

# 控制与决策

Control and Decision

## 带工序跳跃的绿色混合流水车间机器与AGV联合调度

耿凯峰, 叶春明

引用本文:

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.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2021.0318>

---

### 您可能感兴趣的其他文章

#### Articles you may be interested in

##### [求解阻塞混合流水车间调度的双层变异迭代贪婪算法](#)

A double level mutation iterated greedy algorithm for blocking hybrid flow shop scheduling

*控制与决策*. 2022, 37(9): 2323–2332 <https://doi.org/10.13195/j.kzyjc.2021.0607>

##### [区间数可重入混合流水车间调度与预维护协同优化](#)

Collaborative optimization of interval number reentrant hybrid flow shop scheduling and preventive maintenance

*控制与决策*. 2021, 36(11): 2599–2608 <https://doi.org/10.13195/j.kzyjc.2020.0973>

##### [基于改进蛙跳算法的分布式两阶段混合流水车间调度](#)

An improved shuffled frog leaping algorithm for the distributed two-stage hybrid flow shop scheduling

*控制与决策*. 2021, 36(1): 241–248 <https://doi.org/10.13195/j.kzyjc.2019.0472>

##### [超启发式交叉熵算法求解模糊分布式流水线绿色调度问题](#)

Hyper-heuristic cross-entropy algorithm for green distributed permutation flow-shop scheduling problem with fuzzy processing time

*控制与决策*. 2021, 36(6): 1387–1396 <https://doi.org/10.13195/j.kzyjc.2019.1681>

##### [自适应Jaya算法求解多目标柔性车间绿色调度问题](#)

Multi-objective flexible job shop green scheduling problem with self-adaptive Jaya algorithm

*控制与决策*. 2021, 36(7): 1714–1722 <https://doi.org/10.13195/j.kzyjc.2019.1773>

# 带工序跳跃的绿色混合流水车间机器与 AGV 联合调度

耿凯峰<sup>1,2</sup>, 叶春明<sup>1†</sup>

(1. 上海理工大学 管理学院, 上海 200093; 2. 南阳理工学院 信息化建设与管理中心, 河南 南阳 473004)

**摘要:** 针对带工序跳跃的绿色混合流水车间机器和自动引导车 (AGV) 联合调度问题, 提出改进 memetic algorithm (MA) 以同时最小化最大完工时间和总能耗. 首先, 设计基于工序、机器和转速的三层编码策略, 最大程度保证算法在整个解空间中搜索; 然后, 设计混合种群初始化方法以提高初始种群解的质量, 同时设计交叉和变异算子以及两种基于问题的邻域搜索策略来平衡算法的全局搜索和局部搜索能力; 最后, 通过大量仿真实验验证 MA 算法求解该问题的有效性和优越性.

**关键词:** 工序跳跃; AGV 调度; 绿色调度; 顺序相关调整时间

**中图分类号:** TP18      **文献标志码:** A

**DOI:** 10.13195/j.kzyjc.2021.0318

**引用格式:** 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.

## Joint scheduling of machines and AGVs in green hybrid flow shop with missing operations

GENG Kai-feng<sup>1,2</sup>, YE Chun-ming<sup>1†</sup>

(1. School of Business, University of Shanghai for Science and Technology, Shanghai 200093, China; 2. Information Construction and Management Center, Nanyang Institute of Technology, Nanyang 473004, China)

**Abstract:** For the joint scheduling of machines and automated guided vehicles (AGVs) in the green hybrid flow shop problem with missing operations, an improved memetic algorithm (MA) is proposed to minimize both the maximum completion time and the total energy consumption, simultaneously. Firstly, a three-layer encoding method based on operations, machines and speeds is designed to ensure the search of the algorithm in the whole solution space to the greatest extent. Then, a hybrid population initialization method is designed to improve the quality of initial population. Meanwhile, crossovers, mutation operators and two kinds of neighbourhood search strategies based on the problem are designed to balance the global and local search ability of the algorithm. Finally, simulation experiments show the superiority of the proposed algorithm in solving the joint scheduling of machines and AGVs in the green hybrid flow shop problem with missing operations.

**Keywords:** missing operations; AGV scheduling; green scheduling; sequence dependent setup times

## 0 引言

混合流水车间调度问题 (hybrid flow shop scheduling problem, HFSP) 作为一种经典调度问题在实际生产中应用非常广泛, 如电子、纺织、化工及半导体等行业<sup>[1-2]</sup>. 由于 HFSP 涉及工序和设备较多, 是经典的 NP-hard 问题, 开展关于 HFSP 的研究具有重要的科学价值和实际意义.

近年来, 环境问题日益凸显, 减少碳排放, 发展绿色经济, 已成为世界各国的共识. 作为能源密集型领

域, 制造业消耗了全球近三分之一的能源<sup>[3-4]</sup>. 研发节能设备是节能降耗最直接的途径, 然而, 它通常需要较长的周期和巨大的投资, 中小企业很难在短期内获益. 另外, 研究表明<sup>[5]</sup>, 绿色调度是实现节能降耗减排的另一种手段. 国内外学者对其开展了大量研究, 文献 [6] 针对低碳工艺规划与调度集成问题, 建立了考虑关机和重启策略的数学模型; 文献 [7] 研究了分时电价下单机调度问题, 以最小化电能成本为优化目标; 文献 [8] 提出了改进的 NSGA-II 算法求解多目标

收稿日期: 2021-02-24; 录用日期: 2021-07-05.

基金项目: 国家自然科学基金项目 (71840003); 上海市科委“科技创新行动计划”软科学重点项目 (20692104300); 河南省科技攻关项目 (182102210113).

责任编辑: 王凌.

†通讯作者. E-mail: yechm6464@163.com.

HFSP,并以最小化能耗和最大完工时间为优化目标;文献[9]利用改进差分进化算法求解考虑能耗的多目标HFSP;文献[10]提出了改进遗传算法通过调节机器转速来实现绿色调度。

随着多品种、小批量、个性化等生产模式的兴起,工艺稳定性难度增大,设备的专用性降低且柔性更强,严重影响了生产效率。在实际生产中,有些工件可能不需要在某些工位上进行加工,即存在工序跳跃的现象。允许工序跳跃的HFSP在钢铁、塑料以及电子等领域应用较为广泛,已取得部分研究进展。针对带工序跳跃的两阶段HFSP,文献[11]提出了一种三阶段启发式算法,并以最小化最大完工时间为优化目标。针对带工序跳跃的HFSP,文献[12]提出了混合遗传算法,以平均延迟时间最小为优化目标。文献[13]详细分析了带工序跳跃的HFSP的复杂性,并提出几种有效的启发式算法来求解该问题。另外,由于工序的跳跃特性,工件的加工路线更加灵活多样,流水车间中固定的自动化流转系统很难胜任物料的运输任务,这些运输任务通常由数量有限的AGV完成。据统计<sup>[14]</sup>,材料运输成本约占总生产成本的30%,说明运输环节对整个生产系统性能的影响至关重要。在柔性制造系统调度中,关于机器和AGV联合调度的研究已经取得较大进展。文献[15]提出了混合遗传算法求解柔性作业车间与AGV集成调度问题,并设计了二维向量编码方案和多种局部搜索策略,以最小化最大完工时间为优化目标。文献[16]研究了柔性制造系统中机器和AGV的调度问题,同时考虑了最优AGV数量、最短运输时间、路径规划问题和无冲突路径问题。文献[17]提出了禁忌搜索算法求解机器和AGV集成调度问题,并针对性地设计了基于工序和AGV分配的两种邻域搜索策略,有效地提高了算法的性能。分析可知,在以往关于机器和AGV联合调度的研究中,大多数研究集中在柔性作业车间领域,同时优化目标大都以时间相关指标为主,而与能耗、碳排放等环境相关的指标较少。

制造企业调查统计表明<sup>[18]</sup>:在实际生产中,非加工时间占整个生产时间的90%以上。因此,调整时间等辅助时间因素不可忽略。文献[19]对考虑调整时间的作业车间调度问题进行了综述,并分析了批量、非批量、顺序相关、顺序无关调整时间的调度问题。文献[20]提出了一种新型帝国竞争算法求解考虑顺序相关调整时间(sequence dependent setup times, SDST)的低碳柔性作业车间调度问题。文献[21]提出了一种改进的Jaya算法求解考虑运输时间和调整时间的

多目标柔性作业车间调度问题。文献[22]采用混合遗传算法求解同时考虑运输时间和SDST的流水车间调度问题。文献[23]提出一种遗传规划超启发式算法,用于求解考虑SDST的分布式置换流水车间调度问题。

在保证产品质量的前提下,优化资源分配策略、提高资源利用率、探索节能降耗减排机制、开展机器与AGV联合调度研究,对于节约能源成本、提高企业竞争力具有重要意义。从现有文献来看:1)尽管机器与AGV集成调度研究取得了一定的成果,但是从绿色调度视角将AGV与带工序跳跃的HFSP相结合的研究较少。2)在以往的研究中,调整时间通常放在加工时间中考虑或者被忽略,关于考虑SDST的机器和AGV联合调度的研究较少。基于此,本文以同时考虑工序跳跃特性和SDST的HFSP机器及AGV联合调度问题(MAHFSP-MO-SDST)为研究对象,以最小化最大完工时间和总能耗为优化目标。针对问题的特性,提出了改进的MA算法求解该问题。为了提高算法的性能,设计了基于工序、机器和转速的三层编码策略、混合种群初始化方法、交叉和变异算子以及两种基于问题的邻域搜索策略。最后,通过仿真实验验证了该算法的优越性。

## 1 问题描述

MAHFSP-MO-SDST可以描述为: $n$ 个工件需要在 $S$ 个工位上进行加工,在工位 $g$ 中有 $m_g$ 台并行机, $m_g \geq 1$ 。所有工件具有相似的工艺路线,按照同一方向访问一些工位,但部分工件不必在某些工位上进行加工,即工序跳跃。比如,某工件按照工位1-工位2-工位4-工位5的路径进行加工,则该工件跳跃了工位3。工序的调整时间不可忽略,且与相邻工序和所在的机器有关,即SDST。AGV用于在装载区、卸载区和机器之间执行运输任务。每台机器有3种状态,即加工状态、等待状态以及调整状态。每个AGV有3种状态:即负载状态、空载状态以及等待状态。在不同状态下,机器和AGV的单位能耗不同,且在等待状态下AGV单位能耗为0。为了简化模型,本文假设跳跃的工序也需要进行加工,只是加工时间、运输时间以及顺序相关调整时间均为0。

除经典HFSP假设外,本文主要假设条件如下:

- 1) 允许工序跳跃。
- 2) 在任一时刻,一个AGV只能运输一个工件,一个工件也只能被一个AGV运输。
- 3) 同一工位的机器完全相同,每个机器有几种不同的加工速度,加工速度越快单位能耗越高。

4) 缓冲区足够停放AGV和工件,且不考虑机器和AGV故障。

5) AGV在负载和空载状态下以相同的速度移动,且不考虑碰撞、路径重复以及电量影响。

6) 所有工序的标准加工时间、任意两台机器之间的运输时间以及顺序相关调整时间已知。

7) 所有AGV和工件的初始位置均在装载区,所有工件的最终位置均在卸载区。

8) 为了方便计算,在卸载区增加一道虚拟工位,该工序调整时间和加工时间为0,所有机器首道工序调整时间为0。

9) 假设每台机器具有 $H$ 个不同的加工速度 $V = \{V_1, V_2, \dots, V_H\}$ 。如果 $p_{igk}$ 为工件 $i$ 在机器 $M_{gk}$ 上的标准加工时间,则实际加工时间为 $\overline{p_{igkh}} = p_{igk}/V_h$ 。如果 $V_h > V_{h'}$ ,  $h, h' \in \{1, 2, \dots, H\}$ , 则 $\overline{p_{igkh}} < \overline{p_{igkh'}}$ 且 $E_{gkh}^{\text{busy}} \times \overline{p_{igkh}} > E_{gkh'}^{\text{busy}} \times \overline{p_{igkh'}}$ , 其中 $E_{gkh}^{\text{busy}}$ 表示机器以 $V_h$ 速度加工时的单位能耗。

MAHFSP-MO-SDST的主要任务包含生产调度和AGV调度两个子问题,目标是合理安排每台机器上工件的加工顺序和AGV的行程,使最大完工时间 $C_{\max}$ 和总能耗TE最小。

最大完工时间 $C_{\max}$ ,即makespan,指所有工件从装载区被运输至卸载区完成时间的最大值,即

$$C_{\max} = \max C_j, \forall j. \quad (1)$$

其中: $C_j$ 为工件 $j$ 的完工时间, $j$ 为工件编号, $j \in \{1, 2, \dots, N\}$ , $N$ 为工件总数。

总能耗TE主要由机器加工状态总能耗 $\text{TE}^{\text{busy}}$ 、空转状态总能耗 $\text{TE}^{\text{idle}}$ 、调整状态总能耗 $\text{TE}^{\text{setup}}$ 和AGV总能耗 $\text{TE}^{\text{AGV}}$ 四部分组成,即

$$\text{TE} = \text{TE}^{\text{busy}} + \text{TE}^{\text{idle}} + \text{TE}^{\text{setup}} + \text{E}^{\text{AGV}}. \quad (2)$$

$\text{TE}^{\text{busy}}$ 为所有机器处于加工状态时的总能耗,有

$$\text{TE}^{\text{busy}} = \sum_{t=0}^{C_{\max}} \sum_{g=1}^S \sum_{k=1}^{m_g} \gamma_{gkh}^t \cdot E_{gkh}^{\text{busy}}, \forall h. \quad (3)$$

其中:当机器 $M_{gk}$ 在 $t$ 时刻以速度 $V_h$ 处于加工状态时, $\gamma_{gkh}^t$ 取值1,否则,取值0; $M_{gk}$ 为工位 $g$ 中第 $k$ 台机器; $V_h$ 为机器的加工速度,且 $V_1 < V_2 < V_H$ ;  $E_{gkh}^{\text{busy}}$ 为机器 $M_{gk}$ 以 $V_h$ 的加工速度加工工件时的单位能耗; $S$ 为工位总数; $m_g$ 为工位 $g$ 中并行机数量。

$\text{TE}^{\text{idle}}$ 为所有机器处于空转状态时的总能耗,空转能耗主要是指机器完成当前工序加工、等待下一道工序到达这段时间的能耗,有

$$\text{TE}^{\text{idle}} = \sum_{t=0}^{C_{\max}} \sum_{g=1}^S \sum_{k=1}^{m_g} \mu_{gk}^t \cdot E_{gk}^{\text{idle}}. \quad (4)$$

其中:当机器 $M_{gk}$ 在 $t$ 时刻处于空闲状态时, $\mu_{gk}^t$ 取值1,否则,取值0; $E_{gk}^{\text{idle}}$ 为机器 $M_{gk}$ 处于空闲状态时的单位能耗。

$\text{TE}^{\text{setup}}$ 为调整状态总能耗,同一机器加工两个相邻工序进行相关设置(如更换刀具、模具等等)时的能耗,有

$$\text{TE}^{\text{setup}} = \sum_{t=0}^{C_{\max}} \sum_{g=1}^S \sum_{k=1}^{m_g} \eta_{gk}^t \cdot E_{gk}^{\text{setup}}. \quad (5)$$

其中:当机器 $M_{gk}$ 在 $t$ 时刻处于调整状态时, $\eta_{gk}^t$ 取值1,否则,取值0; $E_{gk}^{\text{setup}}$ 为机器处于调整状态时的单位能耗。

$\text{TE}^{\text{AGV}}$ 为运输能耗,是指所有工件通过AGV在装载区、卸载区以及机器之间运输时的总能耗,主要包括负载能耗 $\text{TE}_{\text{load}}^{\text{AGV}}$ 和空载能耗 $\text{TE}_{\text{unload}}^{\text{AGV}}$ ,即

$$\text{TE}^{\text{AGV}} = \text{TE}_{\text{load}}^{\text{AGV}} + \text{TE}_{\text{unload}}^{\text{AGV}}; \quad (6)$$

$$\text{TE}_{\text{load}}^{\text{AGV}} = \sum_{f=1}^F \sum_{t=0}^{C_{\max}} \lambda_f^t \cdot E_f^{\text{agv\_load}}, \quad (7)$$

$$\text{TE}_{\text{unload}}^{\text{AGV}} = \sum_{f=1}^F \sum_{t=0}^{C_{\max}} \beta_f^t \cdot E_f^{\text{agv\_unload}}. \quad (8)$$

其中:当AGV $f$ 在 $t$ 时刻处于负载状态时, $\lambda_f^t$ 取值1,否则,取值0;当AGV $f$ 在 $t$ 时刻处于空载状态时, $\beta_f^t$ 取值1,否则,取值0; $E_f^{\text{agv\_unload}}$ 为AGV $f$ 空载时单位耗能; $E_f^{\text{agv\_load}}$ 为AGV $f$ 负载时单位耗能; $f$ 为AGV编号, $f \in \{1, 2, \dots, F\}$ , $F$ 为AGV总数。

## 2 求解MAHFSP-MO-SDST的改进 memetic algorithm算法

Memetic algorithm是一种基于种群的全局搜索和个体的局部启发式搜索的结合体,采用不同的搜索策略可以构成不同的memetic algorithm算法。通过建立全局搜索和局部搜索之间的平衡,MA广泛应用于各种多目标优化问题并表现出良好的性能。针对MAHFSP-MO-SDST问题的特性,本文基于NSGA-II设计MA算法,主要包括编码、解码、种群混合初始化策略、交叉以及变异算子、邻域搜索算子和快速非支配排序等,MA主要步骤如下:

step 1: 设置种群规模、交叉概率、变异概率以及迭代次数等参数。

step 2: 初始化种群POP(iter),采用最早加工原则初始化50%的种群,另外50%种群根据各种约束完全随机生成,iter = 1。

step 3: 解码种群,计算目标函数值。

step 4: 交叉变异操作,基于种群POP(iter)采用交叉算子得到新种群POP1(iter),基于种群POP1(iter)采

用变异算子得到新种群POP2(iter).

step 5: 邻域搜索, 采用基于 makespan 和 TE 的两种邻域搜索策略进行局部搜索生成新种群POP3(iter).

step 6: 合并种群POP(iter)和POP3(iter)为新种群POP4(iter).

step 7: 基于种群POP4(iter)执行快速非支配排序和拥挤距离计算, 并从中选择 popsize 较优个体得到新种群POP(iter + 1), iter = iter + 1.

step 8: 终止条件, 如果达到最大迭代次数, 则输出POP, 否则, 转到step 3.

## 2.1 编码方法

为了最大程度确保算法在整个解空间中搜索, 本文采用3层编码方式编码, 即个体由3部分组成: 工序排序(OS)、机器选择(MS)和加工速度选择(VS), 每部分的长度为所有工件的工序总数. OS部分采用工件序号按照工位顺序排序, 即先安排第1工位的工序, 然后安排第2工位的工序, 以此类推, 工件第几次出现表明是该工件的第几道工序. MS部分由各对应工序所选择的机器序号组成, 每个工序可由对应工位上任意一台机器加工完成. VS部分表示MS中机器对应的加工速度, 数字越大表明加工速度越快, 加工时间越短, 单位能耗越高. 另外, 所有工件的最后一道虚拟工序不参与编码.

## 2.2 解码方法

上述编码方法仅涉及机器和AGV联合调度问题中的工序排序、机器选择以及速度选择3个子问题, 并未涉及AGV, 因此在解码设计时需要考虑AGV的调度问题. 本文在解码过程中采用最先服务原则安排AGV, 如果有多个AGV符合条件, 则随机选择一个. 具体步骤如下:

step 1: 初始化AGV、机器、工件等相关状态信息, 令 $\alpha = 1$ .

step 2: 在OS中从左至右取第 $\alpha$ 个元素, 获得其工件号 $i$ 和工序号 $q$ , 并且在MS中取出对应的机器序号 $k$ , VS中取出机器 $k$ 对应的加工速度 $h$ , 同时计算出实际加工时间 $p$ .

step 3: 找出当前工序 $O_{iq}$ 前一道工序所在机器 $k'$ . 若 $q = 1$ , 此时工件 $i$ 仍在装区位置, 则机器 $k'$ 为 $M_0$ . 根据AGV的当前位置和最后一次运输任务的结束时间, 从所有AGV中选择最早到达机器 $k'$ 的AGV $f$ . 如果AGV $f$ 已经在机器 $k'$ 处, 则转至step 5, 否则转至step 4.

step 4: 将AGV $f$ 空载移动到机器 $k'$ .

step 5: 使用AGV $f$ 将工件 $i$ 从机器 $k'$ 运输到机器 $k$ . 如果 $O_{iq}$ 为机器 $k$ 的第1道工序, 工序 $O_{iq}$ 的调整时间为0, 则转至step 7, 否则转至step 6.

step 6: 找出机器 $k$ 前一个加工的工件 $i'$ , 并计算出工序 $O_{iq}$ 的顺序相关调整时间.

step 7: 将工序 $O_{iq}$ 的调整时间加上工件 $i$ 前一道工序的完工时间与AGV $f$ 到达机器 $k$ 时间的较大值作为工序 $O_{iq}$ 的开始加工时间.

step 8: 更新工序 $O_{iq}$ 的开始加工时间 $start_{iq}$ , 并计算完工时间.

step 9:  $a = a + 1$ , 若 $a$ 小于等于工序总数则转至step 2, 否则终止.

step 10: 根据工件在工位 $S$ 的完工时间以及AGV的位置信息, 选择最早可以执行运输任务的AGV, 将所有工件运输至卸载区.

step 11: 计算出两个目标函数值 $C_{\max}$ 和TE.

## 2.3 种群初始化

初始解质量的好坏会直接影响算法的性能. 采用完全随机方式可以使初始种群具有良好的多样性, 但是很难保证解的质量. 为了提高算法初始解的质量并保持良好的多样性, 本文采用完全随机和最早加工原则(principle of earliest processing, PEP)两种混合初始化策略. 本文种群中50%的个体采用随机初始化方法生成; 50%的个体采用PEP策略生成. PEP初始化方法如算法1所示.

### 算法1 PEP\_INITIALIZATION().

输入: 工件个数 $n$ ;

输出: 初始解 $X$ .

1) 初始化AGV和所有机器的状态信息, 随机生成工件排序 $\pi$ , 长度为 $n$ , 每个工件出现一次, 初始化工位 $g = 1$ . //遍历工位,  $S$ 为工位总数

2) for  $g = 1 : S$  do

3) if  $g == 1$  then

4) 根据 $\pi$ 顺序加工工件, 随机选择机器及加工速度, 并选择最早可用的AGV参与运输, 同时记录AGV和机器的状态信息.

5) else

6) 依据 $g - 1$ 工位上工件的完工时间升序排列得到 $\pi'$ , 根据 $\pi'$ 依次加工工件, 并且选择最早到达(表示为 $T_1$ )工件 $i$ 在工位 $g - 1$ 上加工时所在机器 $k$ 的AGV $f$ 执行运输任务. //遍历 $\pi'$ 中所有工件

7) for  $i = 1 : n$  do

8) 若工件 $i$ 上一道工序完工时间 $T_2 \leftarrow e_{i(g-1)}$ , 则工件 $i$ 的开始运输时间 $T_3 \leftarrow \max(T_1, T_2)$ . //遍历

工位  $g$  中的所有机器

9) for  $k' = 1 : m_g$  do

10) 计算 AGV  $f$  到达机器  $k'$  的运输时间  $T_4$ , 机器  $k'$  的最早可用时间  $T_5$ , 机器  $k'$  上顺序相关调整时间  $T_6$ .

11) end

12) end

13) 选择使当前工序的开始加工时间最早的机器,  $s_{ig} \leftarrow \min(\max(T_3 + T_4, T_5 + T_6))$ , 同时为该工序随机选择加工速度, 并更新 AGV 和机器的状态信息.

14) end

15) end

16) 根据调度方案构造初始解  $X$ .

### 2.4 交叉和变异算子

交叉操作是进化过程中不可缺少的环节, 它通过两条不同的亲本染色体交换基因来产生新个体. 根据 3 层编码的特点, 本文采用两种不同的交叉算子. 为了保证解的可行性, 在 OS 部分采用 JBX(job-based crossover) 交叉策略. 首先, 将工件随机分为两个集合 set 1 和 set 2; 然后将父代 1 中属于 set 1 的基因复制到相同位置的子代 1 中, 将父代 2 中属于 set 2 的基因复制到相同位置的子代 2 中; 最后, 将父代 2 中属于 set 2 的基因依照原顺序依次添加到子代 1 中, 并将父代 1 中属于 set 1 的基因依照原顺序依次添加到子代 2 中, 具体过程如图 1 所示.

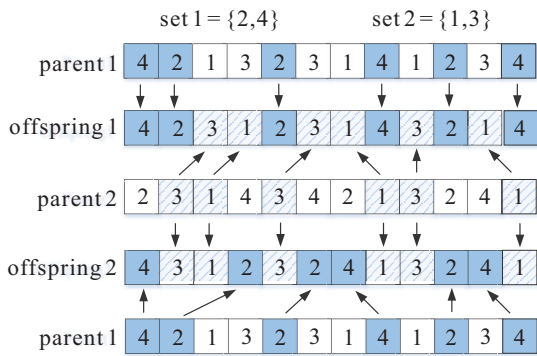


图 1 OS 部分交叉算子

在 MS 和 VS 部分采用的交叉方式与 JBX 原理类似, 随机生成  $n$  个整数组成 set 3, 将其对应位置的元素进行交叉. 例如, 令  $n = 5$ , set 3 = {2, 4, 7, 9, 12}, 以 VS 部分为例, 其交叉过程如图 2 所示.

变异算子可以提高种群的多样性, 本文设计了两种不同的变异算子. 在 OS 部分采用基于工位的随机交换策略. 首先随机选择一个工位  $g$ , 然后随机生成两个不大于工件总数的整数  $r_1$  和  $r_2$ , 最后交换  $r_1$  和  $r_2$  位置的工件. 在 MS 和 VS 部分采用随机变异策略,

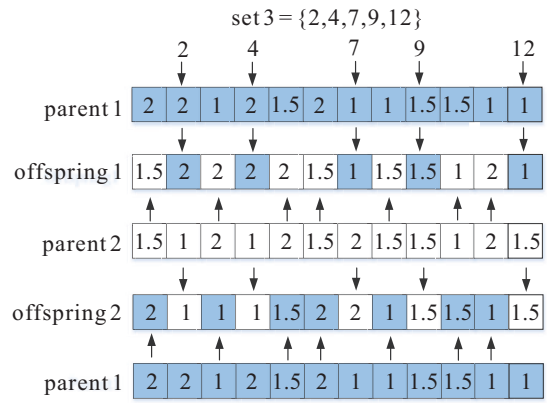


图 2 VS 部分交叉算子

即随机生成两个不大于工序总数的整数  $r_3$  和  $r_4$ , 将  $r_3$  位置的机器随机替换为同工位的其他机器, 同时, 随机改变  $r_4$  位置的加工速度.

### 2.5 邻域搜索

邻域搜索是 MA 算法的重要组成部分, 它可以避免算法陷入局部最优和过早收敛. 基于求解问题的特点, 本文设计了基于 makespan 和 TE 的两种邻域搜索策略.

#### 2.5.1 基于 makespan 的邻域搜索

对于 MAHFSP-MO-SDST, 负载最大的机器 (记作  $M^{\maxload}$ ) 是影响 makespan 的关键因素. 为了优化 makespan, 有必要重新调整  $M^{\maxload}$  与同工位其他机器之间或  $M^{\maxload}$  机器内部的工件分配和排列. 本文基于 makespan 共设计 5 种邻域搜索算子. 在算法运行过程中随机选择一个邻域搜索算子, 如果更新后的个体支配原个体, 则用新个体替换原个体.

NS1 是将负载最大机器上的工件移动到同工位的其他机器上, 具体步骤如下.

step 1: 从种群中随机选择个体  $X$ , 根据解码策略生成的调度方案找出负载最大的机器  $m_{gk}$ .

step 2: 随机选择机器  $m_{gk}$  上的工件  $i$ .

step 3: 尝试将工件  $i$  插入到同工位机器  $m_{g(k+1)}$  上所有可能的插入点, 得到新个体  $X'$ . 如果该工位仅有一台机器  $m_{gk}$ , 则机器  $m_{gk}$  以最快的速度加工工件  $i$ .

step 4: 根据支配关系找出最好的新个体  $X'$ .

NS2 是改变负载最大的机器上工件的加工顺序, 具体步骤如下.

step 1: 从种群中随机选择个体  $X$ , 根据解码策略生成的调度方案找出负载最大的机器  $m_{gk}$ .

step 2: 随机选择机器  $m_{gk}$  上的工件  $i$ .

step 3: 尝试将工件  $i$  插入到  $m_{gk}$  上剩余工件中所有可能的插入点, 得到新个体  $X'$ .

step 4: 根据支配关系找出最好的新个体  $X'$ .

NS3是交换负载最大机器上工件的加工顺序,具体步骤如下.

step 1: 从种群中随机选择个体  $X$ , 根据解码策略生成的调度方案找出负载最大的机器  $m_{gk}$ .

step 2: 随机选择机器  $m_{gk}$  上的两个工件  $i$  和  $j$ , 将工件  $i$  和工件  $j$  交换位置, 得到新的个体  $X'$ .

NS4是将负载最大的机器上的工件与同工位其他机器上的工件进行交换, 具体步骤如下.

step 1: 从种群中随机选择个体  $X$ , 根据解码策略生成的调度方案找出负载最大的机器  $m_{gk}$ .

step 2: 随机选择机器  $m_{gk}$  上的工件  $i$ .

step 3: 尝试将工件  $i$  与同工位机器  $m_{g(k+1)}$  上所有的工件进行交换得到新个体  $X'$ . 如果该工位仅有一台机器  $m_{gk}$ , 则机器  $m_{gk}$  以最快的速度加工工件  $i$ .

step 4: 根据支配关系找出最好的新个体  $X'$ .

NS5是提高负载最大的机器上工件的加工速度以缩短加工时间, 具体步骤如下.

step 1: 从种群中随机选择个体  $X$ , 根据解码策略生成的调度方案找出负载最大的机器  $m_{gk}$ .

step 2: 随机选择机器  $m_{gk}$  上的以较低速度加工的工件  $i$ , 提高其加工速度得到新的个体  $X'$ .

### 2.5.2 基于TE的邻域搜索

一般来说, 机器的加工速度越快, 工序加工时间越短. 然而, 无论工序处理的速度有多快, 都不可避免会出现机器和工件等待的现象, 因此盲目地提高加工速度很可能会造成不必要的能耗浪费. 在不恶化 makespan 的前提下, 通过合理降低机器加工速度, 适当地延长加工时间, 可以减少额外的能耗浪费. 基于此, 本文设计了基于TE的邻域搜索, 在不需要改变机器分配和工件排列的基础上可以有效减少TE, 具体过程如算法2所示.

**算法2** Neighborhoodsearch\_TE().

输入: 个体  $X$ ;

输出: 更新后的个体  $X$ . //遍历工位,  $S$  为工位总数

1) for  $k = 1 : S$  do //遍历工位  $g$  中的所有机器

2) for  $k = 1 : m_g$  do

3) for  $j = 1 : n$  do //如果工件  $j$  先于工件  $i$  在机器  $M_{gk}$  上加工时,  $y_{jigk}$  取值1; 否则, 取值0.

4) if  $y_{jigk} == 1$  and  $V_h > V_1$  then

5) 降低加工速度后工序加工时间的增加值为  $TT' \leftarrow p_{jgk}/V_{h'} - p_{jgk}/V_h, h' < h$ . 在AGV、机器和顺序相关调整时间等约束条件下, 计

算该工序允许延长的加工时间:  $TT' \leftarrow \min(s_{ig} - e_{jg} - u_{jigk}, T_{jgg'} - e_{jg})$ . 其中  $g'$  为工件  $j$  的下一道工序所在工位或者虚拟工位, 因跳跃工序的存在, 则  $g' \geq g + 1$ . 若工件  $j$  为机器  $M_{gk}$  上加工的最后一个工件, 则  $TT' \leftarrow T_{jgg'} - e_{jg}$ .  $s_{ig}$  和  $e_{jg}$  分别为工件  $i, j$  在工位  $g$  上加工的开始和结束时间,  $u_{jigk}$  为在机器  $M_{gk}$  上先后加工工件  $j$  和  $i$  时, 工件  $i$  的顺序相关调整时间.  $T_{jgg'}$  为AGV将工件  $j$  从工位  $g$  运输到  $g'$  的开始时间.

- 6) if  $TT'' \geq TT'$  then
- 7)  $V_h \leftarrow V'_h$ , 同时更新个体  $X$ .
- 8) end
- 9) end
- 10) end
- 11) end
- 12) end

### 3 数值实验

本文采用 Matlab R2017a 实现算法编程, 所有实验均在操作系统为 Windows 10, 处理器 Intel Core i7-4770 M CPU @3.40 GHz, 内存 16 G 的计算机上进行. 另外, 关于 MAHFSP-MO-SDST 的研究较少, 没有现成的文献作对比, 本文选择 HGWOA<sup>[15]</sup>、IEDA<sup>[24]</sup> 以及 NSGA-II<sup>[25]</sup> 3 种算法作对比研究. 这 3 种算法在求解混合流水车间调度问题时都有广泛的应用. 为了公平起见, 3 种算法均采用与 MA 相同的编码以及解码方法. 为了方便起见, 本文假设所有机器和 AGV 在相同状态下单位能耗相同,  $E_{gkh}^{busy} = 3V_h^2, E_{gk}^{idle} = 1, E_f^{agv\_unload} = 1.1, E_f^{agv\_load} = 1.5$  和  $E_{gk}^{setup} = 1.2$ .

#### 3.1 算例构造及参数设置

本文测试算例采用  $j*s*v*$  命名,  $*$  为一个正整数, 例如  $j10s5v3$  表示规模为 10 个工件 5 个工位 3 个 AGV 的算例. 根据工件数量、工位数量以及 AGV 数量不同规模的组合, 本文共生成 45 个算例, 所有变量的取值范围如表 1 所示.

表 1 算例变量取值范围

序号	取值范围
工件数量	10,15,20,25,30
工位数量	3,4,5
AGV 数量	3,4,5
每工位机器数量	2,3
工序标准加工时间	[0, 10] 之间整数
机器间 AGV 运输时间	[1, 6] 之间整数
顺序相关调整时间	[1, 6] 之间整数
跳跃工序占比	10% ~ 15%
加工速度	1, 1.5, 2

MA 中的关键参数主要为种群规模 Popsiz, 算

法迭代次数  $Maxiter$ , 交叉概率  $P_c$  和变异概率  $P_m$ . 通过 Taguchi 方法进行正交实验, 当  $Popsize = 80$ ,  $Maxiter = 150$ ,  $P_c = 0.7$  和  $P_m = 0.1$  时算法性能最优, 其他算法参数设置分别参考文献 [15, 24-25].

### 3.2 性能评价指标

本文采用文献 [26] 中 3 个评价指标 GD、IGD 和  $C$ -metric 来衡量算法的多样性、收敛性和解集覆盖率. 另外, 由于算例的最优 Pareto 前沿 (OPF) 是未知的, 使用近似 Pareto-front (APF) 代替 OPF. 对于每个算例, APF 通过将所有算法得到的非支配解组合成一个集合, 然后从集合中去除支配解的方法得到.

### 3.3 种群初始化的有效性验证

为了证明采用完全随机和最早加工原则 (PEP) 两种混合初始化策略的有效性, 将 MA 与采用完全随机生成初始种群的 MA (记作 MA1) 进行了比较. 随机选择 12 个算例, 每个算例独立运行 20 次所得 GD、IGD 以及  $C$ -metric 的平均值如表 2 所示. 粗体表示每个指标的最优值. 由表 2 可知, 在 12 个测试算例中  $C(MA, MA1)$  均大于  $C(MA1, MA)$ , 因此 MA1 算法所获得的非支配解大部分都被 MA 算法所获得的非支配解支配. 在大部分算例中 MA 算法的 GD 和 IGD 指标都比 MA1 的小, 说明本文提出的种群初始化方案有助于提高算法的性能.

表 2 MA 和 MA1 算法 3 个指标比较

案例	$C$ -metric		GD		IGD	
	$C(MA, MA1)$	$C(MA1, MA)$	MA	MA1	MA	MA1
j10s3v3	<b>0.54</b>	0.23	<b>0.0148</b>	0.0724	<b>0.0102</b>	0.0144
j10s3v4	<b>0.65</b>	0.14	0.0966	<b>0.0843</b>	<b>0.0049</b>	0.0134
j10s3v5	<b>0.61</b>	0.30	<b>0.0351</b>	0.1301	<b>0.0153</b>	0.0189
j15s4v3	<b>0.47</b>	0.18	<b>0.0231</b>	0.0872	<b>0.0128</b>	0.0195
j15s4v4	<b>0.52</b>	0.22	<b>0.0441</b>	0.1005	<b>0.0127</b>	0.0178
j15s4v5	<b>0.56</b>	0.33	<b>0.0298</b>	0.1013	<b>0.0096</b>	0.0155
j20s3v3	<b>0.63</b>	0.29	<b>0.0181</b>	0.0526	0.0118	<b>0.0101</b>
j20s3v4	<b>0.55</b>	0.35	<b>0.0364</b>	0.1438	<b>0.0104</b>	0.0124
j20s3v5	<b>0.70</b>	0.24	<b>0.0302</b>	0.1235	<b>0.0068</b>	0.0103
j30s4v3	<b>0.68</b>	0.19	<b>0.0612</b>	0.0947	<b>0.0107</b>	0.0152
j30s4v4	<b>0.46</b>	0.33	<b>0.0432</b>	0.1126	<b>0.0106</b>	0.0176
j30s4v5	<b>0.57</b>	0.26	<b>0.0327</b>	0.1002	0.0145	<b>0.0134</b>

### 3.4 邻域搜索的有效性验证

为了证明两种邻域搜索算子的有效性, 将 MA 与省略邻域搜索的 MA (记作 MA2) 进行了比较. 随机选择 12 个算例, 每个算例独立运行 20 次所得 GD、IGD 以及  $C$ -metric 的平均值如表 3 所示. 粗体表示每个指标的最优值. 由表 3 可知, 在 12 个测试算例中  $C(MA, MA2)$  均大于  $C(MA2, MA)$ , 说明 MA2 算法所获得的非支配解大部分都被 MA 算法所获得的非支配解支配. 在所有算例中, MA 算法的 GD 和 IGD 指标

都比 MA2 的小. 因此, 本文设计的两种邻域搜索策略充分利用问题的性质对个体进行针对性地改进, 效果非常明显.

表 3 MA 和 MA2 算法 3 个指标比较

案例	$C$ -metric		GD		IGD	
	$C(MA, MA2)$	$C(MA2, MA)$	MA	MA2	MA	MA2
j10s3v3	<b>0.84</b>	0.16	<b>0.0227</b>	0.0413	<b>0.0044</b>	0.0097
j10s3v4	<b>0.70</b>	0.24	<b>0.0081</b>	0.0268	<b>0.0053</b>	0.0235
j10s3v5	<b>1</b>	0	<b>0.0074</b>	0.0122	<b>0.0012</b>	0.0282
j15s4v3	<b>0.55</b>	0.25	<b>0.0063</b>	0.0306	<b>0.0085</b>	0.0361
j15s4v4	<b>0.77</b>	0.21	<b>0.0051</b>	0.0235	<b>0.0052</b>	0.0369
j15s4v5	<b>0.68</b>	0.24	<b>0.0115</b>	0.0296	<b>0.0108</b>	0.0346
j20s3v3	<b>0.69</b>	0.16	<b>0.0036</b>	0.0327	<b>0.0069</b>	0.0232
j20s3v4	<b>0.75</b>	0.25	<b>0.0137</b>	0.0332	<b>0.0084</b>	0.0285
j20s3v5	<b>0.49</b>	0.15	<b>0.0117</b>	0.0201	<b>0.0026</b>	0.0106
j30s4v3	<b>0.88</b>	0.12	<b>0.0086</b>	0.0425	<b>0.0046</b>	0.0109
j30s4v4	<b>1</b>	0	<b>0.0073</b>	0.0274	<b>0.0017</b>	0.0358
j30s4v5	<b>0.85</b>	0.15	<b>0.0026</b>	0.0263	<b>0.0086</b>	0.0132

### 3.5 与其他算法比较

针对工件、工位以及 AGV 数量不同的 45 个算例, 每种算法独立运行 20 次, 每运行一次得到一组 [ $C$ -metric, GD, IGD] 值. 表 4 统计了 4 种算法 20 次运行后 3 个指标的平均值, 粗体字表示每个指标的最优值. 由表 4 可以看出, 除极个别情况下, MA 算法的 3 个评价指标明显优于其他 3 种算法.

另外, 表 4 只能从宏观角度说明算法的优劣, 为了验证算法性能是否确实存在统计意义上的差异, 需要进行 Wilcoxon 符号秩检验 (置信水平为 0.05), 实验结果如表 5 所示.  $p$  值均远小于 0.05, 表明 MA 算法与其他 3 种具有显著差异, 并且求解效果更优.

图 3 为 j10s4v5 算例 4 种算法的独立运行一次的 Pareto 前沿. 由图 3 可知, MA 算法所获得的非支配解要优于其他算法. 图 3 中 B 处 Pareto 解所对应的目标函数值分别为  $C_{max} = 70$  和  $TE = 1035.87$ , 对应调度方案的甘特图如图 4 所示. 图 4 下半部分表示机器调度方案, 2(1) 表示工件 2 的第 1 道非跳跃工序, 另外, 在机器 1 上调整时间块 2-4 表示在机器 1 上工件 2 和工件 4 之间的顺序相关调整时间. 图 4 上半部分为 AGV 调度方案, 箭头两端数字表示起始机器, 箭头上方的数据表示被运输的工件. 以 AGV 2 为例, 其运输路线为:

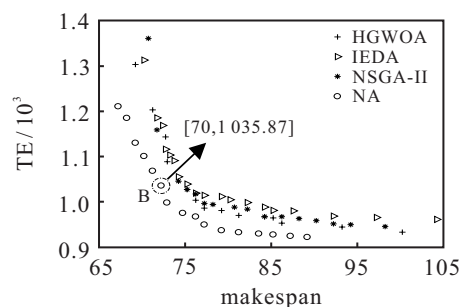


图 3 j10s4v5 算例 4 种算法 Pareto 前沿

表4 4种算法GD、IGD以及C-metric指标对比(A: MA, B: NSGA-II, C: IEDA, D: HGWOA)

算例	GD/IGD				C-metric					
	MA	NSGA-II	IEDA	HGWOA	C(A,B)	C(B,A)	C(A,C)	C(C,A)	C(A,D)	C(D,A)
j10s3v3	<b>0.021 2/0.012 7</b>	0.121 4/0.034 9	0.105 5/0.063 4	0.056 4/0.028 8	<b>0.77</b>	0.18	<b>0.88</b>	0.12	<b>0.69</b>	0.29
j10s3v4	<b>0.020 3/0.021 5</b>	0.064 3/0.052 4	0.064 5/0.068 9	0.092 6/0.032 5	<b>0.52</b>	0.35	<b>0.63</b>	0.16	<b>0.65</b>	0.24
j10s3v5	<b>0.013 6/0.008 6</b>	0.106 5/0.042 2	0.125 8/0.072 4	0.043 7/0.022 1	<b>0.61</b>	0.23	<b>0.81</b>	0.04	<b>0.55</b>	0.34
j10s4v3	<b>0.026 2/0.013 9</b>	0.086 2/0.051 8	0.103 4/0.067 5	0.064 5/0.045 3	<b>0.76</b>	0.21	<b>0.76</b>	0.13	<b>0.46</b>	0.29
j10s4v4	<b>0.008 3/0.014 1</b>	0.087 1/0.048 2	0.049 4/0.061 7	0.020 1/0.039 3	<b>0.79</b>	0.13	<b>0.88</b>	0.12	<b>0.66</b>	0.22
j10s4v5	<b>0.010 2/0.011 7</b>	0.038 2/0.034 4	0.040 8/0.048 7	0.038 6/0.029 7	<b>0.56</b>	0.27	<b>0.73</b>	0.16	<b>0.59</b>	0.25
j10s5v3	<b>0.016 4/0.014 2</b>	0.093 9/0.042 9	0.108 6/0.052 8	0.082 2/0.012 1	<b>0.61</b>	0.39	<b>0.86</b>	0.14	<b>0.58</b>	0.21
j10s5v4	<b>0.015 2/0.006 8</b>	0.066 9/0.028 4	0.080 4/0.061 7	0.062 5/0.021 1	<b>0.68</b>	0.19	<b>0.76</b>	0.14	<b>0.65</b>	0.35
j10s5v5	<b>0.012 6/0.009 4</b>	0.066 4/0.023 4	0.081 1/0.049 5	0.044 1/0.038 1	<b>0.76</b>	0.21	<b>0.64</b>	0.26	<b>0.72</b>	0.17
j15s3v3	<b>0.002 4/0.011 5</b>	0.045 9/0.050 1	0.040 5/0.101 1	0.027 1/0.036 5	<b>0.55</b>	0.45	<b>0.73</b>	0.27	<b>0.52</b>	0.36
j15s3v4	<b>0.024 3/0.005 1</b>	0.106 8/0.020 1	0.105 5/0.039 2	0.030 3/0.029 1	<b>0.72</b>	0.18	<b>0.77</b>	0.16	<b>0.53</b>	0.37
j15s3v5	<b>0.004 9/0.015 7</b>	0.091 3/0.050 9	0.107 6/0.052 5	0.056 6/0.029 7	<b>0.55</b>	0.28	<b>0.83</b>	0.12	<b>0.65</b>	0.29
j15s4v3	<b>0.009 6/0.016 6</b>	0.092 4/0.027 7	0.082 8/0.065 6	0.017 5/0.056 4	<b>0.81</b>	0.19	<b>0.88</b>	0.06	<b>0.48</b>	0.31
j15s4v4	0.024 9/0.012 9	0.105 5/0.042 3	0.132 9/0.075 8	<b>0.017 3/0.037 6</b>	<b>0.73</b>	0.27	<b>0.72</b>	0.18	<b>0.58</b>	0.17
j15s4v5	<b>0.018 7/0.015 9</b>	0.107 1/0.058 9	0.123 4/0.065 8	0.067 6/0.034 6	<b>0.75</b>	0.21	<b>1</b>	0	<b>0.63</b>	0.26
j15s5v3	<b>0.024 1/0.020 3</b>	0.073 4/0.050 5	0.105 8/0.081 2	0.034 6/0.027 5	<b>0.72</b>	0.15	<b>0.76</b>	0.13	<b>0.66</b>	0.34
j15s5v4	<b>0.018 8/0.017 9</b>	0.102 4/0.070 1	0.109 9/0.086 5	0.035 9/0.046 3	<b>0.57</b>	0.19	<b>0.85</b>	0.15	<b>0.51</b>	0.36
j15s5v5	<b>0.011 2/0.015 8</b>	0.049 2/0.064 9	0.053 5/0.071 8	0.017 6/0.039 5	<b>1</b>	0	<b>0.73</b>	0.23	<b>0.70</b>	0.30
j20s3v3	<b>0.020 1/0.025 2</b>	0.104 8/0.033 1	0.110 8/0.053 5	0.030 7/0.038 4	<b>0.65</b>	0.27	<b>0.80</b>	0.20	<b>0.53</b>	0.24
j20s3v4	<b>0.019 6/0.005 6</b>	0.052 5/0.014 3	0.051 8/0.039 6	0.055 4/0.011 4	<b>0.84</b>	0.04	<b>0.75</b>	0.14	<b>0.67</b>	0.22
j20s3v5	<b>0.011 6/0.011 7</b>	0.065 6/0.050 6	0.070 9/0.065 2	0.051 8/0.015 6	<b>0.64</b>	0.36	<b>0.69</b>	0.21	<b>0.65</b>	0.35
j20s4v3	<b>0.021 2/0.003 7</b>	0.119 5/0.051 1	0.101 4/0.062 8	0.044 5/0.033 7	<b>0.74</b>	0.18	<b>0.76</b>	0.24	<b>0.82</b>	0.18
j20s4v4	<b>0.018 4/0.003 2</b>	0.092 2/0.073 1	0.074 9/0.025 4	0.061 8/0.012 3	<b>0.69</b>	0.16	<b>0.75</b>	0.12	<b>0.69</b>	0.25
j20s4v5	0.019 8/0.006 1	0.047 4/0.017 3	0.109 4/0.029 9	<b>0.012 6/0.011 6</b>	<b>0.82</b>	0.15	<b>1</b>	0	<b>0.58</b>	0.37
j20s5v3	<b>0.014 4/0.010 2</b>	0.086 7/0.026 8	0.101 8/0.034 8	0.053 9/0.016 4	<b>0.68</b>	0.26	<b>0.84</b>	0.16	<b>0.44</b>	0.32
j20s5v4	<b>0.016 7/0.009 9</b>	0.065 4/0.034 2	0.085 2/0.066 3	0.023 5/0.026 2	<b>0.85</b>	0.15	<b>0.69</b>	0.21	<b>0.58</b>	0.18
j20s5v5	<b>0.021 3/0.008 8</b>	0.072 7/0.037 5	0.057 2/0.054 4	0.040 2/0.023 2	<b>0.69</b>	0.15	<b>0.76</b>	0.24	<b>0.61</b>	0.23
J25s3v3	<b>0.008 1/0.001 2</b>	0.053 2/0.046 4	0.065 2/0.052 6	0.025 5/0.023 8	<b>0.76</b>	0.17	<b>0.86</b>	0.14	<b>0.57</b>	0.19
j25s3v4	<b>0.013 7/0.016 5</b>	0.102 8/0.030 8	0.118 2/0.038 3	0.032 1/0.024 7	<b>0.66</b>	0.15	<b>0.80</b>	0.16	<b>0.60</b>	0.22
j25s3v5	<b>0.012 5/0.010 9</b>	0.157 2/0.053 1	0.102 2/0.081 8	0.058 6/0.012 7	<b>0.62</b>	0.21	<b>0.72</b>	0.18	<b>0.59</b>	0.24
j25s4v3	<b>0.020 7/0.027 3</b>	0.107 4/0.021 9	0.125 1/0.064 5	0.066 8/0.035 8	<b>0.79</b>	0.18	<b>0.75</b>	0.14	<b>0.49</b>	0.36
j25s4v4	<b>0.017 3/0.015 6</b>	0.066 2/0.068 5	0.088 1/0.070 5	0.066 4/0.029 8	<b>0.61</b>	0.35	<b>0.64</b>	0.13	<b>0.58</b>	0.28
j25s4v5	<b>0.011 4/0.007 4</b>	0.064 6/0.032 1	0.108 9/0.080 7	0.079 6/0.018 7	<b>0.68</b>	0.24	<b>0.72</b>	0.22	<b>0.54</b>	0.32
j25s5v3	<b>0.010 4/0.008 8</b>	0.080 8/0.035 2	0.097 3/0.042 3	0.026 4/0.028 8	<b>0.84</b>	0.16	<b>0.84</b>	0.07	<b>0.63</b>	0.33
j25s5v4	<b>0.017 2/0.012 6</b>	0.080 1/0.058 8	0.105 5/0.063 4	0.052 6/0.042 5	<b>0.54</b>	0.35	<b>0.79</b>	0.14	<b>0.57</b>	0.29
j25s5v5	<b>0.015 1/0.027 5</b>	0.052 2/0.027 6	0.064 5/0.028 9	0.043 7/0.022 1	<b>0.75</b>	0.25	<b>1</b>	0	<b>0.65</b>	0.35
j30s3v3	<b>0.014 5/0.006 3</b>	0.090 7/0.039 8	0.125 8/0.072 4	0.016 6/0.025 3	<b>0.58</b>	0.42	<b>0.91</b>	0.03	<b>0.49</b>	0.34
j30s3v4	<b>0.012 4/0.023 6</b>	0.101 4/0.034 9	0.103 4/0.047 5	0.070 1/0.029 3	<b>0.66</b>	0.19	<b>0.64</b>	0.16	<b>0.55</b>	0.30
j30s3v5	0.025 7/0.014 4	0.064 3/0.042 4	0.049 4/0.051 7	<b>0.018 6/0.029 7</b>	<b>0.78</b>	0.22	<b>0.80</b>	0.15	<b>0.64</b>	0.36
j30s4v3	<b>0.001 8/0.016 1</b>	0.076 5/0.042 2	0.080 8/0.048 7	0.022 2/0.012 1	<b>0.79</b>	0.14	<b>0.83</b>	0.17	<b>0.72</b>	0.28
j30s4v4	0.015 3/0.010 8	0.066 2/0.021 8	0.038 6/0.042 8	<b>0.012 5/0.021 1</b>	<b>0.71</b>	0.21	<b>1</b>	0	<b>0.75</b>	0.16
j30s4v5	<b>0.002 1/0.009 5</b>	0.047 1/0.048 2	0.050 4/0.041 7	0.004 1/0.038 1	<b>0.64</b>	0.26	<b>0.78</b>	0.18	<b>0.64</b>	0.28
j30s5v3	<b>0.001 7/0.012 4</b>	0.061 8/0.034 4	0.071 1/0.059 5	0.047 1/0.036 5	<b>0.78</b>	0.18	<b>0.66</b>	0.34	<b>0.51</b>	0.29
j30s5v4	<b>0.016 1/0.008 3</b>	0.073 9/0.012 9	0.080 5/0.031 1	0.050 3/0.029 1	<b>0.57</b>	0.36	<b>0.84</b>	0.16	<b>0.66</b>	0.22
j30s5v5	<b>0.020 4/0.002 7</b>	0.066 9/0.028 4	0.095 5/0.039 2	0.056 6/0.020 7	<b>0.61</b>	0.25	<b>0.73</b>	0.22	<b>0.61</b>	0.35

表5 MA vs.其他3种算法评价指标Wilcoxon符号秩检验

算法	评价指标 (p-value)		
	GD	IGD	C-metric
MA vs.NSGA-II	3.17e-16	1.61e-14	3.08e-16
MA vs.IEDA	3.16e-16	3.62e-16	3.02e-16
MA vs.HGWOA	2.25e-10	5.79e-11	3.07e-16

$M_0(\text{Job } 2) \rightarrow M_1 \rightarrow M_0(\text{Job } 6) \rightarrow M_1(\text{Job } 3) \rightarrow M_3(\text{Job } 6) \rightarrow M_5 \rightarrow M_7(\text{Job } 5) \rightarrow M_{10}(t = 0)$ , 其中  $\rightarrow$  表示 AGV 处于空载状态,  $\Rightarrow$  表示 AGV 处于负载状态,  $M_0$

表示装载区,  $M_{10}$  表示卸载区,  $t = 0$  表示执行虚拟工序.

以j15s3v4为例, MA算法所得makespan和TE两个目标进化收敛曲线如图5和图6所示. 在每个图中, 最上面曲线表示在进化过程中各代种群makespan和TE两个目标的最大值收敛曲线, 中间和下面曲线分别表示两个目标函数的平均值和最小值收敛曲线. 由图5和图6可知, 在进化前期, 3个曲线均迅速减小, 而在进化后期均值和最小值基本保持不变或者稍

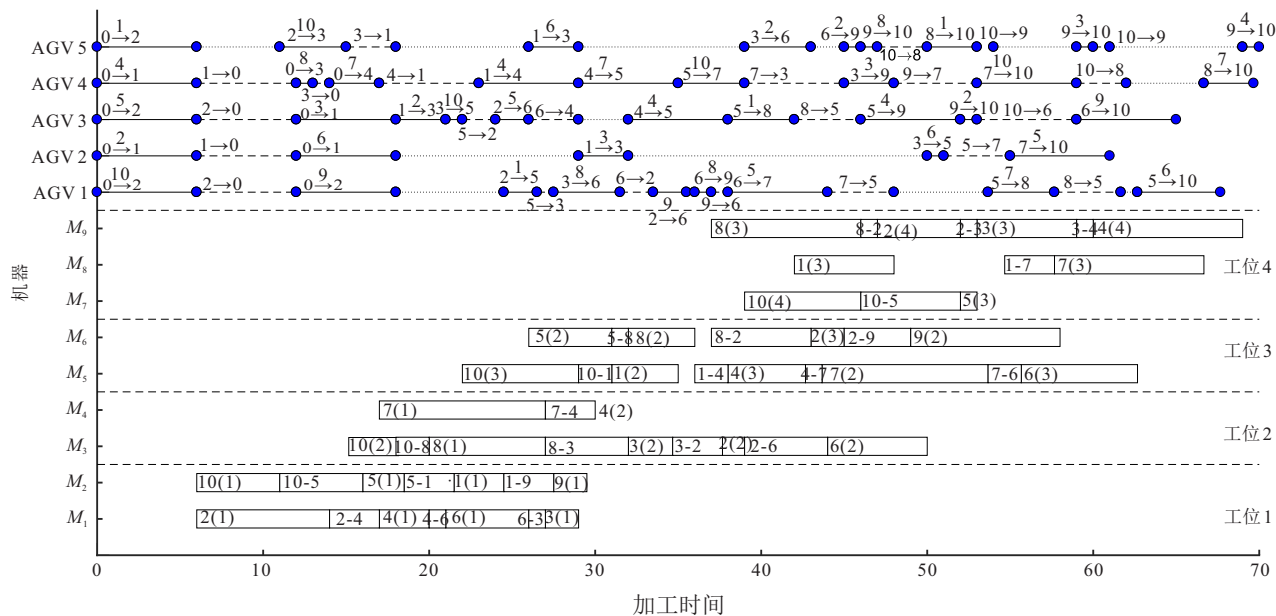


图4 j10s4v5算例甘特图

微减小且均值均接近于最小值,最大值有一定的波动,这表明算法在保持优秀解的同时具有较强的搜索能力和收敛性能. 综上,本文提出的改进MA算法能够有效求解同时考虑工序跳跃和SDST特性的HFSP机器及AGV联合调度问题,且具有较强优越性.

合调度问题,更接近实际生产;然后,基于求解问题的特性,提出了3层编码策略和混合种群初始化方法,设计了交叉变异算子以及基于两个优化目标的邻域搜索策略,极大地提高了算法的寻优能力. 最后,通过大量仿真实验验证了算法的有效性和优越性. 未来将进一步研究绿色调度问题,例如设计更好的智能算法、考虑分布式特点、考虑冲突和碰撞的AGV与机器联合调度等.

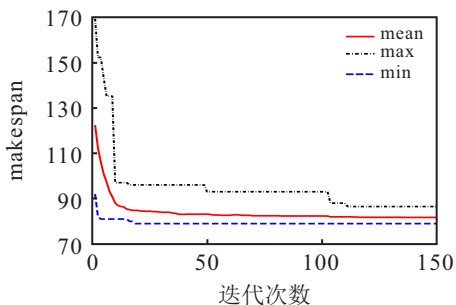


图5 j15s3v4算例makespan目标收敛曲线

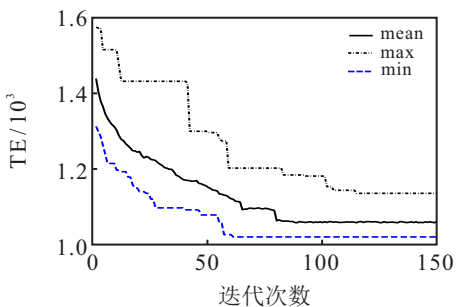


图6 j15s3v4算例TE目标收敛曲线

### 4 结论

本文研究了带工序跳跃的绿色混合流水车间机器和AGV联合调度问题,以最小化最大完工时间和总能耗为优化目标,并提出改进的MA算法求解该问题. 首先,结合混合流水车间小批量多品种的特性,充分考虑SDST和工序跳跃特性,研究了机器和AGV联

### 参考文献(References)

- [1] Ribas I, Leisten R, Framiñan J M. Review and classification of hybrid flow shop scheduling problems from a production system and a solutions procedure perspective[J]. Computers & Operations Research, 2010, 37(8): 1439-1454.
- [2] Ruiz R, Vázquez-Rodríguez J A. The hybrid flow shop scheduling problem[J]. European Journal of Operational Research, 2010, 205(1): 1-18.
- [3] Meng L L, Zhang C Y, Shao X Y, et al. MILP models for energy-aware flexible job shop scheduling problem[J]. Journal of Cleaner Production, 2019, 210: 710-723.
- [4] Meng L L, Zhang C Y, Shao X Y, et al. Mathematical modelling and optimisation of energy-conscious hybrid flow shop scheduling problem with unrelated parallel machines[J]. International Journal of Production Research, 2019, 57(4): 1119-1145.
- [5] Fang K, Uhan N, Zhao F, et al. A new approach to scheduling in manufacturing for power consumption and carbon footprint reduction[J]. Journal of Manufacturing Systems, 2011, 30(4): 234-240.
- [6] 孟磊磊, 张超勇, 邵新宇, 等. 面向节能的工艺规划与调度集成问题建模研究[J]. 机械工程学报, 2019, 55(16): 185-196. (Meng L L, Zhang C Y, Shao X Y, et al. Mathematical

- modeling of energy-efficient integration of process planning and scheduling[J]. *Journal of Mechanical Engineering*, 2019, 55(16): 185-196.)
- [7] Che A D, Zeng Y Z, Lyu K. An efficient greedy insertion heuristic for energy-conscious single machine scheduling problem under time-of-use electricity tariffs[J]. *Journal of Cleaner Production*, 2016, 129: 565-577.
- [8] 宋存利. 求解多目标混合流水车间调度的改进 NSGAI 算法[J]. *计算机集成制造系统*, <http://kns.cnki.net/kcms/detail/11.5946.tp.20201026.1044.022.html>. (Song C L. An improved NSGAI algorithm for hybrid flow shop scheduling problem with multi-objective[J]. *Computer Integrated Manufacturing Systems*, <http://kns.cnki.net/kcms/detail/11.5946.tp.20201026.1044.022.html>.)
- [9] 周炳海, 刘文龙. 考虑能耗和准时的混合流水线多目标调度[J]. *上海交通大学学报*, 2019, 53(7): 773-779. (Zhou B H, Liu W L. Multi-objective hybrid flow-shop scheduling problem considering energy consumption and on-time delivery[J]. *Journal of Shanghai Jiao Tong University*, 2019, 53(7): 773-779.)
- [10] 吴秀丽, 孙阳君. 机器多转速的柔性作业车间绿色调度问题[J]. *计算机集成制造系统*, 2018, 24(4): 862-875. (Wu X L, Sun Y J. Flexible job shop green scheduling problem with multi-speed machine[J]. *Computer Integrated Manufacturing Systems*, 2018, 24(4): 862-875.)
- [11] Tseng C T, Liao C J, Liao T X. A note on two-stage hybrid flowshop scheduling with missing operations[J]. *Computers & Industrial Engineering*, 2008, 54(3): 695-704.
- [12] Saravanan M, Sridhar S, Harikannan N. Minimization of mean tardiness in hybrid flow shop with missing operations using genetic algorithm[J]. *Journal of Advanced Manufacturing Systems*, 2016, 15(2): 43-55.
- [13] Dios M, Fernandez-Viagas V, Framinan J M. Efficient heuristics for the hybrid flow shop scheduling problem with missing operations[J]. *Computers & Industrial Engineering*, 2018, 115: 88-99.
- [14] Eynan A, Rosenblatt M J. An interleaving policy in automated storage/retrieval systems[J]. *International Journal of Production Research*, 1993, 31(1): 1-18.
- [15] 李西兴, 杨道明, 李鑫, 等. 基于混合遗传鲸鱼优化算法的柔性作业车间自动导引车融合调度方法[J]. *中国机械工程*, 2021, 32(8): 938-950. (Li X X, Yang D M, Li X, et al. Flexible job shop AGV fusion scheduling method based on HGWOA[J]. *China Mechanical Engineering*, 2021, 32(8): 938-950.)
- [16] Lyu X F, Song Y C, He C Z, et al. Approach to integrated scheduling problems considering optimal number of automated guided vehicles and conflict-free routing in flexible manufacturing systems[J]. *IEEE Access*, 2019, 7: 74909-74924.
- [17] Zheng Y, Xiao Y J, Seo Y. A tabu search algorithm for simultaneous machine/AGV scheduling problem[J]. *International Journal of Production Research*, 2014, 52(19): 5748-5763.
- [18] Wein L M, Chevalier P B. A broader view of the job-shop scheduling problem[J]. *Management Science*, 1992, 38(7): 1018-1033.
- [19] Sharma P, Jain A. A review on job shop scheduling with setup times[J]. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 2016, 230(3): 517-533.
- [20] 李明, 雷德明. 考虑准备时间和关键目标的柔性作业车间低碳调度研究[J]. *机械工程学报*, 2019, 55(21): 139-149. (Li M, Lei D M. Research on flexible job shop low carbon scheduling with setup times and key objectives[J]. *Journal of Mechanical Engineering*, 2019, 55(21): 139-149.)
- [21] Li J Q, Deng J W, Li C Y, et al. An improved Jaya algorithm for solving the flexible job shop scheduling problem with transportation and setup times[J]. *Knowledge-Based Systems*, 2020, 200: 106032.
- [22] Liou C D, Hsieh Y C. A hybrid algorithm for the multi-stage flow shop group scheduling with sequence-dependent setup and transportation times[J]. *International Journal of Production Economics*, 2015, 170: 258-267.
- [23] Song H B, Lin J. A genetic programming hyper-heuristic for the distributed assembly permutation flow-shop scheduling problem with sequence dependent setup times[J]. *Swarm and Evolutionary Computation*, 2021, 60: 100807.
- [24] 戴敏, 张玉伟, 曾励. 绿色作业车间机器与AGV的集成调度研究[J]. *南京航空航天大学学报*, 2020, 52(3): 468-477. (Dai M, Zhang Y W, Zeng L. Integrated scheduling of machines and AGVs in green job shop[J]. *Journal of Nanjing University of Aeronautics & Astronautics*, 2020, 52(3): 468-477.)
- [25] 黄辉, 李梦想, 严永. 考虑序列设置时间的混合流水车间多目标调度研究[J]. *运筹与管理*, 2020, 29(12): 215-221. (Huang H, Li M X, Yan Y. Research on multi-objective scheduling of hybrid flow production Shop Considering sequence setting time[J]. *Operations Research and Management Science*, 2020, 29(12): 215-221.)
- [26] Wang G C, Gao L, Li X Y, et al. Energy-efficient distributed permutation flow shop scheduling problem using a multi-objective whale swarm algorithm[J]. *Swarm and Evolutionary Computation*, 2020, 57: 100716.

### 作者简介

耿凯峰(1986—), 男, 讲师, 博士生, 从事生产调度、智能优化的研究, E-mail: gkf8605@126.com;

叶春明(1964—), 男, 教授, 博士生导师, 从事工业工程、生产调度等研究, E-mail: yechm6464@163.com.

(责任编辑: 孙艺红)