控制与决策
Control and Decision

基于深度强化学习的模糊作业车间调度问题

朱家政, 张宏立, 王聪, 李新凯, 董颖超

引用本文:
朱家政,张宏立,王聪,李新凯,董颖超. 基于深度强化学习的模糊作业车间调度问题[J]. 控制与决策, 2024, 39(2): 595−603.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2022.1345

基于近端强化学习的股价预测方法

Method of stock prices forecast based on proximal reinforcement learning

控制与决策. 2021, 36(4): 967−973 https://doi.org/10.13195/j.kzyjc.2019.1245

基于深度强化学习的模糊作业车间调度问题

朱家政, 张宏立†, 王 聪, 李新凯, 董颖超

(新疆大学 电气工程学院，乌鲁木齐 830047)

摘 要: 针对具有模糊加工时间和模糊交货期的作业车间调度问题,以最小化最大完工时间为目标,以近端策略优化(PPO)算法为基本优化框架,提出一种LSTM-PPO (proximal policy optimization with Long short-term memory)算法进行求解. 首先,设计一种新的状态特征对调度问题进行建模,并且依据建模后的状态特征直接对工件工序进行选取,更加贴近实际环境下的调度决策过程;其次,将长短期记忆(LSTM)网络应用于PPO算法的行动者-评论者框架中,以解决传统模型在问题规模发生变化时难以扩展的问题,使智能体能够在工件、工序、机器数目发生变化时,仍然能够获得最终的调度解. 在所选取的模糊作业车间调度的问题集上,通过实验验证了该算法能够取得更好的性能.

关键词: 深度学习；强化学习；近端策略优化算法；模糊作业车间调度

中图分类号: TP18 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2022.1345

引用格式: 朱家政,张宏立,王聪,等. 基于深度强化学习的模糊作业车间调度问题[J]. 控制与决策, 2024, 39(2): 595-603.

Fuzzy job shop scheduling problem based on deep reinforcement learning

ZHU Jia-zheng, ZHANG Hong-li†, WANG Cong, LI Xin-kai, DONG Ying-chao

(College of Electrical Engineering，Xinjiang University，Urumqi 830047，China)

Abstract: For the job shop scheduling problem with fuzzy processing time and fuzzy delivery time, this paper uses the proximal policy optimization (PPO) algorithm as the basic optimization framework with the objective of minimizing the maximum completion time. An LSTM-PPO (proximal policy optimization with long short-term memory) algorithm is proposed to solve the problem. Firstly, a new state feature is designed to model the scheduling problem, and the process is selected directly based on the modeled state feature, which is closer to the actual scheduling decision process. Them, the long short-term memory (LSTM) network is applied to the actor-commentator framework of the PPO algorithm, which solves the problem that the traditional model is difficult to scale up when the problem size changes, and enables the intelligent body to obtain the final scheduling solution even when the number of workpieces, processes, and machines changes. On the selected problem set of fuzzy job shop scheduling, it is experimentally verified that the algorithm can achieve better performance.

Keywords: deep learning；reinforcement learning；proximal policy optimization；fuzzy job shop scheduling

0 引 言

作业车间调度问题(job shop scheduling problems, JSSP)是一类典型的NP-hard问题[1],及时地解决现实生活中的作业车间调度问题在工业、管理和经济等领域十分必要. 而在实际的生产环境中,工序具有明确处理时间的情况往往是不切实际的,能够考虑到人为因素造成的模糊处理时间和模糊交货期的作业车间调度问题会更合适,因此模糊作业车间调度问题(fuzzy job shop scheduling problems, FJSSP)得到了广泛的关注[2].

针对求解FJSSP的历史可以追溯到90年代中期,分支定界法被应用于解决小规模的FJSS[3],然而这种精确算法在进行大规模问题求解时会造成极大的计算负担,而启发式和元启发式算法会有较好的表现. 文献[4]提出了一种结合化学反应优化和禁忌搜索的混合算法求解较大规模的FJSSP. 文献[5]采用

收稿日期: 2022-07-27；**录用日期:** 2022-09-21.

基金项目: 国家自然科学基金项目(51967019, 52065064).

责任编委: 王凌.

†通讯作者. E-mail: zhlxju@163.com.

*本文附带电子附录文件,可登录本刊官网该文“资源附件”区自行下载阅览.

了基于工序的编码并设计了一种修补方式使混沌乌鸦搜索算法能够有效地求解FJSSP. 文献[6]提出了一种混合离散果蝇优化算法求解工序加工时间为区间数的分布式调度问题. 文献[7]提出了一种随机密钥遗传算法,通过使用随机密钥表示法的解码策略提高了求解效率. 文献[8]提出了一种新的选择机制以提升差分进化算法求解此类问题的性能. 但是面对情况复杂的实际生产环境,启发式和元启发式算法仍有不足之处.

通过使用深度网络训练的强化学习能够适应复杂的生产状况,生成适合的调度策略. 文献[9]提出了一种基于深度强化学习和迭代贪婪算法的框架用以求解流水车间调度问题. 目前完全基于深度强化学习的调度方法大都使用深度Q网络. 文献[10]提出了一种深度强化学习框架,根据当前输入的生产状态学习调度策略求解JSSP. 文献[11]提出了一种基于时序差分法的深度强化学习,将启发式算法或分配规则作为调度决策的候选行为,解决了非置换流水车间调度问题,但这种使用深度Q网络只能逼近最优动作-价值对函数,并不能直接对调度策略进行优化. 因此,基于策略的PPO算法得到了广泛应用,文献[12-13]使用了适合离散动作空间的PPO算法直接对工序进行选择,验证了求解车间调度问题的优越性,但并未在PPO算法中使用更先进的深度网络对行动者-评论者网络模型进行改进,提高算法的性能.

本文在PPO算法的基础上,考虑调度过程中生产状态变化的复杂性和实时性,同一工件的不同工序具有较强的时序关联. 因此,将在时间序列预测中表现优越的LSTM网络同PPO算法相结合,增强了PPO算法的特征提取能力,针对不同规模的调度问题最终能够获得优良的调度解.

1 问题描述

1.1 问题定义

通常模糊作业车间调度问题被描述为: 在m台机器$M_k(k=1,2,\dots,m)$加工n个具有模糊加工时间的工件$J_i(i=1,2,\dots,n)$,每个工件J_i的第i道工序表示为O_{ij}. 不同的假设和约束条件会导致不同的模糊作业车间调度问题,本文做出如下假设: 1) 每个工件仅能被每台机器加工一次; 2) 每台机器在任意时刻仅能加工一道工序,每个工件也仅能被一台机器加工; 3) 工件的每道工序的加工仅能在上道工序完成后开始.

问题的目标为最小化最大模糊完工时间C,其数学模型如下所示:

$$\min\ C=\min\Big[\max\Big(\sum_{i=1}^{N}C_{ik}\Big)\Big]. \tag{1}$$

$$\text{s.t. } C_{ik}-P_{ik}+M(1-a_{ihk})\geqslant C_{ik}; \tag{2}$$

$$C_{jk}-C_{ik}+M(1-x_{ijk})\geqslant P_{ik}; \tag{3}$$

$$C_{ik}\geqslant 0; \tag{4}$$

$$a_{ihk}=\begin{cases}1,\ \text{工件}\,h\,\text{先于机器}\,k\,\text{加工}\,i;\\0,\ \text{otherwise};\end{cases} \tag{5}$$

$$x_{ihk}=\begin{cases}1,\ \text{工件}\,i\,\text{先于工件}\,k\,\text{在机器}\,j\,\text{加工};\\0,\ \text{otherwise}.\end{cases} \tag{6}$$

其中: 式(1)为目标函数,式(2)为每个工件的工序约束,式(3)为每个工件的机器约束,C_{ik}为工件i在机器k的模糊完工时间,P_{ik}为模糊起始时间,M为一个足够大的整数,a_{ijk}和x_{ijk}为两个变量.

1.2 模糊数操作

求解模糊作业车间调度问题的关键点是在模糊加工时间的基础上求得模糊完成时间,因此对于模糊数的处理至关重要. 模糊加工时间的运算包括: 求和、取大和比较. 求和运算用于求取工件加工工序的模糊完成时间; 取大运算用于确定工件下一工序的模糊开始时间. 在本文中,将使用三角模糊数表示模糊加工时间,因此,在经过求和和取大运算后的每个工件的开始时间以及完成时间也同样使用三角模糊数表示. 对于模糊加工时间的运算操作定义与文献[4]相同.

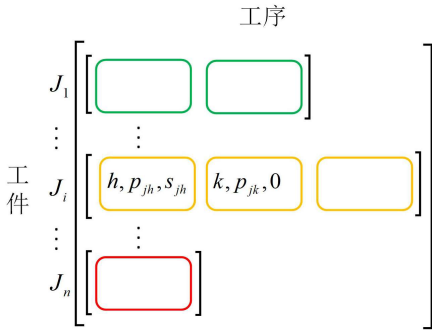
2 基于LSTM-PPO的模糊车间调度问题求解

2.1 调度问题转化

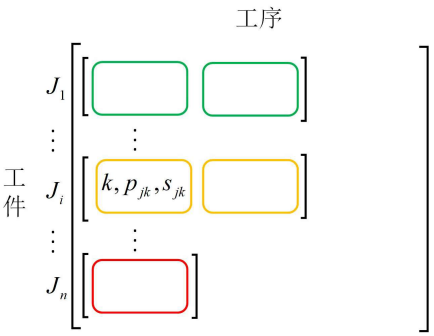
2.1.1 状态特征

状态特征描述了调度环境的全局特征和局部特征,状态空间是一个所有可能状态的集合,智能体需要做出决定的状态被称为决策状态. 对于具有小规模状态空间的问题,每个状态和状态-行动对都可以明确地表示为数组或表格. 在这种情况下,强化学习被称为表格式强化学习. 然而,实际问题中的状态空间往往过大,使用表格式的强化学习算法并不能遍历状态空间中的所有状态,因此表格式强化学习算法将出现“维度灾难”问题. 一个常用的解决方法是泛化价值函数,它可以从过去的不同状态中概括出与当前状态的近似,从而减小状态空间. 根据调度任务的特点,概括能够描述调度问题的特征属性,用于构建强化学习的状态特征.

为了充分地利用深度强化学习从原始输入中的特征提取能力,本文将状态设置为包含所有工件的列表,其中每个工件又是由所有工序组成的列表,如图1(a)所示. 其中p_{jh}表示为工件j的第h道工序的处理时间. 如果一个工序(j,k)是工件下一个加工的工序,则将其最早可能开始的加工时间记作$s_{jk}=\max(C_{ik},C_{jh})$,$C_{ik}$为当前加工工序所选择的加工机器的完工时间,$C_{jh}$为所选工件上一加工工序的完成时间. 在每个决策时间t,都会在某一确定的开始时间采取动作a_t,并选择相对应的工序放入可调度工序的集合中. 该集合A_{st}包含了在状态s_t下所有可选择的动作. 在当前决策时间t从可调度工序集中选取工件J_i的调度流程如图1所示.



(a) 状态s_t



(b) 状态s_t+1

图1　状态转换过程

通过如下所示的一个3×3的FJSSP展示其具体的状态转换过程:

工件1: 机器2 (2,3,4),机器1 (1,3,5),机器3 (3,4,5);

工件2: 机器1 (3,5,7),机器3 (2,6,8),机器2 (1,2,4);

工件3: 机器3 (1,2,3),机器2 (3,6,9),机器1 (2,5,6).

第1行表示工件1需要在机器2、1、3上相继加工,括号内的数字为工序对应的模糊加工时间. 剩余两行与之相似. 该调度问题的状态转换过程如图2所示,表示了调度环境从初始状态S_0开始,经过3次调度操作后,调度环境转化为S_3. 调度环境每次调度后,都会将已完成的操作从状态列表中移除,并对下一工序的开始时间进行更新.

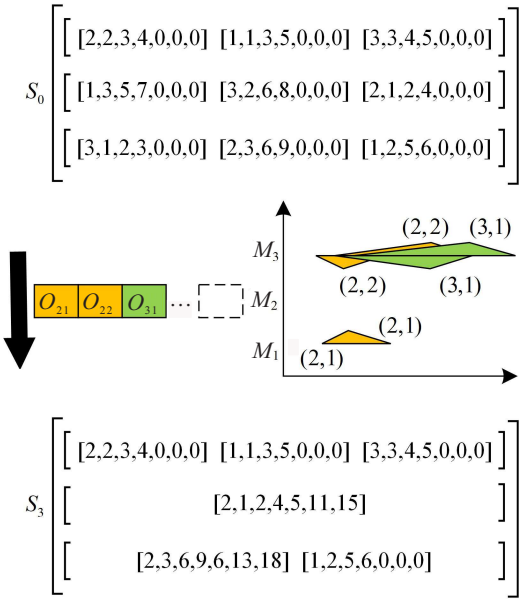


图2　实例状态转换过程

2.1.2　奖励函数

奖励函数能够引导神经网络对所获得的状态特征进行深加工,决定了智能体能否学到预期策略,并直接影响算法的收敛速度和最终性能. 针对以最小化最大模糊完工时间为调度目标的FJSSP,为了与调度问题的求解目标联系更加密切,将单步所获得的奖励设置为:在当前决策时间t,当前加工工件的完工时间C_{jk}与所有工件最大完工时间$C_{\min_t}$的差值负相关. 奖励函数定义为

$$r_{t+1}=\begin{cases}-(C_{jk}-C_{\min_t}), & C_{jk}>C_{\min};\\ 0, & \text{otherwise.}\end{cases}\tag{7}$$

通过奖励函数(7)的设计,可以得到$C_{\max}=-\sum_{t=1}^{k}r_t$.

2.2　LSTM-PPO算法模型

本节对LSTM-PPO算法中使用的神经网络模型进行介绍. LSTM-PPO算法是一种与PPO算法相同的行动者-评论者模式的算法,因此,需要对此模式算法的两个网络进行指定:第1个是行动者网络,用以产生当前状态的行为策略$\pi_\theta(\cdot,s_t)$,第2个是评论者网络,对当前所产生的行为策略进行评价,提供状态-价值函数$V_\omega(s_t)$的估值. 根据前文所定义的状态,在解决不同规模的模糊作业车间调度问题时的难点主要是状态列表在调度过程中其长度不固定,每个决策点的状态列表长度各不相同. 因此,本文选择在处理可变长度序列中表现较好的长短期记忆网络. LSTM是一种递归神经网络,能够有效解决长序列数据处理中出现的自循环梯度爆炸或者自循环梯度消

失问题. 自循环由能够调整信息流的网络所控制,它们将序列作为输入,并返回相同长度的嵌入序列.

图 3 所示为LSTM-PPO算法的网络模型,调度问题所构成的状态输入为图中的三维数据立方体,第1维度为工件数量,第2维度为工序数量,第3维度为工序的相关特征. 将其构造为LSTM网络能够识别的输入形式,得到状态输入列表,并将其作为输入传入到嵌入层中,为每个工件的工序特征产生一个嵌入 y_{ij},将 y_{ij} 和LSTM网络的初始化参数 h_0 输入至第1个LSTM网络中. 每个工件中的工序在时序上都是相继完成的,所以能够将工件的最后一道工序被视为整个工件的压缩表示,因此只考虑每个工件在当前决策时刻最后一道工序的嵌入,并将每个工件得到的工序嵌入组成一个长度为 $|J|$ 的列表作为第2个LSTM的输入. 第2个LSTM网络将输入的工件信息合并为长度为 $|J|$ 的嵌入,得到的输出为 $Z \in \boldsymbol{R}^J$ 的一个向量. 在训练过程中,行动者网络和评论者网络模型共享网络参数,因此第2个LSTM网络的输出 $Z \in \boldsymbol{R}^J$ 由两个网络模型所共用. 在行动者网络模型中,通过结合动作屏蔽机制和Softmax函数,使用全为布尔值的动作屏蔽向量 M 对当前决策时刻状态下的无效动作进行屏蔽过滤,避免算法难以收敛、陷入局部最优等问题的出现. Softmax函数则是将输出嵌入 Z 转化为当前可选工序任务的概率分布,具体公式为

$$\sigma(y,M)_i=\frac{e^{Z_i}M_i}{\sum\limits_{j=1}^{|J|}e^{y_i}M_j},\ i=1,2,\ldots,|J|. \tag{8}$$

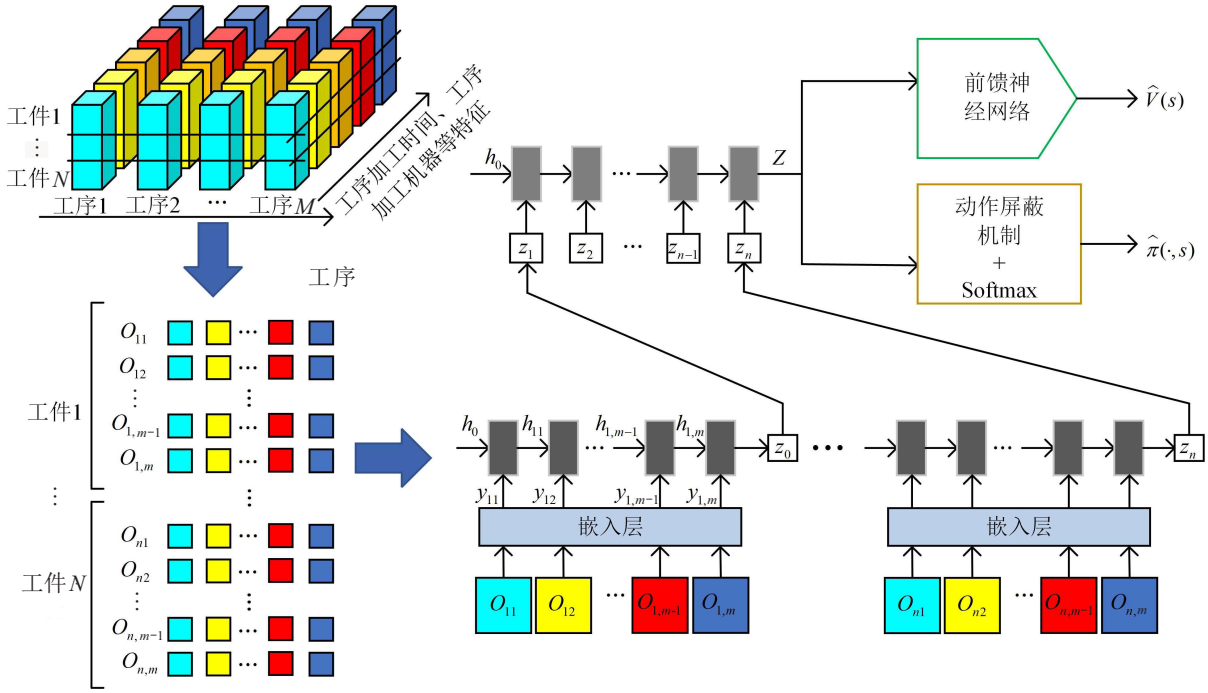


图 3 LSTM-PPO算法网络模型

在评论者网络模型中,向量 Z 作为输入进入前馈神经网络,经过前馈神经网络处理将 Z 收敛为一个标量. 前馈神经网络具有3个全连接的隐藏层,神经元的数量不断减少,使用RELU函数作为它的激活函数,直到最后一层都是线性层,最后评论者网络输出策略评价估值.

LSTM-PPO算法的训练流程如图4所示,详细步骤如下:

step 1: 初始化行动者网络和评价者网络参数,EPSIODE ∈ [0, MAXEPSIODE]为算法训练次数.

step 2: 将调度问题转化为训练智能体需要的状态列表,构造状态特征列表 s_0.

step 3: 将状态特征列表输入到行动者网络,计算当前状态下的动作概率分布 $\log P(a_0|s_0)$,根据概率采样选择确定动作 a_0 返回至环境中,计算奖励 r_1 并更新当前状态特征列表得到 s_1. 通过评价者网络计算状态价值 $v(s_0)$,将本次产生的经验 $[s_0,a_0,r_1,v(s_0),\log P(a_0|s_0)]$ 存放至经验池中.

step 4: 是否达到经验池容量 N,若未达到则转至step 3继续进行轨迹采样.

step 5: 读取经验池中保存的轨迹,依据下式计算优势函数:

$$\hat{A}_t=\delta_t+(\gamma\lambda)\delta_t+\ldots+(\gamma\lambda)^{T-t+1}\delta_{T-1},$$
$$\delta_t=r_t+\gamma v(s_{t+1})-v(s_t). \tag{9}$$

其中: γ、λ为折扣因子, $v(s_t)$为在t时刻状态s的状态价值.

step 6: 从经验池采集数据进行评价者网络更新, 利用下式计算折扣回报和损失函数:

$$G_t = (1+\gamma+\ldots+\gamma^{T-t})r_{t+1} + \gamma^{T+1-t}v(s_{t+1}), \tag{10}$$

$$\text{closs} = \text{mean}(G - v), \tag{11}$$

然后反向传播更新评价者网络.

step 7: 从经验池采集数据输入至行动者网络中计算动作概率分布$\log P_{\text{new}}(a_0|s_0)$.

$$\text{ratio}(\theta) = \frac{\log P_{\text{new}}(a_0|s_0)}{\log P(a_0|s_0)}, \tag{12}$$

$$\text{aloss} = \hat{\boldsymbol{E}}_t[\min(\text{ratio}(\theta)\hat{A}_t, \text{clip}(\text{ratio}_t(\theta), 1-\varepsilon, 1+\varepsilon)\hat{A}_t)]. \tag{13}$$

式(12)为重要性采样因子更新公式, 利用式(12)计算其与经验池中存储的$\log P(a_0|s_0)$的ratio; 利用式(13)计算损失函数. 然后反向传播更新行动者网络, 其中ε为裁剪系数.

step 8: 判断是否达到经验池中的数据复用次数M, 若未达到, 则转至step 5; 否则判断是否达到训练次数, 若未达到, 则转至step 2, 否则结束算法训练.

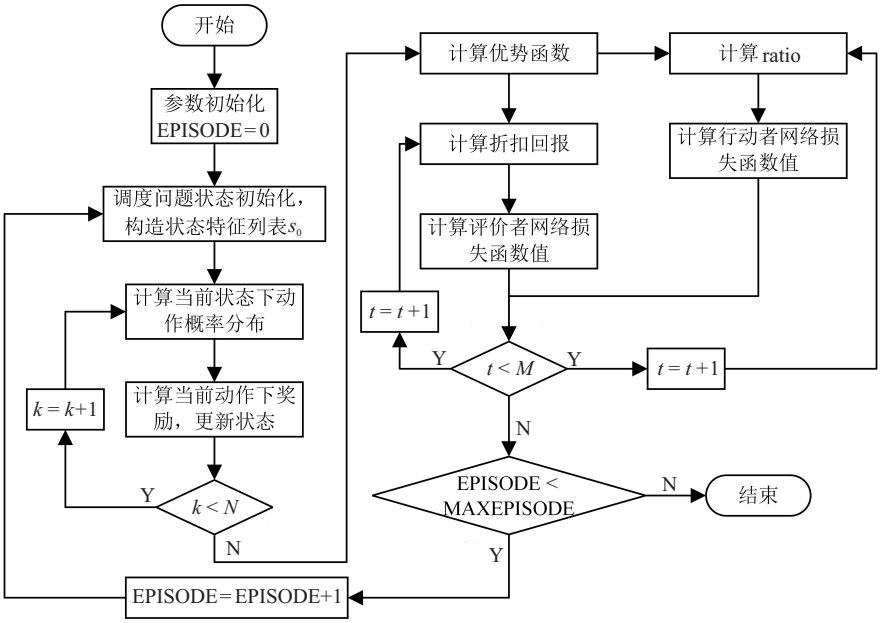


图 4　算法训练流程

3　实例验证

3.1　实例环境及参数设置

实验使用Python编程语言, 在Windows11系统上运行Pycharm进行实现. 硬件条件为AMD Ryzen 5 4600 H 3.00 GHz, 16 GB RAM.

带有截止日期的实验数据集来源于文献[8], 其中包含4个6×6以及4个10×10的实例, 本文利用实例1～实例8进行表示. 通过与文献[8]的实验结果对比, 以评价本文所提出算法的优化性能.

在深度强化学习算法中, 超参数的取值对于算法的性能影响较大. 为了确定这些超参数的取值, 考虑不同学习率、数据复用次数、裁剪系数、采样步数、批次尺寸等因素对实例1的验证结果进行了分析, 研究算法对各个超参数的敏感性. 如图5所示, 坐标系的x和y分别表示训练次数和最小化最大完工时间. 其中浅色区域部分为算法实际训练数据绘制的曲线, 为方便比较, 对曲线进行平滑处理, 即深色实线部分, 后续图中皆进行相同处理方式.

在图5(a)中, 进行了4种不同学习率的比较, 过大或过小的学习率都会使算法的性能下降, 不能很好地对实例进行求解.

图5(b)中进行了数据复用次数的比较, 数据复用次数是指在智能体进行学习更新时, 经验池中保留的每个样本需要使用多少次, 可以看出不同数值下的调度结果的差距不大, 其数值为10时表现最优.

由于LSTM-PPO算法只在信任域内进行更新, 裁剪系数能够对算法中估计优势的函数进行裁剪, 其数值越小, 表示信任域越窄, 策略更新越谨慎. 如图5(c)所示, 其值在0. 3时表现效果最好, 即使继续增大, 也无法提升算法的性能.

采样步数是指算法在单轮更新中, 智能体同环境交互收集数 据的条数. 交互完成后, 智 能体使用所收

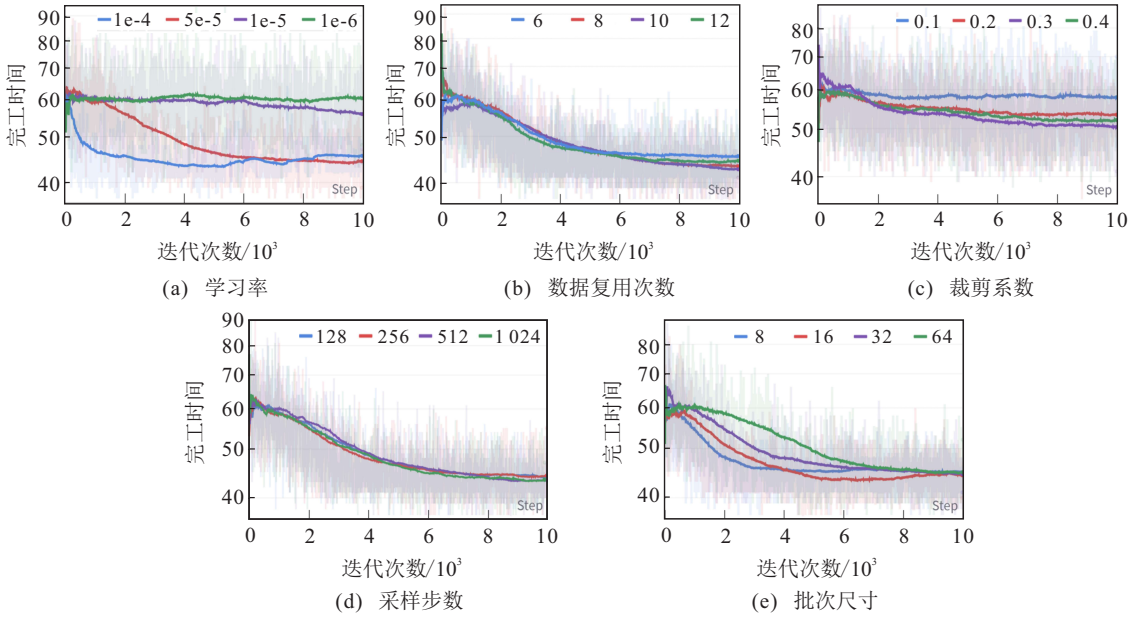


图 5 实例 1 中超参数比较

集的数据训练网络. 从图 5(d) 中可以得出, 当采样步数增大时, 只是会降低数据的利用效率, 使训练的过程变慢, 对算法性能影响较小.

批次尺寸是在每次网络训练中, 从采样步数所收集数据的总条数中单次抽取的数目. 如图 5(e) 所示, 不同的取值会影响算法的收敛速度, 对算法性能影响较小, 选取图中曲线较为平滑的 16 作为该值的取值.

通过超参数的比较可以发现, 在 LSTM-PPO 算法中对其性能影响比较明显的超参数为学习率和裁剪系数, 其余超参数则会对其收敛速度产生一定影响. 综合上述分析, 采用表 1 所示的超参数设置进行后续算法的比较.

表 1 LSTM-PPO 超参数选择

超参数	数值
网络学习率	5e-5
数据复用次数	10
裁减系数	0.3
采样步数	512
批次尺寸	16
训练次数	10 000
神经元个数	256

对于实例 1 的算法训练过程如图 6 所示. 图 6(a) 中回合奖励随着训练的进行, 逐渐增加并收敛到了最优值. 图中产生振荡的原因是由于强化学习在训练过程中以 “寻找最优策略” 为目标, 试图找出最优策略. 因此, 对于最优策略的探索寻找在训练过程中不

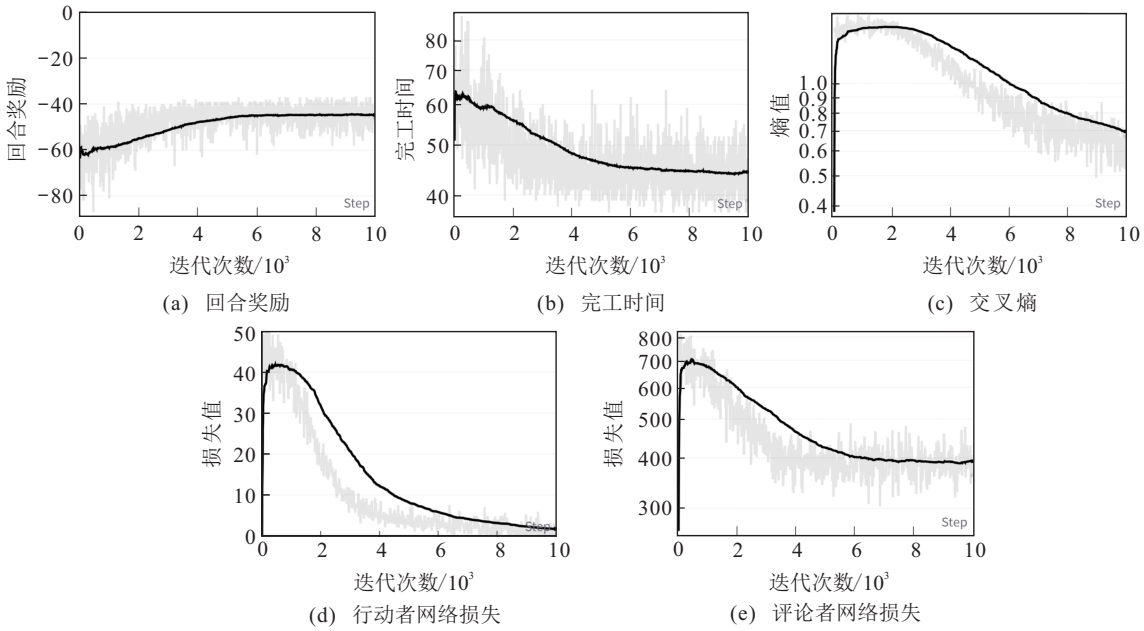


图 6 实例 1 算法训练过程

会停止,导致其图像产生振荡. 图6(b)中完工时间的训练图像恰好与回合奖励曲线相反,随着训练的进行,完工时间逐渐减小. 图6(c)的交叉熵图像代表着智能体在训练过程中对于采取策略的信任程度,随着训练的进行,其值逐渐减小,即所采取的策略将会越来越确定. 图6(d)和6(e)的损失曲线都在逐渐下降并趋于平缓,可以在一定程度上说明对于智能体的训练已经完成.

3.2　实例比较与分析

当前暂无使用深度强化学习算法解决带有工件截止日期的模糊作业车间调度问题,且并无此类的标准数据集. 因此,为了验证本文算法的有效性,同文献[8]提出改进差分进化算法(differential evolution algorithm, DE)相比较. 其中6×6实例1和10×10实例5的调度甘特图如图7和图8所示. 在图7中,横线下方的三角形对应工序的模糊开工时间,横线上方的三角形表示模糊完成时间. 三角形旁的数字对则表示对应加工的工件编号及工件的工序编号.

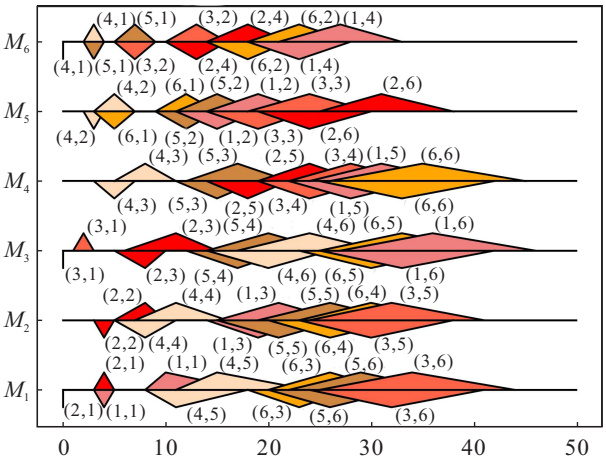


图7　实例1调度甘特图

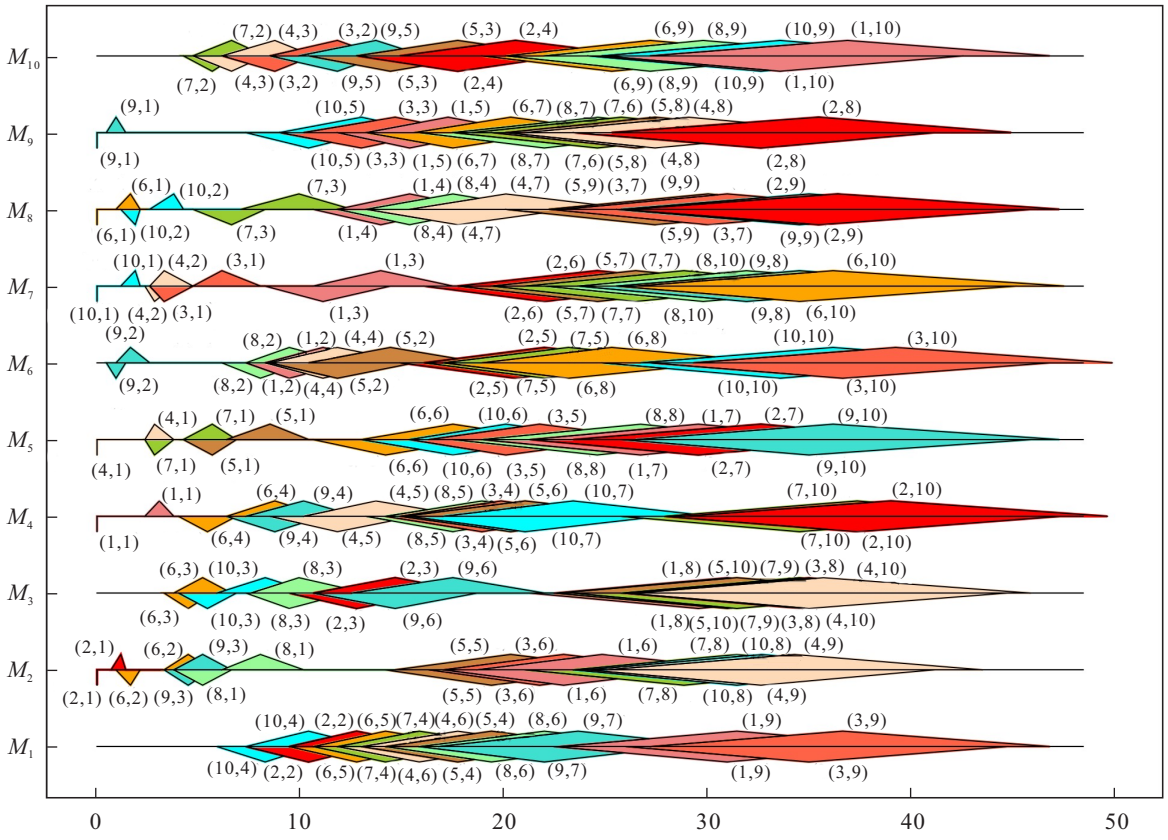


图8　实例5调度甘特图

表2分别给出了启发式调度规则、NSODE算法、D3QN (dueling double DQN)和PPO算法在实例中的比较结果. 启发式调度规则引用自文献[10]. D3QN算法是一种基于DQN的改进算法,它的主要突破点在于利用模型结构将值函数表示成更细致的形式,使智能体模型能够拥有更好的表现. 因此,选用此算法作为值函数类型的深度强化学习算法代表进行比对,其参数设定为: 学习率为0.000 01,记忆容量为100 000;目标网络更新步数为20;动作选择机制为贪婪递减策略. PPO算法参数设置基本与LSTM-PPO算法相同.

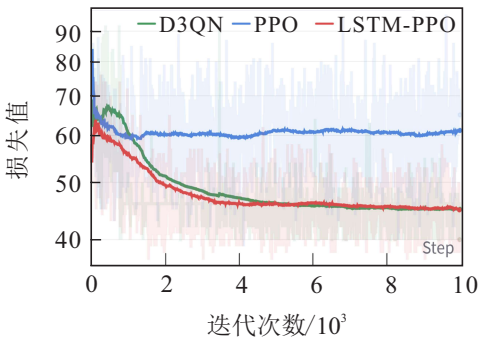
表2中数据为最小化最大模糊完工时间的对比,为了简化形式,选取模糊加工时间中最可能出现的数值作为对比数据. 根据表2中的实例结果对比可以看出,对于8个实例,3种深度强化学习算法在实例中的表现均明显优于文献[8]中的启发式算法,启发式调

表 2 实例结果对比

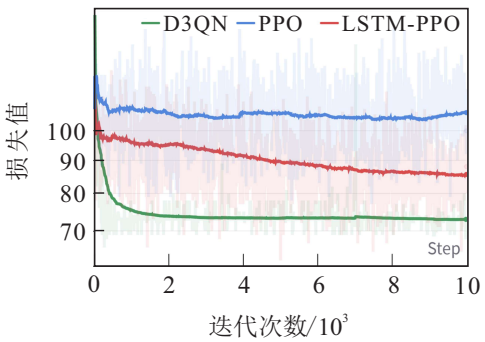
	实例 1	实例 2	实例 3	实例 4	实例 5	实例 6	实例 7	实例 8
SPT	71	176	148	202	589	446	330	182
LPT	90	136	204	171	352	399	372	203
LRM	40	95	121	91	180	173	159	**63**
LRPT	40	95	121	91	180	173	159	**63**
LSO	64	121	174	176	441	440	359	177
SRM	77	214	178	198	599	528	479	246
SRPT	75	214	189	198	656	418	540	271
SSO	75	176	144	159	490	418	301	139
LPT+LSO	81	166	168	135	479	465	349	134
LPT*TWK	77	151	186	150	493	431	475	230
LPT/TWK	85	151	208	134	604	431	532	177
LPT*TWKR	71	95	173	98	357	385	159	96
LPT/TWKR	108	97	241	196	357	503	308	243
SPT+SSO	83	220	188	176	587	512	496	172
SPT*TWK	69	220	178	181	579	537	421	197
SPT/TWK	80	189	162	178	522	406	477	189
SPT*TWKR	98	220	100	173	645	592	500	253
SPT/TWKR	60	128	100	109	248	256	224	83
NSODE	62	116	137	112	251	283	267	112
D3QN	42	84	79	66	175	161	169	65
PPO	39	85	78	69	191	168	174	71
LSTM-PPO	**36**	**81**	**73**	**64**	**166**	**157**	**157**	**63**

度规则在部分实例中的表现能够同深度强化学习算法持平,但其稳定性极差. D3QN 算法在小规模实例上的表现与未改进的 PPO 算法性能差别不大,而在较大规模的 4 个实例上表现出的性能要优于 PPO 算法. 经过改进后的 LSTM-PPO 算法在 8 个实例上所取得的结果均是最优. 以上结果充分表明了本文所提出的 LSTM-PPO 算法的稳定性及对于不同规模问题调度求解的适应性.

为分析 3 种深度强化学习算法在训练中的变化,给出如图 9 所示的训练过程对比图. 从图 9(a) 中可以看出,在解决小规模 6×6 的调度问题中,PPO 算法产生严重震荡,无法很好地收敛. D3QN 算法虽然能够迅速收敛,但容易陷入局部最优解的情况. 增加 LSTM 的 LSTM-PPO 算法在训练过程中收敛更加稳定、迅速,优化结果也更好. 图 9(b) 为解决较大规模 10×10 调度问题的训练过程,可以看出,PPO 算法在较大规模问题中表现较差,不仅产生了严重震荡,且在训练后期有发散的趋势. D3QN 算法能够以较少的迭代次数求得不错的解并且快速地进入收敛状态,但智能体仍然会陷入局部最优,无法找出更优的调度策略. LSTM-PPO 算法在处理较大规模问题时,收敛慢于处理小规模问题,其原因在于训练智能体时,智能体需要不断地尝试探索是否仍存在更优策略,而随着问题规模的增大,导致了智能体需要探索的空间增加,使得智能体需要更多的训练次数来寻找更优策略,因此其在训练过程中,收敛速度变慢. 虽然收敛有所变慢,振荡有所加剧,但其整体的收敛过程依然趋于稳定,且其较强的探索能力使得求解的效果仍然是最优秀的. 综上所述,整体的训练过程表明,相较于 D3QN 算法、PPO 算法,LSTM-PPO 算法在增加了 LSTM 网络后,能够有效地对模糊调度问题状态特征进行数据提取,更好地完成任务分配的要求. LSTM-PPO 算法训练过程相比更加稳定,随着问题规模的增加,算法的收敛速度会随之下降,但其求解能力更强,更能满足对于 FJSSP 求解的需求.



(a) 实例 1 算法对比



(b) 实例 8 算法对比

图 9 深度强化学习算法训练过程对比

本文所提出的算法使用Pytorch框架和Python语言进行编写,算法需要对深度神经网络进行训练,所花费时间较长,但训练好的网络能够以较短的时间输出较优的策略. 因此本文在算法时间上没有进行对比,但从最终的求解结果可以看出,本文所提出的LSTM-PPO算法求解性能优异,能够有效地解决带有工件截止日期的模糊作业车间调度问题.

4 结 论

为了解决以最小化最大完工时间为目标的带有工件截止日期的模糊作业车间调度问题,本文提出了一种LSTM-PPO算法. 针对小规模、较大规模调度问题进行了仿真测试和算法对比,结果表明: 相比于启发式算法,所提出的LSTM-PPO算法能够有效地解决带有截止日期的模糊作业车间调度问题,且提升效果显著. 利用所提出方案建立的状态特征概括性较强,具有较高的灵活性和动态性. 在动作空间的设计中,直接对当前的工序进行选择调度,避免了通过调度规则对工序进行调度,适应性更强. 在奖励函数的设计中,直接将调度目标同奖励函数紧密联系,提升了算法的收敛速度和最终性能. 进一步的工作将在此基础上,尝试使用门控循环单元(gate recurrent unit, GRU)同PPO算法进行结合,以解决LSTM-PPO算法中参数过多、训练时间过长的缺点,提高训练效率. 并继续完善奖励函数,在现有的奖励函数的基础上,添加辅助奖励函数,使智能体能够更快地寻求更优策略.

参考文献(References)

[1] Garey M R, Johnson D S, Sethi R. The complexity of flowshop and jobshop scheduling[J]. Mathematics of Operations Research, 1976, 1(2): 117-129.

[2] Wei Y L, Qiu J B, Lam H K, et al. Approaches to T-S fuzzy-affine-model-based reliable output feedback control for nonlinear itô stochastic systems[J]. IEEE Transactions on Fuzzy Systems, 2017, 25(3): 569-583.

[3] Kuroda M, Wang Z. Fuzzy job shop scheduling[J]. International Journal of Production Economics, 1996, 44(1/2): 45-51.

[4] 李俊青, 潘全科. 求解模糊作业车间调度问题的混合优化算法[J]. 机械工程学报, 2013, 49(23): 142-149.
(Li J Q, Pan Q K. Solving fuzzy job-shop scheduling problems by a hybrid optimization algorithm[J]. Journal of Mechanical Engineering, 2013, 49(23): 142-149.)

[5] 刘凯, 黄辉先, 赵骥. 求解模糊作业车间调度问题的混沌乌鸦搜索算法[J]. 传感器与微系统, 2021, 40(6): 110-113.
(Liu K, Huang H X, Zhao J. Chaotic crow search algorithm for fuzzy job-shop scheduling[J]. Transducer and Microsystem Technologies, 2021, 40(6): 110-113.)

[6] 王凌, 郑洁, 王晶晶. 求解区间数分布式流水线调度的混合离散果蝇优化算法[J]. 控制与决策, 2020, 35(4): 930-936.
(Wang L, Zheng J, Wang J J. A hybrid discrete fruit fly optimization algorithm for distributed permutation flowshop scheduling with interval data[J]. Control and Decision, 2020, 35(4): 930-936.)

[7] Lei D M. Solving fuzzy job shop scheduling problems using random key genetic algorithm[J]. The International Journal of Advanced Manufacturing Technology, 2010, 49(1): 253-262.

[8] Gao D, Wang G G, Pedrycz W. Solving fuzzy job-shop scheduling problem using DE algorithm improved by a selection mechanism[J]. IEEE Transactions on Fuzzy Systems, 2020, 28(12): 3265-3275.

[9] 王凌, 潘子肖. 基于深度强化学习与迭代贪婪的流水车间调度优化[J]. 控制与决策, 2021, 36(11): 2609-2617.
(Wang L, Pan Z X. Scheduling optimization for flow-shop based on deep reinforcement learning and iterative greedy method[J]. Control and Decision, 2021, 36(11): 2609-2617.)

[10] Han B A, Yang J J. Research on adaptive job shop scheduling problems based on dueling double DQN[J]. IEEE Access, 2020, 8: 186474-186495.

[11] 肖鹏飞, 张超勇, 孟磊磊, 等. 基于深度强化学习的非置换流水车间调度问题[J]. 计算机集成制造系统, 2021, 27(1): 192-205.
(Xiao P F, Zhang C Y, Meng L L, et al. Non-permutation flow shop scheduling problem based on deep reinforcement learning[J]. Computer Integrated Manufacturing Systems, 2021, 27(1): 192-205.)

[12] Wang L B, Hu X, Wang Y, et al. Dynamic job-shop scheduling in smart manufacturing using deep reinforcement learning[J]. Computer Networks, 2021, 190: 107969.

[13] Luo S, Zhang L X, Fan Y S. Real-time scheduling for dynamic partial-no-wait multiobjective flexible job shop by deep reinforcement learning[J]. IEEE Transactions on Automation Science and Engineering, 2022, 19(4): 3020-3038.

作者简介

朱家政(1997–), 男, 硕士生, 从事深度强化学习、智能优化调度算法等研究, E-mail: 1078845290@qq.com;

张宏立(1972–), 男, 教授, 博士, 从事复杂生产过程优化与调度等研究, E-mail: zhlxju@163.com;

王聪(1989–), 女, 教授, 博士, 从事智能控制、群智能优化算法等研究, E-mail: 641087385@qq.com;

李新凯(1991–), 男, 讲师, 博士, 从事电力巡检无人机控制、非线性动力学等研究, E-mail: lxk318@foxmail.com;

董颖超(1993–), 男, 博士生, 从事综合能源调度、群智能优化算法等研究, E-mail: 625408917@qq.com.