行指标实验结果, 相应地图 9 和图 10 显示了 200 天精矿品位和精矿产量的实验结果. 图 8 表明采用所提算法 2, 运行指标限制在规定范围之内. 图 9 和图 10 表明精矿品位和精矿产量满足静态约束条件. 图 6 ~

图 10 表明了本文算法的有效性.

为验证本文算法的优势, 做了对比性实验. 表 2 为采用本文方法、文献 [11] 的多执行网络集成算法 (Multi-actor networks ensemble, MAE) 和文献 [33] 的 Reinforce 算法获得的精矿产量和实际精矿产量的对比性结果. 由表 2 和图 10可以看出, 本文方法得到的精矿产量高于实际生产精矿产量. 通过计算平均值, 本文算法 2 相比于实际精矿产量提高了约 1000 吨/天、40 吨/小时. 不同于文献 [11, 33], 本文优化目标为最大化累积产品产量, 不是单次采样时刻的产量, 单次采样时刻产量高不能保证累积时间内产量的最大化. 由表 2 可以看出, 相比于文献 [11] 算法, 本文算法提高 30 天 (按天采样) 和 1 天

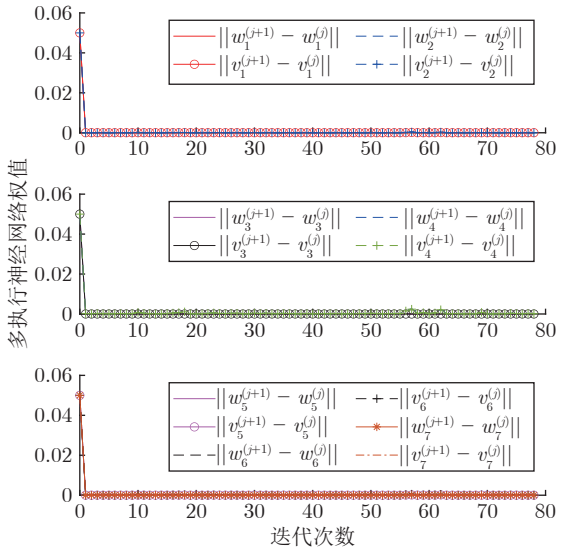

图 6 多执行神经网络权值

Fig. 6 Evolution of weights of multi-actor neural networks

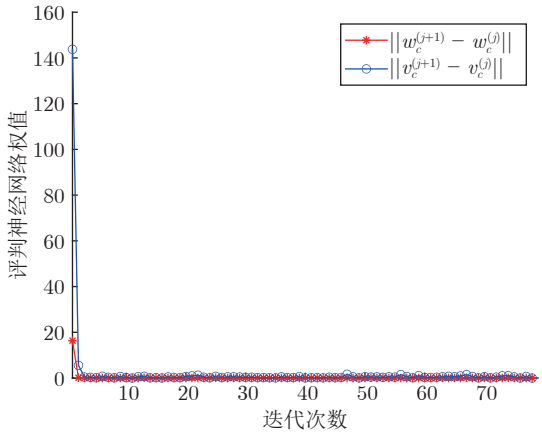

图 7 评判神经网络权值

Fig. 7 Evolution of weights of critic neural network

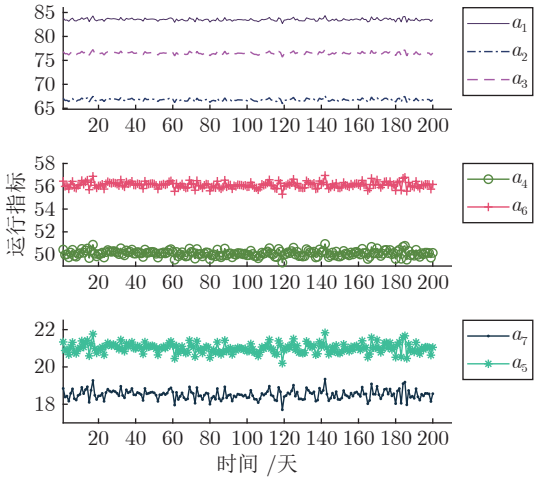

图 8 200 天的运行指标

Fig. 8 200-day operational indices

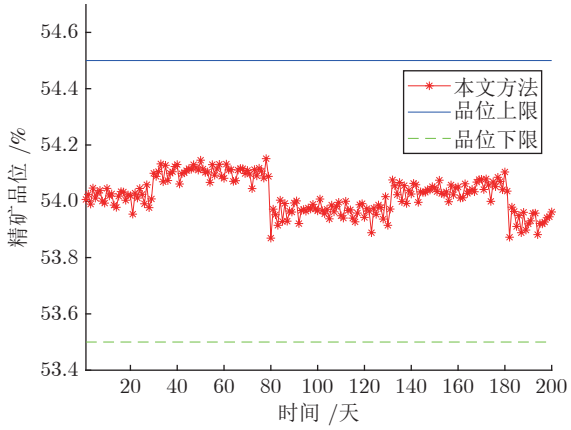

图 9 200 天的精矿品位

Fig. 9 200-day concentrate grade

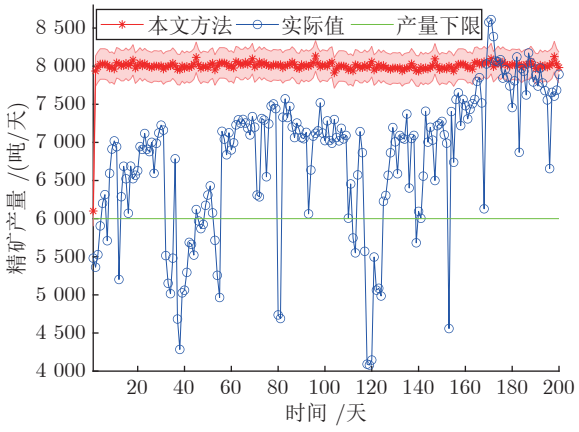

图 10 200 天的精矿产量

Fig. 10 200-day concentrate yield

(按小时采样) 精矿产量分别为 34 167.6 吨和 2 299.5 吨; 相比于文献 [33] 算法, 本文算法提高 30 天和 1 天

表 2　算法的实验结果对比

Table 2　Comparison results between different algorithms

实验	方法	产量 (吨)	品位 (%)
30 天	本文算法	240 369.8	54.13
	多执行网络集成算法[11]	206 202.2	54.10
	Reinforce[11, 33]	203 907.6	54.07
	实际值	199 650.6	52.86
1 天	本文算法	8 030.2	54.17
	多执行网络集成算法[11]	5 730.7	54.15
	Reinforce[11, 33]	5 648.3	52.58
	实际值	5 659.4	52.58

精矿产量分别为 36 462.2 吨和 2 381.9 吨. 执行类似文献 [10−11] 的策略同步更新强化学习算法, 图 11 显示了 10 次运行本文算法 2 和策略同步更新算法的时间消耗. 10 次实验中, 策略异步更新强化学习算法和策略同步更新强化学习算法平均每次执行时间分别为 4.83 秒与 7.80 秒, 表明了本文提出的策略异步更新算法提高了学习效率. 实际选矿过程生产条件动态变化, 针对如下三种生产条件变化均值相同 $\mu = [31.74\ 43.66\ 13.94\ 71.68\ 58.96]$, 不同方差:

工况 1: $\sigma_1^2 = [0.68\ 0.64\ 0.48\ 3.93\ 2.59]$

工况 2: $\sigma_2^2 = [2.68\ 1.67\ 2.44\ 5.79\ 5.42]$

工况 3: $\sigma_3^2 = [2.88\ 3.73\ 4.44\ 8.72\ 8.32]$

执行算法 2, 图 12 显示了考虑工况变化和不考虑工况变化统计结果对比. 结果表明: 未考虑工况变化, 没有根据工况的波动调节运行指标, 精矿产量变化比较平稳. 而本文算法能根据生产条件变化自适应调节运行指标, 优化精矿产量, 平均精矿产量高于同种工况下的未考虑工况变化的值.

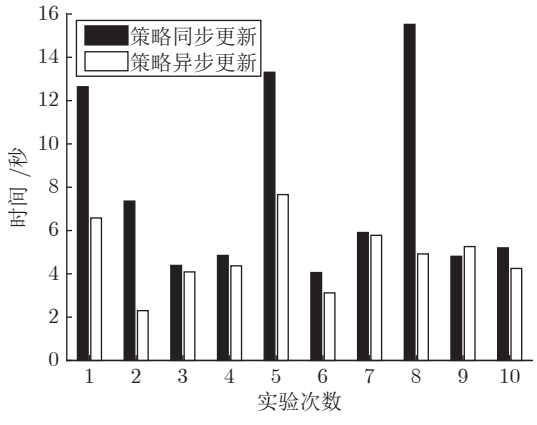

图 11　策略异步更新和策略同步更新强化学习算法时间消耗对比

Fig. 11　Comparison of time consumption between asynchronous policy update and synchronous policy update

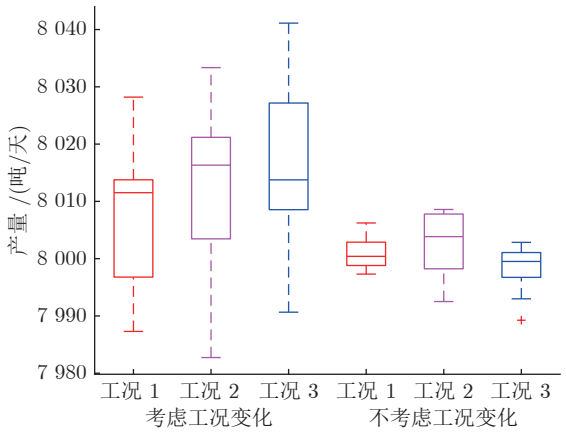

图 12　考虑工况变化和不考虑工况变化统计结果对比

Fig. 12　Statistic results with and without consideration of dynamics of production condition

4　结束语

本文针对不确定工业过程运行指标决策问题, 基于自适应动态规划技术, 提出了一种数据驱动的策略异步更新强化学习算法, 决策运行指标, 并给出了算法收敛性的理论证明. 该算法不要求状态转移概率矩阵已知, 利用样本均值代替计算生产指标状态转移概率矩阵, 采用集中式性能评估和多策略异步更新方式, 利用可测量数据, 自学习决策运行指标. 该算法提高了分布式强化学习的学习效率, 实现了生产条件动态波动环境下, 工业过程生产指标优化并且保证运行指标和生产指标在规定范围之内. 仿真实验验证了方法的有效性.

References

1　Chai Tian-You. Challenges of optimal control for plant-wide production processes in terms of control and optimization theories. *Acta Automatica Sinica*, 2009, **35**(6): 641−649
(柴天佑. 生产制造全流程优化控制对控制与优化理论方法的挑战. 自动化学报, 2009, **35**(6): 641−649)

2　Ding Jin-Liang, Yang Cui-E, Chen Yuan-Dong, Chai Tian-You. Research progress and prospects of intelligent optimization decision making in complex industrial process. *Acta Automatica Sinica*, 2018, **44**(11): 1931−1943
(丁进良, 杨翠娥, 陈远东, 柴天佑. 复杂工业过程智能优化决策系统的现状与展望. 自动化学报, 2018, **44**(11): 1931−1943)

3　Chai Tian-You, Ding Jin-Liang, Wang Hong, Su Chun-Yi. Hybrid intelligent optimal control method for operation of complex industrial processes. *Acta Automatica Sinica*, 2008, **34**(5): 505−515
(柴天佑, 丁进良, 王宏, 苏春翌. 复杂工业过程运行的混合智能优化控制方法. 自动化学报, 2008, **34**(5): 505−515)

4　Huang X, Chu Y, Hu Y, Chai T. Production process management system for production indices optimization of mineral processing. *IFAC Proceedings Volumes*, 2005, **38**(1): 178−183

5　Ochoa S, Wozny G, Repke J U. Plantwide optimizing control of a continuous bioethanol production process. *Journal of Process Control*, 2010, **20**(9): 983−998

6　Ding J, Chai T, Wang H, Wang J, Zheng X. An intelligent factory-wide optimal operation system for continuous production process. *Enterprise Information Systems*, 2016, **10**(3): 286−302

7 Ding J, Modares H, Chai T, Lewis F L. Data-based multi-objective plant-wide performance optimization of industrial processes under dynamic environments. *IEEE Transactions on Industrial Informatics*, 2016, **12**(2): 454−465

8 Chai T, Ding J, Wang H. Multi-objective hybrid intelligent optimization of operational indices for industrial processes and application. *IFAC Proceedings Volumes*, 2011, **44**(1): 10517−10522

9 Ding J, Yang C, Chai T. Recent progress on data-based optimization for mineral processing plants. *Engineering*, 2017, **3**(2): 183−187

10 Li J, Ding J, Chai T, Lewis F L. Nonzero-sum game reinforcement learning for performance optimization in large-scale industrial processes. *IEEE Transactions on Cybernetics*, 2019, **50**(9): 4132−4145

11 Liu C, Ding J, Sun J. Reinforcement learning based decision making of operational indices in process industry under changing environment. *IEEE Transactions on Industrial Informatics*, 2021, **17**(4): 2727−2736

12 Lewis F L, Vrabie D, Vamvoudakis K. Reinforcement learning and feedback control. *IEEE Control Systems*, 2012, **32**(6): 76−105

13 Bertsekas D P, Tsitsiklis J N. *Neuro-Dynamic Programming*. Nashua: Athena Scientific, 1996.

14 Bertsekas D P. Proper policies in infinite-state stochastic shortest path problems. *IEEE Transactions on Automatic Control*, 2018, **63**(11): 3787−3792

15 Liu D, Wang D, Li H. Decentralized stabilization for a class of continuous-time nonlinear interconnected systems using online learning optimal control approach. *IEEE Transactions on Neural Networks and Learning Systems*, 2013, **25**(2): 418−428

16 Na J, Zhao J, Gao G B, Li Z C. Output-feedback robust control of uncertain systems via online data-driven learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2020, **32**(6): 2650−2662

17 Song R, Lewis F L, Wei Q. Off-policy integral reinforcement learning method to solve nonlinear continuous-time multi-player nonzero-sum games. *IEEE Transactions on Neural Networks and Learning Systems*, 2016, **28**(3): 704−713

18 Modares H, Nageshrao S P, Lopes G A D, Babuska R, Lewis F L. Optimal model-free output synchronization of heterogeneous systems using off-policy reinforcement learning. *Automatica*, 2016, **71**: 334−341

19 Bertsekas D P. Multiagent reinforcement learning: Rollout and policy iteration. *IEEE/CAA Journal of Automatica Sinica*, 2021, **8**(2): 249−272

20 Liang M, Wang D, Liu D. Neuro-optimal control for discrete stochastic processes via a novel policy iteration algorithm. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019, **50**(11): 3972−3985

21 Zhang H, Luo Y, Liu D. Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints. *IEEE Transactions on Neural Networks*, 2009, **20**(9): 1490−1503

22 Marvi Z, Kiumarsi B. Safe reinforcement learning: A control barrier function optimization approach. *International Journal of Ro-bust and Nonlinear Control*, 2021, **31**(6): 1923−1940

23 Greene M L, Deptula P, Nivison S, Dixon W E. Sparse learning-based approximate dynamic programming with barrier constraints. *IEEE Control Systems Letters*, 2020, **4**(3): 743−748

24 Bellman R, Åström K J. On structural identifiability. *Mathematical Biosciences*, 1970, **7**(3−4): 329−339

25 Luo B, Yang Y, Liu D. Policy iteration Q-learning for data-based two-player zero-sum game of linear discrete-time systems. *IEEE Transactions on Cybernetics*, 2021, **51**(7): 3630−3640

26 Kiumarsi B, Lewis F L. Actor-critic-based optimal tracking for partially unknown nonlinear discrete-time systems. *IEEE Transactions on Neural Networks and Learning Systems*, 2014, **26**(1): 140−151

27 Zhang R, Tao J. Data-driven modeling using improved multi-objective optimization based neural network for coke furnace system. *IEEE Transactions on Industrial Electronics*, 2017, **64**(4): 3147−3155

28 Wang D, Ha M, Qiao J. Self-learning optimal regulation for discrete-time nonlinear systems under event-driven formulation. *IEEE Transactions on Automatic Control*, 2020, **65**(3): 1272−1279

29 Lewis F L, Liu D. *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*. New York: John Wiley & Sons, 2013.

30 Li J, Ding J, Chai T, Lewis F L, Jagannathan S. Adaptive interleaved reinforcement learning: Robust stability of affine nonlinear systems with unknown uncertainty. *IEEE Transactions on Neural Networks and Learning Systems*, 2022, **33**(1): 270−280

31 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
(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)

32 Lowe R, Wu Y, Tamar A, Harb J, Abbeel P, Mordatch I. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in Neural Information Processing Systems*, 2017: 6379−6390

33 Sutton R S, Barto A G. *Reinforcement Learning: An Introduction*. Cambridge: MIT press, 2018.

李金娜 辽宁石油化工大学教授. 主要研究方向为运行优化控制, 数据驱动控制, 强化学习和多智能体优化控制. 本文通信作者.
E-mail: lijinna_721@126.com
(**LI Jin-Na** Professor at Liaoning Petrochemical University. Her research interest covers optimal operational control, data-driven control, reinforcement learning, and optimal control of multi-agent systems. Corresponding author of this paper.)

袁 林 辽宁石油化工大学硕士研究生. 主要研究方向为运行优化控制, 数据驱动控制和强化学习.
E-mail: lewinyuan@126.com
(**YUAN Lin** Master student at Liaoning Petrochemical University. His research interest covers optimal operational control, data-driven control, and reinforcement learning.)

丁进良 东北大学教授. 主要研究方向为生产全流程运行优化, 智能优化, 神经网络和强化学习.
E-mail: jlding@mail.neu.edu.cn
(**DING Jin-Liang** Professor at Northeastern University. His research interest covers optimization of the whole production process, intelligent optimization, neural networks, and reinforcement learning.)