文章编号:1007－6735(2026)02－0239－14　DOI: 10.13255/j.cnki.jusst.20250116001

柔性路径变节与跨单元协同的调度节能优化

沈景凤[1]，　田　源[1]，　李嘉曜[2]，　仲梁维[1]，　苗　涛[1]
（1. 上海理工大学 机械工程学院，上海 200093；2. 上海理工大学 管理学院，上海 200093）

摘要：针对多柔性路径下单元制造系统生产效率低、生产能耗高的问题，从优化跨单元运输次数的角度出发，建立了以完工时间、总生产能耗和跨单元次数为优化目标的多目标模型，并提出一种变邻域非支配排序遗传算法进行求解。基于混合全局和随机生成解集法，选取跨单元次数较少的个体为初始解，设计基于跨单元运输的解码方式，并融合 3 种变邻域算子，降低运输及空闲时间对加工过程的影响，运用改进交叉和变异非支配排序方法得到高效的进化过程。最后，通过算例仿真验证改进算法的合理性，并在实例部分分析得出：均衡跨单元次数有利于协调优化完工时间与总能耗这两个目标，进一步证实了所建模型的优越性。

关键词：生产能耗；跨单元调度；跨单元运输次数；非支配排序遗传算法
中图分类号：TP 301　　**文献标志码：**A

Optimization for energy-saving scheduling with flexible path variation and intercell collaboration

SHEN Jingfeng[1], TIAN Yuan[1], LI Jiayao[2], ZHONG Liangwei[1], MIAO Tao[1]
(1. *School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China*; 2. *Business School, University of Shanghai for Science and Technology, Shanghai 200093, China*)

Abstract: To solve the problems of low production efficiency and high production energy consumption of cell manufacturing systems under multiple flexible routes, a multi-objective model was established that considered the completion time, total production energy consumption, and the number of intercell transportations from the perspective of optimizing the number of intercell transportations. According to the characteristics of the model, a variable neighborhood non-dominated sorting genetic algorithm was proposed for solving this model. Based on the combination of global and random solution generation methods, individuals with fewer intercell transportations were selected as the initial solutions. A

收稿日期: 2025−01−16
基金项目: 国家自然科学基金资助项目（72174121）
第一作者: 沈景凤（1967—），女，副教授．研究方向：优化算法．E-mail：sh_jf@163.com
通信作者: 仲梁维（1962—），男，教授．研究方向：智能设计．E-mail：zlvcad@126.com

引文格式: 沈景凤，田源，李嘉曜，等．柔性路径变节与跨单元协同的调度节能优化[J]．上海理工大学学报，2026，48（2）：239-252.
Citation: SHEN Jingfeng, TIAN Yuan, LI Jiayao, et al. Optimization for energy-saving scheduling with flexible path variation and intercell collaboration[J]. Journal of University of Shanghai for Science and Technology, 2026, 48(2): 239-252.

decoding method based on intercell transportation was designed, and three variable neighborhood operators were integrated to reduce the impact of transportation and idle time on processing. Then, the crossover and mutation non-dominated sorting methods were improved to achieve a high-quality evolutionary process. Finally, the rationality of the improved algorithm was demonstrated through simulation examples. In the case study section, it was shown that balancing the number of intercell transportations helps coordinate and optimize the two objectives of completion time and total energy consumption, further demonstrating the superiority of the established model.

Keywords: *production energy consumption*; *intercell scheduling*; *intercell transportation times*; *non-dominated sorting genetic algorithm*

在个性化定制推动企业生产向小批量多品种模式转型的背景下，产品多样化使单一单元难以独立完成任务，跨单元制造应运而生。已有研究[1-2]指出，跨单元系统中因设备加工能力重叠而形成的柔性路径，会产生跨单元运输，进而影响生产效率和成本，该问题被描述为柔性路径条件下的跨单元节能调度，即IESCFR（intercell energy-saving scheduling considering flexible routes）。

针对IESCFR问题，现有研究多从降低跨单元运输次数与时间来优化生产目标。Li等[2]提出基于信息素的多智能体调度方法以减少冗余跨单元次数；Subhaa等[3]建立联合单元形成与调度模型，在增加柔性路径的同时最小化加工成本；Feng等[4]通过划分单元数量减少工件流动时间；Deliktas等[5]及Feng等[6]分别从运输时间和遗传算法角度优化完工时间与负载；Kesen等[7]则关注批量运输效率。近年来，邹萌邦等[8]与马梦珂等[9]引入复杂网络理论，通过模块度和网络指标优化路径与成本。

实际调度需兼顾多因素，节能调度日益受到关注。Renna等[10]调整设备空闲策略以节能；Barak等[11]研究自动导引车（automated guided vehicle，AGV）运输能耗；张洪亮等[12]分析工人操作对能耗的影响；徐晨昕[13]在网络化多单元系统中构建了碳排放、时间与成本模型。然而，这些研究大多忽视了跨单元次数对能耗与负载均衡的深层影响。

综上，现有IESCFR研究多聚焦于降低跨单元运输时间，但本研究发现，仅优化时间可能导致跨单元次数不合理：过少会引起跨单元次数过低、设备负载不均、空闲时间增加；过多则增加运输能耗。而当存在瓶颈设备时，适度增加跨单元次数反而有助于优化时间与能耗，此平衡机制尚未得到充分研究。基于资源需求差异的均衡调度对提升资源利用率与降低能耗尤为重要[14]，本文将跨单元运输次数作为第三目标，结合改进的第二代非支配排序遗传算法（non-dominated sorting genetic algorithm II，NSGA-II）和邻域搜索算法，探究其对生产效率与能耗的影响。仿真结果表明，优化跨单元次数能有效均衡完工时间与生产能耗。

1 问题描述与数学模型

IESCFR为柔性作业车间的扩展问题，工件的各道工序首先需要被分配到各个单元，再从单元内分配合适的设备完成加工。

1.1 模型建立

图1所示的IESCFR问题中，路径1的工序O_1、O_2和O_3在同一单元U_1中加工，路径2的工序O_1和O_2在同一单元U_1中加工，O_3则需要在单元U_2中加工，由此产生了跨单元路径。问题描述：工件集$i\in\{1,2,\cdots,n\}$，工件有$j\in\{1,2,\cdots,h\}$道工序，需要在设备集$k\in\{1,2,\cdots,m\}$上加工，每道工序可选择一台或多台设备加工。多台柔性设备被部署在不完全相同的制造单元$u\in\{1,2,\cdots,c\}$中，且具有不同加工能力。调度目标就是为每道工序安排合适的单元和设备并明确工序的加工顺序，通过柔性路径的动态调整，即根据实时状态（设备负载、运输能耗、单元间距离）动态调整工序的加工路径，避免瓶颈设备过载，使给定目标最优。IESCFR问题中，工件在单元间流动产生的跨单元运输不可避免，当单元u内设备k处于瓶颈时，将工序O_{ij}分配至单元u的设备k，虽然增加了运输时

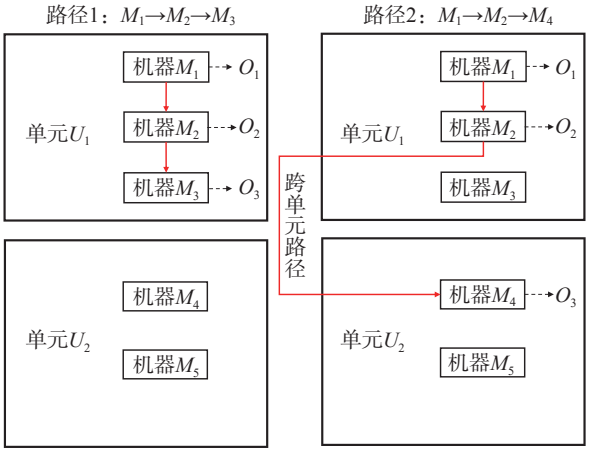


图 1　柔性路径下跨单元节能调度的加工路径

Fig.1　Processing routes of IESCFR

间与能耗，但可能缩短了完成时间。因此，IESCFR 需要考虑运输对目标的影响。

IESCFR 应满足柔性路径下跨单元调度的限制约束：

a. 同一时刻一道工序只能选择一个单元内的一台设备加工；

b. 同一时刻同台设备只能加工一道工序；

c. 工件之间加工优先级相同；

d. 不考虑中断情况；

e. 同一工件不同工序有顺序约束，即工序关系；

f. 加工和运输能力充足，且所有单元运输能力相同；

g. 不考虑工件设置和卸载时间。

1.2　模型建立与参数

由完工时间 f_1、总能耗 f_2 和跨单元运输次数 f_3 共同构成了 IESCFR 的 3 目标优化问题。

完工目标的计算公式为

$$f_1 = \min T_{\max} = \min \sum T_{ku}^{\text{end}} \tag{1}$$

式中：$T_{\max}$ 为整个系统的完工时间；T_{ku}^{end} 指单元 u 中设备 k 上最后一道工序的完工时间。

总能耗目标的计算公式为

$$f_2 = \min E_{\text{total}} = \min\left(\sum_{k=1}^{m} E_{ku}^{\text{pro}} + \sum_{k=1}^{m} E_{ku}^{\text{idle}} + \sum_{u=1}^{c} E_{uu'}^{\text{trans}}\right) \tag{2}$$

式中：E_{total} 为全局能耗；u 和 u' 为单元索引；k 为设备索引；E_{ku}^{pro}、E_{ku}^{idle} 分别为单元 u 中设备 k 的加工能耗和空闲能耗；$E_{uu'}^{\text{trans}}$ 为单元 u 到 u' 间的运输能耗；m 和 c 分别为设备数量和制造单元数量。

加工能耗计算公式为

$$E_{ku}^{\text{pro}} = \sum_{i=1}^{n}\sum_{j=1}^{h}\sum_{k=1}^{m}\sum_{u=1}^{c} t_{ijku} x_{ijku} e_{ku}^{\text{pro}} \tag{3}$$

式中：i 为工件索引；j 为工序索引；n 为工件数量；h 为工序数量；t_{ijku} 为工序加工时间；x_{ijku} 为决策变量；e_{ku} 为单位加工能耗。

空闲能耗表达式为

$$E_{ku}^{\text{idle}} = \sum_{k=1}^{m}\sum_{u=1}^{c} e_{ku}^{\text{idle}} \sum_{k=1}^{m} T_{k}^{\text{idle}} \tag{4}$$

式中：e_k^{idle} 为设备 k 的单位空闲能耗；T_k^{idle} 为设备 k 加工过程中的空闲时间。

运输能耗由电驱引导车进行单元间工件转移产生，计算公式为

$$E_{uu'}^{\text{trans}} = e_{uu'}^{\text{trans}} \sum_{i=1}^{n}\sum_{j=1}^{h}\sum_{k=1}^{m}\sum_{u=1}^{c} t_{uu'} q_{ij(j+1)} \tag{5}$$

式中：$e_{uu'}^{\text{trans}}$ 为单位运输能耗；$t_{uu'}$ 为单元运输时间；$q_{ij(j+1)}$ 为决策变量。

跨单元次数计算公式为

$$f_3 = \min \sum_{i=1}^{n}\sum_{j=1}^{h} R_{ij} \tag{6}$$

$$R_{ij} = \sum_{i=1}^{n}\sum_{j=1}^{h} q_{i(j+1)} \tag{7}$$

式中：R_{ij} 表示工件 i 产生的相邻工序的跨单元移动次数之和（仅当工序 j 与 $j+1$ 相邻并在不同单元加工时计为一次跨单元移动）；$q_{i(j+1)}$=1 时在相邻工序下产生一次跨单元。

文献 [15] 借鉴 Wagner“机床位置”建模思想，通过定义设备位置索引，以设备起止时间等变量确定设备位置，从而在求解中消除冗余工序−设备时间转换，提升算法效率。本文沿用该思想，设定工序开始时间 T_{ij} 和设备开始时间 T_{ku} 的映射关系，算法运行时直接调用 T_{ku}，确定工序开工时间。

决策变量 x_{ijku}、$y_{iji'j'}$ 与 $y_{ij(j+1)}$ 应满足：

$$\sum_{i=1}^{n}\sum_{j=1}^{h}\sum_{k=1}^{m} x_{ijku} \leqslant 1 \tag{8}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{h} y_{ij(j+1)} \leqslant 1 \tag{9}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{h}\sum_{k=1}^{m} y_{iji'j'} \leqslant 1 \tag{10}$$

式(8)表示任一工件 i 的工序 j 只能选择在一个单元 u 中的一台设备 k 上加工；式(9)表示同一工件的紧前紧后工序关系；式(10)指同一设备上前后工序的关系。

$$T_{ijku}+\sum_{i=1}^{n}\sum_{j=1}^{h}\sum_{k=1}^{m}\sum_{u=1}^{c}\left(t_{ijku}x_{ijku}+t_{uu'}q_{i(j+1)}\right)\cdot y_{i(j+1)}\leqslant T_{i(j+1)ku} \tag{11}$$

式(11)为工件紧前紧后工序开始时间的关系，$T_{i(j+1)ku}$为 $j+1$道工序的开始时间，该工序需要跨单元加工时，$t_{uu'}q_{i(j+1)}$的值为0，反之则需计算。

$$T_{ku}\geqslant T_{ij}+L\left(1-x_{ijku}\right),k\in[1,m],u\in[1,c] \tag{12}$$

$$T_{ij}\geqslant T_{ku}+L\left(1-x_{ijku}\right),k\in[1,m],u\in[1,c] \tag{13}$$

式中，L 表示一个极大数。式(12)和式(13)指设备开始时间等于其所加工工序开工时间。

工序和设备的允许开始时间判断：当$T_{ku}\geqslant T_{ij}$时，即工序O_{ij}已允许加工但设备k还没完成先前加工，此时式(12)生效。为了保证最后的开始时间统一，需要式(12)和(13)同时生效，x_{ijku}=1 满足式(12)和(13)，即$T_{ku}=T_{ij}$。

$$t_{ku}=T_{ku}+\sum_{i=1}^{n}\sum_{j=1}^{h}\sum_{k=1}^{m}\sum_{u=1}^{c}t_{ijku}x_{ijku} \tag{14}$$

式中，T_{ku}为设备加工一道工序的完工时间。

$$T_k^{\text{idle}}=\sum_{i=1}^{n}\sum_{k=1}^{m}\left(T_{i'j'ku}-T_{ijku}\right) \tag{15}$$

式中：ij和$i'j'$指在同一设备上分配的前后两道工序。在分配当前工序时，若产生的空闲时间中包含运输时间，根据式(12)～(14)可推得：

$$T_k^{\text{idle}}\geqslant T_{ku}-t_{ku}+t_{ij}+t_{uu'} \tag{16}$$

式中，t_{ij}为工序的加工时间。

当工序需要在t_{ku}大于T_{ku}时，$t_{ij}+t_{uu'}$会缩小前段阈值。

$$R_{ij}\left(t_{uu'}q_{i(j+1)}\right)=T_{i(j+1)ku}-\sum_{i=1}^{n}\sum_{j=1}^{h}\left(T_{ijku}+t_{ijku}x_{ijku}\right) \tag{17}$$

式(17)指跨单元次数R_{ij}根据前后两道工序判断是否产生，其与任一工件的完工时间并非线性关系，后续将对其关系进行实验说明，$q_{i(j+1)}=1$表示跨单元产生。

$$T_{ku}^{\text{start}}=\min\sum_{k=1}^{m}T_{ku} \tag{18}$$

$$T_{ku}^{\text{end}}=\max\sum_{k=1}^{m}t_{ku} \tag{19}$$

式(18)和(19)分别为任一设备在任一时间的最早开始时间和最终完成时间。

$$x_{ijku}=\begin{cases}1,\ O_{ij}\text{被分配到单元}u\text{的机器}k\\0,\ \text{其他}\end{cases} \tag{20}$$

$$y_{iji'j'}=\begin{cases}1,\ O_{ij}\text{在}O_{i'j'}\text{之前加工}\\0,\ \text{其他}\end{cases} \tag{21}$$

$$q_{i(j+1)}=\begin{cases}1,\ O_{ij}\text{和}O_{i(j+1)'}\text{跨单元加工}\\0,\ \text{其他}\end{cases} \tag{22}$$

2 算法设计

NSGA-II算法适用于求解离散调度问题，对IESCFR问题具有较好的适用性[15]。本文结合柔性路径问题特征，提出一种融合非支配排序遗传算法与变邻域搜索的混合方法，即改进的变邻域非支配排序遗传算法(improved variable neighborhood non-dominated sorting genetic algorithm，IVNSGA)。非支配排序遗传算法用于求解多目标调度模型，变邻域搜索方法用于在交叉变异后进一步优化解集，提高搜索精度。算法亦对各部分进行改进以增强整体性能，流程图如图 2 所示，具体步骤如下：

Step 1 设置算法参数，包括最大迭代次数$K_{\max}$，种群数量N，基础交叉率p_{kc}，基础变异率p_{km}，父代种群数量K_{pop}，精英保留概率λ_1和λ_2，邻域搜索退火概率λ_{pns}；

Step 2 混合种群初始化，根据跨单元次数筛选前N代作为初始种群；

Step 3 初始化种群解码，进行非支配排序和拥挤度计算；

Step 4 父代P_t种群根据锦标赛法选择交配对象，进行交叉和变异操作，生成子代种群C_t；

Step 5 根据改进精英保留策略选择规模为N的解集作为进化种群G_{t+1}；

Step 6 对种群G_{t+1}使用变邻域搜索算子进行更新，生成全新种群作为下一次迭代父代种群P_{t+1}。

2.1 初始解生成

种群初始解的质量直接影响进化效果。本文首先通过混合全局选择GS(global searching)和随机选择法生成$(1+n\%)N$数量的个体；随后计算各

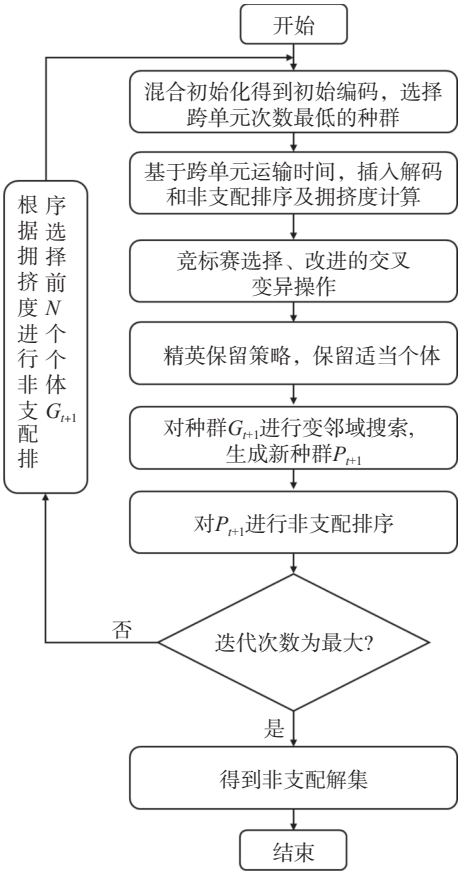


图 2　改进变邻域非支配排序遗传算法流程图

Fig.2　Algorithm flowchart of IVNSGA

初始个体的跨单元次数，按升序排序后选取前若干个个体组成初始种群。

2.2　编　码

编码是算法实现的基础。针对 IESCFR 问题特性，本文采用 Feng 等[4] 提出的 3 层染色体编码结构，以弥补传统两段式编码在单元特征表达上的不足。工序层X_p的数字表示工件编号，其出现次数对应该工件的工序总数。如图 3 中首位数字“3”代表工件 3 的第一道工序。机器层X_m对应工序层，数字表示对应工序在其可用设备集中的选择序号。如图 3 中工件 1 的工序 2 可被分派在[1，2，3]设备，则数字“3”表示工序 2 选择设备 3 作为加工设备。单元层用于标识工序加工所属单元，并在解码时判断是否跨单元及计算运输时间。如图 3 中的“1”表示在单元 1 加工；虚线框内数字为运输时间，“0”表示首道工序无运输，后续“5”表示工件从单元 1 移至单元 2 需 5 个单位运输时间。

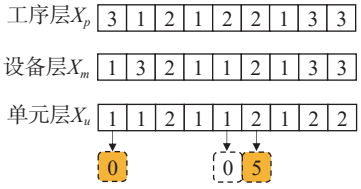


图 3　三层式编码

Fig.3　Three-layer coding

2.3　解　码

解码旨在将编码获得的染色体还原为实际加工过程。本文借鉴张洪亮等[16] 在分布式调度中的解码思想，结合跨单元调度特点，提出一种基于跨单元运输时间的贪婪插入解码方法，有效缩短空闲时间并加快算法收敛。该解码过程根据工序能否插入设备空闲时段分为同单元可插入、跨单元可插入与无法插入 3 种情形；若无法插入，则采用延后插入策略予以处理。

同单元可插入分为两种情景（见图 4）：

情景 1　工序O_{ij}的完工时间大于机器M_i^c原分配的工序$O_{l(h-1)}$的开始时间，且小于O_{lh}的完工时间，$T_{ij}=T_{ku}=C_{l(h-1)}$。

情景 2　紧前紧后工序在同一单元内两台机器上加工，工序O_{ij}的完工时间大于机器M_i^c原分配的工序$O_{l(h-1)}$的开始时间，且小于O_{lh}的完工时间，$T_{ij}=T_{ku}=C_{i(j-1)}$。

跨单元可插入分为两种情景（见图 5）：

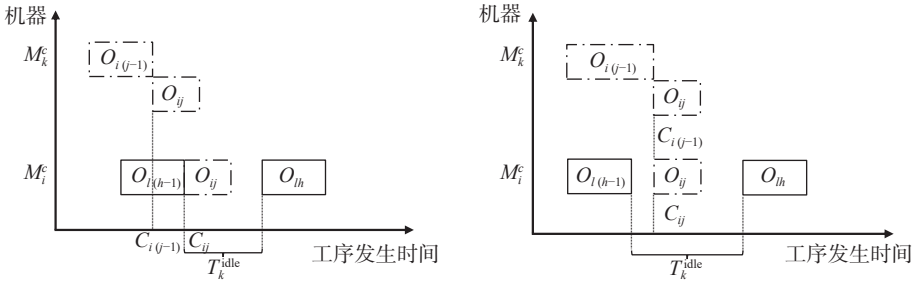


图 4　同单元机器空闲窗可插入

Fig.4　Insertable idle windows of same-cell machines

情景3　工序O_{ij}加工完成产生运输时间t_c，运输完成时间大于机器M_i^c原分配工序$O_{l(h-1)}$的开始时间且小于O_{lh}的完工时间，$T_{ij}=T_{ku}=C_{i(j-1)}+t_c$。

情景4　工序O_{ij}的跨单元运输时间t_c小于机器M_i^c原分配工序O_{lh}的开始时间，且小于O_{lh}的完工时间，$T_{ij}=T_{ku}=C_{l(h-1)}$。

同一设备上工序按序加工，时间窗口无重叠；不同工件工序在同一设备上冲突时，则标记为“无法插入”，并采用“延后处理”解决。无法插入分为两种情景（见图6）：

情景5　工序O_{ij}加工后可能产生运输时间t_{cg}，两台设备处于同一单元时，$t_{cg}=0$，空闲时间T_k^{idle}小于O_{ij}的加工时间，不满足插入条件，$T_{ij}=T_{ku}=C_{l(h-1)}$。

情景6　工序O_{ij}加工后可能产生运输时间t_{cg}，两台设备处于同一单元时，$t_{cg}=0$，空闲时间T_k^{idle}大于O_{ij}的加工时间，满足插入条件，但O_{ij}的开始时间大于$O_{l(h-1)}$，因此无法插入，$T_{ij}=T_{ku}=C_{l(h-1)}$。

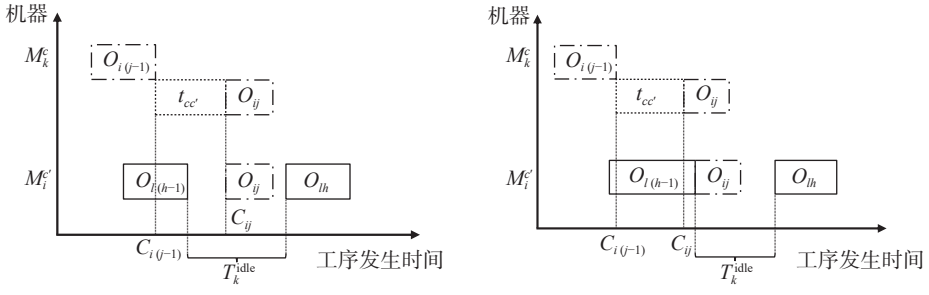


图5　跨单元机器空闲窗可插入

Fig.5　Insertable idle windows of intercell machines

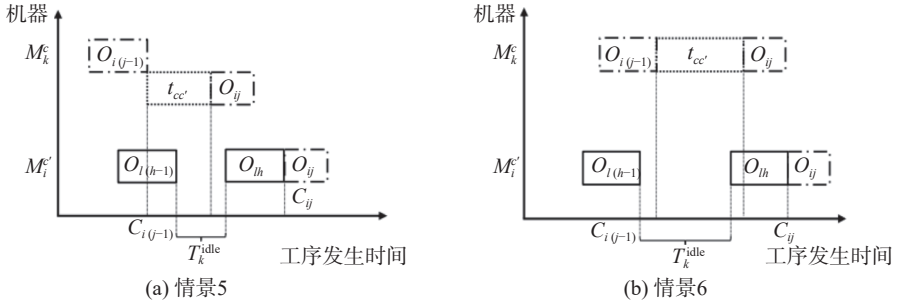


图6　无空闲窗延后插入

Fig.6　Delayed insertion without idle windows

当解码器检测到“无法插入”时，启动延后插入机制：通过回溯至先前调度步骤，尝试其他可行插入位置；若仍失败，则持续循环回溯直至获得可行解。

2.4　交叉和变异

为提升子代质量与种群多样性，本文在模拟二进制交叉（simulated binary crossover，SBX）基础上采用混合交叉（hybrid crossover，HX）[17]策略。针对工序、设备与单元层的不同编码特性，分别设计交叉方法，并以随机数$r=\text{rand}[0,1]$函数决定对工序层进行多点交叉或对设备层进行交叉。若$r\leqslant 0.5$，则对工序层进行多点交叉；若$r>0.5$，则对设备层交叉。在每次交叉过程中对种群中的10%进行单元层交叉，以保留优良单元基因，加速前期收敛。

工序层交叉：如图7所示，从两父代P_1和P_2中随机选取工件子集J，将P_1中属于J工件的信息和P_2中不属于J工件的信息复制生成子代C_1，将P_1中不属于J工件的信息和P_2中属于J工件的信息复制生成子代C_2。

设备层交叉：如图8所示，将随机选择的包含工序码和设备码的父代P_1和P_2按随机位置列表交换设备信息生成子代C_1和C_2，若产生非法设备点（设备编号超出可用范围），则将其修复为随机可行设备，并同步调整单元层编码。图8中，若第6个点位由可行设备1变为可行设备3，但其对

应的工序只拥有可行设备 1 和 2，则 3 为非法设备点，此时将可行设备 3 随机修复为可行设备 1 或 2。单元层需要同时跟随设备码进行修复，图 7 中第二个位置的设备 3 被替换为设备 2，单元 2 替代单元 1。

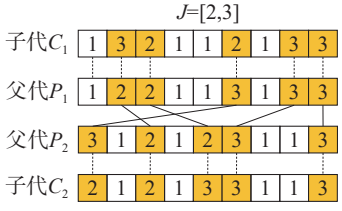


图 7　工序层交叉操作

Fig.7　Operation-level crossover operation

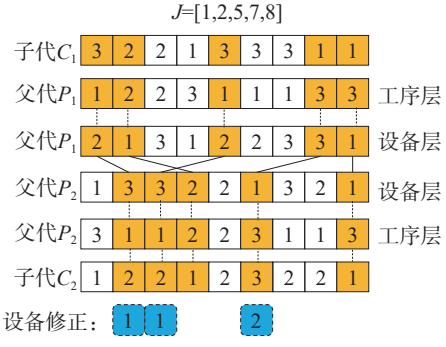


图 8　设备层交叉操作

Fig.8　Equipment-level crossover operation

单元层交叉：如图 9 所示，交换两父代 P_1 和 P_2 中同一工件对应工序的单元编码，生成新子代 C_1 和 C_2。若新单元内无该工序可用设备，则回退至原单元与设备，避免出现非法解并维持解集质量。

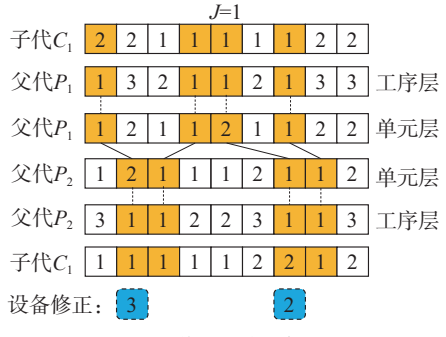


图 9　单元层交叉操作

Fig.9　Cell-level crossover operation

变异操作包括工序层变异与设备单元层变异。工序层变异：如图 10 所示，随机选择工序序列中的两个位置，交换其对应的工序编码，生成新的子代，以改变工序加工顺序，增加种群多样性。设备单元层变异：如图 11 所示，随机选取设备层两个基因位，将其替换为该工序可行的其他

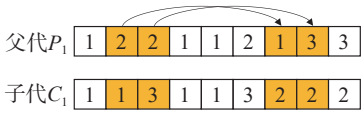


图 10　工序层变异操作

Fig.10　Operation-level mutation operation

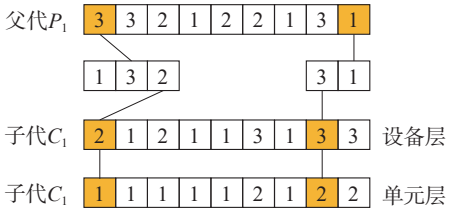


图 11　设备单元层变异操作

Fig.11　Equipment-cell level mutation operation

设备，并同步更新对应的单元编码，确保解的有效性。通过这种双点基因变异方式，可同时变异设备与单元，避免非法解集生成。

在多数生产调度问题中，交叉与变异概率常设为固定值，通常交叉率取 [0.9,1]，变异率不超过 0.1[18]。为提升搜索效率并抑制早熟，研究多采用自适应调整策略，主要包括线性和非线性两种模式。针对复杂调度问题中线性方法稳定性不足的缺点，本文设计一种基于 sigmoid 的自适应两阶段自适应交叉率。根据种群进化阶段调整的自适应交叉率 p_c 和变异率 p_m 增强算法能力，具体为

$$p_c=\begin{cases}\dfrac{0.1}{p_{kc}+\mathrm{e}^{\frac{f^{\mathrm{b}}(n)-f_{\mathrm{avg}}}{f_{\max}-f_{\mathrm{avg}}}}}+0.9, & f(n)>f_{\mathrm{avg}}\\ 0.9, & f(n)\leqslant f_{\mathrm{avg}}\end{cases} \tag{23}$$

$$p_m=\begin{cases}0.1, & f(n)>f_{\mathrm{avg}}\\ 0.1-\dfrac{0.1}{p_{km}+\mathrm{e}^{\frac{f(n)-f_{\mathrm{avg}}}{f_{\max}-f_{\mathrm{avg}}}}}, & f(n)\leqslant f_{\mathrm{avg}}\end{cases} \tag{24}$$

式中：基础交叉率 $p_{kc}\in[0,0.5]$，基础变异率 $p_{km}\in[0.4,1]$；$f(n)$ 为当前个体适应度；f_{avg}、$f_{\max}$、$f_{\min}$ 分别为种群适应度均值、最大值与最小值；$f^{\mathrm{b}}(n)$ 为交叉个体中较优适应度。当个体适应度较差（即 $f(n)$ 较大）时，采用较高交叉与变异率以促进进化；当个体较优时，则适当降低概率。若 $f(n)\leqslant f_{\mathrm{avg}}$，说明种群已收敛至较优，此时以稳定概率进化以提升多样性，总适应度 f_{fit} 计算为

$$f_{\mathrm{fit}}=\sum_{n=1}^{3}\frac{f(n)-f_{\min}}{f_{\max}-f_{\min}} \tag{25}$$

式中，$f(n)$ 分别表示当前个体的完工时间 f_1、总能耗 f 和跨单元次数 f_3。

2.5 改进精英保留策略

传统 NSGA-II 算法的保留策略采用固定比例对父代种群进行一定数量的解集进行保留，迭代后期易出现大量一代支配解集，本文设计一个分布函数条件父代精英种群数量，如下所示：

$$G_{i+1}=\begin{cases}F_1^{\mathrm{par}}+\lambda_1 F_n^{\mathrm{par}}, F_1^{\mathrm{off}}\leqslant 0.5G_i\\ \mathrm{round}\left(0.1F_1^{\mathrm{par}}\right)+\lambda_2 F_n^{\mathrm{par}}, F_1^{\mathrm{off}}>0.5G_i\end{cases} \tag{26}$$

式中：F_1^{par}与F_n^{par}分别为父代 1 和 n 级非支配解集，F_1^{off}与F_n^{off}为对应子代解集；G_i为当前迭代种群；G_{i+1}为保留后种群；条件系数$\lambda_1=0.5$、$\lambda_2=0.6$。

当前期$F_1^{\mathrm{off}}\leqslant 0.5G_i$时，优先保留父代精英解；当 $F_1^{\mathrm{off}}>0.5G_i$时，则提高劣解比例并少量保留最优解，以防局部收敛。精英保留策略步骤如下：

a. 建立外部存档文件 Outfit，用于存放当前可选非支配排序解集；

b. 按式 (24) 计算种群并存入 Outfit；

c. 合并父代与子代，去重后重新进行非支配排序与拥挤度计算，结果存入 Outfit；

d. 若 Outfit 规模超过设定值，则通过非支配排序与拥挤度计算保留前若干优秀个体，作为下一代种群。

2.6 变邻域搜索算子

当个体目标值相同或处于同一帕累托前沿时，邻域搜索可通过局部插入与变换改进解的结构，提升解的质量。本文设计 3 种局部搜索算子，以降低随机搜索盲目性，增强算法跳出局部最优的能力，并结合模拟退火思想设置接受概率以避免种群退化。

a. 邻域结构VS_1: 跨单元次数优化。通过降低冗余跨单元次数，减少由之引发的空闲与运输时间，从而优化完工时间与能耗。

b. 邻域结构VS_2: 负载均衡优化。针对工序集中少数单元导致的负载不均问题，重新分配工序，以提升单元与设备间的加工平衡性。

c. 邻域结构VS_3: 关键工序优化。在迭代后期重点优化最后一道工序的完工时间，通过调整其加工单元或设备以压缩最大完工时间。

针对柔性路径中“动态设备选择”与“多单元分配”特性，VS_1基于式(7)的跨单元次数定义，VS_2基于式(14)～(16)的空闲时间模型定义，VS_3基于式(18)～(19)的关键路径识别。结合工序在多单元设备间的可分配性，实现动态路径优化、负载均衡与关键路径缩短。在搜索过程中，优先处理具有多单元可选设备的工序，充分发挥柔性路径的调度潜力。

3 数值实验和分析

3.1 算例构造和评价指标分析

当前暂无基准算例集用于 IESCFR，因此，本文选取 Brandimarte 算例集中的 MK01～MK08 以及 Hurink 算例中的 LA16 和 LA26 为基础，根据文献 [4] 设备单元划分方式对算例进行不同的单元划分且不考虑并行机。考虑到算例多样性，算例集具有不同单元和设备数量。本文考虑绿色环境，拓展能耗信息为：设备单位加工能耗$e_{ku}\in[10,20]$，单位空闲能耗$e_k\in[1,4]$，单元间单位运输能耗$e_{uu'}\in[5,10]$，生成新算例的参数如表 1 所示。

表 1 算例信息

Tab.1 Example information

算例	工件数	工序数	加工时间/min	设备数	单元数	运输时间/min
MK01-2	10	[5,6]	[1,6]	6	2	8
MK02-3	10	[5,7]	[1,6]	6	3	8,10
MK03-3	15	[5,8]	[1,6]	8	3	20,30
MK04-2	15	[5,8]	[3.19]	8	4	20,25,30
MK05-2	15	[7,10]	[3,19]	4	2	20
MK06-4	10	[9,10]	[32,96]	15	4	30,40,50
MK07-3	20	[8,10]	[19,52]	10	3	20,25
MK08-4	20	[8,10]	[19.52]	10	4	15,20,25
LA16-4	20	[9,12]	[35,95]	10	4	20,25,30
LA26-5	20	[9,12]	[35,95]	10	5	20,25,30

本文采用 IGD（反世代距离）、最优值（$F_{\min}$）、平均值（F_{avg}）和最差值（$F_{\max}$）对算法性能进行评估。$F_{\min}$反映算法的求解上限，F_{avg}体现算法稳定性，$F_{\max}$则用于分析算法对算例的适应能力，平均最差值越大说明算法越存在不合理之处。IGD 是综合评价算法收敛性与分布性的指标，计算方式如式(27)所示，其值越小，说明所得非支配解集越接近真实帕累托前沿。

$$\mathrm{IDG}(F,F^*)=\frac{1}{|F^*|}\sum_{S_1\in F^*}\min_{S_2\in F}d(S_1,S_2) \tag{27}$$

式中，F^*表示真实的帕累托前沿。由于设计的算例暂无真实帕累托前沿解，本文对所有算法运行

得出的最优结果作为前沿解进行对比，F是由算法找到的第一帕累托前沿解，$d(S_1,S_2)$是帕累托前沿解F和F^*中两个个体的最短欧式距离。最优值和均值指标能直观反映算法性能。

3.2　算法性能对比

本文选择 NSGA-II 算法、TCGA(三层编码遗传算法)[4]、IWOA(鲸群算法)[19] 作为对比算法。其中，TCGA 用于求解跨单元调度问题，IWOA 为生成调度问题常用算法。参数设置影响着算法性能，IVNSGA-II（第二代改进变邻域非支配遗传算法）和 NSGA-II 中基础交叉率p_{kc}设置为 0.9，变异率p_{km}设置为 0.1。TCGA 根据原文最优取值，交叉率p_c为 0.85，变异率p_m为 0.01，精英保留率为 0.05。IWOA 的参数规定根据文献 [19]，位置向量纬度为 42°，螺旋系数 b=1，选择概率为 0.6，适应惯性权重分别为 0.9 和 0.5，权重因子为 0.3 和 0.7。每种算法的父代种群规模设置为 200。由于算例规模不同，越大规模的算例可能需要更大迭代次数以接近真实 Pareto 解，因此，为各算例设置不同的迭代次数。小规模设置为 200 代，中规模设置为 300 代，大规模设置为 450 代。每组算例再规定独立运行 20 次，通过 IGD 和部分算例的均值迭代解集进行对比分析，最优值加粗显示，结果如表 2 所示。

由表 2 可知，在 MK01-2、MK02-3 等小规模算例中，IWOA 的 IGD 最优，IVNSGA-II 与传统算法结果接近，说明工件数较少时本文算法优势不明显。对于中规模算例(工件数 15，工序数 75～120)，IVNSGA-II 的 IGD 显著低于其他算法，收敛性与分布性表现优异。在 MK06-4 这类设备量高于工件量的算例中，IVNSGA-II 的 IGD 虽非最优但接近 TCGA 与 INSGA-II，表明此时单元运输约束影响减弱，空闲时间对优化作用更突出。而在 LA16-4、LA26-5 等大规模算例中，IVNSGA-II 的 IGD 趋近于 0，显著优于对比算法，说明其 Pareto 解集更接近真实前沿，全局优化能力更强。

表 2　平均 IGD

Tab.2　Average IGD

问题	规模	IVNSGA-II	NSGA-II	TCGA	IWOA
MK01-2	10×6×2	0.2346	0.2541	0.2500	**0**
MK02-3	10×6×3	0.0641	0.0851	0.1810	**0**
MK03-3	15×8×3	**0.0833**	0.1395	0.3521	0.6096
MK04-2	15×8×4	**0.0976**	0.3660	0.2297	0.5163
MK05-2	15×4×2	**0.0995**	0.6286	0.2800	0.7810
MK06-4	10×15×4	0.1643	0.2048	**0.1581**	0.6936
MK07-3	20×10×3	**0.1015**	0.7239	0.2612	0.7796
MK08-4	20×10×4	**0.0857**	0.6408	0.2565	0.8261
LA16-4	20×10×4	**0.0917**	0.5714	0.3225	0.8078
LA26-5	20×10×5	**0.0258**	0.6357	0.2898	0.9310

进一步通过箱型图分析中大规模算例(表 3、图 12)显示，IVNSGA-II 在完工时间、总能耗与跨单元次数的均值多数情况下低于其他算法，且在大规模实例中显著优于对比方法。尤其在跨单元次数与总能耗方面，IVNSGA-II 明显优于基础 NSGA-II，验证了算法改进的有效性与综合性能优势，能够有效求解 IESCFR 问题。

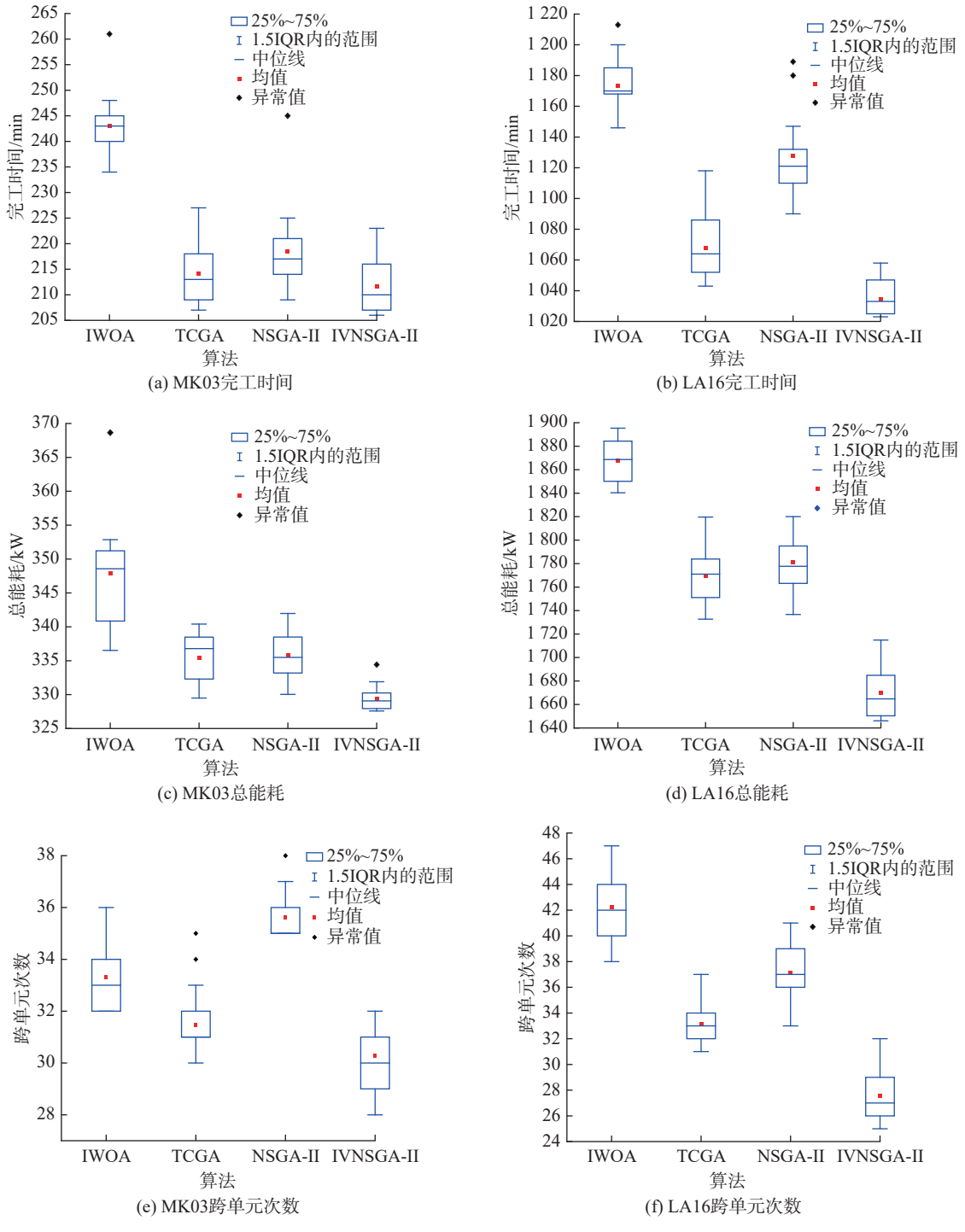
表 3　MK03 和 LA16 算例结果

Tab.3　Example results of MK03 and LA16

算例	优化目标	IVNSGA-II			NSGA-II			TCGA			IWOA		
		F_{max}	F_{avg}	F_{min}	F_{max}	F_{avg}	F_{min}	F_{max}	F_{avg}	F_{min}	F_{max}	F_{avg}	F_{min}
MK03-3	T_{max}	223.00	211.68	206.00	245.00	218.42	209.00	227.00	214.11	207.00	261.00	243.00	234.00
	E_{total}	334.42	329.40	327.58	339.81	335.84	332.23	340.40	335.38	329.49	368.65	347.90	337.52
	R	32.00	30.26	28.00	38.00	35.63	35.00	36.00	32.00	33.31	35.00	31.47	30.00
	T_{max}	1078.00	1035.26	1023.00	1189.00	1127.63	1090.00	1118.00	1067.52	1044.00	1213.00	1173.26	1146.00
LA16-4	E_{total}	1714.85	1665.31	1646.07	1820.04	1777.70	1736.65	1819.69	1765.13	1732.69	1895.35	1864.66	1843.51
	R	31.00	27.57	28.00	41.00	37.15	33.00	37.00	33.15	32.00	45.00	42.26	38.00

4　案例分析

为了进一步验证本文提出的多目标模型对实际案例的有效性，同时说明跨单元运输时间与其他调度目标的相关性，根据文献 [20-22] 中某个卡车零件生产车间实例作为对象，将车间数据整理



(a) MK03完工时间　(b) LA16完工时间

(c) MK03总能耗　(d) LA16总能耗

(e) MK03跨单元次数　(f) LA16跨单元次数

图 12　算法性能对比箱型图

Fig.12　Box plots for algorithm performance comparison

为 15×8×4(15 个零件，8 台设备和 4 个单元)的中规模生产数据，表 4 为设备在单元内归置数据，工厂内各个单元运输能力相同，运输工具速率为 2 m/s；表 5 为 Pareto 前沿解的目标值，表 6 为加工零件各项数据。

4.1　Pareto 解集分布

根据图 13 所示的 15×8×4 案例 Pareto 解集分布可见，跨单元次数在 12～14 范围内解集最为密集，且最优完工时间与总能耗均出现在跨单元次数为 12 的解集中。综合分析表明，跨单元运输次数越多，对完工时间和能耗的影响越大，合理控制跨单元次数有助于调度目标的优化收敛。

4.2　跨单元次数相关性分析

根据式 (14)、(15) 分析，跨单元运输次数与设

表 4　单元间距离和设备归置数据

Tab.4　Intercell distance and equipment placement data

生产单元	单元距离				包含设备
	U_1	U_2	U_3	U_4	
U_1	0	—	—	—	M_1、M_8
U_2	10	0	—	—	M_2、M_3
U_3	16	16	0	—	M_4、M_7
U_4	22	24	30	0	M_5、M_6、M_9、M_{10}

备完工时间呈非线性关系，且运输过程会通过增加空闲时间影响总能耗。为验证该关系，本文运用 IVNSGA-II 对实例运行 10 次，分析不同跨单元次数下各指标的均值相关性，结果如图 14 所示。由图可见，跨单元次数在 16～40 次范围内，完工时间总体呈下降趋势，其中 36～40 次间降幅显著；次数在 10～13 次时，完工时间出现波动，表明适当增加次数反而有助于缩短完工时间。总能耗在 19～40 次区间内随次数减少显著下降，而在 10～18 次区间内变化平缓，说明此时运输能耗不再是由总能耗主导，加工与空闲能耗占比更大。空闲时间在次数为 10～14 次时有所上升，反映适度跨单元可协助工序由瓶颈设备向空闲设备转移，优化调度结果。

表 5　Pareto 前沿解目标

Tab.5　Pareto frontier solution target

编号	完工时间/min	总能耗/kW	跨单元次数
1	49.15	74.15	12
2	51.56	72.97	12
3	49.18	74.67	11
4	50.41	74.68	10
5	50.65	73.83	11
6	53.50	73.57	13
7	52.75	74.50	14

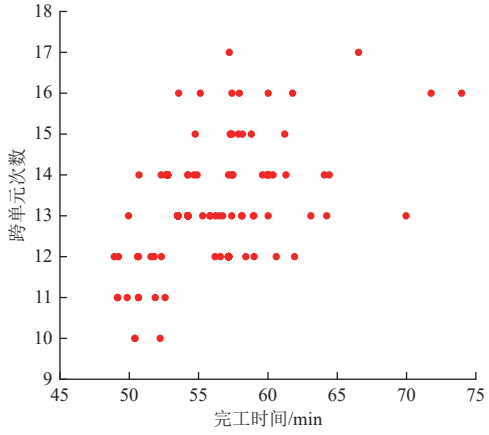
综上，跨单元次数在一定区间内降低有利于优化完工时间、总能耗与空闲时间；次数越多，运输能耗越高与时间越长，对其他目标影响越大。为更深入揭示运输次数与其他指标的联合作用，下文将结合两种情况的甘特图进一步说明，如图 15 所示。

分析表明，适度跨单元调度可在能耗与时间间实现平衡。图 15 中，$p(10,8)$=2.81 表示第 10 道工序在设备 8 上加工需要 2.81 min，工序 10-8 代表工序 10 在设备 8 上加工，表示某次最后完成的工序为 10-8。当工序 10-8 被调度至空闲设备 8，虽导致其与紧前工序 10-7 增加跨单元运输

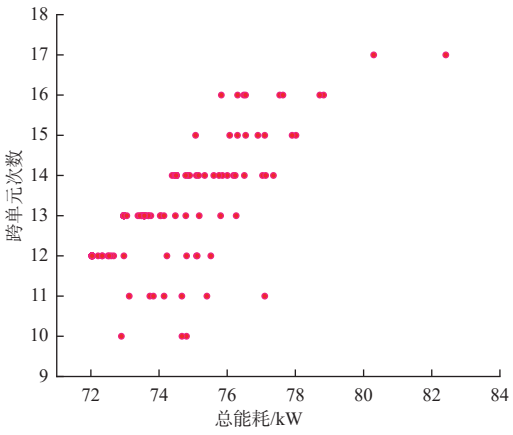
表 6　零件加工时间和加工能耗

Tab.6　Processing time and energy consumption of parts

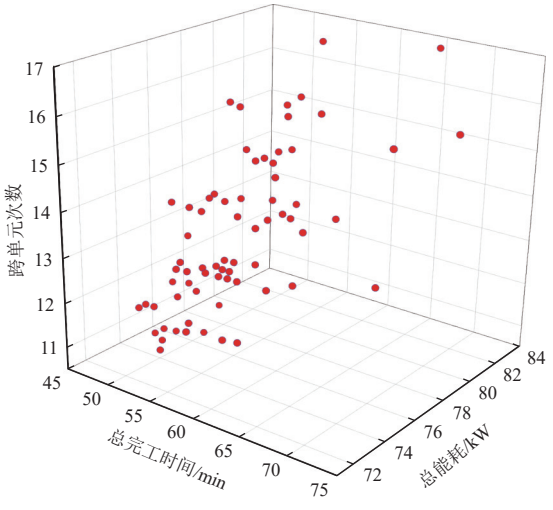
工件	工序	加工时间/min、设备能耗/kW									
		M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}
J_1	O_1	3.38、8	3.21、14	—	2.52、5.5	4.31、9.5	—	—	2.43、12	—	—
	O_2	—	4.61、14	5.06、10	—	—	2.90、10.75	2.90、11.75	—	—	3.83、11
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	O_7	2.96、8	—	4.61、10	3.48、5.5	—	2.52、10.75	3.87、11.75	3.34、12	—	—
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
J_7	O_1	3.99、8	4.70、14	2.02、10	—	3.88、9.5	3.03、10.75	4.84、11.75	—	—	—
	O_2	—	3.03、14	—	—	—	2.51、10.75	4.70、11.75	4.90、12	—	3.65、11
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	O_6	—	5.95、14	3.02、10	—	3.04、9.5	4.46、10.75	5.11、11.75	3.29、12	—	—
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
J_{14}	O_1	3.07、8	5.82、14	4.93、10	—	3.80、9.5	3.98、10.75	—	2.41、12	—	—
	O_2	—	5.47、14	—	—	2.45、9.5	3.87、10.75	—	—	4.53、8.75	3.86、11
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	O_5	—	—	2.27、10	2.30、5.5	2.49、9.5	—	—	2.05、12	—	—
J_{15}	O_1	—	—	—	—	—	—	4.93、11.75	3.13、12	5.19、8.75	—
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	O_6	—	4.30、14	—	2.02、5.5	2.40、9.5	4.43、10.75	2.29、11.75	—	—	—
	O_7	2.43、8	—	5.96、10	—	3.19、9.5	—	2.34、11.75	4.08、12	—	—



(a) 完工时间与跨单元次数关系图



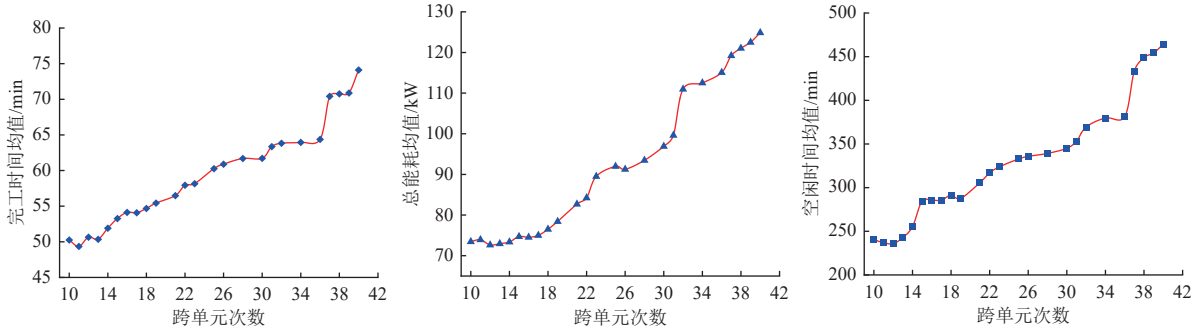
(b) 总能耗与跨单元次数关系图



(c) 完工时间、总能耗与跨单元次数关系

图 13 Pareto 散点图集合

Fig.13 Set of Pareto scatter plots



(a) 跨单元次数与完工时间相关性　(b) 跨单元次数与总能耗相关性　(c) 跨单元次数与空闲时间相关性

图 14 跨单元次数与其他指标间相关性

Fig.14 Correlation between intercell transportation times and other indicators

(11 min，0.57 kW)，但全局完工时间反由 49.15 min 降至 48.85 min，证明该调度可释放系统效率。因此，将跨单元次数作为决策变量有助于平衡效率与能耗。

4.3 模型对比

上述实验表明，跨单元次数可有效协调完工时间与总能耗间的冲突。为进一步验证跨单元次数对解质量的提升，本文将 IESCFR 模型(三目标)与仅考虑完工时间和总能耗的双目标模型进行对比，运用 IVNSGA-II 算法分别求解两模型。双目标模型中的目标函数值虽不包含跨单元次数，但能够通过解码过程记录产生的跨单元加工得到跨

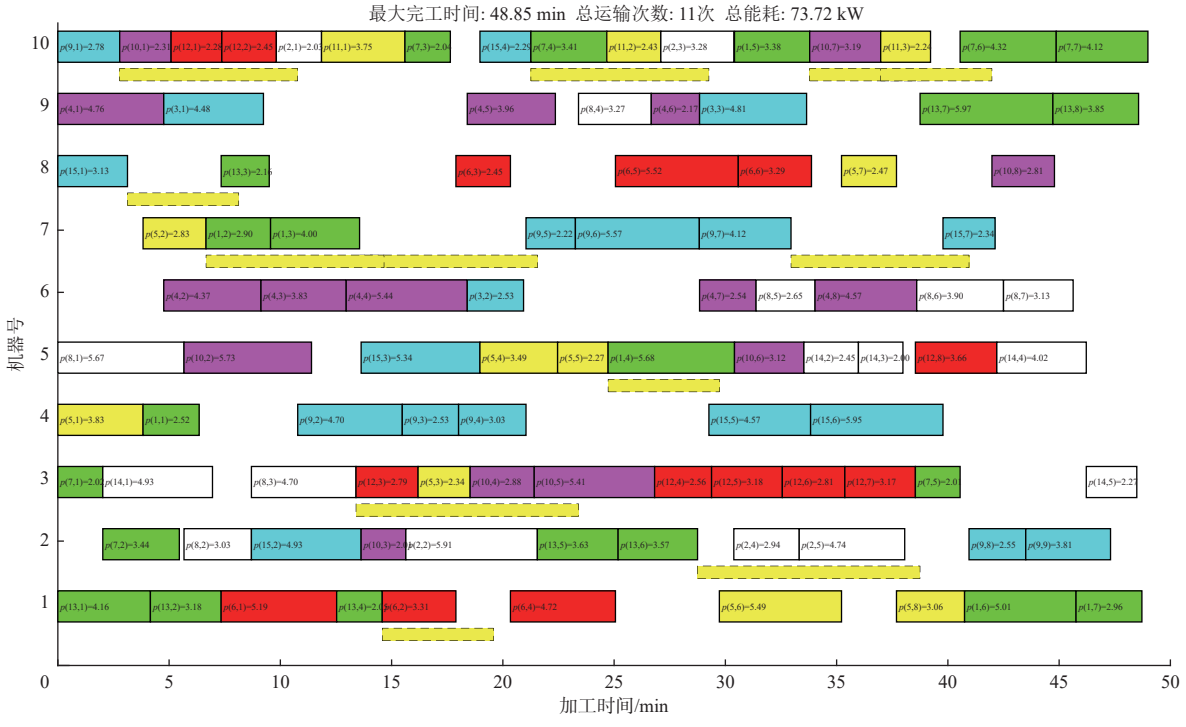


图 15　情景 2 甘特图

Fig.15　Gantt chart of case 2

单元次数。表 7 展示了两种模型的求解结果对比。

从表 7 可以看出，本文提出的 IESCFR 模型由于考虑跨单元次数，在求解过程中针对跨单元次数进行了优化，最终得出的完工时间和总能耗值均优于双目标模型。完工时间提前 13.48%，总能耗降低 1.89%，空闲时间减少 3.56%。其中，最大完工时间$T_{\max}$优化程度更突出。在上文已验证跨单元次数对于空闲时间的影响，可以看出 IESCFR 模型的完工时间改进程度最高，总能耗和空闲时间也有一定改进。

表 7　两种模型求解值对比

Tab.7　Comparison of the solution values of two models

IESCFR模型			双目标模型		
$T_{\max}$/min	E_{total}/kW	R	$T_{\max}$/min	E_{total}/kW	R
50.37	73.12	12	57.16	74.50	14

5　结　论

本文结合单元制造系统多柔性路径的结构特征，构建多目标数学模型，提升模型的工程实用性与求解精度。选取适配该模型的变邻域非支配遗传算法完成优化求解，并结合实例，深入剖析跨单元运输频次与生产效率、能耗之间的内在关联。研究表明，在调度后期合理增加跨单元运输次数，能够有效缩短最大完工时间。该研究为单元制造跨单元协同调度提供了决策参考与优化思路。未来可围绕生产扰动场景，进一步探究动态工况下跨单元运输策略对生产运行的影响。同时，考虑更加贴合实际生产的能耗构成要素，完善能耗计算体系，为跨单元车间节能调度优化问题提供更全面的理论支撑与技术依据。

参考文献：

[1] KRISHNAMOORTHY B, KAMATH M. Scheduling in a cellular manufacturing environment: a review of recent research[J]. Engineering Valuation and Cost Analysis, 2000, 2(5): 409−423.

[2] LI D N, WANG Y, XIAO G X, et al. Dynamic parts scheduling in multiple job shop cells considering intercell moves and flexible routes[J]. Computers & Operations Research, 2013, 40(5): 1207−1223.

[3] SUBHAA R, JAWAHAR N, PONNAMBALAM S G. An improved design for cellular manufacturing system associating scheduling decisions[J]. Sādhanā, 2019, 44(7): 155.

[4] FENG H X, XIA T B, DA W, et al. Concurrent design of cell formation and scheduling with consideration of duplicate machines and alternative process routings[J]. Journal of Intelligent Manufacturing, 2019, 30(1):

275−289.

[5] DELIKTAS D, TORKUL O, USTUN O. A flexible job shop cell scheduling with sequence-dependent family setup times and intercellular transportation times using conic scalarization method[J]. International Transactions in Operational Research, 2019, 26(6): 2410−2431.

[6] FENG Y L, LI G, SETHI S P. A three-layer chromosome genetic algorithm for multi-cell scheduling with flexible routes and machine sharing[J]. International Journal of Production Economics, 2018, 196: 269–283.

[7] KESEN S E, GÜNGöR Z. How important is the batch splitting activity in scheduling of virtual manufacturing cells (VMCs)?[J]. International Journal of Production Research, 2011, 49(6): 1645−1667.

[8] 邹萌邦, 刘琼, 尹勇. 基于改进小世界遗传算法的网络环境下跨单元调度 [J]. 计算机集成制造系统, 2019, 25(8): 1991−1999.

[9] 马梦珂, 倪静. 基于度值和聚类系数的跨单元调度问题优化 [J]. 计算机应用研究, 2021, 38(9): 2651−2656.

[10] RENNA P, MATERI S. Switch off policies in job-shop manufacturing systems including workload evaluation[J]. International Journal of Management Science and Engineering Management, 2021, 16(4): 254−263.

[11] BARAK S, MOGHDANI R, MAGHSOUDLOU H. Energy-efficient multi-objective flexible manufacturing scheduling[J]. Journal of Cleaner Production, 2021, 283: 124610.

[12] 张洪亮, 徐静茹, 谈波, 等. 考虑交货期的双资源柔性作业车间节能调度 [J]. 系统仿真学报, 2023, 35(4): 734−746.

[13] 徐晨昕. 网络环境下面向低碳的跨单元调度优化 [D]. 武汉: 华中科技大学, 2018.

[14] 徐雨婷, 陈世平. 面向云数据中心资源均衡分配需求的聚类调度算法研究 [J]. 上海理工大学学报, 2020, 42(4): 404−410.

[15] 孟磊磊, 张超勇, 肖华军, 等. 面向加工时间可控的柔性作业车间节能调度问题建模 [J]. 计算机集成制造系统, 2019, 25(5): 1062−1074.

[16] 张洪亮, 徐公杰, 鲍蔷, 等. 考虑运输时间的分布式柔性作业车间绿色调度 [J]. 中国机械工程, 2022, 33(21): 2554−2563.

[17] 宋存利. 求解多目标混合流水车间调度的改进 NSGA-Ⅱ[J]. 计算机集成制造系统, 2022, 28(6): 1777−1789.

[18] 王嵘冰, 徐红艳, 郭军. 自适应的非支配排序遗传算法 [J]. 控制与决策, 2018, 33(12): 2191−2196.

[19] 栾飞, 蔡宗琰, 吴书强, 等. 求解低碳车间调度问题的改进鲸鱼算法 [J]. 机械科学与技术, 2020, 39(5): 721−728.

[20] DELIKTAŞ D, öZCAN E, USTUN O, et al. Evolutionary algorithms for multi-objective flexible job shop cell scheduling[J]. Applied Soft Computing, 2021, 113: 107890.

[21] HAN Y Y, CHEN X H, XU M M, et al. A multi-objective flexible job-shop cell scheduling problem with sequence-dependent family setup times and intercellular transportation by improved NSGA-II[J]. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 2022, 236(5): 540−556.

[22] 张洪亮, 童超, 丁倩兰. 带有动态到达工件的分布式柔性作业车间调度问题研究 [J]. 安徽工业大学学报 (自然科学版), 2024, 41(5): 573−582.

（编辑：丁红艺）