

控制与决策

Control and Decision

基于混合学习策略进化算法的柔性作业车间节能分批调度研究

张国辉, 伦伟航, 李亮, 沈克伟, 李峥峰

引用本文:

张国辉, 伦伟航, 李亮, 等. 基于混合学习策略进化算法的柔性作业车间节能分批调度研究[J]. 控制与决策, 2025, 40(12): 3655-3666.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2025.0594>

您可能感兴趣的其他文章

Articles you may be interested in

面向全局搜索的自适应领导者樽海鞘群算法

Global search-oriented adaptive leader salp swarm algorithm

控制与决策. 2021, 36(9): 2152-2160 <https://doi.org/10.13195/j.kzyjc.2020.0090>

基于地标特征和元学习方法推荐最适用优化算法

Recommending best suitable metaheuristic based on landmarking feature and meta-learning approach

控制与决策. 2021, 36(5): 1223-1231 <https://doi.org/10.13195/j.kzyjc.2019.0993>

基于局部搜索的反向学习竞争粒子群优化算法

Opposition-based learning competitive particle swarm optimizer with local search

控制与决策. 2021, 36(4): 779-789 <https://doi.org/10.13195/j.kzyjc.2019.1150>

求解约束优化问题的改进果蝇优化算法及其工程应用

Improved fruit fly optimization algorithm for solving constrained optimization problems and engineering applications

控制与决策. 2021, 36(2): 314-324 <https://doi.org/10.13195/j.kzyjc.2019.0557>

l_p -范数约束下MKL-OC-ELM的装备故障检测

MKL and OC-ELM fault detection based on l_p -norm constraint

控制与决策. 2021, 36(10): 2379-2388 <https://doi.org/10.13195/j.kzyjc.2020.0443>

基于混合学习策略进化算法的柔性作业车间 节能分批调度研究

张国辉^{1†}, 伦伟航¹, 李亮², 沈克伟², 李峥峰³

(1. 郑州航空工业管理学院 管理工程学院, 郑州 450046;

2. 新乡航空工业(集团)有限公司, 河南 新乡 453049;

3. 中原工学院 智能机电工程学院, 郑州 451191)

摘要: 首先, 针对柔性作业车间节能分批调度问题, 构建一个以最小化最大完工时间和机器加工总能耗为目标的优化模型; 同时, 提出一种规则数字化分批方法, 合理划分批次, 以进一步提升调度效率. 然后, 为高效解决该问题, 提出一种融合强化学习的非支配排序遗传算法 II (RLNSGA-II), 该算法通过自适应策略调整交叉率和变异率, 能够有效增强算法的全局搜索能力. 接着, 结合分批问题特性设计 3 种邻域搜索策略, 可显著提升算法的局部搜索能力. 最后, 通过对比实验验证所设计强化学习参数自适应策略和邻域搜索策略的有效性, 并将 RLNSGA-II 与其他多目标优化算法进行对比, 验证其在求解 EFJSP-BS 中的显著优越性.

关键词: 柔性作业车间调度; 节能分批调度; 多目标优化; 强化学习; RLNSGA-II; 局部搜索

中图分类号: TH165 文献标志码: A

DOI: 10.13195/j.kzyjc.2025.0594

引用格式: 张国辉, 伦伟航, 李亮, 等. 基于混合学习策略进化算法的柔性作业车间节能分批调度研究 [J]. 控制与决策, 2025, 40(12): 3655-3666.

A hybrid learning strategy evolutionary algorithm for energy-efficient flexible job shop batch scheduling problem

ZHANG Guo-hui^{1†}, LUN Wei-hang¹, LI Liang², SHEN Ke-wei², LI Zheng-feng³

(1. School of Management Engineering, Zhengzhou University of Aeronautics, Zhengzhou 450046, China;

2. Xinxiang Aviation Industry Co., Ltd., Xinxiang 453049, China; 3. School of Intelligent Mechatronics

Engineering, Zhongyuan University of Technology, Zhengzhou 451191, China)

Abstract: This paper addresses the energy-efficient flexible job shop batch scheduling problem (EFJSP-BS) by formulating an optimization model aimed at minimizing both the makespan and the total energy consumption of machines. A rule-based digital batching method is proposed to reasonably divide jobs into batches, further improving scheduling efficiency. To efficiently solve the problem, a nondominated sorting genetic algorithm II integrated with reinforcement learning (RLNSGA-II) is proposed. This algorithm adaptively adjusts the crossover and mutation rates through a reinforcement learning strategy, significantly enhancing global search capabilities. Moreover, three neighborhood search strategies tailored to the characteristics of the batching problem are designed to substantially improve local search performance. Comparative experiments are conducted to validate the effectiveness of the reinforcement learning-based adaptive parameter strategy and the neighborhood search strategies. Additionally, the performance of the RLNSGA-II is compared with other multi-objective optimization algorithms, demonstrating its significant superiority in solving the EFJSP-BS.

Keywords: flexible job shop scheduling; energy-efficient batch scheduling; multi-objective optimization; reinforcement learning; RLNSGA-II; local search

0 引言

随着数字化转型浪潮持续推动, 智能制造技术

变革正重塑产业格局. 在此转型过程中, 制造业能耗

问题日益凸显, 如何实现高效节能生产已成为行业

收稿日期: 2025-06-05; 录用日期: 2025-08-21.

基金项目: 国家自然科学基金面上项目 (52575603); 河南省重点研发专项项目 (231111221200); 教育部人文社会科学规划基金项目 (23YJAZH193).

责任编辑: 李新宇.

[†]通信作者. E-mail: zgh09@zua.edu.cn.

关注焦点^[1]. 车间调度作为制造系统的核心环节, 通过优化资源配置和工序安排, 在提升生产效率的同时能够显著降低能耗. 而在车间调度研究领域中, 柔性作业车间调度问题 (FJSP) 是车间调度问题的扩展^[2]. 求解 FJSP 问题中随着机器、工件以及工序的柔性提升, 导致生产任务的分配方案急剧增加. 因此, 在柔性车间实际生产过程中, 通常会采用技术改进或组织优化的方式来提高资源利用率, 如工序并行、分批调度等, 其中分批调度方法能够更有效地解决柔性作业车间问题, 合理的批量划分方法有助于在降低生产成本的同时缩短生产周期^[3]. 有关车间分批调度问题研究中, 周靖雯等^[4]针对具有多品种小批量特征的复杂混合流水车间调度问题, 提出了等量一致分批策略, 同时, 实现了工件在不同工序上的并

行加工; Shao 等^[5]提出了一种涉及切换时间和切换操作员限制的可变子批次柔性车间多目标最优调度方法; 刘雪红等^[6]为了将可变批次的调度策略应用于生产, 以提高大规模柔性作业车间的生产效率和设备利用率, 针对柔性作业车间可变子批问题的特点建立了多目标柔性作业车间调度模型和析取图模型; Wang 等^[7]针对制造业中结合批次流和有限运输资源的柔性装配作业车间调度问题, 提出了一种集成仿真优化方法, 通过自修复遗传算法优化批次策略. 如图 1 所示, 存在两类工件均有 3 道工序, 且每类工件数量若干, 采用分批调度策略后, 完工时间得到改善. 这表明分批调度方法能够有效提升生产效率, 验证了其在柔性作业车间调度中的实用性.

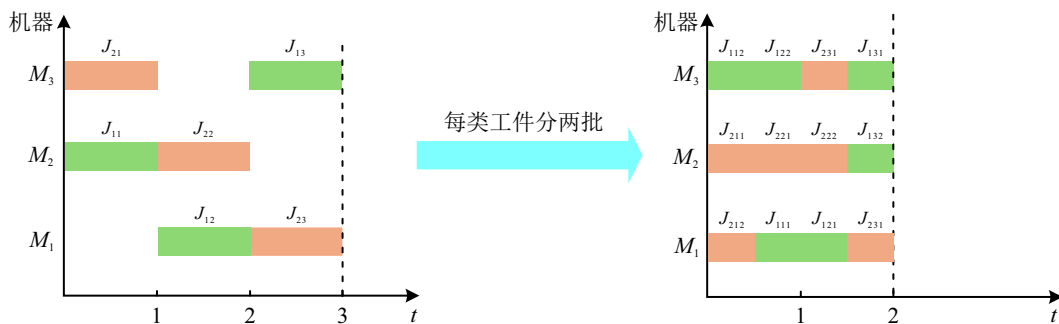


图1 是否考虑分批情况对比

全球资源短缺问题日益凸显, 推动低碳制造发展已成为制造业实现可持续发展的关键^[8]. 作为能源消耗和碳排放的重要领域, 我国制造业实施绿色转型不仅是对国家“双碳”战略(碳达峰、碳中和)的积极响应, 更是应对国际资源竞争新格局的战略选择. 因此, 在研究柔性车间调度问题时, 需要考虑节能问题, 这使得调度问题更具挑战性. Wang 等^[9]针对峰值功率约束下的柔性作业车间调度问题, 构建了以电力成本、完工时间和能耗为优化目标的多目标调度模型; 王建华等^[10]针对多目标柔性作业车间绿色调度问题 (MO-FJGSP), 建立了优化目标为最大完工时间、机器总负荷和能耗最小的多目标数学模型; Wang 等^[11]提出了考虑机器重启的节能低碳柔性作业车间调度问题, 同时, 设计了一种增强的模因算法并采用机器重启策略来平衡能耗与设备寿命.

求解 FJSP 问题通常主要分为 3 类: 精确算法、启发式算法和智能优化算法. 智能优化算法相较于精确算法与启发式算法, 在求解柔性作业车间调度问题时表现出更好的适应性和灵活性, 但是, 其关键参数的设置对于解的质量影响显著. 因此, 如何实现算法关键参数的智能调整, 已成为当前研究中的一

个重要挑战. Chen 等^[12]提出了一种基于强化学习 (RL) 的自适应遗传算法, 该算法以遗传算法为优化框架, 通过强化学习机制实现了关键参数的动态智能调节; Zhang 等^[13]结合 Q 学习算法和 Sarsa 算法改进了算法, 解决了柔性作业车间调度问题, 根据种群的当前状态利用 Q 学习算法和 Sarsa 算法自适应调整交叉和变异的概率; 尹爱军等^[14]融合了多个多样性度量指标, 利用强化学习动态优化种群迭代过程中的拆分比例参数来保持种群多样性, 改善了算法的收敛性能.

综上所述, 在考虑柔性作业车间节能分批调度研究方面, NSGA-II 及其改进算法和 RL 的改进方法在该问题上已取得了一定进展, 但是仍然均存在不足之处, 尤其是将强化学习中的 Q-learning 算法与 NSGA-II 融合来解决柔性作业车间节能分批调度问题研究较少. 因此, 本文针对柔性作业车间节能分批调度问题 (EFJSP-BS), 提出 RLNSGA-II 算法. 该算法以最小化最大完工时间和机器加工总能耗为优化目标, 构建相应的优化模型, 为提升初始种群质量设计 3 种初始化规则. 同时加入 Q-learning 机制, 使得交叉率和变异率能够根据种群进化状态实现自适应

调整. 最终通过对比实验验证了所提出算法在 EFJSP-BS 问题求解中的优越性能.

1 问题描述与模型建立

1.1 EFJSP-BS 问题描述

柔性作业车间节能分批调度 (EFJSP-BS) 问题描述如下: 有 m 台机器, 机器集合 $M = \{M_1, M_2, \dots, M_m\}$. 共有 n 种类型的加工工件, 工件集合 $J = \{J_1, J_2, \dots, J_n\}$ ($i = 1, 2, \dots, n$), 每类工件 J_i 的数量为 Q_i , 且每类工件 J_i 均可分为 B_i 个子批, 同时, 各子批的工件总数量应为同类工件 J_i 的工件数量 Q_i . 每类工件 J_i 有若干个工序 $O_{ij} = \{O_{i1}, O_{i2}, \dots, O_{ip}\}$ ($i = 1, 2, \dots, p$), O_{ij} 为第 i 类工件第 j 道工序.

在 EFJSP-BS 问题中, 调度的基本单位不同于传统的 FJSP, 后者通常以工件为单位, 而 EFJSP-BS 则是以子批为单位. 具体而言, 调度过程涉及为每个子批的工序选择合适的加工机器, 并确定各机器上子批的加工顺序, 使得机器加工能耗最小以及最大完工时间最小化.

EFJSP-BS 问题满足以下假设条件:

- 1) 在 0 时刻机器可以加工任一个子批;
- 2) 每台机器同一时刻只能加工一个子批;
- 3) 每个子批开始加工后不能中断;
- 4) 根据工件工序的紧前紧后约束, 各类工件的各子批次间也有工序的紧前紧后约束;
- 5) 不考虑各批次运输时间和调整时间的影响;
- 6) 每个子批所含工件数量必须为严格大于 0 的整数, 同时, 同一类型工件在所有子批中数量的和应等于该类型工件的总量.

1.2 模型建立

对于柔性作业车间节能分批调度问题, 建立数学模型所定义的变量如表 1 所示.

本文构建以最大完工时间和机器加工总能耗最小化为目标的优化模型. 目标函数和约束如下.

- 1) 最小化最大完工时间为

$$\text{Min} f_1 = \text{MinMax}(C_i), \quad 1 \leq i \leq n. \quad (1)$$

- 2) 最小化机器加工总能耗为

$$f_2 = \text{Min} \left\{ \sum_{i=1}^n \sum_{j=1}^{j_i} \sum_{z=1}^{B_i} \sum_{h=1}^m T_{ijzh} E_{ijzh} X_{ijzh} \right\}. \quad (2)$$

- 3) 约束条件如下所示:

$$\sum_{h \in M_{ijz}} F_{ijzh} \leq \sum_{h \in M_{i(j+1)z}} S_{i(j+1)zh}; \quad (3)$$

$$F_{ijzh} = S_{ijzh} + T_{ijzh}; \quad (4)$$

表1 相关符号说明

变量	含义
i	工件序号
j	工序序号
z	批次序号
n	工件类型数, 工件集合 $J = \{J_1, J_2, \dots, J_n\}$
m	机器总数
h	机器序号, 机器集合 $M = \{M_1, M_2, \dots, M_m\}$
j_i	第 i 类工件总工序数
J_i	第 i 类工件集合
Q_i	第 i 类工件总数
B_i	第 i 类工件子批个数
O_{ijz}	第 i 类工件的第 j 道工序的第 z 个子批
P_{ijz}	第 i 类工件的第 j 道工序的第 z 个子批批量数
M_{ijz}	第 i 类工件的第 j 道工序的第 z 个子批可用机器集
C_i	第 i 类工件的完工时间
S_{ijzh}	第 i 类工件的第 j 道工序的第 z 个子批在机器 h 上的开始加工时间
F_{ijzh}	第 i 类工件的第 j 道工序的第 z 个子批在机器 h 上的完工时间
T_{ijzh}	第 i 类工件的第 j 道工序的第 z 个子批在机器 h 上的加工时间
E_{ijzh}	第 i 类工件的第 j 道工序的第 z 个子批在机器 h 上加工产生的单位时间能耗
X_{ijzh}	若工序 O_{ijz} 在 M_h 上加工, 则 $X_{ijzh} = 1$; 否则, $X_{ijzh} = 0$
M	一个无穷大的正数
$Y_{ijz, i'j'z', h}$	$Y_{ijz, i'j'z', h} \in \{0, 1\}$: 若在机器 h 上, 工序 O_{ijz} 先于任务 $O_{i'j'z'}$ 加工, 则 $Y_{ijz, i'j'z', h} = 1$; 否则, $Y_{ijz, i'j'z', h} = 0$

$$\sum_{h \in M_{ijz}} X_{ijzh} = 1; \quad (5)$$

$$S_{ijzh} \geq F_{i'j'z'h} - M(1 - Y_{ijz, i'j'z', h}); \quad (6)$$

$$S_{i'j'z'h} \geq F_{ijzh} - MY_{ijz, i'j'z', h}; \quad (7)$$

$$Y_{ijz, i'j'z', h} + Y_{i'j'z', ijz, h} = 1, \quad \forall (ijz) \neq (i'j'z); \quad (8)$$

$$B_i \leq B_i^{\text{Max}}, \quad B_i \in \mathbb{Z}^+. \quad (9)$$

其中: 式 (3) 表示子批的当前工序必须在上一道工序完工后才能开始, 式 (4) 表示各工序的完工时间由其开始加工时间与相应的加工时长的和决定, 式 (5) 表示任意工序的子批在机器上只能加工一次, 式 (6) ~ (8) 表示任一时刻一台机器只能加工一个子批, 式 (9) 表示各工件子批批量的约束.

为便于理解 EFJSP-BS 问题, 表 2 提供了 2 个工件、3 台机器的例子. 需要说明的是, 机器能耗指的

是机器加工该工序所消耗的总能耗,而非单位时间的能耗.同时,由于车间工具资源有限,每类工件的最大子批次数被严格限制在不超过3.

表2 2 × 3调度方案处理信息

工件	数量	批次	工序	加工时间/机器能耗		
				M_1	M_2	M_3
J_1	10	[2, 3]	$O_{1.1}$	3/184	1/167	2/170
			$O_{1.2}$	—	3/174	2/186
			$O_{1.3}$	2/187	1/175	3/169
J_2	10	[2, 3]	$O_{2.1}$	2/185	3/176	—
			$O_{2.2}$	3/171	2/169	1/168

2 EFJSP-BS 模型求解方法设计

2.1 RLNSGA-II 算法框架

在解决柔性作业车间节能分批调度问题中,需要同时解决最小化最大完工时间和最小化机器总能耗问题.为了更好地解决复杂环境下的调度问题,本文设计一种融合强化学习的RLNSGA-II算法,改善传统NSGA-II中交叉率和变异率一成不变的限制,同时提出3种邻域搜索策略,以避免算法过早收敛.算法终止条件为迭代至最大迭代次数,算法流程如图2所示.

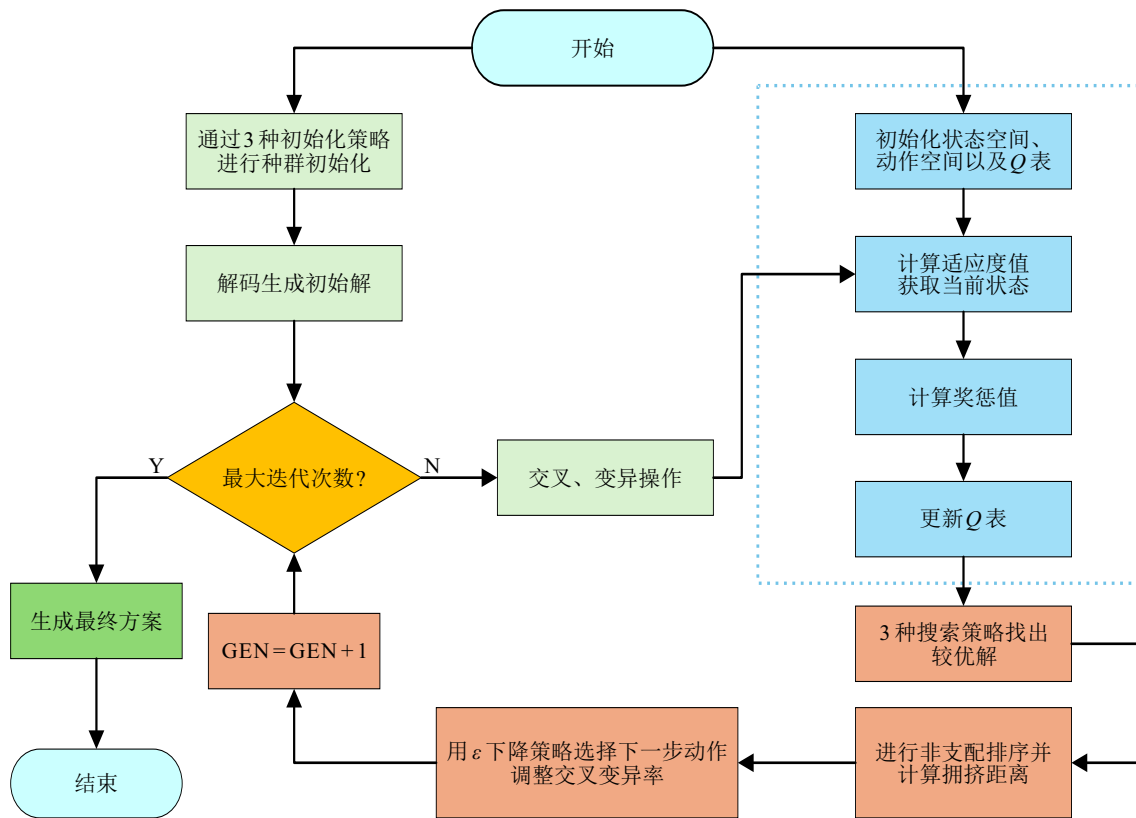


图2 算法流程

2.2 分批方法设计

引入分批策略有助于有效降低整体加工时间和能耗,从而提升资源的综合利用率.为此,本文在文献[15]中提出的随机数字分割方法的基础上进行改进,提出一种规则数字化分批方法.以表2中的问题为例,假设每类工件数量均为10.

每个小格表示一类工件,每个格子里的数字表示该类工件的分批方案,同时,将每类工件按照工件

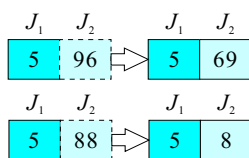


图3 分批方案情况

数量依次编号,如图3所示.

1) 第1类情况: J_1 格子数字为5,表示编号“1~5”的工件为一批,编号“6~10”的工件为一批,共两批; J_2 格子数字为96,十位比个位大,先调换位置,因此, J_2 的分批为编号“1~6”一批、编号“7~9”一批、编号“10”为一批,共3批.

2) 第2类情况: J_1 同第1类情况; J_2 格子数字为88,因十位与个位相同,仅保留个位“8”作为分批依据,因此, J_2 批次编号“1~8”、编号“9~10”,共两批.

2.3 编码与解码

1) 3层编码.

根据EFJSP-BS问题的特性,本文采用3层编码

的方式: 第1层为每类工件分批方案, 第2层为机器选择部分, 第3层为工序排序部分。

对于分批方法部分参见第2.2节规则数字化分批方法设计。对于机器选择部分, 针对每个子批, 首先找出其可选加工机器集合, 编码中的数值即表示选择该集合中的第几台机器; 对于工序子批排序部分, 每个格子数字有两位, 十位表示第几类工件, 个位表示这类工件的第几道工序。如图4所示, 以表2中 2×3 调度方案为例。染色体分为3层: 第1层决定工件的批次数, 参见第2.2节规则数字化分批方法设计; 第2层为机器选择部分, J_1 有3道工序且分批数为2, 因此, 每道工序有两个子批需要安排机器, 设备层第1行的2表示 J_1 工件的第1道工序第1个子批, 在可选机器集合中需要选择第2台机器进行加工, 以此类推; 第3层为工序排序部分, 第1个21为工件 J_2 的第1道工序第1个子批, 第2个21为工件 J_2 的第1道工序第2个子批, 以此类推, 同时考虑了工序间的约束, 工序层不是随机生成, 在排列一类工件的下一道工序时, 前面必须有这一类工件足够的紧前工序。

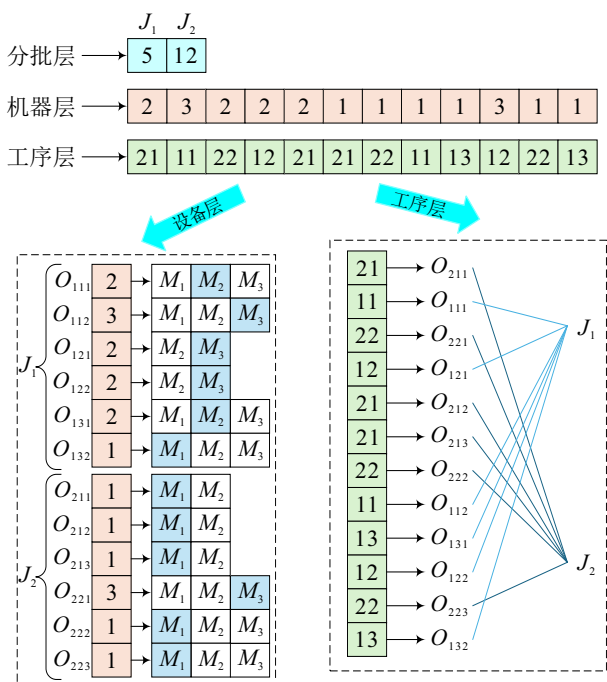
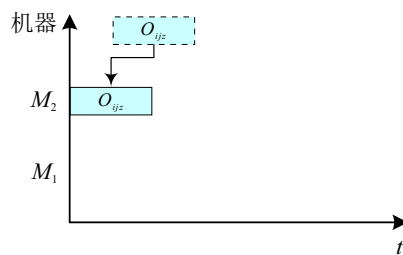


图4 3层式编码

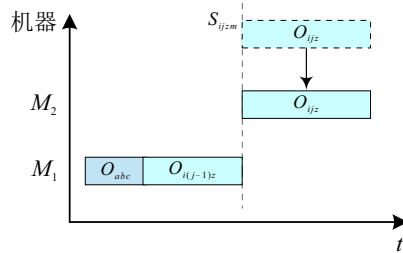
2) 解码部分。

为了提升资源利用率、缩短工序完工时间, 本文引入一种左移插入式解码策略, 该解码方式以工件加工顺序为主线, 在机器选择、工序排序的基础上引入批量问题机制, 通过左移插入策略将待加工工序尽可能安排到满足条件的最靠前空闲时间段, 具体如下。

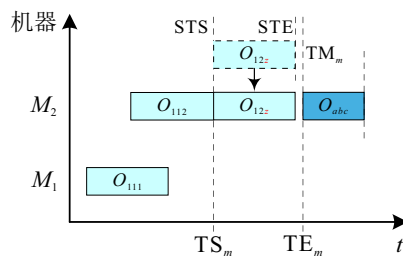
① 若 O_{ijz} 为工件*i*的首道工序子批, 则直接在机器 M_2 上从0时刻开始加工, 如图5(a)所示。



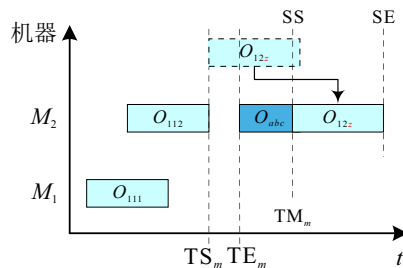
(a) 首序直加工



(b) 前序即加工



(c) 空闲段插入



(d) 末尾段加工

图5 4种解码情况

② 若 O_{ijz} 为工件*i*在机器 M_2 上首道工序任意子批, 则在其前序任意子批 $O_{i(j-1)z}$ 完成后立即加工, 如图5(b)所示。

③ 若 O_{ijz} 不属于以上两种情况, 则在机器 M_2 上已调度子批间查找所有空闲时间段 $[TS_m, TE_m]$ 。其中: TS_m 为空闲时间段开始时间, TE_m 为空闲时间段结束时间。计算子批最早可加工时间 S_{ijzm} , 并按照下文式(10)计算插入空闲时间段的 earliest start time STS; 若该段长度满足加工需求式(11), 则分配时间段 $[STS, STE]$; 否则, 按照式(12)安排在 $[SS, SE]$ 加工:

$$STS = \text{Max}(S_{ijzm}, TS_k); \quad (10)$$

$$STE = T_{ijzh} + S_{ijzm} \leq TE_m; \quad (11)$$

$$SS = \text{Max}(S_{ijzm}, TM_m), SE = SS + T_{ijzh}. \quad (12)$$

这里 TM_m 为在机器上对应最后子批加工结束时间, 如图 5(c) 和图 5(d) 所示.

2.4 初始化

为提高初始化种群的质量, 本文设计 3 种生产初始化规则来确保种群的质量和多样性. 优先确定批次划分的编码, 采用规则数字化分批方法来确定方案, 且后续机器编码和工序编码均是按照批次划分来考虑的; 对于机器和工序子批调度的编码, 以完全随机、最短加工时间、最小加工能耗的规则来生成初始种群, 且 3 者按照 50%、25%、25% 分配, 具体如下:

1) 完全随机规则: 以完全随机的形式生成机器选择和工序子批排序的初始解. 为每个工序子批选择一个机器, 对工序子批的顺序按照工序间的约束排序.

2) 最短加工时间规则: 在生成机器选择初始解时, 优先为每个工序子批安排可选机器中加工时间最短的机器, 工序子批排序同上.

3) 最小加工能耗规则: 在生成机器选择初始解时, 优先为每个工序子批安排可选机器中加工能耗最小的机器, 工序子批排序同上.

2.5 交叉变异

1) 交叉算子.

机器层: 通过引入多个随机交叉点, 交换交叉点上两个父代对应的基因, 产生两个新的子代. 工序层: 首先, 根据几类工件随机产生一个随机数, 根据这个随机数来进行交叉操作, 利用这个随机数提取两个父代中所有该类工件工序子批; 然后, 将父代 1 的该类工件所有工序子批依次与父代 2 中的该类工件进行交换, 产生两个新的子代, 如图 6 所示.

2) 变异算子.

机器层: 通过引入多个随机变异点, 改变父代对应的变异点可选机器号, 生成新的子代. 工序层: 随机选择工序层的两个变异点, 交换两个变异点的基因, 由于工序层编码的情况, 变异操作易产生不可行解, 因此, 工序层变异后对其子代进行了检查排除不可行解, 如图 7 所示.

2.6 局部搜索

为增强 RLNSGA-II 算法的搜索能力, 本文设计 3 种邻域搜索策略, 并在迭代过程中随机选择其中一

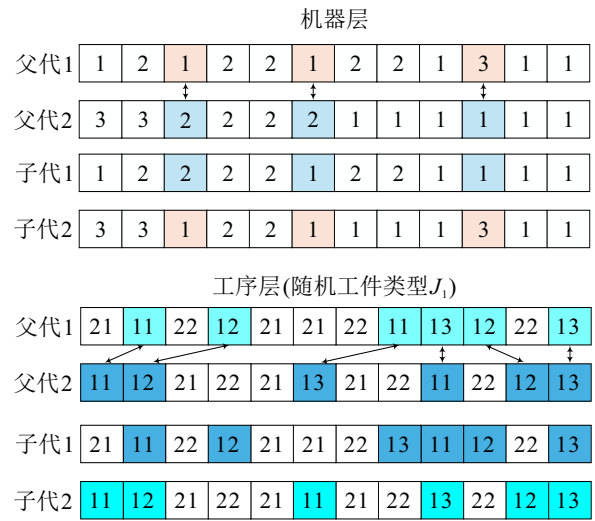


图6 交叉算子

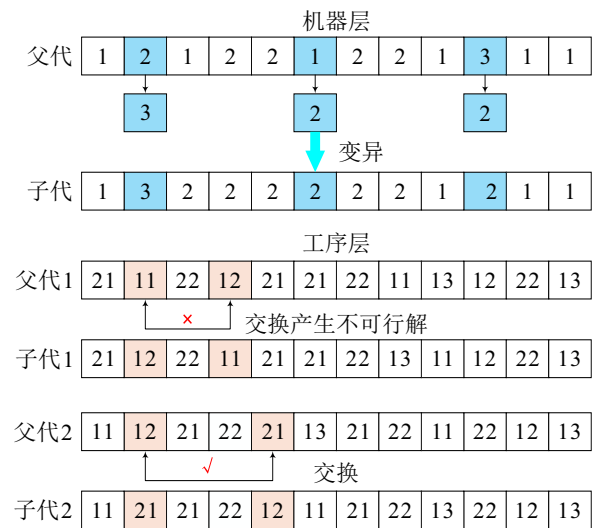


图7 变异算子

种策略, 以保证解的多样性的同时, 提升收敛速度和精度. 规则如下.

1) 批次突变邻域规则.

首先对当前染色体进行解码, 构造对应的甘特图, 从中识别出最后完成的工序, 即决定最大完工时间的关键任务. 记录该工序批次的所属工件信息, 对此工件执行批次突变操作, 将其在染色体中的批次编码基因突变为一个不同的合法随机整数值, 具体操作如算法 1 所示. 以 J_2 为例, 详情如图 8 所示.

算法1 批次突变邻域规则.

输入: 个体 S_1 , 最大完工时间 Time_1 , 最小能耗 EC_1 ;
输出: 新个体 S_3 .

1. 对 S_1 进行解码, 查找关键任务 L
2. 根据关键任务 L 搜寻工件类型 J_1 , 找出对应 FP_1 数
3. 随机工件 J_2 的 FP , 直至 $FP_2 \neq FP_1, FP = FP_2$
4. 根据 FP 重新对 J_2 进行编码, 同时保留 S_1 中工件 J_2 以外工件编码

5. 生成新个体 S_2 , 对 S_2 进行解码, 得到最大完工时间 Time_2 、最小能耗 EC_2
6. 利用线性加权法计算 S_1 得分 scores_1 、 S_2 得分 scores_2
7. if $\text{scores}_1 < \text{scores}_2$
8. $S_3 = S_1$
9. end
10. if $\text{scores}_1 > \text{scores}_2$
11. $S_3 = S_2$
12. end
13. return S_3

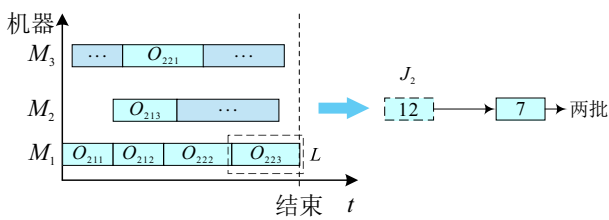


图8 批次突变示意图

2) 最大完工时间最小化邻域规则.

首先, 对当前染色体进行解码, 构建对应的甘特图, 并识别每台机器上最后一道工序子批, 将其视为关键任务. 然后, 根据加工时间矩阵判断每个关键任务是否在对应的最短加工时间机器上加工, 若已在最优机器上加工, 则保留该基因段编码; 若未在最优机器上加工, 则将关键任务调整至加工时间最短的机器.

3) 加工能耗最小邻域规则.

首先, 对当前染色体进行解码, 构建对应的甘特图, 并识别每台机器上最后一道工序子批, 将其视为关键任务. 然后, 根据能耗矩阵判断每个关键任务是否在对应的最小能耗机器上加工, 若已在最优机器上加工, 则保留该基因段编码; 若未在最优机器上加工, 则将关键任务调整至加工时间最短的机器.

3 基于 Q-learning 的参数适应策略

在进化算法中, 交叉和变异操作对于算法性能具有重要影响, 其中 P_c 和 P_m 是最关键的两个参数. 若 P_c 和 P_m 过于小, 则算法很难生成新的种群个体从而导致解空间大小受限; 若 P_c 和 P_m 过于大, 则会导致父代中基因优秀的个体难以保留, 不利于算法的收敛. 因此, 本文引入强化学习中 Q-learning 算法, 帮助算法在迭代过程中选择最优参数.

3.1 Q-learning 算法介绍

Q-learning 算法核心思想是智能体与环境互动, 采样状态、动作和奖励数据, 并以此迭代更新其价值函数, 从而优化策略以实现长期累积回报的最大化.

其基本更新公式为

$$Q^{\text{new}}(s_t, a_t) \leftarrow (1 - \alpha)Q(s_t, a_t) + \alpha(r_{t+1} + \gamma \max_a Q(s_{t+1}, a)). \quad (13)$$

其中: $Q(s_t, a_t)$ 表示在状态 s_t 下采取动作 a_t 的期望收益, α 为学习率, γ 为折扣因子.

3.2 状态定义

在 EFJSP-BS 问题中, 以最小化最大完工时间、机器加工总能耗为目标建立状态空间前, 需要将个体适应度值定义为状态空间. 由于所研究的问题为多目标问题, 直接评估解的优劣十分复杂. 一个高质量的帕雷托解集, 不仅需要具有良好的收敛性, 还需具备良好的分布性. 针对这些问题, 本文引入 3 个评价指标组合对解的质量进行评价: 集合覆盖率 (SC)、间距 (S) 和多样性 (D).

1) 集合覆盖率 (SC) 如下所示:

$$SC(A, B) = \frac{|\{b \in B \mid \exists a \in A : a \prec b\}|}{|B|}. \quad (14)$$

其中: $a \prec b$ 表示解 a 在所有目标上不劣于解 b , 且至少在一个目标上优于 b ; $|B|$ 为解集 B 的大小; $SC(A, B) \in [0, 1]$, 值越大, A 的优势越明显, $SC(A, B) = 1$ 表示 A 中的一个解支配 B 中的所有解.

2) 间距指标 (S) 如下所示:

$$S(A) = \frac{1}{N-1} \sum_{i=1}^N |c_i - \bar{c}|. \quad (15)$$

其中: N 为 A 中解的数量; c_i 为第 i 个解与解集合 A 中其他解间的最小距离; \bar{c} 为所有 c_i 的平均值; $S(A)$ 的值越小, 解的分布越均匀.

3) 多样性指标 (D) 如下所示:

$$D(B) = \frac{1}{(N-1)\bar{d}} \sum_{i=1}^{N-1} |d_i - \bar{d}|. \quad (16)$$

其中: d_i 为第 i 个解与其相邻解间的距离, \bar{d} 为所有 d_i 的平均值, N 为非支配解集 B 的规模.

在进化过程中, 经交叉和变异操作形成的中间种群 F_1 , 与进一步通过邻域结构以及非支配排序选择后的种群 F_2 , 在个体构成和质量上存在显著差异, 其状态也随之发生变化. 因此, 为避免 Q-learning 中由种群快速变动带来的状态评估偏差, 本文采用首代种群 P_1 作为参考种群. 同时, 为了综合衡量解集的收敛性与多样性, 参考式 (14) ~ (16) 构建多指标适应度函数. 第 i 代种群 P_i 的多目标适应度值定义^[13] 如下所示:

$$F(P_i) =$$

$$l_1 \times \text{SC}(P_1, P_i) + l_2 \times \frac{S(P_i)}{S(P_1)} + l_3 \times \frac{D(P_i)}{D(P_1)}. \quad (17)$$

其中: $\text{SC}(P_1, P_i)$ 为第 i 代种群相对于初始种群的集合覆盖率; $S(P_i)$ 、 $D(P_i)$ 分别为第 i 代种群的间距和多样性; l_1 、 l_2 、 l_3 为 3 个评价指标的权重体系, 且满足 $l_1 + l_2 + l_3 = 1$, 同时设 l_1 、 l_2 、 l_3 的权重分别为 0.35、0.3、0.35.

为构建适用的 Q -learning 的状态空间, 本文根据总体适应度将状态空间划分为 21 个状态. 其中: $S = \{S(1), S(2), \dots, S(21)\}$, 将适应度值区间 $[0, 1]$ 按照步长 0.05 划分为 20 个子区间, 当适应度大于 1 时为第 21 个状态空间.

3.3 动作定义

在本文 Q -learning 的框架下, 智能体通过选择动作来调节进化算法中的交叉概率 P_c 和变异概率 P_m , 以适应不同状态下的搜索策略需求, 从而提高种群的多样性和收敛性. 为构建高效且灵活的动作空间, 分别构建对应的参数动作集合, 具体如下.

1) 交叉概率动作集合 A_{P_c} .

交叉概率 P_c 的常用范围为 $[0.4, 0.9]$, 本文将其均匀划分为 10 个动作, 间隔为 0.05, 形成集合形式如下所示:

$$A_{P_c} = \{0.40, 0.45, 0.50, \dots, 0.90\}. \quad (18)$$

2) 变异概率动作集合 A_{P_m} .

变异概率的常用范围为 $[0.01, 0.21]$, 同样划分为 10 个动作, 间隔为 0.02, 构建如下集合:

$$A_{P_m} = \{0.01, 0.03, 0.05, \dots, 0.21\}. \quad (19)$$

3.4 奖惩机制

当智能体执行一项行动后, 环境会提供正或负的奖励, 以不断更新其行为策略. 奖励机制的设计对于算法能否有效学习具有关键作用. 在本研究中, 奖励函数的构建以种群在多目标适应度函数上的提升为核心, 旨在引导参数策略朝向更优的搜索方向演化.

奖励函数根据当前代种群 P_i 与下一代种群 P_{i+1} 的适应度值进行差值比较, 其基本形式定义为

$$r = f(F(P_{i+1}) - F(P_i)). \quad (20)$$

其中: $F(P_i)$ 为第 i 代种群的多目标适应度值; $f(\cdot)$ 为激励函数, 用于将适应度变化转化为有限的奖励值.

考虑到实际进化过程中的适应度差异幅度有限, 为避免 Q 值更新不稳定, 本文将奖励值限制在有限区间内, 具体定义如下所示:

$$r = \begin{cases} +1, & F(P_{i+1}) - F(P_i) > k; \\ 0, & |F(P_{i+1}) - F(P_i)| \leq k; \\ -1, & F(P_{i+1}) - F(P_i) < -k. \end{cases} \quad (21)$$

其中 k 为一个设定的阈值, 用于过滤微小波动, 避免噪声干扰学习过程. 该机制鼓励智能体选择能够显著提升种群质量的参数组合, 并惩罚导致种群退化的动作.

3.5 动作选择策略

动作选择策略是智能体的决策机制, 它定义了特定状态下选择动作的依据. 这一策略对于学习效率和全局最优策略的获得具有重要影响, 尤其是在状态空间较大或环境反馈不确定的优化问题中尤为关键.

本文采用改进的 ϵ -贪心策略^[16], 该策略在确保探索新动作与利用当前最优策略间取得平衡的同时, 具备实现全局最优搜索的潜力, 具体如下所示:

$$\pi(s_t, a_t) = \begin{cases} \text{Randomly select } a \in A, \text{ rand} \leq \epsilon; \\ \text{Max}_a Q(s_t, a_t), \text{ rand} > \epsilon; \end{cases} \quad (22)$$

$$\epsilon = \epsilon_1 - z((\epsilon_1 - 0.05) / \max z). \quad (23)$$

3.6 Q -table 更新

在确定上述所有策略后, 便可以开始进行交叉变异率的自适应策略. 智能体的决策主要依赖 Q 表更新, 该表以状态为行、动作为列, 记录其对应值. 每次迭代中, 各 $Q(s, a)$ 根据学习经验和未来奖励预测进行更新, 具体更新方式如式 (13) 所示.

4 仿真实验

本文使用 Matlab 2021b 对所提出 RLNSGA-II 算法进行实例仿真实验, 运行环境为 Intel(R) Core(TM) i5-14600KF 3.50 GHz 和 32.00 GB RAM. 首先使用 Matlab 2021b 对所提出算法进行多因子正交实验以确定算法的多个参数的取值. 由于分批问题没有标准算例, 本文随机生成 10 个不同规模的算例进行求解, 将 RLNSGA-II 算法与 NSGA-II、DQNSGA、IGA 进行实验对比, 验证所提出算法的有效性. 相关测试算例下载链接为 https://pan.baidu.com/s/1KzREF_V32JO0C-Ki_iIAjg, 提取码为 xkf9.

4.1 评价指标

在求解 EFJSP-BS 问题时, 采用解集覆盖率 (SC) (见第 3.2 节状态定义)、反世代距离 (IGD) 两个指标, 共同评价帕累托前沿 (PF) 的质量.

IGD 是衡量获得解集质量的重要指标, 能够综合反映多目标优化算法在收敛性和多样性两方面的

表现. 具体而言, 对于真实帕累托最优前沿集PF*中的每个解y, 找到其在算法所得解集PF中距离最近的解x, 计算欧氏距离并求和, 最终对PF*中所有解求平均值且不开方, 形成IGD值. 当PF*中点数多于PF时, IGD能够完整地反映算法所得到的解相对于真实帕累托前沿的代表性和广度. 因此, IGD值越小, 算法多样性和收敛性越好. 定义如下所示:

$$IGD(PF, PF^*) = \frac{1}{|PF^*|} \sum_{y \in PF^*} d(y, x). \quad (24)$$

其中: $d(y, x)$ 为解y与解z间的欧氏距离, $|PF^*|$ 为真实累托前沿中解的个数.

4.2 参数设置

在Q-learning算法中学习率 α 、折扣因子 γ 以及初始贪婪因子 ε 是影响算法性能和收敛行为的核心参数, 其合理设置对于算法在复杂问题中的表现具有关键作用. 因此, 对3个参数设计田口正交实验(DOE), 并以IGD为指标选取参数组合. 3个参数区间选择为(0.5, 0.6, 0.7, 0.8, 0.9), 生成 $L_{25}^{(5)}$ DOE正交阵列, 将25组进行实验, 每组实验进行10次取平均值生成如图9所示的均值主效应, 根据IGD值来评判每组参数匹配的好坏. 由图9得出最优参数组合为 $\alpha = 0.8$ 、 $\gamma = 0.9$ 、 $\varepsilon = 0.5$.

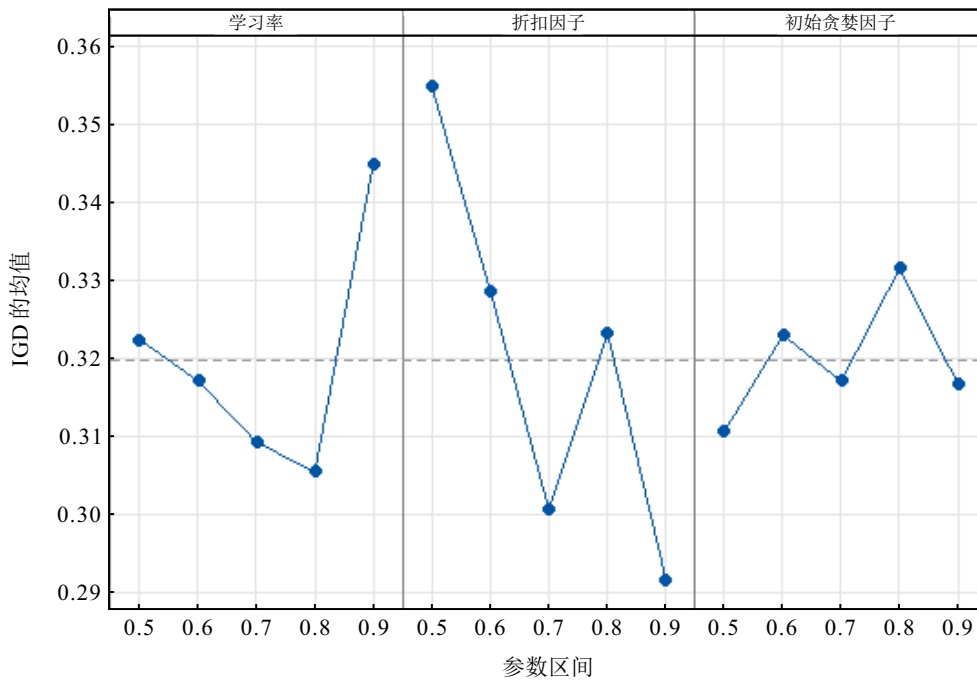


图9 均值主效应

4.3 有效性实验

为验证RLNSGA-II中所提出策略的有效性, 本文构建两种变体算法: ARLNSGA-II (不包含局部搜索策略) 和 BRLNSGA-II (未引入Q-learning自适应策略). 采用第4.1节所述两种评价指标对3种算法

表3 IGD值对比

实验算例	RLNSGA-II	ARLNSGA-II	BRLNSGA-II
BSC01	0.220486	0.351586	0.343769
BSC02	0.250121	0.303637	0.412753
BSC03	0.236028	0.345819	0.639138
BSC04	0.179586	0.377610	0.274976
BSC05	0.239105	0.368030	0.300618
BSC06	0.195121	0.240990	0.305349
BSC07	0.150072	0.297856	0.223117
BSC08	0.110561	0.207764	0.141692
BSC09	0.168002	0.300342	0.242449
BSC10	0.063972	0.158390	0.186725

性能进行比较. 为减少随机性的影响, 每个算例均独立运行5次, 并取平均结果进行统计分析. 表3和表4分别为在10个测试算例下, RLNSGA-II、

表4 SC值对比

实验算例	RLNSGA-II vs. ARLNSGA-II		RLNSGA-II vs. BRLNSGA-II	
	SC(R, A)	SC(A, R)	SC(R, B)	SC(B, R)
	BSC01	0.82508	0.08902	0.57322
BSC02	0.5	0.43164	0.88522	0.06364
BSC03	0.6	0.37846	0.86668	0
BSC04	0.8	0.2	0.59	0.4
BSC05	0.50984	0.3875	0.75682	0.1875
BSC06	0.5497	0.37262	0.76666	0.18888
BSC07	0.8	0.2	0.79166	0.19376
BSC08	0.84048	0.18932	0.78182	0.18462
BSC09	0.43076	0.35426	0.57334	0.35966
BSC10	0.49372	0.44424	0.98182	0.0091

ARLNSGA-II 和 BRLNSGA-II 三种算法在 IGD 和 SC 评价指标上的性能对比, 其中最优结果用粗体表示.

表 3 和表 4 中对比 RLNSGA-II 与 ARLNSGA-II 可得到, RLNSGA-II 在 IGD 值的对比下均为较小于 ARLNSGA-II, 且 SC 值均大于 ARLNSGA-II, 验证了所设计的 3 种邻域搜索策略的有效性. 对比 RLNSGA-II 与 BRLNSGA-II 可得到, RLNSGA-II 在 IGD 值的对比下均为较小于 BRLNSGA-II, 且 SC 值均大于 BRLNSGA-II, 验证了使用 Q-learning 算法来自适应调整 P_c 、 P_m 的值是有效的.

4.4 对比实验

为了更全面地评估所提出 RLNSGA-II 算法, 将其与现有文献中所研究的其他 3 种算法进行比较, 分别为 DQNSGA^[17]、IGA^[18] 和 NSGA-II^[19]. 为公平比较, 所有对比算法的参数均基于文献推荐值或通过实验调优确定. 表 5 和表 6 分别为 4 种算法 (RLNSGA-II、DQNSGA、IGA 和 NSGA-II) 在 10 个算例下基于 IGD 和 SC 指标的评估结果, 最优值均已加粗标示. 为了更直观地呈现数据分布特征和算法差异, 基于表 5 数据绘制了箱线图 (如图 10 所示), 基于表 6 数据绘制了箱线图 (如图 11 所示), 通过箱

表5 4种算法在 IGD 值上的比较

实验算例	RLNSGA-II	DQNSGA	IGA	NSGA-II
BSC01	0.339756	0.341273	0.369242	0.355040
BSC02	0.290761	0.318495	0.342513	0.299812
BSC03	0.249833	0.279447	0.304051	0.290702
BSC04	0.272795	0.336331	0.405558	0.317179
BSC05	0.191432	0.284933	0.332827	0.244438
BSC06	0.201988	0.242228	0.234578	0.222089
BSC07	0.205399	0.267329	0.254507	0.224915
BSC08	0.153394	0.175332	0.345798	0.172862
BSC09	0.170985	0.185736	0.362030	0.205159
BSC10	0.133659	0.189534	0.339874	0.194971

表6 4种算法在 SC 值上的比较

实验算例	RLNSGA-II vs. DQNSGA		RLNSGA-II vs. IGA		RLNSGA-II vs. NSGA-II	
	SC(R,D)	SC(D,R)	SC(R,I)	SC(I,R)	SC(R,N)	SC(N,R)
BSC01	0.6299	0.30865	0.59167	0.35615	0.57965	0.36041
BSC02	0.52111	0.34077	0.7	0.19485	0.50345	0.32479
BSC03	0.47692	0.42364	0.6	0.19886	0.45449	0.36303
BSC04	0.56428	0.34273	0.6	0.27441	0.46777	0.38963
BSC05	0.50494	0.38348	0.9	0.04706	0.45559	0.41514
BSC06	0.51666	0.37809	0.53651	0.2164	0.47167	0.39938
BSC07	0.4905	0.4111	0.6	0.32776	0.5	0.47036
BSC08	0.45066	0.37645	0.96667	0.02143	0.36899	0.41971
BSC09	0.56666	0.30059	0.8125	0.0125	0.575	0.26458
BSC10	0.53182	0.30345	0.74833	0.07839	0.45421	0.38735

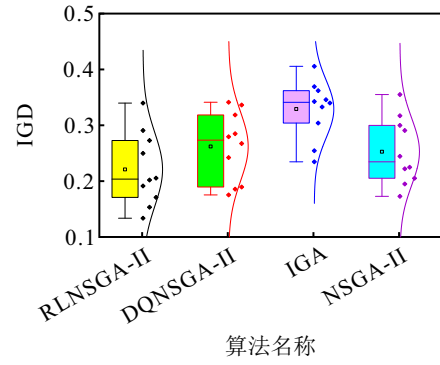


图10 IGD 值箱线图对比

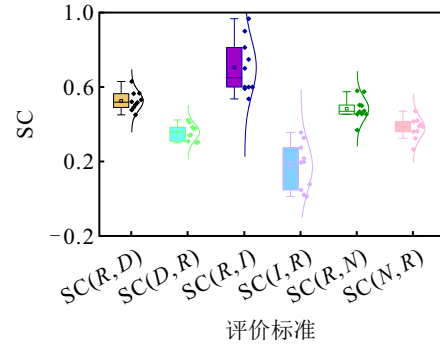


图11 SC 值箱线图对比

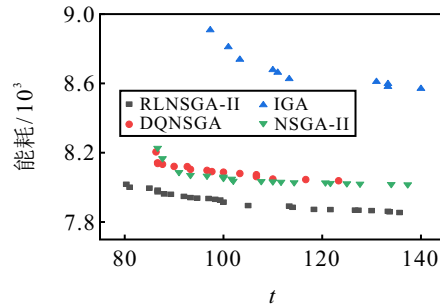
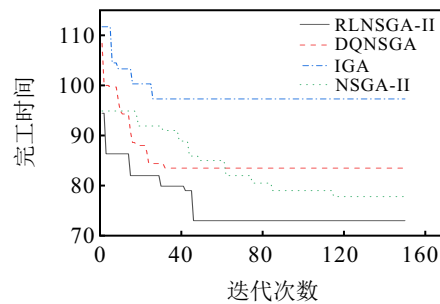
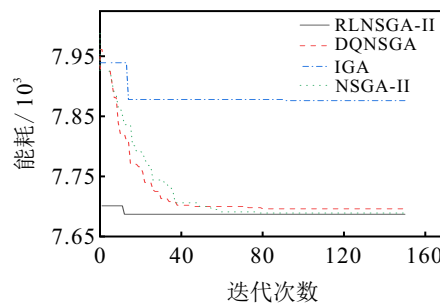


图12 4种算法 BSC05 算例的非支配前沿对比



(a) 最大完工时间收敛曲线



(b) 机器加工总能耗收敛曲线

图13 BSC05 算例收敛曲线对比

线图可以更清晰地观察各算法性能的稳定性和离散程度。

由表 5 的 IGD 指标结果可以看出, RLNSGA-II 在所有测试算例中均优于 DQNSGA、IGA 与 NSGA-II, 表明其在收敛性和多样性方面均具有更好的综合性能。图 10 进一步显示, RLNSGA-II 整体明显优于 IGA, 并略优于 NSGA-II 与 DQNSGA。表 6 的 SC 指标对比中, RLNSGA-II 在大多数算例中也优于其他 3 种算法。图 11 中, $SC(R, D)$ 、 $SC(R, I)$ 和 $SC(R, N)$ 均显著高于对应的 $SC(D, R)$ 、 $SC(I, R)$

与 $SC(N, R)$, 进一步验证了 RLNSGA-II 的优越性。

为了更清晰地展示不同算法的性能差异, 本文以 BSC05 算例为代表进行以下算法对比。图 12 对比了 RLNSGA-II、IGA、NSGA-II 与 DQNSGA 四种算法的非支配前沿, 结果显示 RLNSGA-II 的非支配前沿分布最为均匀。在解的质量方面, RLNSGA-II 明显优于 DQNSGA、IGA 与 NSGA-II。图 13 为两个目标下 4 种算法的收敛曲线。图 14 为 BSC05 算例使用 RLNSGA-II 算法求解得到的甘特图。

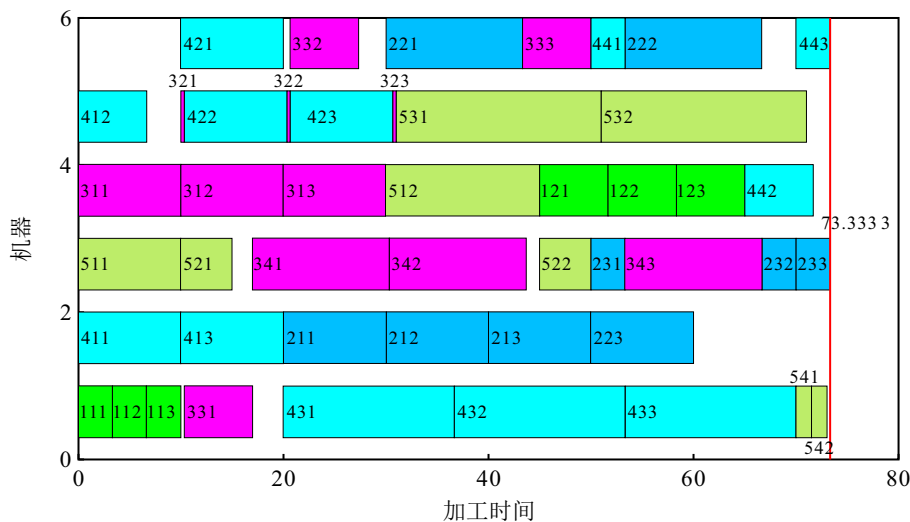


图14 BSC05 算例甘特图

5 结论与展望

本文研究了柔性作业车间节能分批调度问题 (EFJSP-BS), 总结得出以下结论:

- 1) 分批方法设计, 提出了一种规则数字化分批法划分方案;
- 2) 为提升初始种群的质量, 针对该问题模型制定了 3 种不同的种群初始化策略;
- 3) 根据分批调度问题特性设计了 3 种邻域搜索策略, 以保证目标在迭代的过程中能够迅速、精确地向最优解收敛, 同时, 算法引入了 Q -learning 使得交叉率和变异率能够根据种群进化状态自适应调整。

本文研究了分批调度问题中一致分批的问题, 然而, 在柔性作业车间调度的过程中存在很多不同的分批问题, 如不一致分批问题、动态调度中的分批问题等。因此, 未来的研究可进一步扩大研究范围, 探索针对不一致分批、动态调度分批问题等的优化策略。此外, 在算法方面, 后续工作将进一步对深度强化学习进行学习研究, 针对不同分批问题的特性, 设计更加智能和高效的算法框架。

参考文献 (References)

[1] 罗聪, 龚文引. 混合分解多目标进化算法求解绿色置

换流水车间调度问题[J]. 控制与决策, 2024, 39(8): 2737-2745.
 (Luo C, Gong W Y. A hybrid multi-objective evolutionary algorithm based on decomposition for green permutation flow-shop scheduling problem[J]. Control and Decision, 2024, 39(8): 2737-2745.)
 [2] Brandimarte P. Routing and scheduling in a flexible job shop by tabu search[J]. *Annals of Operations Research*, 1993, 41(3): 157-183.
 [3] 徐新胜, 吴松泽, 杜文, 等. 考虑低碳的柔性作业车间分批调度方法[J]. 中国机械工程, <https://link.cnki.net/urlid/42.1294.TH.20250325.1330.015>.
 (Xu X S, Wu S Z, Du W, et al. Consideration of low-carbon flexible job shop batch scheduling methods[J]. Chinese Journal of Mechanical Engineering, <https://link.cnki.net/urlid/42.1294.TH.20250325.1330.015>.)
 [4] 周靖雯, 胡志栋, 彭万超. 基于改进灰狼算法的批量流混合流水车间节能调度问题研究[J]. 机械设计, 2025, 42(4): 86-94.
 (Zhou J W, Hu Z D, Peng W C. Research on energy-saving scheduling of lot-streaming hybrid flow shop based on improved grey wolf optimization algorithm[J]. *Journal of Machine Design*, 2025, 42(4): 86-94.)
 [5] Shao S, Xu G C, Li J X, et al. A job assignment scheduling algorithm with variable sublots for lot-streaming flexible job shop problem based on

- NSGAI[J]. *Computers & Operations Research*, 2025, 173: 106866.
- [6] 刘雪红, 段程, 王磊. 基于改进候鸟算法的柔性作业车间分批调度问题[J]. *计算机集成制造系统*, 2021, 27(11): 3185-3195.
(Liu X H, Duan C, Wang L. Flexible job shop scheduling with lot streaming based on improved migrating birds optimization algorithm[J]. *Computer Integrated Manufacturing Systems*, 2021, 27(11): 3185-3195.)
- [7] Wang H, Peng T, Li X Y, et al. An integrated simulation-optimization method for flexible assembly job shop scheduling with lot streaming and finite transport resources[J]. *Computers & Industrial Engineering*, 2025, 200: 110790.
- [8] 耿凯峰, 叶春明. 带工序跳跃的绿色混合流水车间机器与 AGV 联合调度[J]. *控制与决策*, 2022, 37(10): 2723-2732.
(Geng K F, Ye C M. Joint scheduling of machines and AGVs in green hybrid flow shop with missing operations[J]. *Control and Decision*, 2022, 37(10): 2723-2732.)
- [9] Wang J H, Wu C Y, Peng Y T. Multi-objective scheduling for an energy-efficient flexible job shop problem with peak power constraint[J]. *Applied Soft Computing*, 2024, 167: 112330.
- [10] 王建华, 潘宇杰, 孙瑞. 自适应 Jaya 算法求解多目标柔性车间绿色调度问题[J]. *控制与决策*, 2021, 36(7): 1714-1722.
(Wang J H, Pan Y J, Sun R. Multi-objective flexible job shop green scheduling problem with self-adaptive Jaya algorithm[J]. *Control and Decision*, 2021, 36(7): 1714-1722.)
- [11] Wang W T, Zhao J. An enhanced memetic algorithm for energy-efficient and low-carbon flexible job shop scheduling problem considering machine restart[J]. *Journal of Manufacturing Systems*, 2025, 80: 457-478.
- [12] Chen R H, Yang B, Li S, et al. A self-learning genetic algorithm based on reinforcement learning for flexible job-shop scheduling problem[J]. *Computers & Industrial Engineering*, 2020, 149: 106778.
- [13] Zhang G H, Yan S F, Song X H, et al. Evolutionary algorithm incorporating reinforcement learning for energy-conscious flexible job-shop scheduling problem with transportation and setup times[J]. *Engineering Applications of Artificial Intelligence*, 2024, 133: 107974.
- [14] 尹爱军, 闫文涛, 张厚望. 面向多目标柔性作业车间调度的强化学习 NSGA-II 算法[J]. *重庆大学学报*, 2022, 45(10): 113-123.
(Yin A J, Yan W T, Zhang H W. Reinforcement learning NSGA-II for multi-objective flexible job shop scheduling[J]. *Journal of Chongqing University*, 2022, 45(10): 113-123.)
- [15] 闫富乾, 石致远, 王立闻. 基于改进灰狼算法的柔性作业车间动态节能分批调度问题[J]. *现代制造工程*, 2024(1): 24-32.
(Yan F Q, Shi Z Y, Wang L W. Dynamic energy-saving batch scheduling of flexible job shop problem based on improved grey wolf optimization algorithm[J]. *Modern Manufacturing Engineering*, 2024(1): 24-32.)
- [16] 张国辉, 李志霄, 张利平, 等. 基于强化学习协同进化算法求解柔性作业车间节能调度问题[J]. *计算机应用研究*, 2025, 42(7): 2039-2047.
(Zhang G H, Li Z X, Zhang L P, et al. Reinforcement learning based co-evolutionary algorithm for solving flexible job shop energy efficient scheduling problem[J]. *Application Research of Computers*, 2025, 42(7): 2039-2047.)
- [17] Tang H T, Xiao Y, Zhang W, et al. A DQL-NSGA-III algorithm for solving the flexible job shop dynamic scheduling problem[J]. *Expert Systems with Applications*, 2024, 237: 121723.
- [18] Zhang G H, Hu Y F, Sun J H, et al. An improved genetic algorithm for the flexible job shop scheduling problem with multiple time constraints[J]. *Swarm and Evolutionary Computation*, 2020, 54: 100664.
- [19] Yuan M H, Li Y D, Zhang L Z, et al. Research on intelligent workshop resource scheduling method based on improved NSGA-II algorithm[J]. *Robotics and Computer-Integrated Manufacturing*, 2021, 71: 102141.

作者简介

张国辉 (1980-), 男, 教授, 博士, 主要研究方向为智能优化算法、车间调度, E-mail: zgh09@zua.edu.cn;

伦伟航 (2000-), 男, 硕士生, 主要研究方向为智能优化算法与智能制造, E-mail: 2928252627@qq.com;

李亮 (1986-), 男, 中级经济师, 主要研究方向为精益生产、制造数字化转型, E-mail: liwqqn@126.com;

沈克伟 (1980-), 男, 研究员级高级工程师, 主要研究方向为生产过程优化与效率提升、供应链协同与生产一体化管理, E-mail: 30708790@qq.com;

李峥峰 (1976-), 男, 副教授, 博士, 主要研究方向为智能制造、智能优化算法, E-mail: zhengfengli@zut.edu.cn.