

基于NSGAII的多目标随机加工时间柔性车间作业调度

李淑月*, 唐璐玮

南京邮电大学物联网学院, 江苏 南京

收稿日期: 2025年3月22日; 录用日期: 2025年4月18日; 发布日期: 2025年4月28日

摘要

新兴信息技术的飞速发展与全球经济一体化格局加速国际强国间新一轮的激烈角逐, 智能制造成为我国应对瞬息万变的市场需求, 向数字化、自动化、智能化、低碳化转型升级的重大战略决策。柔性车间作业调度是智能制造车间生产计划管理的重要问题之一, 因其NP-hard本质而备受关注, 此外实际运营环境存在的不确定因素使现有解决方案面临巨大的挑战。文章以最小化最大完工时间、最小化误工 - 提前成本、最小化机器总负载为优化目标, 结合作业加工时间随机变化的特性, 围绕多目标随机加工时间的柔性车间作业调度问题展开研究, 提出一种改进NSGAII的多目标随机加工时间柔性车间作业调度方法(N-MOSFJ)。该方法依据各目标偏好设计种群初始化方法, 提升初始解质量和多样性, 同时, 通过构造参考下界在具有随机加工时间动态变化的解空间中为算法迭代引导可参考的搜索方向。为深入挖掘局部最优解, 生成感知绩效因子, 并设置动态定位因子以加速全局靶向探索, 基于此建立个体综合适应度值评估机制确保算法客观量化个体质量。此外, 采用差异优先与质量优先混合选择模式, 协同质量约束的“优生多生”交叉变异策略, 有效平衡多目标间冲突。通过在仿真数据集和领域公开数据集上对算法进行比较验证, 实验结果表明提出的算法具有较好的有效性和稳定性。

关键词

柔性车间作业调度, 随机加工时间, 多目标优化, NSGAII, 智能制造

Multi-Objective Flexible Job Shop Scheduling with Stochastic Processing Times Based on NSGAII

Shuyue Li*, Luwei Tang

School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing Jiangsu

*通讯作者。

Abstract

The rapid development of emerging information technologies and the acceleration of global economic integration have intensified competition among international powers. Intelligent manufacturing has become a critical strategy for China to address dynamic market demands and achieve transformation toward digitization, automation, intelligence, and low-carbon development. Flexible job shop scheduling is a vital problem in production planning within intelligent manufacturing systems. Its NP-hard nature and the uncertainties in real-world operational environments present significant challenges to existing solutions. This study investigates a multi-objective flexible job shop scheduling problem with stochastic processing times. The objectives are to minimize the makespan, tardiness-early cost, and total machine load. A novel method based on an improved NSGAII, called N-MOSFJ, is proposed. The method designs a preference-based population initialization strategy to enhance the quality and diversity of initial solutions. Reference lower bounds are constructed to guide the algorithm's iterations in a dynamically changing solution space with stochastic processing times. A performance-aware factor is introduced to improve local exploitation, while a dynamic positioning factor accelerates global exploration. An integrated fitness evaluation mechanism ensures an objective assessment of individual solution quality. To balance conflicts among multiple objectives, a hybrid selection strategy combining diversity and quality priorities is adopted. This is integrated with a quality-constrained “elite reproduction and multiple reproduction” crossover and mutation strategy. By comparing and validating the algorithm on simulation datasets and publicly available domain datasets, experimental results show that the proposed algorithm demonstrates good effectiveness and stability.

Keywords

Flexible Job Shop Scheduling, Stochastic Processing Time, Multi-Objective Optimization, NSGAII, Intelligent Manufacturing

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

全球经济一体化加速国际竞争, 各国企业为争夺市场份额不断寻求技术创新提升自身竞争力, 通过在全球范围进行资源配置和生产布局来应对挑战。工业 4.0 [1]为全球经济一体化提供更加敏捷高效的生产基础, 智能制造[2]为工业 4.0 提供所需的实施技术和方法, 支持物理制造加工过程的实时监控与动态扰动因素驱动的应急决策。

柔性车间作业调度[3]是智能制造车间生产计划管理的重要问题之一, 然而这种灵活性导致调度问题可行解空间范围呈指数级增大, 造成求解方案的难度急剧上升, 使其成为比传统车间作业调度更复杂的 Non-deterministic Polynomial-time hard (NP-hard)问题, 备受学术界关注[4]。鉴于精确算法在处理这一问题时效率上的局限性, 众多群智能和进化算法应运而生[5], 但这类算法理论依据源于对生物群落社会性的模拟, 数学分析较为薄弱, 导致算法对具体问题和应用环境的依赖性比较大。由于柔性车间作业调度的复杂性, 传统的数学优化方法难以在合理的时间内找到高质量的可行解, 为此越来越多学者将关注点

转向群智能进化算法, 其中较为典型的包括遗传算法[6]、蚁群算法[7]、粒子群算法[8]、分布估计算法[9]以及一些新兴算法, 如候鸟算法[10]、果蝇算法[11]、帝国主义竞争算法[12]、免疫算法[13]。

实际智能制造环境中需求多样化且不确定因素普遍存在, 为柔性车间作业调度提出更大的挑战[14]。为此现有研究从时间、成本、质量、碳排放等多角度综合考虑智能制造系统需求, 结合变批量多品种[15]、分散布局并行加工[16]、异地传输[17]、不可再生资源[18]等行业特性, 构建多目标柔性车间作业调度模型并提出群智能进化调度优化算法。现有研究通常采用预测式、反应式、预测反应式三种方法应对加工过程中的不确定因素, 但预测式调度算法紧密依赖估计结果的准确性, 造成调度方案的可行性差。反应式调度算法难以控制重调度的范围, 会产生级联反应, 使得重调度范围不断扩大导致调度算法的效率降低。预测反应式调度算法通过对不同估计结果设置及时的反应措施, 在一定程度上可以提高算法的抗干扰性, 但如何有效地对不确定因素建模是影响算法性能的一个关键问题。相对而言, 非支配排序遗传算法[19]基于遗传算法框架嵌入多目标问题本质特征后解决各类多目标柔性车间作业调度问题表现突出[20]-[22], 分析各目标函数最优性条件并获取到目标函数偏好后, 在算法迭代过程中逐步权衡多目标间的冲突, 利用非支配排序与拥挤度有效判定局部收敛并增强个体多样性, 为实际智能制造生产环境调度优化提供支撑。

2. 问题模型

2.1. 问题描述

给定 n 个作业集合 $J = \{J_i | 1 \leq i \leq n\}$, m 台机器集合 $M = \{m_k | 1 \leq k \leq m\}$, 任意作业 $J_i \in J$ 包含具有指定先序约束关系的 r_i 个操作 o_{ij} ($1 \leq j \leq r_i$), 且有确定的截止期 d_i 。每一操作 o_{ij} 对应一候选加工机器集 M_{ij} , o_{ij} 在可选加工机器 k 上的加工时间为 t_{ijk} 。由于智能车间资源损耗、原材料质量、工艺变更、加工参数、物流运输等对作业操作加工时间的稳定性具有较大影响, 难以使用常量对操作加工时间准确描述。文章为精确刻画符合智能车间实际运营环境的操作加工时间, 将 t_{ijk} 表示为随机变量, 即 $t_{ijk} \sim f(\theta_{ijk}, \sigma_{ijk}^2)$, 其中 f 为特定概率分布, θ_{ijk} 和 σ_{ijk}^2 分别表示操作加工时间的均值与方差。

设 Θ 为智能车间多目标随机柔性作业调度问题的解空间, 其中一个确定的多目标随机柔性作业调度方案 $S (S \in \Theta)$ 需要解决两方面问题: 首先为每个操作 o_{ij} 分配加工机器, 然后对分配到同一机器上的操作进行排序。进一步地, S 若为一个可行多目标随机柔性作业调度方案, 则需要满足以下约束:

- (1) 每台机器同一时间只能加工一个操作;
- (2) 每个作业同一时间只能在一台机器上加工;
- (3) 同一作业不同操作需满足先序约束关系, 不同作业间操作无先序约束关系;
- (4) 操作一旦开始不可以被抢占。

智能车间多目标随机柔性作业调度目的在于高效地在解空间 Θ 中确定一个最优调度解集 S_{opt} , 使得 $\forall S \in S_{opt}$ 可行且在优化目标值上满足 Pareto 占优。

智能车间多目标随机加工时间柔性作业调度问题模型相关符号描述如表 1 所示:

Table 1. Symbols and descriptions

表 1. 符号与描述

符号	描述
n	作业数量
m	机器数量
J_i	第 i 个作业

续表

r_i	第 <i>i</i> 个作业包含的操作数量
o_{ij}	第 <i>i</i> 个作业的第 <i>j</i> 个操作, $0 < j \leq r_i$
M_{ij}	操作 o_{ij} 的候选加工机器集
t_{ijk}	操作 o_{ij} 在第 <i>k</i> 台机器上的加工时间
x_{ijk}	决策变量, 操作 o_{ij} 在第 <i>k</i> 台机器加工 $x_{ijk} = 1$, 否则 $x_{ijk} = 0$
d_i	第 <i>i</i> 个作业的截止期
D_i	第 <i>i</i> 个作业实际完成的延迟时间
E_i	第 <i>i</i> 个作业实际完成的提前时间
α_i	第 <i>i</i> 个作业的误工惩罚单位成本
β_i	第 <i>i</i> 个作业的提前库存单位成本
ct_{ij}	操作 o_{ij} 的完成时间
CT_i	第 <i>i</i> 个作业的完工时间

2.2. 数学模型

智能车间调度决策需要同时考虑用户、任务、资源多方面的需求, 对用户而言耗费成本越低越好, 对任务而言完工时间最早越好, 对资源而言机器负载越小越好。可见用户、任务、资源三方面的需求相互间存在较大冲突, 因此为灵活地适应动态变化的市场趋势并同时兼顾多方面的业务需求, 本文以任务完工时间、基于用户截止期的误工-提前时间、资源总负载为优化目标, 结合智能车间作业操作加工时间的随机性, 构建多目标随机柔性作业调度问题数学模型。

$$\min_{S \in \Theta} F(S) = \{f_1(S), f_2(S), f_3(S)\} \quad (1)$$

$$f_1(S) = E(\max_{1 \leq i \leq n} \{CT_i\}) \quad (2)$$

$$f_2(S) = E\left(\sum_{i=1}^n (\alpha D_i + \beta E_i)\right) \quad (3)$$

$$f_3(S) = E\left(\sum_{k=1}^m \sum_{i=1}^n \sum_{j=1}^{r_i} x_{ijk} * t_{ijk}\right) \quad (4)$$

s.t.

$$ct_{ij} + t_{i(j+1)k} \leq ct_{i(j+1)}, \quad 0 < i \leq n, \quad 0 < j < r_i, \quad x_{i(j+1)k} = 1 \quad (5)$$

$$(ct_{ij} + t_{ijk} \leq ct_{ij'}) \vee (ct_{ij'} + t_{ij'k} \leq ct_{ij}),$$

$$0 < i \leq i' \leq n, \quad 0 < j \leq r_i, \quad 0 < j' \leq r_{i'}, \quad x_{ijk} = 1 \wedge x_{ij'k} = 1 \quad (6)$$

$$\sum_{k=1}^m x_{ijk} = 1, \quad 0 < i \leq n, \quad 0 < j < r_i \quad (7)$$

上述模型中, 式(1)表示优化目标, 每个目标具体由式(2)~(4)描述。式(2)为作业总完工时间期望值。式(3)表示基于用户截止期的误工-提前时间期望值, 由误工惩罚与提前库存两部分组成, 其中 $D_i = \max\{0, CT_i - d_i\}$ 表示作业 *i* 相对截止期的误工时间, $E_i = \max\{0, d_i - CT_i\}$ 表示作业 *i* 相对截止期提前完成任务时间, α 和 β 分别表示两部分惩罚的权重。式(4)表示机器总负载期望值。为得到可行的调度方案, 需要满足式(5)~(7)的约束条件, 式(5)表示同一作业不同操作间的先序约束, 式(6)表示同一机器各操作间的串行约束, 式(7)表示同一操作仅能在一台机器上加工。

3. 基于 NSGAII 的多目标随机加工时间柔性作业车间调度算法

具有随机加工时间的多目标柔性作业车间调度难点在于问题解空间规模巨大, 作业加工时间的随机性致使最优解位置在解空间中动态变化, 传统多目标进化算法善于对解空间中固定位置的最优解进行探索, 但难以应对最优解位置的不断变更。针对这一挑战, 提出一种多目标随机加工时间柔性作业车间调度方法 N-MOSFJ。设计动态定位因子对解空间中难以预估的最优解区域变更进行识别与追踪, 构造感知绩效因子平衡多个目标间的冲突并在动态变化环境中对个体质量进行综合评估。基于动态定位因子与感知绩效因子间的交互联动, 通过多次迭代逐步校正与动态可变最优解间的同频共振。

3.1. N-MOSFJ 整体框架

为提升算法对个体操作的效率, 采用双层编码结构对个体进行描述, 上层编码表示操作在所分配机器上的执行顺序, 下层编码指定操作对应的加工机器。算法 1 所示为 N-MOSFJ 的整体框架。

算法 1 分为起始准备与迭代改进两个阶段。在起始准备阶段考虑问题多目标特性, 设计具有偏好的种群初始化贪心策略, 通过提高初始解的质量与多样性为算法搜索最优解提供基础。然后构造参考下界, 通过提供理论参考目标为算法引导有潜质的搜索方向, 进而提高算法搜索效率。在迭代改进阶段, 综合利用感知绩效因子和动态定位因子合理控制协调多目标间冲突的同时推进算法对最优解区域的探寻。每次迭代过程中, 基于混合模式的选择机制维护个体优良基因的同时增强种群的多样性, 设计具有质量约束的交叉变异机制高效裁剪邻域范围, 采用可伸缩的新种群生成策略确保同时维护个体与群体的共同利益。整体框架中涉及的关键元素将在以下小节中详细解释。

算法 1. N-MOSFJ 整体框架

输入: 机器数 m , 作业数 n , 各作业操作数 r_i

输出: Pareto 最优解集

Step 1. 基于多目标偏好的种群初始化(算法 2);

Step 2. 构造参考下界(算法 3);

Step 3. while 未满足算法终止条件 do

Step 4. for 种群中每一个个体

 基于参考下界的感知绩效因子生成;

 基于感知绩效因子的动态定位因子构建;

 基于感知绩效因子与动态定位因子的综合适应度值评估;

end for

Step 5. 基于混合模式的选择机制;

Step 6. 基于质量约束的交叉与变异;

Step 7. 基于可伸缩筛选的新种群生成(算法 4);

Step 8. end while

Step 9. 输出 Pareto 最优解集。

3.2. 基于多目标偏好的种群初始化

为能够给算法提供高质量的初始解, 提出基于多目标偏好的种群初始化贪心策略, 如算法 2 所示。首先结合实际运营环境的需求, 由领域专家为各优化目标设置合理的目标权重。然后基于各操作的概率分布, 采样生成各操作在各候选机器上的加工时间。对于种群中的每一个个体, 为各操作在机器候选集中随机分配操作的加工机器, 根据式(8)~(10)计算各操作在各目标上的偏好, 基于式(11)计算各操作的多目标综合偏好, 以各操作多目标综合偏好为贪心准则, 按综合偏好由大到小为优先级为各操作安排在加工机器上的操作顺序。

算法 2. 基于多目标偏好的种群初始化贪心策略

输入: 机器数 m , 作业数 n , 各作业操作数 r_i , 种群规模 $pNum$

输出: 种群 Pop

Step 1. 基于实际运营环境需求设置多目标权重向量 $w_q (q=1,2,3)$, 满足 $\sum_{q=1}^3 w_q = 1$;

Step 2. 基于概率分布采样生成各操作在各候选机器上的加工时间;

Step 3. for ($l=1$ to $pNum$)

 为第 l 个体 x_l 的各个操作 o_{ij} 随机分配加工机器;

 基于式(8-10)计算个体 x_l 的操作 o_{ij} 在各目标函数值上的偏好 $\delta_{ij}^q (q=1,2,3)$;

 基于式(11)计算个体 x_l 的操作 o_{ij} 在分配机器上的加工优先级 ρ_{ij} ;

 end for

Step 4. for ($l=1$ to $pNum$)

 for ($k=1$ to m)

 按加工优先级 ρ_{ij} 由大至小对机器 k 上分配的操作进行排序;

 end for

 基于优先级排序结果为个体 x_l 的各操作 o_{ij} 指定加工顺序;

 end for

Step 5. 输出种群 Pop .

式(8)~(10)分别表示操作 o_{ij} 在完工时间、提前-误工成本、机器负载三个目标函数上的偏好。式(8)中 CT 表示为所有操作随机分配加工机器后预估完工时间, 若操作 o_{ij} 在分配的加工机器上执行时间越短, 则其对优化完工时间的贡献度越大。式(9)以作业 i 的截止时间 d_i 为衡量标准, 若作业 i 的完成时间与截止时间 d_i 越接近, 其对优化提前 - 误工成本的贡献度越大。式(10)则以所有机器的负载总和为评估依据, 用以衡量操作 o_{ij} 在优化机器总负载时的贡献度。

$$\delta_{ij}^1 = 1 - \frac{t_{ijk}}{CT}$$

$$CT = \max_{1 \leq k \leq m} \left\{ \sum_{i=1}^n \sum_{j=1}^{r_i} x_{ijk} * t_{ijk} \right\} \quad (8)$$

$$\delta_{ij}^2 = 1 - \frac{|d_i - \sum_{j=1}^{r_i} t_{ijk}|}{d_i} \quad (9)$$

$$\delta_{ij}^3 = 1 - \frac{t_{ijk}}{\sum_{k=1}^m \sum_{i=1}^n \sum_{j=1}^{r_i} x_{ijk} * t_{ijk}} \quad (10)$$

$$\rho_{ij} = \sum_{q=1}^3 \omega_q * \delta_{ij}^q \quad (11)$$

在对操作 o_{ij} 在三个优化目标上的偏好进行合理评估后, 结合多目标在实际需求中的权重, 根据式(11)计算操作 o_{ij} 多目标综合偏好, 以多目标综合偏好作为优先级安排操作加工顺序, 优先级高的优先执行, 由此得到具有较好质量的初始解集。

3.3. 构造参考下界

如何在 MOSFJSSP 规模巨大的解空间中快速探索最优解的潜在区域是决定求解算法性能的关键因素。MOSFJSSP 中各操作加工时间随机变化导致最优解在解空间中会不断变化, 为避免盲目搜索, 有一个可获取且能够随同操作随机加工时间同步变更的参考下界显得至关重要。为此如算法 3 所示, 通过寻找和利用三个目标函数在理想状态下的最优解来构造能够合理引导种群进化方向的参考下界, 根据随机

变化的操作加工时间设置不断变更的搜索参考点, 由此提高算法的搜索效率。

算法 3. 构造参考下界

输入: 操作 o_{ij} 加工时间 t_{ijk} ($1 \leq i \leq n, 0 \leq j \leq n_i, 1 \leq k \leq m$)

输出: 参考下界 F_q^* ($1 \leq q \leq 3$)

Step 1. for ($i = 1$ to n)

 设置操作 o_{ii} 的开始时间 $s_{ii} = 0$;

 for ($j = 1$ to r_i)

 获取操作 o_{ij} 加工时间 $p_{ij} = \min_{1 \leq k \leq m} \{t_{ijk}\}$;

 end for

 计算作业 i 的最早结束时间 $e_i = \sum_{j=1}^{n_i} p_{ij}$;

end for

Step 2. 计算 $F_1^* = \max_{1 \leq i \leq n} \{e_i\}$;

Step 3. 计算 $F_2^* = \min_{1 \leq i \leq n} \{|d_i - e_i|, 0\}$;

Step 4. 计算 $F_3^* = \sum_{i=1}^n e_i$;

Step 5. 输出参考下界 F_3^* ($1 \leq q \leq 3$).

算法 3 构造的参考下界反映实际加工车间的理想状态。作业理想完工时间是将各操作分配至具有最短加工时间的机器上, 并按照各操作间的先序约束关系无空闲地依次执行, 因此所有作业最小完工时间参考下界设置为 $F_1^* = \max_{1 \leq i \leq n} \{e_i\}$ 。构造用户截止期的误工 - 提前最优成本参考下界 $F_2^* = \min_{1 \leq i \leq n} \{|d_i - e_i|, 0\}$ 表示所有作业都严格按照预定计划执行, 无延误也无提前, 不产生任何因作业完工时间与规划截止期偏差而产生的额外成本。 $F_3^* = \sum_{i=1}^n e_i$ 评估理想最优机器总负载的参考下界, 通过选择每个操作在所有机器上的最小加工时间并进行累加而获得。

3.4. 基于参考下界的感知绩效因子生成

在算法迭代过程中, 设计基于参考下界的感知绩效因子, 用以评估第 k 代种群及种群中个体在各优化目标上的性能, 进而从宏观与微观多角度反映算法演进成效。

$$\text{avg}_q^k = \frac{\sum_{i=1}^{p\text{Num}} f_q^k(x_i)}{p\text{Num}} \quad (12)$$

$$\varphi_{lq}^k(x_i) = \frac{f_q^k(x_i) - F_q^*}{\text{avg}_q^k - F_q^*} \quad (13)$$

$$\varphi_{lq}^k(Pop) = \frac{\text{avg}_q^k(x_i) - F_q^*}{\text{avg}_q^0 - F_q^*} \quad (14)$$

式(12)计算第 k 代种群所有个体在第 q 个优化目标上的性能均值。式(13)表示第 k 代个体 x_i 在第 q 个优化目标上的绩效因子, 分子用于评估个体 x_i 在第 q 个优化目标上与参考下界间的差距, 分母用于评估种群性能均值与参考下界间的差距, 通过两者间的比率从微观角度衡量经过 k 次迭代个体 x_i 在第 q 个优化目标上的改进程度。式(14)表示第 k 代种群整体在第 q 个优化目标上的绩效因子, 分子计算当前种群性能均值与参考下界间的差距, 分母表示初始种群性能均值 avg_q^0 与参考下界间的差距, 通过两者间的比率从宏观角度衡量经过 k 次迭代种群整体的改进程度。

3.5. 基于感知绩效因子的动态定位因子构建

为避免陷入局部最优, 在评估种群与个体感知绩效因子基础上, 仍需进一步确定种群与个体在解空间中的位置, 从而在每一次迭代过程中为搜索全局最优解探索可能的搜索方向。当算法在多次迭代后仍裹足不前, 为辅助算法及时跳出局部最优, 构建基于绩效因子的动态定位因子, 控制算法竭尽全力向参考下界靠近。

$$\varphi_{2q}^0(Pop) = \varphi_{2q}^0(x_i) = 1/L \quad (15)$$

$$\varphi_{2q}^{k+1}(x_i) = \frac{\varphi_{1q}^k(x_i)}{\sum_{q=1}^L \varphi_{1q}^k(x_i)} \quad (16)$$

$$\varphi_{2q}^{k+1}(Pop) = \frac{\varphi_{1q}^k(Pop)}{\sum_{q=1}^L \varphi_{1q}^k(Pop)} \quad (17)$$

算法初始阶段还未累积任何搜索经验, 难以辨识良好搜索方向, 在多目标的立体搜索空间中, 种群与个体在多个目标方向上的定位选择如式(15)所示平均分配。随着算法的迭代, 可以通过种群与个体的感知绩效因子逐步累积搜索经验, 可以根据感知绩效因子挖掘向参考下界快速演进的搜索路径。式(16)、(17)分别为个体动态定位因子与种群动态定位因子的构建方法, 可以看出第 $k+1$ 代动态定位因子受第 k 代种群感知绩效因子影响, 利用逐步迭代累积的历史信息, 判定多维解空间中某一区域是否值得探索。在每一次迭代过程中动态校正定位因子, 在宏观上通过整个种群的搜索偏向与微观上每个个体的搜索偏向加速算法快速的向最优解靠近。

3.6. 个体综合适应度值评估

评估个体感知绩效因子并获取其相应的动态定位因子后, 为改善子代基因做准备, 需进一步对个体进行综合适应度值评估。如式(18)所示, 个体综合适应度值同时考虑感知绩效因子与动态定位因子对个体质量的影响, 同理, 式(19)表示对种群进行综合适应度值评估。

$$CEI(x_i) = \sum_{q=1}^{q=L} (\varphi_{2q}^{k+1}(x_i) * \varphi_{1q}^{k+1}(x_i)) \quad (18)$$

$$CEI(Pop) = \sum_{q=1}^{q=L} (\varphi_{2q}^{k+1}(Pop) * \varphi_{1q}^{k+1}(Pop)) \quad (19)$$

感知绩效因子可以反映个体在各优化目标上具有的特性, 但对于多目标优化问题, 个体在单一目标上的优良程度并不等同于个体整体的优良程度, 在很多情况下个体在某一优化目标上达到最优值的同时可能在其他目标上处于最差值。为能够协调多个目标间的冲突, 避免算法最终寻找到的最优解集遍布于前沿面的边缘, 式(18)汇聚个体在多个优化目标上的感知绩效因子对个体性能进行综合评估, 提升算法找到的最优解集均匀分散在前沿面上的可能性。

个体性能优劣的评判除与其在各个优化目标上的特性有关, 还与实际运营场景对多个目标的应用需求相关。若实际应用环境中在最小完工时间、最小误工-提前成本、最小机器总负载三个优化目标中最为关注最小完工时间, 而个体却在最小误工-提前成本目标上具有最优特性, 此时个体相比于在最小完工时间目标上具有最优特性的其他个体而言, 综合评估值应相对降低。为能够表达这一物理含义, 式(18)基于动态定位因子对个体在不同优化目标上的感知绩效因子进行调节, 使得个体特性与实际需求能够交融叠加, 增强个体综合适应度值评估的合理性。

算法演进过程中, 除考虑优良个体对搜索方向的引导, 也要考虑个体与群体间的协同。若群体中所

有个体综合适应度水平与群体平均综合适应度水平差异较小, 此时算法很可能陷入局部最优且多样性较差。为能够敏锐查觉这一现象, 式(19)从群体角度给出迭代过程中群体综合适应度值的评价方法, 基于群体平均标准可以判定个体在群体环境中的相对评级, 有助于具有不同综合适应度值的个体间进行协同并合理提升群体性能。

3.7. 新种群生成

3.7.1. 基于混合模式的选择机制

在迭代改进的过程中, 为能够维护种群个体的质量并增强个体多样性, 设计混合模式选择机制, 充分利用优良个体基因进行局部挖掘, 合理配置潜力个体基因进行全局探索。混合模式选择机制包括质量优先、差异优先两种方式。质量优先按个体综合适应度值对个体进行排序, 每次选择综合适应度值最优的两个个体进行繁殖。差异优先根据式(20)计算种群中任意两个个体 x_i 和 x_j 的多维空间距离, 基于个体多维空间距离对个体对进行排序并优先选择多维空间距离大的个体对。

$$dis_{ij} = \sqrt{\sum_{q=1}^L (f_q(x_i) - f_q(x_j))^2} \quad (20)$$

质量优先选择模式旨在维护种群个体的优良基因, 个体综合适应度值优表示其在多个优化目标上具有较好的特性, 且在解空间的位置靠近最优解的可能性较大, 将适应度值优良的个体优先选择出来繁殖后代, 相当于通过优良个体构建邻域进行局部搜索, 可以进一步提升解的质量。差异优先选择模式关注种群个体多样性, 通过优先选择多维空间距离较大的个体对探索新的有前景区域, 基于具有不同优化目标特性的个体间协同, 有效避免陷入局部最优的同时引导算法向更有前景的区域探索。

3.7.2. 基于质量约束的交叉与变异

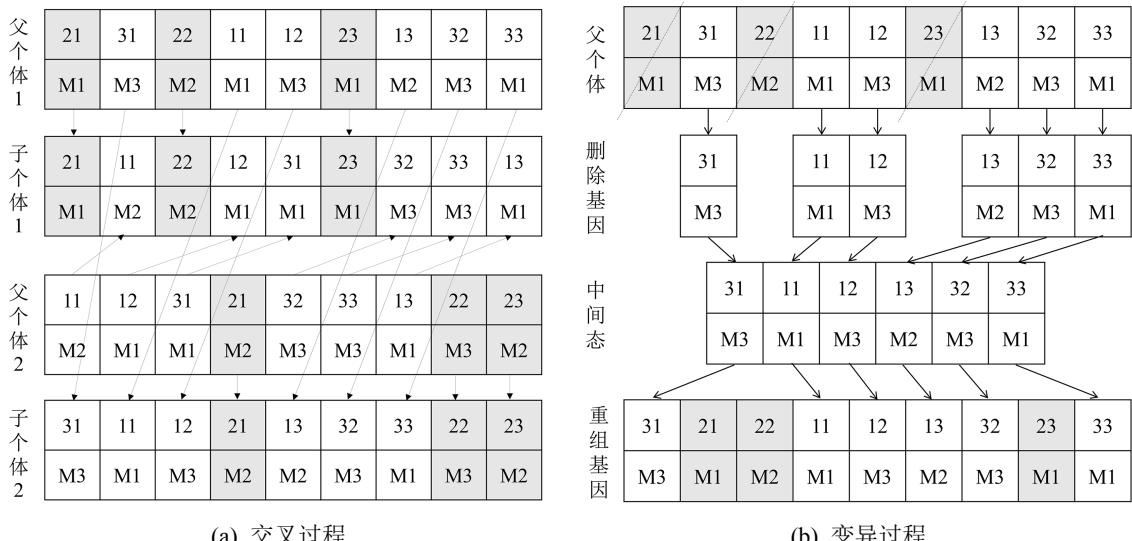


Figure 1. Crossover and mutation operators with quality constraints
图 1. 质量约束的交叉变异算子

基于混合模式选择出父个体后, 按照图 1(a)所示过程对父个体进行交叉。随机选定一个作业, 将在父个体 1 中选定作业的各操作及对应分配机器基因位置不变地拷贝至子个体 1 中, 将在父个体 2 中选定作业的各操作及对应分配机器同样位置不变地拷贝至子个体 2 中。此后将父个体 2 中其余作业各操作及

对应分配机器相对位置不变地插入子个体 1 中, 将父个体 1 中其余作业各操作及对应分配机器相对位置不变地插入子个体 2 中。变异操作通过删除基因与重组基因完成个体的突变。如图 1(b)所示, 对于待变异父个体, 随机选定某个作业, 在父个体中将选定作业所有操作对应的基因删除, 形成变异中间态, 再将所删除的基因按相对位置不变的条件下随机重新插入到中间态中, 形成变异后的子代。

种群经过以上交叉变异操作后, 产生的子个体能够充分融合不同父个体的特性, 在完成局部搜索的同时向新的区域进行探索。在此基础上, 若要进一步提升算法搜索效率, 需保证交叉变异后子个体质量的提高且增强探索新区域的必要性。为此利用群体与个体综合适应度值设置“优生多生”控制策略。父代个体 P_i 和 P_j 交叉后生成子代个体 C_s 和 C_t , 若 $CEI(C_s)$ 和 $CEI(C_t)$ 均优于 $CEI(Pop)$, 则 P_i 和 P_j 继续繁殖子代至不满足条件为止。变异操作则与交叉操作相反, 为跳出局部最优对新区域进行探索, 尝试对当前较差个体进行变异, 增强探索区域的必要性, 即在当前种群中, 对所有 $CEI(P_i)$ 劣于 $CEI(Pop)$ 的个体进行随机选择执行变异操作。

3.7.3. 基于筛选的可伸缩新种群生成

新种群生成过程如算法 4 所示。首先计算父代个体综合适应度值以及任意两个体间的多维空间距离为交叉做准备。每次交叉操作前随机生成选择模式, 基于选定的选择模式在综合适应度值列表或多维空间距离列表中获取交叉的父个体。交叉后判定生成子个体质量, 如果所生成的子个体质量全部优于种群平均质量, 则相应父个体具有一次再生的机会。交叉结束后选择子个体质量劣于种群平均质量的所有个体进行变异。最后父代种群 μ 个个体与生成子种群 λ_e 个个体合并, 对所有个体进行非支配排序并计算个体拥挤度, 择优选择 μ 个个体作为新生成种群输出。

算法 4. 基于筛选的可伸缩新种群生成

输入: 第 k 代父种群 Pop_k

输出: 第 $k+1$ 代新种群 Pop_{k+1}

Step 1. 计算第 k 代父种群中每个个体的综合适应度值并从优至劣对个体排序;

Step 2. 计算第 k 代父种群中个体间多维空间距离并从大到小对个体对进行排序;

Step 3. while (交叉次数 $< \lceil \mu/2 \rceil$) do

Step 4. 随机确定选择模式;

if(质量优先)

 提取综合适应度值最优的两个个体进行交叉;

else(差异优先)

 提取多维空间距离最大的个体对进行交叉;

Step 5. 获取生成子个体 y_i 和 y_j ;

Step 6. if ($CEI(y_i) > CEI(Pop) \ \& \ CEI(y_j) > CEI(Pop)$)

y_i 和 y_j 的父个体基于优生多生策略再次交叉生成子个体 y_p 和 y_q ;

Step 7. end while

Step 8. 获取交叉后生成的子种群 λ_e ;

Step 9. for ($i = 1$ to λ_e)

 if $CEI(y_i) < CEI(Pop)$

y_i 应用变异操作;

 end for

Step 10. 合并父代种群与子种群 $\mu + \lambda_e$;

Step 11. 对合并种群 $\mu + \lambda_e$ 进行非支配排序, 并计算个体拥挤度;

Step 12. 基于个体综合适应度值择优选取 μ 个体;

Step 13. 输出第 $k+1$ 代新种群 Pop_{k+1} .

为能够深度挖掘父代个体特性并在有限时间内探索较大范围的解空间, 探索过程若察觉某一解空间区域存在高质量解概率较大, 如 Step 6 所示采用优生多生策略围绕此区域进行更为细致的局部搜索。这一操作会使得实际生成子个体数量超过初始规划生成子个体数量, 即 $\lambda_e \geq \mu$, 使种群在演进过程因寻优产生伸展。但若每次迭代新种群规模都伸展会导致算法性能随迭代次数增加明显降低, 为避免这一缺陷, 算法 Step 10~12 将父代种群与子代种群合并后利用非支配排序与拥挤度择优选择高质量个体, 将伸展种群收缩至原种群规模后输出, 这一基于筛选的伸缩操作在提升新种群的群体质量的同时确保算法性能的稳定。

4. 实验结果分析

为合理且全面地评估提出算法的性能, 文章设置多样性的应用场景对算法进行测试, 将得到的结果与 GA、NSGAII 进行比较, 分析不同算法的运行时间、收敛情况及解的质量。所有算法采用 Python v3.11 进行编码, 实验在 AMD Ryzen 7 7840HS with Radeon 780M Graphics 3.8GHz 处理器、32G 内存、64 位 windows11 操作系统环境中运行。

4.1. 实验设置

为精确刻画实际智能车间运营环境, 设置如表 2 所示三种规模问题实例, 各场景作业数量在 $[3, 30, 300]$ 范围内变化, 机器数量在 $[4, 8, 16]$ 范围内变化。

Table 2. Instance scale (number of jobs \times number of operations \times number of machines)
表 2. 实例规模(作业数 \times 操作数 \times 机器数)

小规模	中规模	大规模
$3 \times 3 \times 4$	$30 \times 3 \times 8$	$300 \times 3 \times 16$

对于每种规模场景, 基于表 3 所示的概率分布随机生成三组实例, 分别具有截断正态分布、均匀分布和指数分布的操作加工时间。 t_{ijk} 表示第 i 个作业第 j 个操作 O_{ij} 在机器 M_k 上的加工时间, $t_{ijk} \sim f(\theta_{ijk}, \sigma_{ijk}^2)$ 是符合概率分布 f 的随机变量。当 f 为正态分布时, θ_{ijk} 和 σ_{ijk}^2 分别表示操作 O_{ij} 的加工时间的均值与方差; 当 f 为均匀分布时, 操作加工时间均匀分布于区间 $[\theta_{ijk} - 3 \times \sigma_{ijk}^2, \theta_{ijk} + 3 \times \sigma_{ijk}^2]$; 当 f 符合指数分布时, 指数分布的均值取 θ_{ijk} 。获取操作 O_{ij} 的加工时间后, 为寻求符合实际市场需求的作业截止期, 基于历史数据统计分析作业截止期与作业各操作加工时间的关系, 设置作业 i 的截止期 d_i 为 $\sum_{j=0}^n \left(\frac{1}{m} \sum_{k=1}^m \theta_{ijk} \right)$ 。

Table 3. Stochastic processing time

表 3. 随机加工时间 t_{ijk}

正态分布	均匀分布	指数分布
$t_{ijk} \sim N(\theta_{ijk}, \sigma_{ijk}^2)$	$t_{ijk} \sim U[\theta_{ijk} - 3 \times \sigma_{ijk}^2, \theta_{ijk} + 3 \times \sigma_{ijk}^2]$	$t_{ijk} \sim Exp(\theta_{ijk})$

4.2. 参数调优

实际车间中, 不同作业复杂程度各异且同一作业的不同操作加工难度参差不齐, 异构机器加工同一操作的性能也相差悬殊, 因此基于对实际车间作业操作加工时间的历史数据统计分析, θ_{ijk} 和 σ_{ijk}^2 的取值区间分别设置为 $(0, 100]$ 和 $[50, 200]$ 。综合评估市场竞争环境下产品需求的紧迫性和企业资金链管理的

稳定性, 在领域专家指导下设置作业误工惩罚成本权重 α 及作业提前库存成本权重 β 分别为 0.4 和 0.6。

为能够全面细致地展现算法性能, 通过在三种规模数据实例上对不同种群规模与迭代次数的参数组合进行多次参数测试后, 选取具有代表性且实验结果稳定的参数组合进行分析说明。经多次参数测试最终将种群规模固定为三个水平 10, 50, 100, 迭代次数固定为三个水平 10, 100, 500。此外为严格确保新种群个体的质量, 在可伸缩种群过程中确保 λ_e 对算法效率的影响可控, 将“优生多生”策略设置为只有在父个体初次交叉产生的两个子个体质量全部优于种群平均质量时会获得一次多生机会。

4.3. 仿真场景结果分析

基于以上实验数据及参数设置模拟的智能车间柔性作业调度场景, 采用提出的 N-MOSFJ 算法进行求解, 并与传统的 GA、NSGAII 进行对比, 从算法所获得的非支配最优解数量、最优解目标函数值、最优解特征分布三个方面验证提出算法的有效性和可行性。

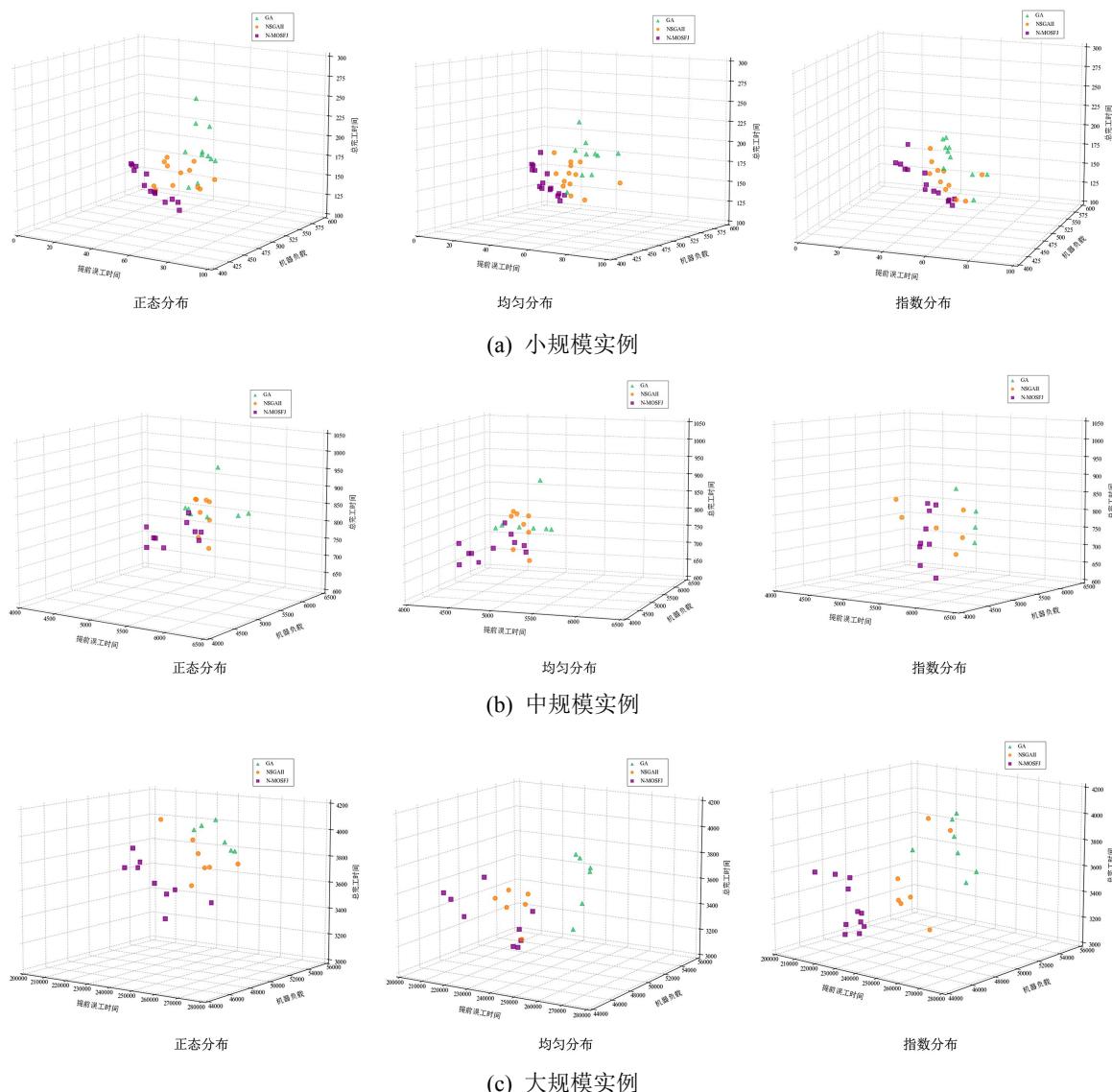


Figure 2. Pareto non-dominated front
图 2. Pareto 非支配前沿面

图 2 所示为种群规模 $N = 50$, 迭代次数为 500 时, 算法得到的 Pareto 非支配前沿面。对于小规模实例, 三种算法在三种分布实例上获得的最优解相对集中, 整体上 N-MOSFJ 表现出更优的目标函数值, 处于理论最优解区域的比例大。特别的是, N-MOSFJ 具有更好的多目标协调能力, 获得的解在目标空间中呈现出有序的分布形态, 实现多个目标间的均衡优化。对于中规模实例, N-MOSFJ 相较于 NSGAII 和 GA 在三种分布的目标空间中能够实现更全面的探索, 但同时三种算法找到的最优解个数均有所减少, 其原因在于解空间随问题规模呈指数级变化, 使得非支配最优解在解空间中的密度变小, 导致三种算法在全局搜索过程中具有盲目性, 但明显地 N-MOSFJ 利用构造参考点作为下界增强了全局搜索能力, 克服搜索过程中遇到的局部最优困境。对于大规模实例, 三种算法在三种分布实例上产生的解遍布范围明显扩大, 并且不同算法搜索到的最优解呈现出明显的分层结构, 其中 N-MOSFJ 表现出显著的汇聚性和非支配性, 而 NSGAII 和 GA 的解分散稀疏, 由此进一步表明 N-MOSFJ 在大规模问题中依然能够有效地平衡解的质量与多样性。类似地当实验过程中增大种群规模时, N-MOSFJ 在 500 次迭代时趋于稳定, 但与小规模、中规模种群不同的是, 其找到的最优解个数显著多于 NSGAII 和 GA, N-MOSFJ 获得的解数量在总数中的平均占比超过 70%。

为进一步深入比较三种算法结果的分布特性, 针对具有三种加工时间分布的大规模实例, 选取种群规模 $N = 50$, 迭代次数 500, 基于图 3 所示的箱线图展示三种算法结果的集中趋势和离散程度。由图可见对于正态分布实例, N-MOSFJ 在三个目标函数上的最大值、最小值、中位数均优于 NSGAII 和 GA, 且从四分位数可以看出, N-MOSFJ 在误工提前成本和机器负载两个目标上均呈左偏分布, 在完工时间上呈对称分布, 说明 N-MOSFJ 搜索到的非支配最优解集中大多数解在三个目标函数上都得到较好的优化效果。对于均匀分布实例, N-MOSFJ 在误工提前成本和完工时间两个目标函数值上的五个关键特征值均优于 NSGAII 和 GA, 较为特殊的是在机器负载目标函数上, NSGAII 的中位数优于 N-MOSFJ, 主要原因在于 NSGAII 产生的解的个数偏少而使得其在特定目标函数上有相对好的解, 但从中位数在箱体中的位置可以看出 N-MOSFJ 在机器负载上呈左偏分布, 说明在机器负载目标函数上 N-MOSFJ 的大部分最优解与 NSGAII 的解持平且解的分布范围更广、最优值更小。对于指数分布实例, N-MOSFJ 的最小值、中位数依然优于 NSGAII 和 GA, 即便在误工提前成本目标函数上的最优值与 GA 相近, 在机器负载目标函数上的最优值与 NSGAII 相近, 但从两个目标函数的中位数在箱体中的位置表明从所产生的非支配最优解集整体角度来说, N-MOSFJ 的质量最优。通过以上分析可以看出, N-MOSFJ 独立于正态分布实例, 但对均匀分布和指数分布实例具有一定的依附性, 但可通过适当增大种群规模和迭代次数协调优化过程中三个目标函数间的冲突。

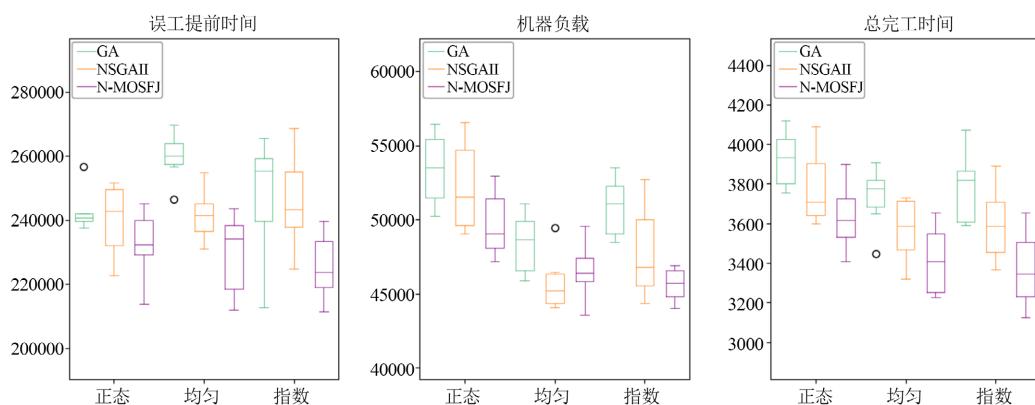


Figure 3. Box plot of algorithm results

图 3. 算法结果箱线图

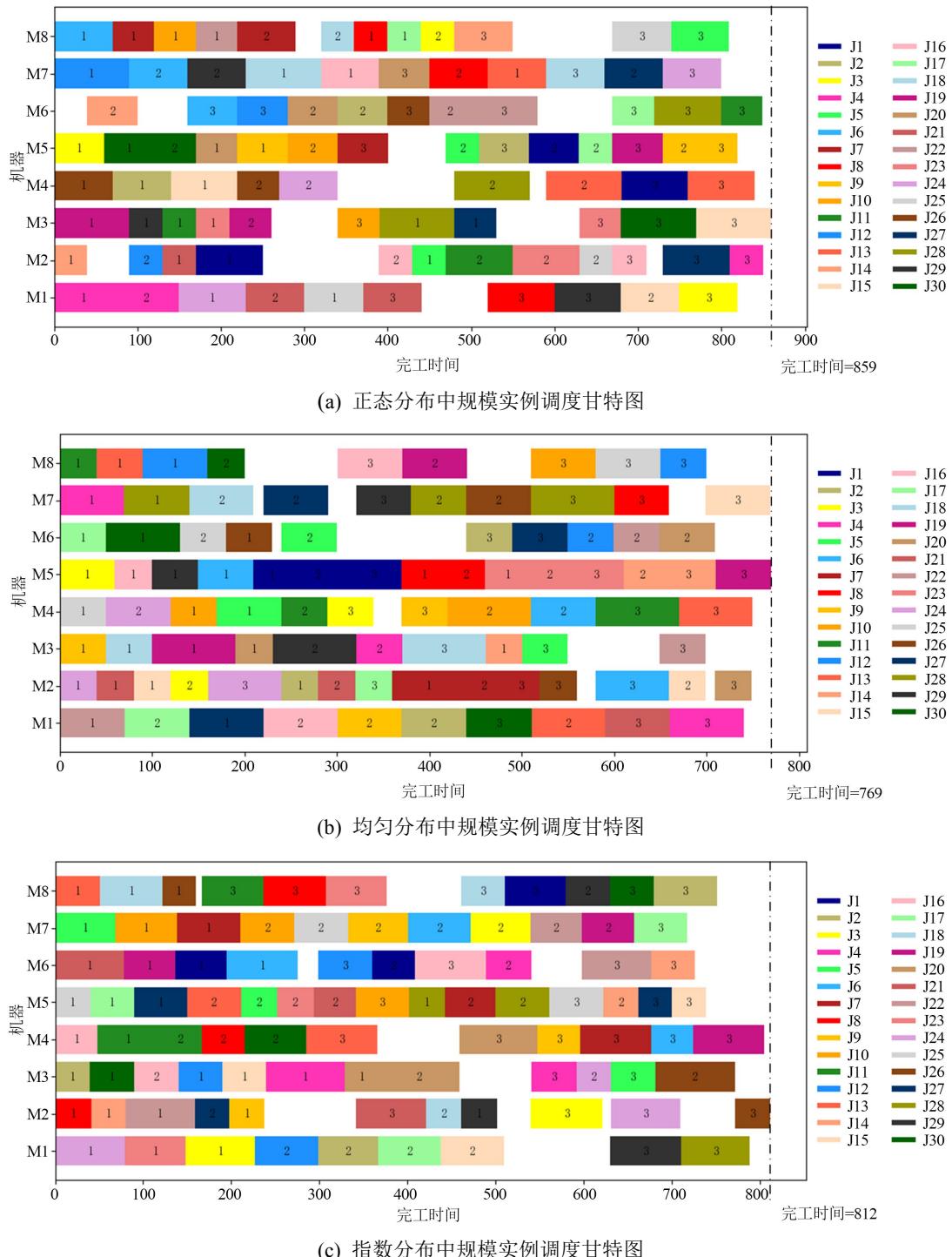


Figure 4. Gantt chart of stochastic flexible job shop scheduling based on N-MOSFJ
图 4. 基于 N-MOSFJ 的随机柔性作业车间调度甘特图

图 4 列举了 N-MOSFJ 在三种分布中规模实例的非支配最优解集中选取的最优解对应的甘特图, 可以更直观地展示 N-MOSFJ 得到的调度方案。图中可见 N-MOSFJ 在调度优化过程中能够合理安排作业操作执行顺序, 尽管少数机器相邻操作间存在可用空闲时间, 其主要意图是在均衡利用机器资源、减少总

完工时间的同时尽可能兼顾优化误工提前成本, 由此反映 N-MOSFJ 在多目标协同优化过程中较强调调节能力, 最终寻找到不同目标间的平衡点。

5. 总结

本文围绕智能制造柔性车间实际运营环境中存在的多目标需求与不确定性, 对柔性车间作业调度展开研究。考虑参与智能制造的三方需求, 建立多目标随机柔性车间作业调度数学模型, 通过将随机变量引入模型应对多类不确定因素对调度方案造成的干扰。基于 NSGAII 提出多目标随机柔性车间作业调度优化方法, 设计具有偏好的种群初始化方法提高初始解的质量, 构建参考下界指导算法高效搜索, 引入感知绩效因子和动态定位算子实现广泛的全局探索和深入的局部挖掘, 设计个体综合适应度值评价机制平衡多目标间的冲突, 设置可伸缩新种群生成机制增强算法搜索的敏感度。为验证提出算法的有效性与高效性, 全面客观地对算法进行的实验比对。首先仿真具有不同规模(小、中、大)、不同概率分布类型(正态、均匀、指数)的 9 种智能制造场景, 将 N-MOSFJ 与经典的 GA、NSGAII 进行比较, 实验结果表明相比于 GA 和 NSGAII, N-MOSFJ 在 9 种应用场景中都具有更好的稳定性。

基金项目

江苏省研究生科研与实践创新计划项目(No. KYCX23_1055)。

参考文献

- [1] Zhang, C., Chen, Y., Chen, H. and Chong, D. (2021) Industry 4.0 and Its Implementation: A Review. *Information Systems Frontiers*, **26**, 1773-1783. <https://doi.org/10.1007/s10796-021-10153-5>
- [2] Shojaeinabas, A., Charter, T., Jalayer, M., Khadivi, M., Ogunfowora, O., Raiyani, N., et al. (2022) Intelligent Manufacturing Execution Systems: A Systematic Review. *Journal of Manufacturing Systems*, **62**, 503-522. <https://doi.org/10.1016/j.jmsy.2022.01.004>
- [3] Dauzère-Pérès, S., Ding, J., Shen, L. and Tamssouet, K. (2024) The Flexible Job Shop Scheduling Problem: A Review. *European Journal of Operational Research*, **314**, 409-432. <https://doi.org/10.1016/j.ejor.2023.05.017>
- [4] Jiang, B., Ma, Y., Chen, L., Huang, B., Huang, Y. and Guan, L. (2023) A Review on Intelligent Scheduling and Optimization for Flexible Job Shop. *International Journal of Control, Automation and Systems*, **21**, 3127-3150. <https://doi.org/10.1007/s12555-023-0578-1>
- [5] Gao, K., Cao, Z., Zhang, L., Chen, Z., Han, Y. and Pan, Q. (2019) A Review on Swarm Intelligence and Evolutionary Algorithms for Solving Flexible Job Shop Scheduling Problems. *IEEE/CAA Journal of Automatica Sinica*, **6**, 904-916. <https://doi.org/10.1109/jas.2019.1911540>
- [6] Wei, F.F., Cao, C.Y. and Zhang, H.P. (2021) An Improved Genetic Algorithm for Resource-Constrained Flexible Job-Shop Scheduling. *International Journal of Simulation Modelling*, **20**, 201-211. <https://doi.org/10.2507/ijsm20-1-co5>
- [7] Zhang, S., Li, X., Zhang, B. and Wang, S. (2020) Multi-Objective Optimisation in Flexible Assembly Job Shop Scheduling Using a Distributed Ant Colony System. *European Journal of Operational Research*, **283**, 441-460. <https://doi.org/10.1016/j.ejor.2019.11.016>
- [8] Mao, C.L. (2021) Production Management of Multi-Objective Flexible Job-Shop Based on Improved PSO. *International Journal of Simulation Modelling*, **20**, 422-433. <https://doi.org/10.2507/ijsm20-2-co11>
- [9] Du, B., Han, S., Guo, J. and Li, Y. (2024) A Hybrid Estimation of Distribution Algorithm for Solving Assembly Flexible Job Shop Scheduling in a Distributed Environment. *Engineering Applications of Artificial Intelligence*, **133**, Article ID: 108491. <https://doi.org/10.1016/j.engappai.2024.108491>
- [10] 王秋莲, 段星皓. 基于高维多目标候鸟优化算法的柔性作业车间调度[J]. 中国机械工程, 2022, 33(21): 2601-2612.
- [11] 吴迎晨, 肖彪, 赵正彩, 等. 柔性作业车间调度多策略果蝇优化算法研究[J]. 现代制造工程, 2023(5): 22.
- [12] Guo, J., Lei, D. and Li, M. (2021) Two-Phase Imperialist Competitive Algorithm for Energy-Efficient Flexible Job Shop Scheduling. *Journal of Intelligent & Fuzzy Systems*, **40**, 12125-12137. <https://doi.org/10.3233/jifs-210198>
- [13] Kamali, S.R., Banirostam, T., Motameni, H. and Teshnehlab, M. (2023) An Immune-Based Multi-Agent System for Flexible Job Shop Scheduling Problem in Dynamic and Multi-Objective Environments. *Engineering Applications of Artificial Intelligence*, **123**, Article ID: 106317. <https://doi.org/10.1016/j.engappai.2023.106317>

- [14] 彭建刚, 刘明周, 张铭鑫, 等. 多目标柔性作业车间调度算法研究综述[J]. 中国机械工程, 2014, 25(23): 3244-3254.
- [15] 单忠德, 汪俊, 张倩. 批量定制柔性生产的数字化, 智能化, 网络化制造发展[J]. 物联网学报, 2021, 5(3): 1-9.
- [16] 周佳军, 姚锡凡, 刘敏, 等. 几种新兴智能制造模式研究评述[J]. 计算机集成制造系统, 2017, 23(3): 624.
- [17] Márquez, C.R.H. and Ribeiro, C.C. (2022) Shop Scheduling in Manufacturing Environments: A Review. *International Transactions in Operational Research*, **29**, 3237-3293. <https://doi.org/10.1111/itor.13108>
- [18] Geurtsen, M., Didden, J.B.H.C., Adan, J., Atan, Z. and Adan, I. (2023) Production, Maintenance and Resource Scheduling: A Review. *European Journal of Operational Research*, **305**, 501-529. <https://doi.org/10.1016/j.ejor.2022.03.045>
- [19] Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002) A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, **6**, 182-197. <https://doi.org/10.1109/4235.996017>
- [20] Liu, Z., Liu, H., Liu, H. and Tan, J. (2024) Many-Objective Flexible Job-Shop Scheduling Based on a Loose Non-Dominated Sorting Genetic Algorithm. *Engineering Optimization*. <https://doi.org/10.1080/0305215x.2024.2354889>
- [21] Sun, J., Zhang, G., Lu, J. and Zhang, W. (2021) A Hybrid Many-Objective Evolutionary Algorithm for Flexible Job-Shop Scheduling Problem with Transportation and Setup Times. *Computers & Operations Research*, **132**, Article ID: 105263. <https://doi.org/10.1016/j.cor.2021.105263>
- [22] Shi, S. and Xiong, H. (2024) Solving the Multi-Objective Job Shop Scheduling Problems with Overtime Consideration by an Enhanced NSGA-II. *Computers & Industrial Engineering*, **190**, Article ID: 110001. <https://doi.org/10.1016/j.cie.2024.110001>