華東理工大學学报 自然科学版 Natural Science Edition
JOURNAL OF EAST CHINA UNIVERSITY OF SCIENCE AND TECHNOLOGY

考虑设备故障的铝卷批处理可重入两阶段协同调度问题

白云峰 徐震浩 顾幸生

Collaborative Two-Stage Reentrant Batch Scheduling Problem for Aluminum Coils Considering Equipment Failures

BAI Yunfeng, XU Zhenhao, GU Xingsheng

在线阅读 View online: http://doi.org/10.14135/j.cnki.1006-3080.20260110001

您可能感兴趣的其他文章

Articles you may be interested in

离散水波优化算法求解带批处理的混合流水车间批量流调度问题

Discrete Water Wave Optimization Algorithm for Hybrid Flowshop Lot-Streaming Scheduling Problem with Batch Processing

华东理工大学学报（自然科学版）. 2021, 47(5): 598-608 http://doi.org/10.14135/j.cnki.1006-3080.20200831001

基于头脑风暴算法的多处理机组合生产批量调度问题

Multi-Processor Combined Production Batch Scheduling Problem Based on Brain Storm Optimization Algorithm

华东理工大学学报（自然科学版）. 2022, 48(5): 685-695 http://doi.org/10.14135/j.cnki.1006-3080.20210427005

基于Pareto支配的两阶段多目标优化算法

Two Stage Multi-Objective Optimization Algorithm Based on Pareto Dominance

华东理工大学学报（自然科学版）. 2022, 48(6): 806-815 http://doi.org/10.14135/j.cnki.1006-3080.20210530001

高层综合中面向运算器电源门控技术的低能耗调度算法

A Novel Low Power Scheduling Algorithm for Functional Unit Power Gating in High-Level Synthesis

华东理工大学学报（自然科学版）. 2022, 48(4): 543-548 http://doi.org/10.14135/j.cnki.1006-3080.20210308002

基于多目标离散正弦优化算法的混合零空闲置换流水车间调度

Mixed No-Idle Permutation Flow Shop Scheduling Problem Based on Multi-Objective Discrete Sine Optimization Algorithm

华东理工大学学报（自然科学版）. 2022, 48(1): 76-86 http://doi.org/10.14135/j.cnki.1006-3080.20201201005

基于强化学习的正弦优化算法求解能耗分布式流水车间节能调度问题

Sine Optimization Algorithm Based on Reinforcement Learning for Energy-Efficient Distributed Flow Shop Scheduling Problem

华东理工大学学报（自然科学版）. 2025, 51(5): 623-632 http://doi.org/10.14135/j.cnki.1006-3080.20250117002

扫码关注公众号，获取更多信息！

文章编号：1006-3080(2026)05-0647-11 DOI：10.14135/j.cnki.1006-3080.20260110001

考虑设备故障的铝卷批处理可重入两阶段协同调度问题

白云峰， 徐震浩， 顾幸生
（华东理工大学能源化工过程智能制造教育部重点实验室，上海 200237）

摘要：针对铝卷冷轧和退火阶段设备故障导致的预调度方案失效问题，建立数学模型，提出基于种群状态自适应调节的开普勒优化算法(Kepler Optimization Algorithm with Adaptive Adjustment Based on Population State, KOABPS)，设计了基于种群状态的自适应调节机制，使得算法具备感知搜索环境、动态调整行为的自主决策能力。在工厂数据集上进行了不同重调度方法的比较，并通过消融实验验证了KOABPS算法的有效性。为更深入研究算法性能，设计不同规模的随机实验，并将KOABPS算法与其他优化算法进行对比分析，结果表明该算法在求解质量和收敛速度方面均具有优势。

关键词：铝卷生产；批处理；可重入调度；设备故障；开普勒优化算法(KOA)

中图分类号：TP301 **文献标志码：**A

可重入调度问题是指工件在加工流程中需重复回到之前经过的某一加工阶段或工位再次处理的调度问题。Wu等[1]提出一种改进型多智能体近邻策略优化算法，融入了自适应信任域裁剪方法。Liu等[2]设计了融合长期与短期收益的奖励塑造技术，引入注意力机制网络以提升特征提取能力和决策效率。Wu等[3]整合毕达哥拉斯模糊集、黄金分割因子和莱维飞行机制，用于求解运输时间不确定下考虑碳排放的调度问题。He等[4]提出迭代构建局部搜索-重构算法，该算法通过策略性整合具有不同功能定位的构建、局部搜索和重构过程以实现优化。批处理调度问题是指多工件可在同一台机器上同时加工，通过组批优化以提升设备利用率。Lei等[5]针对织物染色过程中含机器资格约束的并行批量处理机调度问题，设计了自适应混合蛙跳算法。Song等[6]通过融入贪心启发式和交叉熵思想，对传统免疫遗传算法进行改进。Sun等[7]提出融合初始化启发式和非支配排序的反馈式教与学优化算法，用于求解柔性装配作业车间调度问题。Rocholl等[8]提出结合分组遗传算法或局部搜索的迭代分解方法，用于求解含批量处理机的两阶段柔性流水车间调度问题。

尽管上述研究在可重入或批处理调度方面取得进展，但在面向铝卷生产的复杂工业场景中仍存在不足：多数研究仅聚焦单一加工阶段，而铝卷实际生产中，冷轧与退火高度耦合，同一铝卷需多次往返于两阶段才能完成加工，形成可重入流程；现有工作较少同时考虑可重入与批处理特性，在退火阶段，铝卷可能按材质与规格进行组批，形成可重入与组批共存的生产特性；对设备故障扰动的考虑不足，退火炉作为高负荷长周期运行设备，其故障概率显著高于其他设备[9]。一旦发生故障，不仅会使当前操作中断，还会影响后续操作。由于退火炉组批的特性，设备故障还可能拆散原本组成的批次，对后续操作产生较大影响。

开普勒优化算法(Kepler Optimization Algorithm, KOA)是一种新型元启发式算法。Hu等[10]提出融合自适应函数和正弦混沌引力策略的改进型KOA，在CEC-2017和CEC-2020基准测试集上进行了全面验

收稿日期：2026-01-10
基金项目：国家自然科学基金(62076095)
作者简介：白云峰(2000—)，男，山西阳泉人，硕士生，主要研究方向为生产计划与调度、智能优化算法。E-mail：y30230936@mail.ecust.edu.cn
通信联系人：徐震浩，E-mail：xuzhenhao@ecust.edu.cn
引用本文：白云峰，徐震浩，顾幸生. 考虑设备故障的铝卷批处理可重入两阶段协同调度问题[J]. 华东理工大学学报(自然科学版)，2026，52(5)：647-657.
Citation: BAI Yunfeng, XU Zhenhao, GU Xingsheng. Collaborative Two-Stage Reentrant Batch Scheduling Problem for Aluminum Coils Considering Equipment Failures[J]. Journal of East China University of Science and Technology, 2026, 52(5): 647-657.

证。Li 等[11] 通过结合索伯尔序列种群初始化、正余弦算法与变异-对立学习机制，设计了改进型 KOA，并在 10 个基准测试函数中验证了其性能。然而，鲜有研究将 KOA 应用于可重入批处理调度问题，尤其在含设备故障扰动的动态环境下。因此，本文针对铝卷生产中的冷轧和退火阶段的协同调度问题，考虑工艺流程限制和退火炉故障等影响因素，首次构建了相应的调度模型，提出了一种基于种群状态自适应调节的开普勒优化算法 (Kepler Optimization Algorithm with Adaptive Adjustment Based on Population State, KOABPS)，并进行了消融实验和不同规模的对比实验，结果表明 KOABPS 算法在求解质量和收敛速度方面均具有优势。

1 问题描述和数学模型

1.1 问题描述

冷轧和退火阶段是铝卷实际生产过程中的关键步骤。铝卷经热轧后进入冷轧和退火阶段。冷轧阶段配备多台冷轧机，但只能单卷处理；退火阶段配备多卷退火炉与单卷退火炉两类设备，前者可批量加工铝卷、后者仅能单卷处理。为了提高效率，退火阶段需进行组批，组批时要满足与退火工艺号一致、材料类型相同，入口宽度处于同一区间，且退火开始前各铝卷末次冷轧完成时间间隔不超过上限等条件。铝卷的工艺流程不同，部分铝卷会重复进行冷轧或退火操作，形成可重入加工路径。由于产能和库存的限制，一段时间内单台冷轧机的冷轧操作和所有退火炉的退火操作数量有上限，不同材料类型铝卷的产量也存在限制。铝卷在完成一次操作后要冷却一段时间。冷轧和退火操作对应的设备任选，且各自加工时间相同。表 1 示出了部分铝卷的信息，铝卷加工按照工艺流程从左到右依次进行，其中 C 表示冷轧操作，A 表示退火操作。工艺号与操作一一对应，材料类型有 P、T、F 3 种，入口宽度区间划分为 [1 000,1 250) 和 [1 250,1 500)。在实际生产中，退火炉的故障概率显著高于其他设备。当故障发生时，不仅直接干扰当前铝卷的生产，更会通过可重入流程的耦合性和批处理的约束关联性产生连锁影响，从而提高调度问题的复杂度。

表 1 铝卷部分信息表

Table 1 Table of partial information on aluminum coils

Coil	Process route	Process number	Material type	Entry width/mm
1	C, A, C, C	500, 590, 600, 610	P	1 470
2	C, A	500, 600	P	1 470
3	C, A, A	500, 600, 610	F	1 050
4	C, C, A, C, A	500, 510, 520, 580, 590	T	1 250
…	…	…	…	…

1.2 数学模型

根据生产特性，进行如下假设：所有设备在零时刻均处于就绪状态；任意加工阶段的缓冲区容量无限；当设备发生故障时，未完成操作需转移至其他同类型可用设备重新加工；设备可在特定时间内完成修复。

式 (1) 展示了目标函数：重调度后的最大完成时间 $C_{\max}$；式 (2) 表示设备故障结束时间；式 (3) 表示在故障期间无铝卷加工处理；式 (4) 表示冷轧和退火操作需冷却一段时间之后进行下一操作；式 (5)~(16) 表示在预调度和重调度下均适用的工艺特性，即式 (5) 和式 (6) 表示冷轧和退火操作的完成时间；式 (7) 表示单个冷轧机或单卷退火炉，同一时间不能进行多个冷轧或退火操作，其中 A 为正无穷大；式 (8) 表示单个多卷退火炉同一时间内不能加工多个批次；式 (9) 表示工艺号相同且入口宽度处于同一区间的铝卷才能组批进行退火操作；式 (10) 表示对于同一退火批次的铝卷，退火操作开始前各自的最后一次冷轧操作完成时间的时间间隔不超过最大时间间隔；式 (11) 表示每个操作 o_{ir} 最多只能分配到一台设备上进行加工；式 (12) 表示每个退火操作 o_{ir} 最多只能分配到一个批次；在时间范围 $t_0 \sim t$ 内，式 (13) 表示单台冷轧机的冷轧操作数量不超过其产能上限；式 (14) 表示所有退火炉的总退火操作次数不超过产能上限；式 (15) 表示生产不同材料类型的铝卷数量不超过产量上限；式 (16) 表示在 t 时刻，位于设备 m_{sj} 上加工的操作数量不超过设备容量。

$$C_{\max} = \max(T'_{\mathrm{EO}_{ir}}) - \min(T'_{\mathrm{SO}_{ir}}), \forall i \in I, r \in D_i \tag{1}$$

$$T_{sj,\mathrm{start}} + F_{sj} D_{sj} = T_{sj,\mathrm{end}}, \forall s \in S, \forall j \tag{2}$$

$$Z'_{irsj} T'_{\mathrm{SO}_{ir}} \geqslant T_{sj,\mathrm{end}} \& Z'_{irsj} T'_{\mathrm{EO}_{ir}} \leqslant T_{sj,\mathrm{start}}, \forall i \in I, r \in D_i \tag{3}$$

$$T_{\mathrm{SO}_{ir}} \geqslant \begin{cases} T_{\mathrm{EH}i} + T_{\mathrm{WH}}, \forall i \in I, r = 1, \\ T_{\mathrm{EO}_{i(r-1)}} + T_{\mathrm{W}}, \forall i \in I, r = 2,3,\ldots,n_i \end{cases} \tag{4}$$

$$T^{(')}_{\mathrm{EO}_{ir}} = T^{(')}_{\mathrm{SO}_{ir}} + X_{ir} T_s, \forall i \in I, r \in D_i \tag{5}$$

$$\begin{cases} T^{(')}_{\mathrm{E}k^{(')}} = T^{(')}_{\mathrm{S}k^{(')}} + Y_{ir} T_s, \text{if } W^{(')}_{irk^{(')}} = 1 \\ T^{(')}_{\mathrm{EO}_{ir}} = T^{(')}_{\mathrm{SO}_{ir}} + Y_{ir} T_s \\ \forall i \in I, r \in D_i, k^{(')} \in B^{(')} \end{cases} \tag{6}$$

$$\begin{cases} T^{(')}_{\mathrm{Si}_a r_a} \geqslant T^{(')}_{\mathrm{Ei}_b r_b} - A(1 - H^{(')}_{i_a r_a i_b r_b}) \\ T^{(')}_{\mathrm{Si}_b r_b} \geqslant T^{(')}_{\mathrm{Ei}_a r_a} - A H^{(')}_{i_a r_a i_b r_b} \end{cases} \tag{7}$$
$$Z^{(')}_{i_a r_a s j} = Z^{(')}_{i_b r_b s j} = H^{(')}_{i_a r_a i_b r_b} = 1, \forall i_a, i_b \in I, r_a, r_b \in D_i$$

$$\begin{cases} T^{(')}_{\mathrm{Sk}^{(')}_a} \geqslant ET^{(')}_{k^{(')}_b} - A(1 - HH^{(')}_{k^{(')}_a k^{(')}_b}) \\ T^{(')}_{\mathrm{Sk}^{(')}_b} \geqslant ET^{(')}_{k^{(')}_a} - A HH^{(')}_{k^{(')}_a k^{(')}_b} \end{cases} \tag{8}$$
$$U^{(')}_{k^{(')}_a s j} = U^{(')}_{k^{(')}_b s j} = HH^{(')}_{k^{(')}_a k^{(')}_b} = 1, \forall k^{(')}_a, k^{(')}_b \in B$$

$$\begin{cases} N_{Oi_a r_a} = N_{Oi_b r_b} \\ W_{IDi_a r_a} = W_{IDi_b r_b} \\ W^{(')}_{i_a r_a k^{(')}} + W^{(')}_{i_b r_b k^{(')}} = 2 \end{cases} \tag{9}$$
$$\forall i_a, i_b \in I, r_a, r_b \in D_i, \forall i_a \neq i_b$$

$$\begin{cases} \max\{\left| T^{(')}_{\mathrm{EO}_{i_a l}} - T^{(')}_{\mathrm{EO}_{i_b l'}} \right|\} \leqslant AT \\ W_{i_a l k^{(')}} + W_{i_b l' k^{(')}} = 2 \\ \begin{cases} l = \max\{r_a | X_{ir_a} = 1\} \\ l' = \max\{r_b | X_{ir_b} = 1\} \end{cases} \end{cases} \tag{10}$$
$$\forall i_a, i_b \in I, r_a, r_b \in D_i, \forall i_a \neq i_b$$

$$\sum_{s,j=1,1}^{N_s, n_s} Z^{(')}_{irsj} = 1, \forall i \in I, r \in D_i \tag{11}$$

$$\sum_{k^{(')}=1}^{N_{B^{(')}}} W^{(')}_{irk^{(')}} \leqslant 1, \forall i \in I, r \in D_i \tag{12}$$

$$\sum_{i,r \in \Omega} X_{ir} Z^{(')}_{irsj} \leqslant O_{Msj(t_0 \sim t)}, \forall j, \Omega = \{i, r | t_0 \leqslant T^{(')}_{\mathrm{EO}_{ir}} \leqslant t\} \tag{13}$$

$$\sum_{i,r \in \Omega} \sum_{j=1}^{n_s} Y_{ir} Z^{(')}_{irsj} \leqslant \sum_{j=1}^{n_s} O_{Msj(t_0 \sim t)}, \Omega = \{i, r | t_0 \leqslant T^{(')}_{\mathrm{EO}_{ir}} \leqslant t\} \tag{14}$$

$$\sum_{i \in \Omega} V_{ai} \leqslant P_{Ma(t_0 \sim t)}, \forall a, \Omega = \{i | t_0 \leqslant T^{(')}_{\mathrm{EO}_{ir}} \leqslant t\} \tag{15}$$

$$\sum_{i,r \in \Omega} Z^{(')}_{irsj} \leqslant C_{sj}, \forall s \in S, \forall j, \Omega = \{i, r | T_{\mathrm{SO}} \leqslant t \leqslant T^{(')}_{\mathrm{EO}_{ir}}\} \tag{16}$$

上述模型中, $\cdot$ 表示预调度参数, $\cdot'$ 表示重调度参数。模型中存在一些预调度和重调度下的形式完全相同的约束。为了高效表示这类约束, 采用 $\cdot^{(')}$ 来表示 $\cdot$ 和 $\cdot'$。

2　基于种群状态自适应调节的开普勒优化算法

本文基于基础 KOA 算法, 提出了 KOABPS 算法。图 1 示出了 KOABPS 算法的流程图。KOABPS 算法结合铝卷冷轧退火阶段交叉耦合、复杂的可重入和组批等特点, 提出了基于分级的动态解码, 改进了初始化种群方法, 设计了基于种群状态自适应机制, 采用了 Levy 飞行增强探索。

2.1　编码

针对该调度问题特性, 采用实数编码方式: 将所有铝卷的全部操作串行排列, 操作编码的整数部分表示加工设备, 小数部分表示加工顺序, 整数相同则按小数升序加工。鉴于冷轧与退火工序交叉耦合, 退火阶段生产安排依赖前序冷轧和退火流程, 若提前编码退火操作易违反工艺流程生成不可行解、降低算法收敛性, 则仅对冷轧操作进行实数编码, 退火操作统一置 0。

2.2　基于分级的动态解码

多卷退火炉的操作特性导致每次铝卷进入退火阶段时都需要考虑组批, 但组批需兼顾当前退火操作及前序冷轧安排。如果没有从冷轧和退火阶段整体进行协调, 有可能造成不可行解。因此本文提出了基于分级的动态解码 (Hierarchy-based Dynamic Decoding, HDD)。根据铝卷的退火操作, 依次设定处理级别。铝卷的处理级别设为多级, 每一级包含当前次退火操作以及在此之前的所有冷轧操作, 以此类推, 解码过程如下: (1) 冷轧操作设备分配: 优先对最低级次的冷轧操作进行设备分配; (2) 退火操作设备分配: 优先对与最低级次的退火操作对应的铝卷进行组批, 若成功组批, 则将其退火操作分配到多卷退火炉, 并升级未成功组批的退火操作及后续操作; 若组批失败, 则随机选取当前级次的退火操作分配到单卷退火炉, 并升级除该铝卷外的其他操作, 转到过程 (1)。

2.3　拉丁超立方采样初始化种群

拉丁超立方采样 (Latin Hypercube Sampling, LHS) 是一种统计抽样的方法, 适用于高维空间中获取样本点。LHS 初始化种群分为分层抽样和维度置换两个步骤[12]。分层抽样是指将种群的每个维度都按照其取值范围划分为若干个大小相等且不重叠的区间, 并且在每个区间内仅抽取一个样本点。维度置换则是指对每个维度已抽取的样本点进行随机重排, 再将重排后的各维度样本点按位置组合, 最终形成分布更均匀的种群。

2.4　基于种群状态自适应调节机制

KOA 算法采用随机搜索策略, 无法适配不同迭代阶段的需求, 也难以应对种群分布异常, 因此引入基于种群状态自适应调节机制 (Adaptive Regulation Mechanism Based on Population State, ARMBPS)。该机制主要分为状态感知、探索因子调节和参数适配 3 部分。状态感知通过种群多样性和收敛改进量表

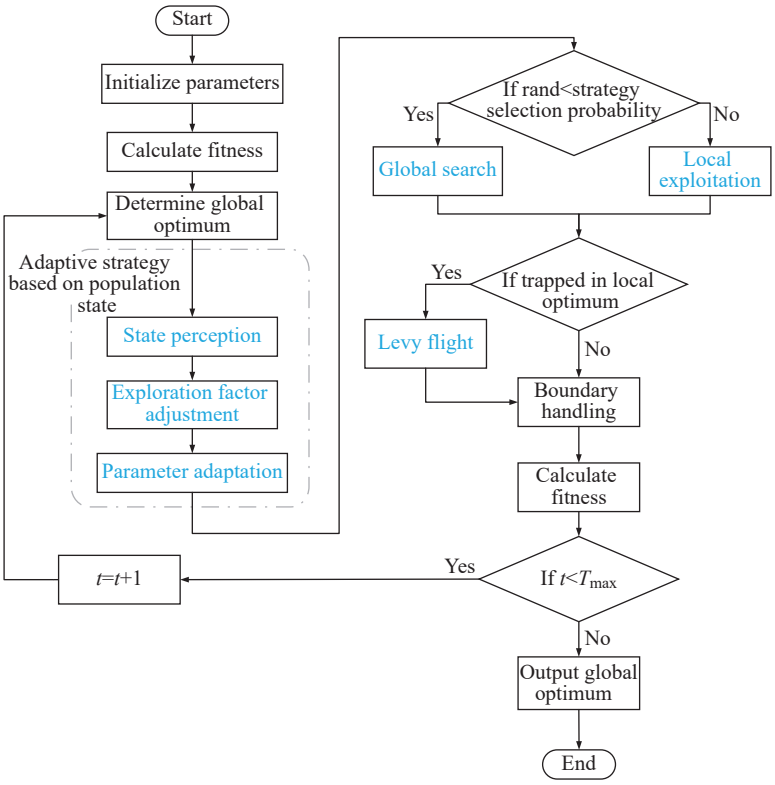


图 1 KOABPS 算法流程图

Fig. 1 Flow chart of the KOABPS algorithm

示。种群多样性如式 (17) 所示，收敛改进量如式 (18) 所示。

$$\mathrm{DIV}_{(t)}=\frac{\sum_{j=1}^{d}\sigma_{j(t)}}{\sum_{j=1}^{d}(ub_j-lb_j)} \tag{17}$$

$$\mathrm{RI}_{(t)}=\begin{cases}\left|\dfrac{\mathrm{fit}_{(t-30)}-\mathrm{fit}_{(t)}}{\mathrm{fit}_{(t-30)}}\right|, t>\dfrac{T_{\max}}{3}\\ 1, t\leqslant\dfrac{T_{\max}}{3}\end{cases} \tag{18}$$

其中，$\mathrm{DIV}_{(t)}$ 表示第 t 代的种群多样性；$\sigma_{j(t)}$ 表示第 t 代第 j 维的标准差；ub_j 和 lb_j 分别表示第 j 维的上限和下限，通过种群多样性可以反映种群的分散程度；$\mathrm{RI}_{(t)}$ 表示第 t 代的收敛改进量；$\mathrm{fit}_{(t)}$ 表示迭代到第 t 代的适应度值；$T_{\max}$ 表示最大迭代次数。通过监测收敛改进量，可以判断算法是否陷入停滞状态。

探索因子如式 (19) 所示，通过探索因子可以平衡算法的探索和开发。初始探索因子为 1.0，表示二者平衡。边界设置为 [0.5,2.0]，既保持了基础的开发能力，又避免了过度探索。参数适配是指基于调整后的探索因子，动态适配算法核心参数。具体包括：(1) 偏心率 $e_{(t)}$，当探索因子增大时，偏心率增大，同时保证不超过 1.0，保证其物理意义；当探索因子减小时，偏心率缩小；当处于平衡状态时，不改变偏心率；(2) 引力系数 $\mathrm{GF}_{(t)}$，当探索因子增大时，减弱引力，避免行星被局部最优解束缚；当探索因子减小时，加速向全局最优解聚集；(3) 半长轴 $A_{i(t)}$，探索因子越大，半长轴越小，轨道周期更短，行星运动更活跃；探索因子越小，半长轴越大，轨道更稳定。策略选择概率决定全局探索与局部开发的执行，KOABPS 通过式 (20) 引入探索因子的线性变化来决定：策略选择概率以平衡值 0.5 为基准，探索因子每增加 1.0，概率提升 0.3，同时限制在 [0.1,0.9] 区间内。

$$\mathrm{EF}_{(t)}=\begin{cases}\min(\mathrm{EF}_{(t-1)}+0.1,2.0), \text{ if } \mathrm{DIV}_{(t)}<0.2\\ \max(\mathrm{EF}_{(t-1)}-0.05,0.5), \text{ if } \mathrm{RI}_{(t)}<0.01\\ \mathrm{EF}_{(t-1)}\end{cases} \tag{19}$$

其中，$\mathrm{EF}_{(t-1)}$ 表示第 $t-1$ 代的探索因子。

$$\mathrm{SP}_{(t)}=\max(\min(0.5+0.3\times(\mathrm{EF}_{(t)}-1),0.9),0.1) \tag{20}$$

2.5 Levy 飞行

Levy 飞行是一种随机行走策略[13]。KOABPS 算法在跳出局部最优的过程中引入 Levy 飞行策略，如式 (21) 所示，其中 $X_{i(t)}$ 表示第 t 次迭代时的个体 i，s 表示 Levy 飞行步长，rand 表示随机方向。当 $\mathrm{RI}_{(t)}<\mathrm{RI}_{th}$ 时，认为算法已陷入局部最优状态，算法将以较高概率启动 Levy 飞行更新过程。

$$X_{i(t+1)}=X_{i(t)}+s\cdot(X_{i(t)}-\mathrm{rand}) \tag{21}$$

3　仿真实验

3.1　构建数据集

本文基于企业实际生产数据构建了两类数据集, 即工厂数据集和随机数据集。工厂数据集共 20 个算例, 记为 FCase 1~FCase 20, 铝卷数量在 [30, 200] 取值, 冷轧机、多卷退火炉和单卷退火炉的数量分别为 4、10、2 台。随机数据集划分为大、中、小 3 种不同规模, 共 60 个算例。若铝卷总数用 J 表示、可用设备数量用 (a, b, c) 表示, 其中 a、b 和 c 分别指冷轧机、多卷退火炉和单卷退火炉的数量, 则算例可表示为 Case J×(a, b, c)。表 2 示出了铝卷工艺流程及比例, 随机数据集算例的铝卷按照该比例选择。随机数据集算例的数据范围为: 小规模数据集: $J \in [15,55]$、$a \in [2,5]$、$b \in [3,6]$、$c \in [2,3]$; 中规模数据集: $J \in [80,120]$、$a \in [3,6]$、$b \in [4,7]$、$c \in [3,5]$; 大规模数据集: $J \in [170,210]$、$a \in [4,7]$、$b \in [5,8]$、$c \in [4,7]$。本文中所有程序均在 MATLAB R2023b 上进行编写并运行, 操作系统为 Windows 10, 处理器为 Intel Core i5-8300。

表 2　铝卷工艺流程及比例

Table 2　Process flow and proportion of aluminum coils

Process route	Ratio/%	Process route	Ratio/%	Process route	Ratio/%
C×1	0.3	C×3, A×1, C×1	4.2	C×1, A×1, C×5, A×1	0.3
C×3	0.2	C×4, A×1, C×1	7.7	C×4, A×1, C×1, A×1	1.0
C×1, A×1	33.6	C×5, A×1, C×1	1.7	C×1, A×1, C×1, A×1, C×1	0.8
C×2, A×1	11.4	C×6, A×1, C×1	0.2	C×3, A×1, C×2, A×1, C×1	0.2
C×3, A×1	12.0	C×8, A×1, C×1	0.2	C×1, A×1, C×5, A×2	0.2
C×4, A×1	8.2	C×9, A×1, C×1	0.5	C×1, A×2	1.7
C×5, A×1	2.5	C×1, A×1, C×1, A×1	0.3	C×2, A×2	2.8
C×6, A×1	0.8	C×1, A×1, C×2, A×1	1.8	C×3, A×2	1.5
C×1, A×1, C×1	1.0	C×1, A×1, C×3, A×1	1.0	C×1, A×2, C×2	0.2
C×2, A×1, C×1	3.3	C×1, A×1, C×4, A×1	0.2	C×3, A×3	0.2

3.2　参数分析

在 KOABPS 算法中, 种群数量 N、迭代次数 $T_{\max}$、搜索精度的初始值 μ_0、下降速度 γ 以及自适应因子 z 会影响算法的性能。响应面法是一种通过多次实验来预测最优参数组合的研究方法[14], 本文采用响应面法, 针对不同规模数据集确定合适的参数组合, 参数范围设置为 $N \in [20,100]$、$T_{\max} \in [200,1\,000]$、$\mu_0 \in [1,4]$、$\gamma \in [1,30]$、$z \in [1,10]$。在小规模数据集的模型方差分析中, “Model”的 F-value 为 2.42, 表示模型具有显著性; “Lack of Fit”的 F-value 为 1.13, 表明失拟相对于纯误差并不显著。图 2 示出了小规模数据集模型的残差正态概率分布, 表明显著性较好。图 3 示出了小规模数据集模型预测值与实际值的比较, 数据点均匀分布在直线两侧。式 (22) 为小规模数据集的模型拟合方程(其中 A、B、C、D、E 分别表示参数 N、$T_{\max}$、μ_0、γ、z)。中大规模的分析过程与小规模类似, 通过分析得到的 A、B、C、D、E 最优参数组合为: 小规模依次为 50、431、3.86、17.67、9.26; 中规模依次为 40、603、3.77、20.39、3.42; 大规模依次为 75、406、1.26、9.46、2.67。

$$\begin{aligned} Y =& 8.38 - 0.038\,7A - 0.010\,0B + 0.008\,3C + 0.025\,0D - \\ & 0.040\,0E - 0.004\,9AB - 0.015\,0AC + \\ & 0.006\,7AD + 0.021\,7AE - 0.070\,8BC + \\ & 0.033\,3BD - 0.015\,8BE - 0.094\,1CD + \\ & 0.053\,3CE + 0.024\,1DE \end{aligned} \tag{22}$$

3.3　不同重调度方法对比

在组批生产的场景中, 完全的右移重调度难以实现, 故采用部分重调度或完全重调度。基于工厂数据集, 采用 KOA 算法并设置相同的参数进行计算, 性能指标采用最优适应度值 (Best Fitness, BF) 和平均适应度值 (Mean Fitness, MF), 通过仿真实验展开分析, 结果如表 3 所示。其中, Partial 表示部分重调度, Complete 表示完全重调度。完全重调度在 20 个工厂数据集算例中的 18 个算例中取得了最优适应度值, 仅在 2 个算例中未达最优。在 MF 中, 完全重调度在 15 个算例中优于部分重调度, 而在未取得最优值的算例中, 其 MF 和最优值的差距不超过 0.4, 因此在后续实验中均采用完全重调度方法。

3.4　KOABPS 消融实验及收敛性分析

为了考察 KOABPS 算法的改进机制的有效性, 针对 KOA 算法和其 3 种变体算法设计了消融实验。即基于 KOA 和拉丁超立方采样的 IKOA-1, 基于 KOA 和种群状态自适应机制的 IKOA-2 以及基于 KOA 和 Levy 飞行的 IKOA-3。图 4 示出了 IKOA-1 和 KOA 算法针对工厂数据集的 3 个算例的初始种群分布, 采用主成分分析法将解空间降维至二维平面进行展示。直观可见, KOABPS 的种群分布区域

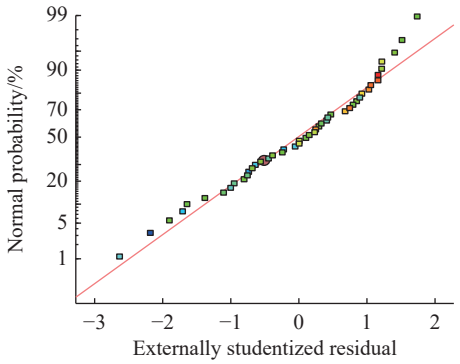


图 2 残差正态概率分布图

Fig. 2 Normal probability plot of residuals

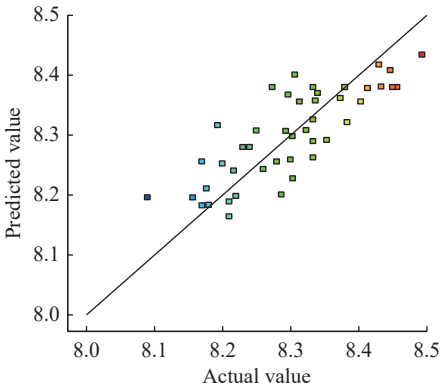


图 3 模型预测插值图

Fig. 3 Prediction interpolation plot of the mode

更广, 个体空间分布更分散, 表明基于拉丁超立方抽样的初始化策略提升了初始化种群的多样性。

表 4 示出了 KOA 和 3 种算法变体的消融实验结果, 评价指标为 BF、标准差(STD)和平均收敛迭代次数(AVGITER)。基础 KOA 表现最差, 仅 4 个算例获得最优 BF, 寻优不充分; 平均 STD 最大, 表明搜索不确定性高, 且绝大多数算例 AVGITER 最高, 收敛速度慢。3 种 KOA 算法变体均优于基础 KOA, IKOA-1 的 BF 平均改进 2.9%, 平均 STD 降至 0.12, 在 4 个算例取得了最优 AVGITER, 表明 LHS 能够初始化提升种群多样性、降低随机性; IKOA-3 的 BF 平均改进 3.1%, 平均 STD 降至 0.11, 在 7 个算例上取得了最优 AVGITER, 反映了 Levy 飞行能增强跳出局部最优能力; IKOA-2 表现最优: BF 平均改进 5.7%, 在 16 个算例上取得了最优 BF, 平均 STD 降至 0.09, 在 8 个算例上取得了最优 AVGITER, 表明自适应调节机制能有效平衡探索开发, 降低搜索偏差、提升效率。

3.5 算法性能比较

为了分析 KOABPS 算法性能, 选取 KOABPS 算法与基础 KOA 算法[15], 解决设备故障调度问题的遗传混沌莱维非线性金枪鱼群优化算法 (GCLNTSO)[16], 解决可重入调度问题的改进灰狼算法 (IGWA)[17], 解决批处理调度问题的融合邻域搜索的遗传算法 (GANS)[18], 解决柔性作业车间调度问题的改进粒子群算法 (IPSO)[19] 和混合帝国主义算法 (EICA-VNS)[20] 进行对比测试。KOABPS 算法的参数如 3.2 节设置; 基础 KOA 算法的参数设置为 $\overline{T}$ =3、γ =15、μ_0 =0.1; GCLNTSO 算法的参数设置为 z =0.05、μ =2; IGWA 算法的参数设置为 a =2; GANS 算法的参数设置为 λ = 0.3、P_c =0.8、P_m =0.3; IPASO 算法的参数设置为 W = 0、y =0.5、ac =2; EICA-VNS 算法的参数设置为 p = 0.3, β_i =4、β_f =10。实验中每个算例均运行 10 次。评价指标为 BF、完成时间的平均延长率 (Average Extension Rate of Completion Time, AECT) 和设备稳定性指标 (Equipment Stability, ES), AECT 和 ES 的计

表 3 不同重调度方法结果对比

Table 3 Comparison of results of different rescheduling methods

FCase	BF		MF		FCase	BF		MF	
	Partial	Complete	Partial	Complete		Partial	Complete	Partial	Complete
1	13.881	**13.281**	14.832	**14.683**	11	34.665	**34.065**	35.630	**35.440**
2	**8.996**	**8.996**	9.964	**9.348**	12	28.296	**25.760**	29.257	**27.136**
3	14.040	**11.197**	14.989	**12.517**	13	33.412	**33.184**	33.875	**33.546**
4	13.060	**10.060**	14.030	**11.380**	14	39.320	**38.355**	40.282	**39.745**
5	15.369	**14.450**	16.317	**15.752**	15	**39.598**	40.270	**40.564**	41.623
6	15.180	**12.880**	16.106	**14.196**	16	48.560	**47.866**	49.468	**49.224**
7	**19.125**	**19.125**	**20.113**	20.487	17	43.913	**43.410**	44.874	**44.757**
8	23.413	**23.358**	**24.344**	24.733	18	**44.629**	46.399	**45.595**	47.743
9	21.860	**21.629**	22.822	**22.032**	19	38.700	**38.468**	**39.662**	39.815
10	23.509	**22.433**	24.471	**23.825**	20	44.695	**43.295**	45.657	**44.637**

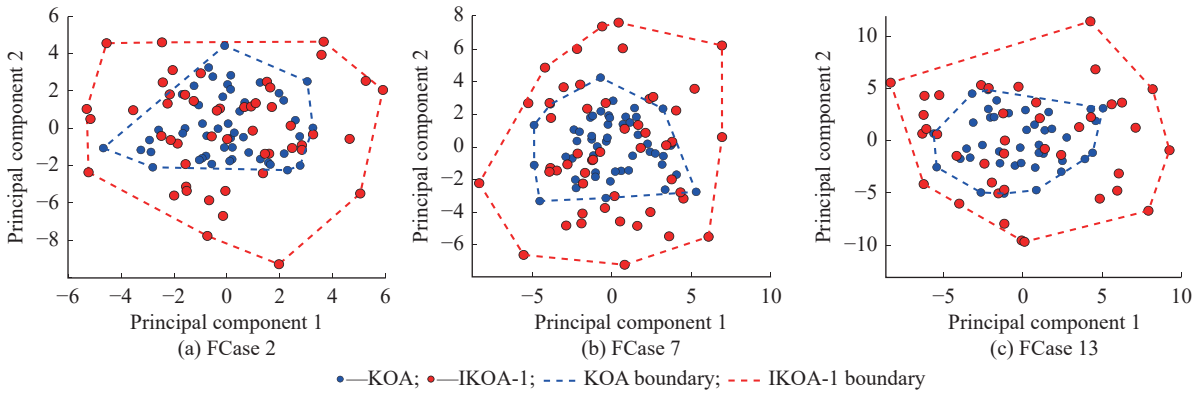


图 4　IKOA-1 和 KOA 算法在 FCase 2、FCase 7 和 FCase 13 的初始种群分布

Fig. 4　Initial population distributions of the IKOA-1 and KOA algorithms for FCase 2, FCase 7, and FCase 13

表 4　KOA 及其变体算法消融实验结果

Table 4　Results of ablation experiments on KOA and its variant algorithms

FCase	BF				STD				AVGITER			
	KOA	IKOA-1	KOA	IKOA-3	KOA	IKOA-1	IKOA-2	IKOA-3	KOA	IKOA-1	IKOA-2	IKOA-3
1	13.281	12.420	0.186	12.848	0.186	0.133	**0.099**	0.150	223.2	239.5	**174**	214
2	**8.996**	**8.996**	0.103	**8.996**	0.103	0.062	0.052	**0.051**	73.4	**72.1**	93.4	92.9
3	11.797	11.640	0.257	11.392	0.257	0.144	**0.080**	0.123	298.2	275.7	**257.3**	233.7
4	**10.060**	10.660	0.049	10.730	0.049	0.054	**0.036**	0.043	377.5	391.4	373.7	**369.5**
5	14.450	14.169	0.155	14.169	0.155	0.074	0.067	**0.061**	373.5	343.5	379.4	**321.2**
6	12.880	13.980	0.155	12.960	0.155	0.136	**0.043**	0.103	279.8	280.7	282.4	**258.5**
7	**19.125**	**19.125**	0.098	**19.125**	0.098	0.051	**0.040**	0.050	**31.2**	38.2	45.2	62.8
8	23.358	22.513	0.076	22.513	0.076	**0.044**	0.047	0.047	373.4	330.5	**225.1**	251.4
9	21.629	20.809	**0.056**	20.854	**0.056**	0.057	0.059	0.057	423.4	426.7	446.5	**411.4**
10	22.433	21.489	0.106	**19.169**	0.106	**0.046**	0.082	0.063	395.2	389.5	413.6	**386.2**
11	34.065	34.657	0.203	34.043	0.203	0.201	**0.080**	**0.080**	350.8	309.2	298.6	**223.7**
12	**25.760**	**25.760**	0.146	**25.760**	0.146	0.091	**0.075**	0.092	419.8	420.2	389.4	**330.4**
13	33.184	**31.524**	0.119	32.724	0.119	0.084	**0.069**	0.082	411.5	**380.6**	457.4	435.2
14	38.355	38.375	**0.081**	38.375	**0.081**	0.082	**0.081**	**0.081**	369.6	338.7	**327.6**	463.3
15	40.270	38.398	0.130	38.398	0.130	0.212	**0.120**	0.133	308.3	384.3	**313.2**	334
16	47.866	44.960	0.337	**44.256**	0.337	0.210	**0.118**	0.169	405	473	**325.7**	399.2
17	43.410	43.905	**0.113**	43.404	**0.113**	0.148	0.142	0.142	341	**237.3**	401	269.9
18	46.399	38.429	0.274	41.460	0.274	0.269	**0.147**	0.217	211.5	233	**190.5**	211.5
19	38.468	38.819	**0.167**	**38.462**	**0.167**	0.181	0.174	0.168	368.3	431.6	**361.6**	396.4
20	43.295	42.095	0.291	42.730	0.291	0.216	**0.134**	0.152	342.5	**283.4**	402.2	342.5

算式分别如式 (23) 和式 (24) 所示。

$$\mathrm{AECT}=\frac{1}{n}\sum_{k=1}^{n}\left|\frac{f_k-f}{f}\right| \tag{23}$$

$$\mathrm{ES}=\frac{\sum_{i=1}^{N_I}\sum_{r=1}^{n_i}\sum_{s,j=1,1}^{N_S,n_s}\left|z'_{irsj}-z_{irsj}\right|}{\sum_{i=1}^{N_I}\sum_{r=1}^{n_i}1} \tag{24}$$

其中，f_k 表示第 k 次的适应度值，f 表示原先调度方案的适应度值，n 表示运行次数。

表 5 示出了小规模数据集的求解结果，其中最优结果采用加粗的方式进行标注。KOABPS 算法在 15 个 Case 中取得全局最优，其 AECT 值在 7 种算法中最小，表明该算法平均解的延长率最低，以及取得结果可以落在全局最优值附近。对于 ES 值，KOABPS 算法在 11 个 Case 取得了最小值，由于重新调度很可

表 5 小规模算例算法性能比较

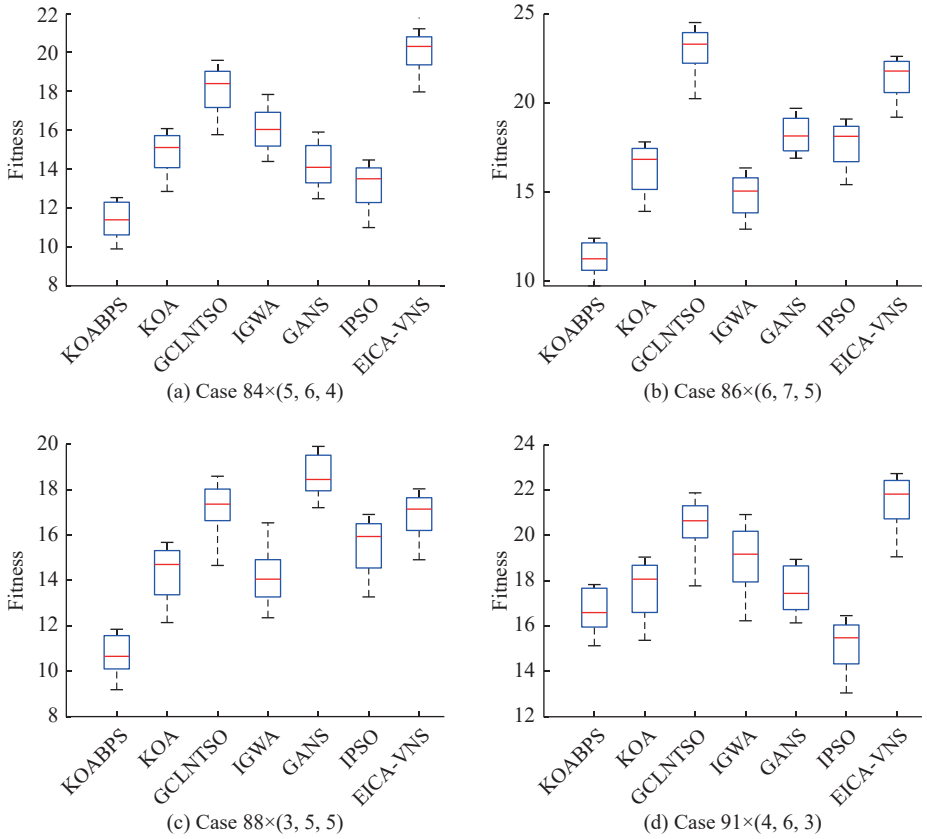
Table 5 Algorithm performance comparison on small-scale dataset

Case	BF							AECT							ES						
$J\times(a,b,c)$	KOA-BPS	KOA	GCLN-TSO	IGWA	GANS	IPSO	EICA-VNS	KOA-BPS	KOA	GCLN-TSO	IGWA	GANS	IPSO	EICA-VNS	KOA-BPS	KOA	GCLN-TSO	IGWA	GANS	IPSO	EICA-VNS
15×(2,3,2)	**5.849**	5.873	5.873	6.031	5.938	5.871	5.871	0.554	0.657	0.699	0.688	**0.503**	0.678	0.735	0.536	0.512	0.552	0.510	0.480	0.466	**0.461**
17×(3,4,2)	**3.272**	4.772	5.272	**3.272**	**3.272**	4.992	5.252	0.157	0.206	0.343	0.384	**0.096**	10.740	0.274	**0.473**	0.480	0.492	0.523	0.489	0.594	0.519
20×(4,5,2)	**3.826**	5.020	4.212	**3.826**	4.051	**3.826**	**3.826**	0.141	0.131	0.156	0.168	0.183	**0.129**	0.154	**0.137**	0.165	0.160	0.157	0.139	0.189	0.218
22×(5,6,2)	4.580	4.424	4.424	**4.310**	4.537	4.424	4.640	0.184	**0.172**	0.248	0.192	0.205	0.188	0.230	**0.161**	0.219	0.231	0.204	0.196	0.274	0.204
25×(2,4,2)	**4.721**	7.121	6.431	5.372	4.983	7.121	7.121	**0.091**	0.483	0.459	0.438	0.192	0.431	0.447	0.411	**0.373**	0.503	0.399	0.480	0.392	0.459
26×(3,5,2)	7.266	7.678	7.386	**7.041**	7.318	8.706	8.666	0.286	0.384	0.376	0.297	**0.274**	0.605	0.637	0.605	0.632	0.598	**0.520**	0.571	0.573	0.629
28×(4,6,2)	5.180	5.180	5.180	5.180	4.957	**4.340**	6.500	**0.128**	0.281	0.322	0.294	0.281	0.272	0.481	0.184	0.203	0.265	**0.154**	0.199	0.230	0.311
30×(5,3,2)	**4.486**	6.166	5.308	4.900	4.738	5.308	6.508	**0.098**	0.252	0.193	0.357	0.127	0.149	0.382	0.543	0.551	0.448	0.509	0.480	0.510	**0.415**
33×(2,5,2)	**5.760**	5.808	6.292	6.292	7.304	6.292	6.648	0.201	0.195	0.100	0.156	0.137	0.133	**0.073**	0.497	0.520	0.474	0.453	**0.425**	0.453	0.493
35×(3,6,2)	**4.640**	**4.640**	**4.640**	4.830	**4.640**	**4.640**	4.792	0.204	0.136	0.126	0.172	0.177	0.119	**0.113**	**0.408**	0.497	0.503	0.611	0.435	0.485	0.467
37×(4,3,2)	5.780	6.336	**5.778**	6.031	6.450	6.662	6.822	**0.077**	0.088	0.102	0.083	0.082	0.110	0.166	**0.506**	0.537	0.671	0.537	0.554	0.565	0.520
38×(5,4,2)	**8.414**	11.088	11.088	8.935	11.700	12.288	10.088	**0.202**	0.668	0.695	0.683	0.504	0.824	0.608	0.620	0.520	0.565	0.591	0.507	**0.499**	0.533
42×(2,6,2)	**7.580**	9.980	9.980	9.366	10.043	11.340	10.908	**0.066**	0.362	0.379	0.592	0.351	0.527	0.573	**0.472**	0.532	0.480	0.591	0.490	0.528	0.499
44×(3,3,2)	**7.168**	13.822	14.022	9.392	10.058	13.822	13.822	**0.078**	0.684	0.856	0.846	0.481	0.712	0.761	0.253	0.296	0.270	0.284	0.301	0.282	**0.246**
47×(4,4,2)	**6.220**	9.428	9.428	9.686	9.861	10.628	10.628	0.239	**0.118**	0.135	0.147	0.145	0.251	0.270	**0.201**	0.214	0.281	0.256	0.221	0.218	0.281
49×(5,5,2)	**9.471**	12.723	11.871	11.074	13.783	12.243	14.271	**0.060**	0.342	0.257	0.285	0.324	0.276	0.527	0.574	**0.451**	0.541	0.557	0.499	0.593	0.515
51×(2,3,2)	**7.160**	8.250	8.328	**7.160**	9.790	10.248	10.088	**0.071**	0.090	0.149	0.103	0.166	0.361	0.339	**0.501**	0.527	0.644	0.661	0.624	0.604	0.629
52×(3,4,2)	**8.908**	12.908	10.796	10.853	9.758	14.396	12.916	**0.060**	0.470	0.251	0.310	0.405	0.611	0.491	0.595	0.657	0.603	0.593	**0.554**	0.667	0.636
54×(4,5,2)	6.950	**6.396**	7.556	6.991	6.840	9.756	8.796	0.332	0.260	0.147	0.125	0.628	0.063	**0.043**	0.718	0.657	0.654	**0.574**	0.601	0.676	**0.624**
55×(5,6,2)	**8.399**	10.799	10.799	11.940	8.964	10.799	10.799	0.220	**0.059**	0.065	0.083	0.092	0.060	0.078	**0.306**	0.397	0.375	0.379	0.334	0.333	0.312

能会打断原来的组批方案，因此 KOABPS 算法的 ES 值可能较高，这同时也体现了该算法在设备故障时不会只在原有的调度方案上做改动，而是寻找新的更好的调度方案。图 5 示出了中规模数据集中 4 个 Case 的箱型图，可以看出，在多数 Case 中 KOABPS 算法的箱体长度相较于其他算法要短，而且箱体总体位置要低于其他算法，说明 KOABPS 取得的最优值要更好，稳定性也较好。图 6 示出了大规模数据集中 4 个 Case 的收敛曲线，可以看出 KOABPS 算法最终均近似收敛到最优值附近，表明 KOABPS 的寻优能力较强。

4 结 论

本文针对设备故障的铝卷生产中冷轧和退火阶段的协同调度问题进行研究，建立了数学模型。针对调度问题兼具可重入、批处理与设备故障的特点，提出了一种 KOABPS 算法：构建了基于分级的动态解码方式，采用拉丁超立方采样初始化种群，设计了基于种群状态自适应调节机制，引入 Levy 飞行跳出局部最优。在工厂数据集中对比了不同重调度方法；设计消融实验分析了算法改进模块的贡献度；在 3 种不同规模的随机数据集上将 KOABPS 算法与 KOA、GCLNTSO 等算法进行仿真比较算法性能，结果表明在解的质量上，KOABPS 算法的效果较好。值得指出的是，其他加工行业，例如半导体晶圆加工、钢铁热处理或纺织印染等，与本文所研究的问题具有类似的特征，所以本文所提出的 KOABPS 算法可以迁移到这些具有共性特征的调度问题中进行应用。当然，针对不同行业的特定约束，可能需对解码或局部搜索模块进行适配。未来工作将聚焦于 KOABPS 在多类可重入批处理调度问题中的泛化能力。

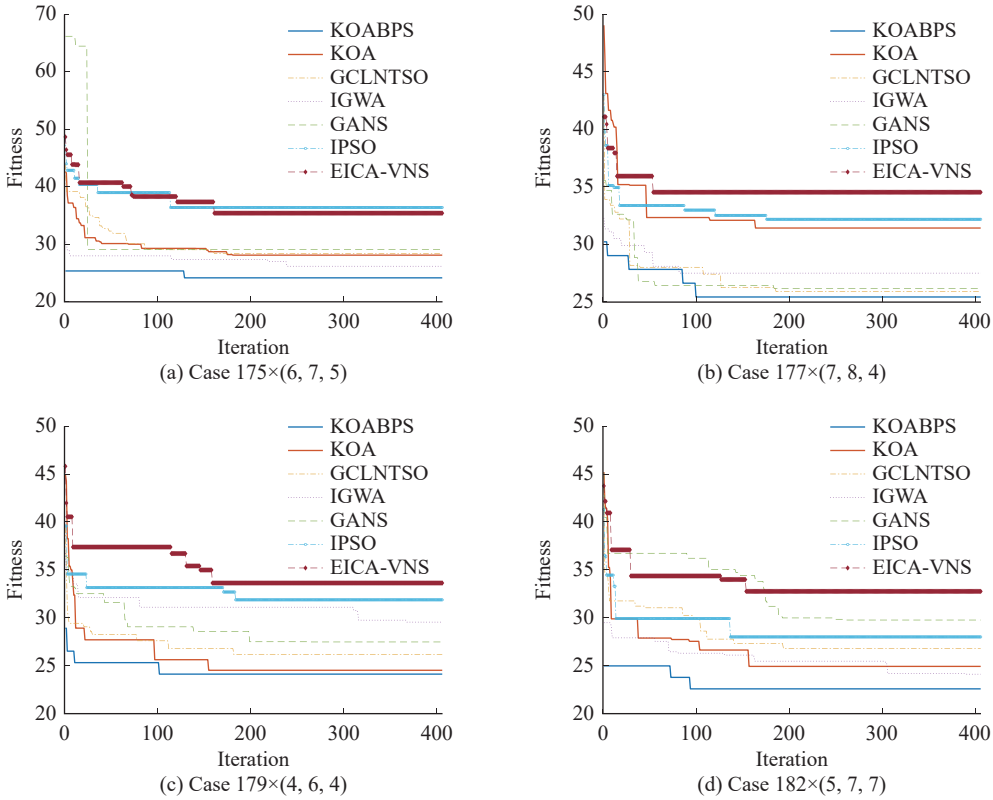


(a) Case 84×(5, 6, 4)　(b) Case 86×(6, 7, 5)

(c) Case 88×(3, 5, 5)　(d) Case 91×(4, 6, 3)

图 5　中规模数据集部分算例结果箱型图

Fig. 5　Box plot of partial instance results on medium-scale dataset



(a) Case 175×(6, 7, 5)　(b) Case 177×(7, 8, 4)

(c) Case 179×(4, 6, 4)　(d) Case 182×(5, 7, 7)

图 6　大规模数据集部分算例收敛曲线图

Fig. 6　Convergence curve plot of partial instance results on large-scale dataset

符号说明:

AT —同一退火批次的铝卷,在退火操作开始前各自最后一次冷轧操作完成时间的最大间隔

B —预调度批次集合, $k=1,2,\cdots,N_B$

B' —重调度批次集合, $k'=1,2,\cdots,N'_B$

C_{sj} —加工阶段 s 的第 j 台设备的容量

D_i —铝卷 i 的操作集合

D_{sj} —阶段 s 的第 j 台设备故障持续时间,服从正态分布

F_{sj} —二值变量,若阶段 s 的第 j 台设备发生故障,取值为1;反之为0

$H_{i_ar_ai_br_b}$ —二值变量,若预调度操作 $o_{i_ar_a}$ 先于 $o_{i_br_b}$ 加工,取值为1;反之为0

$H'_{i_ar_ai_br_b}$ —二值变量,若重调度操作 $o_{i_ar_a}$ 先于 $o_{i_br_b}$ 加工,取值为1;反之为0

$HH_{k_ak_b}$ —二值变量,若预调度批次 k_a 先于 k_b 加工,取值为1;反之为0

$HH'_{k'_ak'_b}$ —二值变量,若重调度批次 k'_a 先于 k'_b 加工,取值为1;反之为0

I —铝卷集合, $i=1,2,...,N_I$

M —所有设备集合,其中 m_{sj} 表示阶段 s 的第 j 台设备

$N_{O_{ir}}$ —铝卷 i 的第 r 个操作的工艺号

$O_{Msj(t_0\sim t)}$ —在时间范围 $t_0\sim t$ 内,加工阶段 s 的第 j 台设备的产能上限

O —所有铝卷的操作集合,其中 o_{ir} 表示第 i 个铝卷的第 r 个操作

$P_{Ma(t_0\sim t)}$ —在时间范围 $t_0\sim t$ 内, a 类材料铝卷的产量上限

P —铝卷的材料集合, $a=1,2,...,N_P$

S —加工阶段集合, $s=1,2,...,N_s$

T_{EH_i} —铝卷 i 的热轧操作完成时间

$T_{EO_{ir}}$ —预调度操作 o_{ir} 的完成加工时间

$T'_{EO_{ir}}$ —重调度操作 o_{ir} 的完成加工时间

T_{Ek} —预调度批次 k 的完成加工时间

$T'_{Ek'}$ —重调度批次 k' 的完成加工时间

$T_{SO_{ir}}$ —预调度操作 o_{ir} 的开始加工时间

$T'_{SO_{ir}}$ —重调度操作 o_{ir} 的开始加工时间

T_{Sk} —预调度批次 k 的开始加工时间

$T'_{Sk'}$ —重调度批次 k' 的开始加工时间

T_s —加工阶段 s 的操作时间

$T_{sj,\text{start}}$ —阶段 s 的第 j 台设备的故障开始时间,服从威布尔分布

$T_{sj,\text{end}}$ —阶段 s 的第 j 台设备故障结束时间

T_{WH} —铝卷在热轧操作后需要冷却的时间

T_W —铝卷前一个操作完成后冷却的时间

U_{ksj} —二值变量,若预调度批次 k 位于设备 m_{sj} 上加工,取值为1;反之为0

$U'_{k'sj}$ —二值变量,若重调度批次 k' 位于设备 m_{sj} 上加工,取值为1;反之为0

V_{ai} —二值变量,若铝卷 i 的类型是 a ,取值为1;反之为0

$W_{ID_{ir}}$ —若入口宽度在[1 000,1 250)之间,取值为1;若入口宽度在[1 250,1 500)之间,为0

W_{irk} —二值变量,若预调度操作 o_{ir} 分配给批次 k ,取值为1;反之为0

$W'_{irk'}$ —二值变量,若重调度操作 o_{ir} 分配给批次 (k') ,取值为1;反之为0

X_{ir} —二值变量,若 o_{ir} 是冷轧操作,取值为1;反之为0

Y_{ir} —二值变量,若 o_{ir} 是退火操作,取值为1;反之为0

Z_{irsj} —二值变量,若预调度操作 o_{ir} 位于设备 m_{sj} 上加工,取值为1;反之为0

Z'_{irsj} —二值变量,若重调度操作 o_{ir} 位于设备 m_{sj} 上加工,取值为1;反之为0

参考文献:

[1] WU J, LIU Y. A modified multi-agent proximal policy optimization algorithm for multi-objective dynamic partial-re-entrant hybrid flow shop scheduling problem[J]. Engineering Applications of Artificial Intelligence, 2025, 140: 109688.

[2] LIU Y, FAN J, ZHAO L, *et al*. Integration of deep reinforcement learning and multi-agent system for dynamic scheduling of re-entrant hybrid flow shop considering worker fatigue and skill levels[J]. Robotics and Computer-Integrated Manufacturing, 2023, 84: 102605.

[3] WU Y, LIN X, ZHU G. Solving many-objective reentrant hybrid flowshop scheduling problem considering uncertainty factors in thin-film transistor liquid crystal display[J]. Computers & Industrial Engineering, 2025, 204: 111117.

[4] HE P, ZHANG B, LU C, *et al*. Optimizing distributed reentrant heterogeneous hybrid flowshop batch scheduling problem: Iterative construction-local search-reconstruction algorithm[J]. Swarm and Evolutionary Computation, 2024, 90: 101681.

[5] LEI D, DAI T. A shuffled frog-leaping algorithm with cooperations for distributed assembly hybrid-flow shop scheduling with factory eligibility[J]. Symmetry, 2023, 15(4): 786.

[6] SONG L, LIU C, SHI H, *et al*. An improved immune genetic algorithm for solving the flexible job shop scheduling problem with batch processing[J]. Wireless Communications and Mobile Computing, 2022, 2022(1): 2856056.

[7] SUN M, CAI Z, ZHANG H. A teaching-learning-based optimization with feedback for LR fuzzy flexible assembly job shop scheduling problem with batch splitting[J]. Expert Systems with Applications, 2023, 224: 120043.

[8] ROCHOLL J, MÖNCH L. Metaheuristics for solving a flexible flow-shop scheduling problem with s-batching machines[J]. International Transactions in Operational Research, 2025, 32(1): 38-68.

[9] 吴杞昊. 热镀锌生产线退火炉过程监测与故障诊断系统研究与设计 [D]. 沈阳: 东北大学, 2013.
[10] HU G, GONG C, LI X, *et al*. CGKOA: An enhanced Kepler optimization algorithm for multi-domain optimization problems[J]. Computer Methods in Applied Mechanics and Engineering, 2024, 425: 116964.
[11] LI J, NOTO M, ZHANG Y. Improved Kepler optimization algorithm based on mixed strategy [C]//Advances in Swarm Intelligence. Singapore: Springer, 2024: 157-170.
[12] PENG G, MA L, HONG S, *et al*. Optimization design and internal flow characteristics analysis based on Latin hypercube sampling method[J]. Arabian Journal for Science and Engineering, 2025, 50(4): 2715-2754.
[13] 陈飞跃, 徐震浩, 顾幸生. 基于离散布谷鸟搜索算法的带阻塞有差速混合流水车间调度 [J]. 华东理工大学学报(自然科学版), 2017, 43(3): 425-435.
[14] 王玉超, 吴彬斌, 贾丽丽, 等. 基于 Box-Behnken 响应面法优化 SC1 清洗热氧化膜工艺 [J]. 华东理工大学学报 (自然科学版), 2025, 51(5): 589-595.
[15] ABDEL-BASSET M, MOHAMED R, AZEEM S A A, *et al*. Kepler optimization algorithm: A new metaheuristic algorithm inspired by Kepler's laws of planetary motion[J]. Knowledge-Based Systems, 2023, 268: 110454.
[16] FAN C, WANG W, TIAN J. Flexible job shop scheduling with stochastic machine breakdowns by an improved tuna swarm optimization algorithm[J]. Journal of Manufacturing Systems, 2024, 74: 180-197.
[17] YU Y, HU Y, GE Y. Optimization of reentrant hybrid flow shop scheduling problem based on improved grey wolf algorithm: Taking automatic chemiluminescence immunoassay analyzer as an example [C]// 2024 7th International Conference on Advanced Algorithms and Control Engineering. Shanghai: IEEE, 2024: 1286-1290.
[18] ZHANG H, LV S, XIN D, *et al*. A genetic algorithm enhanced with neighborhood structure for general flexible job shop scheduling with parallel batch processing machine[J]. Expert Systems with Applications, 2025, 265: 125888.
[19] SANA M U, LI Z, JAVAID F, *et al*. Improved particle swarm optimization based on blockchain mechanism for flexible job shop problem[J]. Cluster Computing, 2023, 26(5): 2519-2537.
[20] 冯晓宇. 基于混合帝国竞争算法的柔性车间调度研究 [D]. 哈尔滨: 哈尔滨工程大学, 2023.

Collaborative Two-Stage Reentrant Batch Scheduling Problem for Aluminum Coils Considering Equipment Failures

BAI Yunfeng, XU Zhenhao, GU Xingsheng
(Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, East China University of Science and Technology, Shanghai 200237, China)

Abstract: Equipment failures in aluminum coil production disrupt normal manufacturing operations, and how to adjust scheduling schemes dynamically in response to such failures has become a critical practical issue. To address this problem, this paper establishes a corresponding mathematical model and proposes a Kepler Optimization Algorithm with Adaptive Adjustment Based on Population State (KOABPS). Specifically, a hierarchy-based dynamic decoding method is constructed, Latin hypercube sampling is adopted for population initialization to ensure sample diversity, a population state-based adaptive adjustment mechanism is designed, and Levy flight is introduced to help the algorithm escape local optima effectively. To verify the effectiveness of KOABPS, a factory dataset is built using actual production data from enterprises, and random datasets of different scales are further constructed according to the characteristics of practical production data. Comparisons between partial rescheduling and complete rescheduling are conducted on the factory dataset, and ablation experiments are carried out to demonstrate the validity of the proposed key modules in KOABPS. In addition, small-, medium- and large-scale random experiments are designed to compare KOABPS with other typical optimization algorithms. The results show that the proposed algorithm outperforms its counterparts in both solution quality and convergence speed.

Key words: aluminum coil production; batch processing; reentrant scheduling; equipment failure; Kepler Optimization Algorithm (KOA)

（责任编辑：李娟）