



中国科技期刊卓越行动计划项目入选期刊

# 控制与决策

CONTROL AND DECISION



## 基于差分进化算法的FMS中机器与多载AGV调度

王子实, 吴耀华

引用本文:

王子实, 吴耀华. 基于差分进化算法的FMS中机器与多载AGV调度[J]. 控制与决策, 2024, 39(9): 3108–3116.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2023.0597>

## 您可能感兴趣的其他文章

### Articles you may be interested in

#### 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>

#### 基于视觉引导多AGV系统的改进A\*路径规划算法

Improved A\* path planning algorithm for vision-guided multi-AGV system

控制与决策. 2021, 36(8): 1881–1890 <https://doi.org/10.13195/j.kzyjc.2019.1670>

#### 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>

#### 带峰值能耗约束流水线调度的协同群智能优化

Cooperative memetic optimization for flowshop scheduling with peak power consumption constraint

控制与决策. 2021, 36(10): 2350–2358 <https://doi.org/10.13195/j.kzyjc.2020.0429>

#### 基于正态云模型的状态转移算法求解多目标柔性作业车间调度问题

State transition algorithm based on normal cloud model for solving multi-objective flexible job shop scheduling problem

控制与决策. 2021, 36(5): 1181–1190 <https://doi.org/10.13195/j.kzyjc.2019.1233>

# 基于差分进化算法的FMS中机器与多载AGV调度

王子实, 吴耀华<sup>†</sup>

(山东大学控制科学与工程学院, 济南 250061)

**摘要:** 针对柔性制造系统中机器与混合多载自动导引搬运车 (automated guided vehicle, AGV) 集成调度问题, 以最小化最大完工时间为目标, 提出一种改进混合离散差分进化算法, 通过离散操作进行差分进化算法的变异、交叉, 并对每次迭代的最优个体进行变邻域搜索. 设计基于搬运任务、机器和 AGV 的 3 层编码结构, 并进行合理化调整; 设计基于关键路径的变邻域搜索, 有针对性地调整关键路径上任务的次序、对应工序的机器和 AGV, 提高局部搜索效率; 设计基于外部档案的个体更新策略, 将种群中较差的个体从外部档案中进行替换, 提高全局搜索效率. 最后, 通过实验验证算法改进策略的有效性以及所提出算法的有效性, 分析 AGV 数量、AGV 载位容量以及 AGV 电池可用时间对算法求解结果和改进效果的影响.

**关键词:** 柔性制造系统; 多载位 AGV; 差分进化算法; 变邻域搜索; 关键路径; 最大完工时间

中图分类号: TP2 文献标志码: A

DOI: 10.13195/j.kzyjc.2023.0597

引用格式: 王子实, 吴耀华. 基于差分进化算法的 FMS 中机器与多载 AGV 调度 [J]. 控制与决策, 2024, 39(9): 3108-3116.

## An improved differential evolution algorithm for simultaneous scheduling of machines and multi-load AGVs in an FMS

WANG Zi-shi, WU Yao-hua<sup>†</sup>

(College of Control Science and Engineering, Shandong University, Jinan 250061, China)

**Abstract:** In this paper, we address the simultaneous scheduling of machines and mixed multi-load automated guided vehicles (AGVs) in flexible manufacturing systems. To minimize the maximum completion time, an improved mixed discrete differential evolution algorithm with variable neighborhood search is proposed. The variation and crossover of the differential evolution algorithm are performed by discrete operations, and the optimal individual of each iteration is updated by variable neighborhood search. A three-layer coding structure based on tasks, machines and AGVs is proposed and rationalized. Variable neighborhood search based on critical path is proposed, and the order of tasks on the critical path, machines and AGVs of corresponding processes are adjusted in a targeted manner to improve the local search efficiency. A colony update strategy based on the external archiving is designed to replace the poorer individuals in the population from the external archiving to improve the global search efficiency. Finally, through experiments, the effectiveness of the algorithm improvement strategy and the effectiveness of the algorithm are proved, the effects of the number of AGVs, the capacity of AGV loading and the available time of AGV batteries on the algorithm solution results and the improvement are analyzed.

**Keywords:** flexible manufacturing systems; multi-load AGVs; differential evolution algorithm; variable neighborhood search; critical path; maximum completion time

## 0 引言

随着生产模式向多品种、小批量、个性化的模式转变, 柔性制造系统 (flexible manufacturing system, FMS) 将成为制造业未来发展的驱动力. 柔性作业车间调度问题广泛存在于生产实践中<sup>[1]</sup>. 自动导引搬运车 (automated guided vehicle, AGV) 凭借其自动化、

高柔性和并行作业等优势, 能较好地满足物流需求, 是实现车间智能物流的关键设备之一. 在实际生产中, 机器与 AGV 之间存在不同时空的相互依赖<sup>[2]</sup>, 因此在柔性制造系统的生产调度中, 不可忽略 AGV 在不同机器之间的搬运. 多载 AGV 能够同时搬运多个工件, 有助于缩小制造车间中 AGV 的数量规模, 提高

收稿日期: 2023-05-06; 录用日期: 2023-08-08.

基金项目: 国家自然科学基金项目 (62273204).

责任编委: 王凌.

<sup>†</sup>通讯作者. E-mail: yaohua.wu@sdu.edu.cn.

AGV的利用率<sup>[3]</sup>,在机械加工、模具加工、钣金加工等行业应用广泛.在实际生产车间中,还会存在载位不同的多载AGV混合使用,以及AGV充电的情况,因此,在柔性制造系统调度中,有必要同时考虑机器资源和混合多载AGV资源,并结合充电约束进行集成调度,构建一个高效灵活的智能化系统.本文正是研究考虑充电的FMS中机器与混合多载AGV集成调度的问题.

在FMS中机器与AGV集成调度问题的研究中,早期,Bilge等<sup>[4]</sup>针对小规模问题,通过精确算法取得全局最优解,但随着问题复杂度的增加,无法在合理时间内搜索到满意解.近年来,胡晓阳等<sup>[5]</sup>提出一种改进迭代局部搜索算法,通过“先到先服务”的启发式规则选择AGV. Satich等<sup>[6]</sup>提出差分进化算法,设计启发式规则选择机器和AGV,但缺乏对算法本身的改进.然而,上述研究对象均为单载AGV,对于多载AGV车间调度问题的研究较少.与单载AGV相比,多载AGV可同时搬运多个工件,对工件的装载和配送有更强的柔性,导致多载AGV调度问题的解空间更复杂,求解难度更大.

在多载AGV的车间调度问题的研究中,早期主要通过建立启发式规则<sup>[7-9]</sup>求解.近年来,Du等<sup>[10]</sup>研究纺织车间中多载AGV路径规划问题,以最大化机器效率为目标,提出一种混合遗传粒子群算法. Zou等<sup>[11]</sup>研究矩阵车间中多载AGV实时配送的问题,以最小化总成本为目标,提出一种改进的迭代贪婪算法. Zhang等<sup>[12]</sup>研究汽车装配车间中混合多载AGV与机器集成调度问题,以最小化物流总成本为目标,通过遗传算法进行求解.然而,目前尚缺乏FMS中机器与多载AGV集成调度问题的研究.

FMS中机器与混合多载AGV集成调度问题作为一种复杂组合优化问题,可以由许多智能算法求解,例如遗传算法<sup>[13-14]</sup>、文化基因算法<sup>[15]</sup>等.其中,差分进化算法作为一种种群进化的群智能算法,具有全局搜索能力强、操作简单等优点,广泛应用于组合优化问题.吴秀丽等<sup>[16]</sup>通过设计的染色体组改进差分进化算法,求解带批处理机的混合流水线调度问题;王开等<sup>[17]</sup>提出融合外部档案等策略的差分进化算法,求解一种非线性方程组系统;Pan等<sup>[18]</sup>提出一种混合邻域搜索的离散差分进化算法,求解无等待流水车间调度问题;伍乐等<sup>[19]</sup>提出一种混合变邻域搜索的离散差分进化算法,求解FMS中机器与AGV同时调度的问题.

对于本文最小化最大完工时间的车间调度问题,

采用基于关键路径的邻域搜索可以提高邻域搜索质量<sup>[20-21]</sup>.基于关键路径的邻域搜索已广泛应用于作业车间调度<sup>[22-23]</sup>、柔性作业车间调度<sup>[24]</sup>等问题的研究,然而,尚缺乏在关键路径中考虑AGV的搬运时间的研究.

本文针对FMS中机器与混合多载AGV集成调度问题,以最小化最大完工时间为目标,同时考虑充电约束,提出一种改进混合变邻域离散差分进化算法,通过离散操作进行差分进化算法的变异、交叉,并对每次迭代的最优个体进行变邻域搜索.设计基于搬运任务、机器和AGV的3层编码结构,并进行合理化调整;设计基于关键路径的变邻域搜索,针对性地调整关键路径上任务的次序、对应工序的机器和AGV;设计基于外部档案的个体更新策略,将种群中较差的个体从外部档案中替换.最后,通过实验验证算法改进策略的有效性以及所提出算法的有效性,分析AGV数量、AGV载位容量以及AGV电池可用时间对算法求解结果和改进效果的影响.

## 1 问题描述与建模

### 1.1 问题描述

某个柔性作业车间有 $N$ 个待加工工件位于起点处,需要在 $M$ 台机器上加工,每个工件有 $u_i$ 道不同的工序,每道工序可选择的加工机器不唯一.若干台AGV从起点开始,负责工件在机器之间的搬运,包括为每道工序进行取货和卸货. AGV的载位容量不尽相同,其中多载AGV可以同时装载多个工件.考虑AGV充电问题,当AGV当前可用电量小于电量阈值时,需要返回起点充电.要求在满足一定约束条件下,以最大完工时间最小为优化目标,为每道工序的取货和卸货任务进行排序,为每道工序选择合适的AGV与机器,确定每个搬运任务的开始时间和完成时间,以及每道工序加工的开始时间和完成时间.

模型假设:1)工件加工完成最后一道工序后离开系统;2)零时刻,所有机器与AGV都可用,机器与AGV连续工作;3)任意时刻,每台机器一次只能加工一个工件;4)相同工件的工序有先后约束,不同工件之间的工序无先后约束;5)AGV行驶速度恒定;6)AGV在任意两台机器之间的运输路线是固定的,运输时间仅与机器之间的距离有关;7)不考虑AGV可能发生冲突、故障等问题;8)AGV的装卸时间可以忽略.

### 1.2 数学模型

为建立问题模型,引入如下符号及变量: $J_i$ 表示工件 $i$ , $i \in \{1, 2, \dots, N\}$ ;  $M_k$ 表示机器 $k$ , $k \in \{1, 2,$

$\dots, M\}$ ;  $R_a$  表示 AGV $_a$ ,  $a \in \{1, 2, \dots, A\}$ ;  $O_{ij}$  表示工件  $i$  的第  $j$  道工序,  $j \in \{1, 2, \dots, u_i\}$ ;  $Q_a$  表示 AGV $_a$  的载位容量;  $o$  表示起点;  $t_{lk}$  表示从任务点  $l$  到任务点  $k$  的运输时间;  $p_{ijk}$  表示工序  $O_{ij}$  在机器  $k$  的加工时间;  $V$  表示一个足够大的正数;  $t_{ar}^s$  和  $t_{ar}^e$  分别表示 AGV 第  $r$  个搬运任务的开始时间和完成时间;  $t_{ij}^s$  和  $t_{ij}^e$  分别表示工序  $O_{ij}$  的开始加工时间和完成时间;  $c_m$  和  $c_f$  分别表示 AGV 的最大电量可用时间和电量阈值;  $c_{ar}$  表示 AGV $_a$  开始执行第  $r$  个搬运任务时的电量可用时间;  $c_{ar}^t$  表示 AGV $_a$  执行第  $r$  个搬运任务时的充电时间;  $v_c$  表示 AGV 的充电速率;  $l_{ar}$  和  $q_{ar}$  表示 AGV $_a$  完成第  $r$  个搬运任务时所在的位置和载位上的工件数量;  $l_k$  表示任务点的位置;  $s_a$  表示 AGV $_a$  的搬运任务总数.

定义如下决策变量:  $x_{ijar}$  为二进制变量, 当工序  $O_{ij}$  的取货是 AGV $_a$  的第  $r$  个搬运任务时为 1, 否则为 0;  $y_{ijar}$  为二进制变量, 当工序  $O_{ij}$  的卸货是 AGV $_a$  的第  $r$  个搬运任务时为 1, 否则为 0;  $z_{ijk}$  为二进制变量, 当工序  $O_{ij}$  由机器  $k$  加工时为 1, 否则为 0;  $\alpha_{ijug}$  为二进制变量, 当工序  $O_{ug}$  先于  $O_{ij}$  卸货时为 1, 否则为 0.

建立数学模型如下:

目标函数为

$$\min C_{\max} = \max (t_{ij}^e). \quad (1)$$

约束条件如下:

1) 限制 AGV 搬运任务的开始时间. 即

$$t_{ar}^s \geq 0, \quad (2)$$

$$t_{ar}^s \geq t_{a(r-1)}^e. \quad (3)$$

其中: 式(2)限制 AGV 从 0 时刻开始搬运, 式(3)限制 AGV 在上次搬运任务完成后开始搬运.

2) 限制 AGV 搬运任务的完成时间. 即

$$t_{ar}^e \geq t_{ar}^s + t_{l_{a(r-1)}l_{ar}}, \quad c_{ar} \geq c_f; \quad (4)$$

$$t_{ar}^e \geq t_{ar}^s + t_{l_{a(r-1)}l_o} + t_{l_o l_{ar}} + c_{ar}^t, \quad c_{ar} < c_f; \quad (5)$$

$$t_{ar}^e \geq t_{i(j-1)}^e - V(1 - x_{ijar}). \quad (6)$$

其中: 式(4)和(5)表示 AGV 搬运任务的完成时间在 AGV 行驶到目的地之后, 若 AGV 不需要充电, 则直接行驶到目的地; 否则 AGV 需要先返回起点充电. 式(6)表示 AGV 取工件任务的完成时间需要在该工件上道工序加工完成之后.

3) 限制工序的开始加工时间. 即

$$t_{ij}^s \geq t_{ar}^e - V(1 - y_{ijar}), \quad (7)$$

$$t_{ij}^s \geq t_{ug}^e - V(1 - \alpha_{ijug} z_{ijk} z_{ugk}). \quad (8)$$

其中: 式(7)限制工序的开始加工时间在该工序到达

机器之后, 式(8)限制工序的开始加工时间在机器允许加工之后.

4) 限制工序的加工完成时间为开始加工时间与加工时间之和. 即

$$t_{ij}^e \geq t_{ij}^s + p_{ijk} - V(1 - z_{ijk}). \quad (9)$$

5) 限制 AGV 载位上工件数量不超过载位容量. 即

$$q_{ar} \leq Q_a. \quad (10)$$

6) AGV 电量的限制. 即

$$c_{ar}^t = (c_m - c_{ar} - t_{l_{a(r-1)}l_o})/v_c, \quad c_{ar} < c_f; \quad (11)$$

$$c_{ar}^t = c_{a(r-1)} - t_{l_{a(r-1)}l_{ar}}, \quad c_{ar} \geq c_f. \quad (12)$$

其中: 式(11)表示 AGV 的充电时间为当前电量充到满电的时间; 式(12)表示 AGV 不需要充电时, 任务完成时电量可用时间需要减去 AGV 的行驶时间.

7) 决策变量限制. 即

$$\sum_{a=1}^A \sum_{r=1}^{s_a} x_{ijar} = 1; \quad (13)$$

$$\sum_{a=1}^A \sum_{r=1}^{s_a} y_{ijar} = 1; \quad (14)$$

$$x_{ijar} = y_{ijar'}, \quad r < r'; \quad (15)$$

$$\sum_{k=1}^M z_{ijk} = 1. \quad (16)$$

其中: 式(13)和(14)限制每个工序的取货和卸货由 AGV 通过一次搬运完成, 式(15)限制同一个工序的取货和卸货分别为同一个 AGV 的前后两个搬运任务, 式(16)限制每个工序只由一台机器加工.

## 2 改进混合变邻域搜索离散差分进化算法

本文提出一种改进混合变邻域搜索离散差分进化算法(improved discrete differential evolution-variable neighborhood search, IDDE-VNS), 求解 FMS 中机器与混合多载 AGV 集成调度问题. 混合变邻域搜索离散差分进化算法将差分进化算法的变异、交叉操作改进为离散操作, 并对算法每次迭代的最优个体进行变邻域搜索. 在此基础上, 本文设计基于任务、机器和 AGV 的 3 层编码结构, 基于关键路径的变邻域搜索算法以及基于外部档案的个体更新策略. 总体步骤如下.

step 1: 输入调度信息. 包括: 工件的加工信息, AGV 数量  $A$ , AGV 的载位容量  $Q_a$ , 系统布局. 设置算法控制参数. 包括: 种群规模  $P$ , 变异概率  $F$ , 交叉概率  $C$ , 初始温度  $T$ , 最大迭代次数  $G_{\max}$ , 最大运行时间  $\text{endtime}$ . 初始化  $g = 0$ ,  $\text{iter} = 0$ , 外部档案为空集.

step 2: 初始化种群. 根据设计的编码结构, 随机生成  $P$  个可行的调度方案作为初始种群.

step 3: 若算法达到终止条件, 即当前迭代次数  $iter$  达到  $G_{max}$ , 或当前运行时间  $nowtime$  达到  $endtime$ , 则输出搜索到的最优解  $X_{best}$ ; 否则, 进入 step 4, 记录当前最优解  $X'_{best}$ .

step 4: 执行离散差分进化算法. 根据  $F$  和  $C$ , 依次生成变异个体和试验个体, 将试验个体与种群个体的适应度值进行比较, 以一定的概率接受试验个体进入下一代, 未接受的试验个体进入外部档案集.

step 5: 执行基于关键路径的变邻域搜索. 将种群中的最优个体作为初始解, 基于关键路径生成邻域结构, 进行变邻域搜索, 将获得的结果作为当前最优个体  $X_{best}$ , 通过  $X_{best}$  更新原种群的最优解.

step 6: 执行基于外部档案的更新策略. 若  $X_{best} = X'_{best}$ , 则  $g = g + 1$ ; 否则  $g = 0$ . 若  $g = 2$ , 即最优解连续迭代2次不变, 则将部分较差个体替换为外部档案中较优的个体. 更新完成后, 将种群中的最优个体作为当前最优个体  $X_{best}$ .

step 7:  $iter = iter + 1$ , 进入 step 3.

### 2.1 编码与解码

#### 2.1.1 编码

根据数学模型, 调度方案的确定取决于  $x_{ijar}$ 、 $y_{ijar}$ 、 $z_{ijk}$  和  $\alpha_{ijug}$  四个决策变量的值, 因此个体的编码需要反映它们的信息. 本文设计3层编码结构, 第1层为搬运任务编码, 编码长度为  $2 \sum_{i=1}^N u_i$ , 每道工序的工件号占据前后两个基因位, 分别表示该道工序的取

货和卸货任务. 对于第  $j$  次出现的工件, 若  $j$  为奇数, 则对应该工件第  $(j + 1)/2$  道工序的取货任务; 否则, 对应该工件第  $j/2$  道工序的卸货任务. 第2层为机器编码, 编码长度为  $\sum_{i=1}^N u_i$ , 每个基因位表示每道工序选择的机器.

第3层为AGV编码, 编码长度为  $\sum_{i=1}^N u_i$ , 每个基因位表示每道工序选择的AGV.

如图1所示, 在任务编码中, 第1个基因位表示工件2第1道工序的取货任务; 在机器/AGV编码中, 第1个基因位表示工件1第1道工序选择机器3/AGV1, 以此类推.

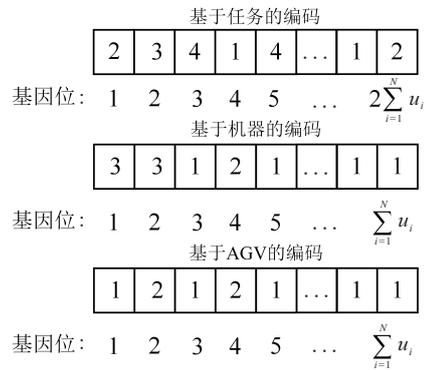


图1 基于任务、机器和AGV的编码结构

在对种群个体进行随机生成、交叉、变换以及邻域搜索时, 可能产生非法个体, 具体体现在: 对于第1层染色体, 某台AGV载位上的工件数达到其载位容量时仍执行取货; 对于第2层染色体, 对某道工序未选择该工序的可行机器. 因此, 需要针对非法个体进行基因调整设计, 具体过程如图2所示.

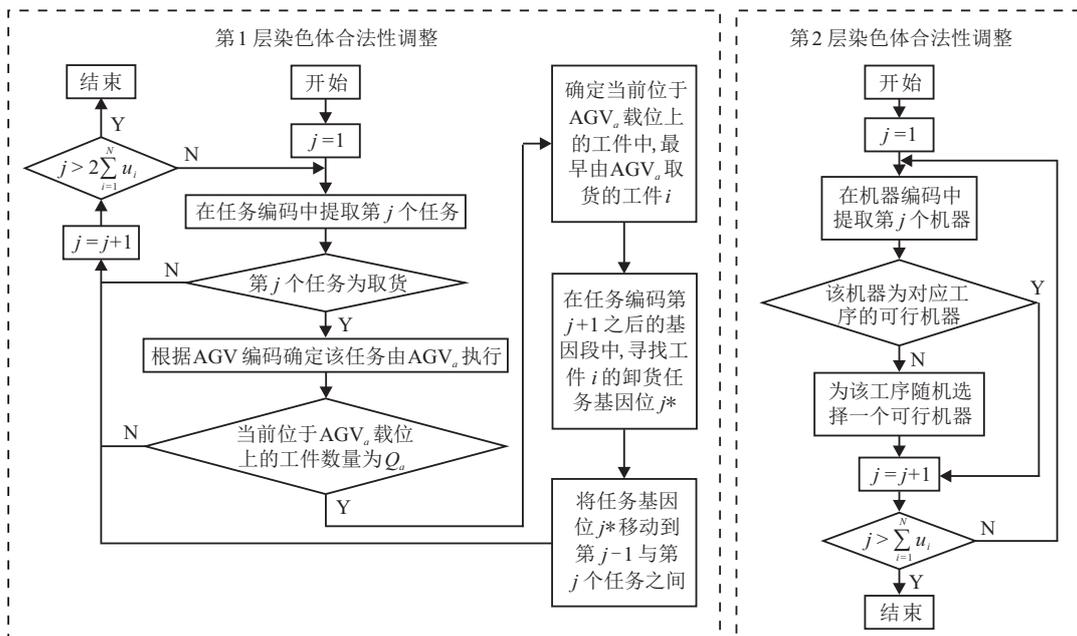


图2 基因合法性调整流程

### 2.1.2 解码

个体解码是根据编码结构产生对应的调度方案,并得到个体的目标值,具体步骤如下。

step 1: 初始化操作. 设各个工件第0道工序的加工完成时间为0,所在位置为起点;各台AGV第0个任务的完成时间为0,所在位置为起点。

step 2: 根据机器染色体、AGV染色体的信息,从左到右读取搬运任务基因,确定任务 $r$ 对应的工序 $O_{ij}$ , $O_{ij}$ 对应的机器 $k$ 和AGV $a$ 。

step 3: 读取当前系统状态. 包括: 工序 $O_{i(j-1)}$ 的加工完成时间和所在位置; 机器 $k$ 允许加工的时间; AGV $a$ 上一次任务的完成时间、所在位置以及当前电量。

step 4: 若任务为取货任务,则AGV需要从当前位置行驶到 $O_{i(j-1)}$ 加工完成的位置; 否则,需要从当前位置行驶到 $O_{ij}$ 加工的机器,搬运任务开始与结束时间解码时考虑约束(2)~(6)。

step 5: 若任务 $r$ 为取货任务,则 $O_{ij}$ 无需加工; 否则, $O_{ij}$ 在任务 $r$ 完成且机器允许加工时开始加工,加工的开始时间和完成时间解码时考虑约束(7)~(9)。

step 6: 重复step 2~step 5,直到所有的任务基因解码完成,生成调度方案,并得到最大完工时间。

## 2.2 离散差分进化算法

### 2.2.1 变异操作

从目标种群中随机选择3个不同的个体 $X_{r1}^{iter}$ 、 $X_{r2}^{iter}$ 和 $X_{r3}^{iter}$ ,按下式生成变异个体 $V^{iter}$ :

$$V^{iter} = \begin{cases} G(\sigma^{iter}, X_{r1}^{iter}), & \text{rand} < F; \\ \sigma^{iter}, & \text{otherwise.} \end{cases} \quad (17)$$

其中: rand为一个处于0~1之间的随机数; $G(\cdot)$ 为变换操作,本文对基于任务编码的染色体和基于机器/AGV编码的染色体,分别采用IPOX和MPX交叉<sup>[19]</sup>;  $\sigma^{iter}$ 为差分矢量,即

$$\sigma^{iter} = \begin{cases} G(X_{r2}^{iter}, X_{r3}^{iter}), & \text{rand} < F; \\ X_{r2}^{iter}, & \text{otherwise.} \end{cases} \quad (18)$$

重复上述过程直到变异个体的规模为 $P$ 。

### 2.2.2 交叉操作

选择第 $i$ 个变异个体 $V_i^{iter}$ 与第 $i$ 个种群个体进行交叉,生成试验个体 $U_i^{iter}$ ,有

$$U_i^{iter} = \begin{cases} G(V_i^{iter}, X_i^{iter}), & \text{rand} < C; \\ V_i^{iter}, & \text{otherwise.} \end{cases} \quad (19)$$

### 2.2.3 选择操作

在试验个体和种群个体中,选择适应度高的个体进入下一代,并引入模拟退火算法的接受准则,以一

定的概率接受较差的个体. 将不被接受的试验个体放入外部档案中,具体操作如下:

当 $C_{\max}(U_i^{iter}) < C_{\max}(X_i^{iter})$ 时, $X_i^{iter+1} = U_i^{iter}$ ; 当 $C_{\max}(U_i^{iter}) \geq C_{\max}(X_i^{iter})$ 且 $\text{rand} < \exp((C_{\max}(X_i^{iter}) - C_{\max}(U_i^{iter}))/T)$ 时, $X_i^{iter+1} = U_i^{iter}$ ; 其他情况下, $X_i^{iter+1} = X_i^{iter}$ , $U_i^{iter}$ 进入外部档案. 其中: $C_{\max}(\cdot)$ 为求解调度方案的目标函数值; $T$ 为退火温度,随迭代次数或运行时间的增大而减小,即

$$T = T(1 - \text{iter}/(G_{\max} + 1)), \quad (20)$$

$$T = T(1 - \text{nowtime}/(\text{endtime} + 1)). \quad (21)$$

## 2.3 基于关键路径的变邻域搜索

### 2.3.1 关键路径

关键路径是调度方案中工序间无时间间隔的最长路径<sup>[22]</sup>,相比于随机产生邻域结构,基于关键路径生成的邻域结构能提高邻域搜索效率<sup>[23]</sup>. 因此,本文首先确定关键路径;再通过调整关键路径中的任务产生邻域结构,进行变邻域搜索。

本文的关键路径需要考虑AGV的搬运,因此,将关键路径定义扩展为调度方案中工序间或搬运任务间无时间间隔的最长路径,即一条机器或AGV连续运行的关键任务序列. 首先,将最晚加工完成的工序对应的卸货任务作为第1个关键任务,设计4种搜索关键路径的规则:

1) 基于AGV连续运行任务优先规则. 从当前关键任务开始,向前搜索AGV连续运行时执行的任务作为关键任务,直到AGV运行时出现中断,若中断发生时间不为0,则说明AGV需要等待某工序加工完成时取货,将该工序对应的卸货任务作为下次搜索开始的关键任务,重复上述过程直到AGV运行时中断发生时间为0。

2) 基于机器连续运行任务优先规则. 从当前关键任务开始,向前搜索机器连续运行时执行的任务作为关键任务,直到机器运行时出现中断,从中断位置处的任务开始,基于AGV连续运行任务优先规则搜索关键任务以及下次搜索开始的关键任务,重复上述过程直到AGV运行时中断发生时间为0。

3) 基于紧前搬运优先规则. 从当前关键任务开始,搜索紧前搬运任务作为关键任务,重复上述过程直到当前任务无紧前搬运任务。

4) 基于紧前加工优先规则. 从当前关键任务开始,搜索紧前加工任务作为关键任务,若该任务无紧前加工任务,则将紧前搬运任务作为关键任务,重复上述过程直到当前任务无紧前搬运任务。

### 2.3.2 基于关键路径的变邻域搜索

变邻域搜索通过改变邻域结构可以进行更大范围的搜索,能够有效避免算法陷入局部最优,提高算法的局部搜索能力.然而,从全部任务集中选择任务进行调整具有一定的随机性.为此,本文设计基于关键路径的变邻域搜索,针对性地从关键路径中选择关键任务进行调整,针对性地搜索能使最大完工时间发生改变的邻域解,一定程度上可以避免产生无效搜索,提高搜索效率.

设计3种邻域结构  $N_1$ 、 $N_2$  和  $N_3$ .  $N_1$  通过任务交换的方式构建邻域解,在任务编码中,分别在关键路径的任务集和全部任务集中,随机选择两个对应工件不同的任务进行交换;  $N_2$  通过改变机器编码构建邻域解,确定关键路径的任务集对应的工序集,在该工序集中随机选择一个工序,为其随机选择一个机器;  $N_3$  通过改变 AGV 编码构建邻域解,确定关键路径的任务集对应的工序集,在该工序集中随机选择一个工序,并为其随机选择一个 AGV.

基于上述邻域结构,本文变邻域搜索的具体步骤如下.

step 1: 将种群中的最优解作为初始解  $X$ , 初始化  $r = 1, r_{\max} = 3, c_{\max} = 40$ .

step 2: 若  $r > r_{\max}$ , 则将当前解  $X$  作为当前搜索到的最优解  $X_{\text{best}}$ , 并通过  $X_{\text{best}}$  更新原种群中的最优解; 否则, 进入 step 3.

step 3: 随机扰动. 将当前解  $X$  通过  $N_r$  产生一个扰动解  $X'$ .

step 4: 局部搜索. 具体步骤如下:

1) 将扰动解  $X'$  作为初始解, 设  $c = 1$ .

2) 若  $c > c_{\max}$ , 则输出局部最优解  $X'$ ; 否则, 进入步骤 3).

3) 将当前解  $X'$  通过  $N_r$  产生一个新解  $X''$ , 若  $C_{\max}(X'') < C_{\max}(X')$ , 则  $X' = X''$ .

4)  $c = c + 1$ , 进入步骤 2).

step 5: 若  $C_{\max}(X') < C_{\max}(X)$ , 则  $X = X'$ , 进入 step 4; 否则,  $r = r + 1$ , 进入 step 2.

### 2.4 基于外部档案的更新策略

为提高种群的质量, 保证种群的多样性, 提高差分进化算法的搜索效率, 需要对种群中较差的个体进行更新. 然而, 随机生成解进行种群个体更新会带来不确定性, 同时需要计算随机解的目标函数值, 使计算量增大. 因此, 为提高求解质量, 充分利用计算资源, 本文通过外部档案策略更新种群个体, 将种群中较差的个体从外部档案中替换. 其中, 外部档案中存

储的是已搜索到但未被接受的试验个体.

首先, 确定种群中的较差个体, 若种群中个体的目标函数值相同, 则随机选择种群中一半的个体作为较差个体; 否则, 选择目标函数值高于平均值的个体作为较差个体. 然后, 确定较差个体数量  $X$ , 以及当前外部档案中个体数量  $Y$ , 若  $X > Y$ , 则将较差个体中前  $Y$  个个体通过外部档案中的个体进行替换, 剩余个体通过随机产生的个体替换; 否则, 取外部档案中目标函数值较好的前  $X$  个个体替换较差个体.

## 3 实验分析

### 3.1 算法对比实验

为验证算法改进策略以及 IDDE-VNS 算法的有效性, 本文对 10 个 Brandimarte 标准算例<sup>[25]</sup> 进行实验, 机器随机分布在  $5 \times 4$  的矩阵车间中, 起点位于车间一侧的居中位置, 车间中相邻节点的距离为 1. 为严格比较算法的性能, 本文选用 AGV 数量较少, 载位容量较少, 电池可用时间较长的场景: 设置 AGV 数量为 2, 载位容量为 1 和 2, 电池可用时间为 800. 参考文献 [26], 将每个算例限制算法的最大运行时间设置为一个与算例规模有关的函数,  $\text{endtime} = \sum_{i=1}^N u_i \times (M/2) \times 60(\text{ms})$ , 算法运行 10 次的平均结果作为参考值. 通过数据实验, 将 IDDE-VNS 的参数设置为能够使其性能最好的值: 种群规模  $P = 80$ , 变异概率  $F = 0.7$ , 交叉概率  $C = 0.5$ , 初始温度  $T = 20$ .

#### 3.1.1 IDDE-VNS 变体的对比实验

为验证算法改进策略对算法的改进效果, 构建以下 IDDE-VNS 变体.

1) 变体 1: 基于 AGV 连续运行任务优先规则生成关键路径, 基于外部档案更新种群个体;

2) 变体 2: 基于 AGV 连续运行任务优先规则生成关键路径;

3) 变体 3: 基于机器连续运行任务优先规则生成关键路径;

4) 变体 4: 基于紧前搬运优先规则生成关键路径;

5) 变体 5: 基于紧前加工优先规则生成关键路径;

6) 变体 6: 基于外部档案更新种群个体;

7) 变体 7: 无改进策略.

其中: 变体 2~变体 5、变体 7 无外部档案策略, 通过随机生成的解更新种群个体; 变体 6 和变体 7 无关关键路径策略, 通过随机选择任务进行变邻域搜索.

表 1 展示了各个算法在各个算例中的计算结果、均值以及相对变体 7 优化的算例个数. 如表 1 所示, 变体 2~变体 5 在多数算例中优于变体 7, 并且 4 种变

表1 算法改进策略的有效性验证

算例	变体1	变体2	变体3	变体4	变体5	变体6	变体7
MK01	94.2	95.2	97.8	94.2	96.8	95.3	96.0
MK02	92.6	96.2	90.8	85.8	89.8	95.6	97.8
MK03	372.0	376.0	384.8	383.4	380.0	390.0	387.2
MK04	157.2	164.8	161.6	154.2	160.80	164.6	162.0
MK05	276.2	317.8	322.4	315.2	319.6	275.8	326.6
MK06	315.6	317.8	322.4	315.2	319.6	317.0	326.6
MK07	265.2	266.2	262.4	267.2	267.8	265.6	272.2
MK08	672.6	676.2	665.2	672.0	668.8	684.0	698.6
MK09	598.0	596.6	625.4	619.4	615.6	609.4	611.2
MK10	538.8	550.2	549.0	561.2	573.2	570.0	559.2
均值	338.2	341.6	343.6	342.6	344.7	346.7	348.9
优化个数	10	9	8	8	7	7	-

体的均值均优于变体7,表明4种生成关键路径的规则均具有较好的适用性和有效性.其中,变体2基于AGV连续运行任务优先规则生成关键路径的方式最有效,故后续实验均通过该方式生成关键路径.变体6在多数算例中优于变体7,且均值优于变体7,表明外部档案策略具有较好的适用性和有效性.变体1的求解结果优于变体2和变体6,表明关键路径策略与外部档案策略同时使用会比分别单独使用有着更高

的求解效率,故后续实验将IDDE-VNS按变体1设置.

### 3.1.2 不同算法的对比实验

为进一步验证IDDE-VNS算法的有效性,将IDDE-VNS与混合变邻域搜索离散差分进化算法DDE-VNS<sup>[19]</sup>、离散差分进化算法DDE<sup>[19]</sup>、混合变邻域搜索离散遗传算法GA-VNS<sup>[13]</sup>和遗传算法GA<sup>[13]</sup>作比较.算法参数均参考算法出处,设置DDE的种群规模为80,变异概率为0.7,交叉概率为0.5,初始温度为20;GA的种群规模为80,交叉概率为0.9,变异概率为0.2;VNS的循环次数为40.各个算法的编码和运行时间与本文一致,初始种群通过随机生成.

表2展示了不同算法在各个算例中的计算结果,以及IDDE-VNS相对于对比算法的优化比例.如表2所示,IDDE-VNS在各个算例的求解结果均优于其他算法,表明IDDE-VNS具有有效性.其中,DDE算法的求解效率较GA略低,然而,通过VNS能得到明显的改进,同时DDE-VNS的求解效率较GA-VNS略高.IDDE-VNS通过两种改进策略能较DDE-VNS进一步提高求解效率.

表2 不同算法优化效果对比

算例	IDDE-VNS	DDE-VNS	DDE	GA-VNS	GA
MK01	94.2	96.0(1.9%)	116.6(19.2%)	99.8(5.6%)	113.2(16.8%)
MK02	92.6	97.8(5.3%)	112.7(17.8%)	97.4(4.9%)	115.4(19.8%)
MK03	372.0	387.2(3.9%)	478.8(22.3%)	390.0(4.6%)	485.2(23.3%)
MK04	157.2	162.0(3.0%)	197.6(20.4%)	163.0(3.6%)	196.4(20.0%)
MK05	276.2	277.8(0.6%)	330.4(16.4%)	284.1(2.8%)	324.8(15.0%)
MK06	315.6	326.6(3.4%)	402.2(21.5%)	331.0(4.7%)	418.6(24.6%)
MK07	265.2	272.2(2.6%)	322.4(17.7%)	270.5(2.0%)	324.3(18.2%)
MK08	672.6	698.6(3.7%)	788.6(14.7%)	674.4(0.3%)	750.0(10.3%)
MK09	598.0	611.2(2.2%)	748.4(20.1%)	612.2(2.3%)	740.0(19.2%)
MK10	538.8	559.2(3.6%)	699.0(22.9%)	572.8(5.9%)	690.2(21.9%)
均值	338.2	348.9(3.0%)	419.7(19.3%)	349.5(3.7%)	415.8(18.9%)

图3展示了不同算法求解MK06时的收敛曲线.如图3所示:VNS能给DDE和GA在算法运行的前期带来较高的收敛速度,从而使DDE-VNS和GA-VNS能收敛到一个较低的值;IDDE-VNS在此基础上有着更高的收敛速度,并且能收敛到一个更低的值.

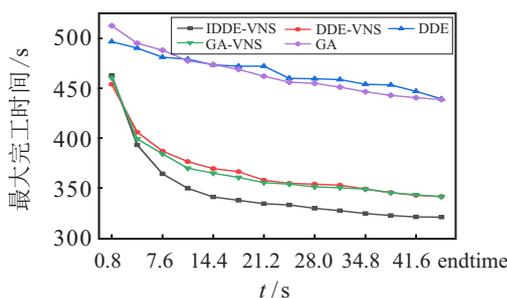


图3 不同算法收敛曲线

### 3.2 AGV数量影响分析

以MK06为例,分析不同AGV数量对IDDE-VNS的求解结果,IDDE-VNS相对于DDE-VNS改进效果的影响,控制AGV的载位容量取值区间为[1, 2],AGV的电池可用时间为800.如图4所示,随着AGV数量

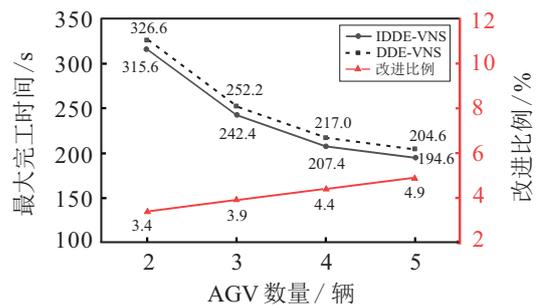


图4 AGV数量的影响

的提高,系统可调度性逐渐增强,算法的求解结果逐渐降低,改进效果逐渐增加.然而,当AGV的数量达到4时,AGV数量趋于饱和,算法的求解结果下降并趋于平稳,再增加AGV数量时,对求解结果的影响程度减弱.

### 3.3 AGV载位容量影响分析

以MK06为例,分析不同AGV载位容量的影响.控制AGV的数量为2,AGV的电池可用时间为800.如图5所示,相比于单载AGV,多载AGV能让求解结果更低,改进效果更高.当载位容量为(2,2)时求解结果最好,当载位容量为(1,2)时改进效果最好.当载位容量为(2,3)时,求解结果变差,这是因为载位容量较大时算法的搜索空间更大,所以在指定的时间内难以得到更有效的调度方案.

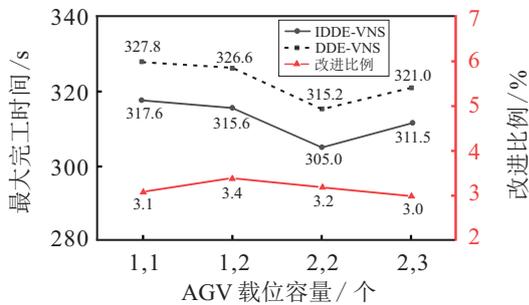


图5 AGV载位容量的影响

### 3.4 AGV电量影响分析

以MK06为例,分析不同AGV电池可用时间的影响.控制AGV的数量为2,AGV的载位容量为(1,2).如图6所示,随着AGV电池可用时间的降低,算法的求解结果逐渐升高,改进效果逐渐增加,这是因为AGV电池可用时间降低使得AGV会增加其搬运时间,从而使AGV的调度对结果的影响变大.当AGV电池可用时间达到600时,AGV的电池充电基本不影响AGV的搬运时间,算法的求解效果和改进效果趋于稳定.

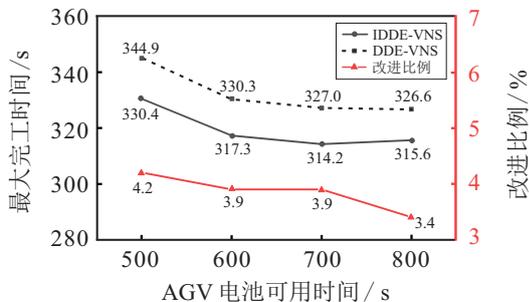


图6 AGV电池可用时间的影响

## 4 结论

针对考虑充电约束的FMS中机器与混合多载AGV集成调度问题,以最小化最大完工时间为目标,

本文提出了一种改进混合离散差分进化算法,通过离散操作进行差分进化算法的变异、交叉,并对每次迭代的最优个体进行变邻域搜索.设计了基于搬运任务、机器和AGV的3层编码结构,并进行合理化调整;设计了基于关键路径的变邻域搜索,针对性地调整关键路径上任务的次序、对应工序的机器和AGV;设计了基于外部档案的个体更新策略,将种群中较差的个体从外部档案中进行替换.最后,通过实验验证了算法改进策略以及所提出算法的有效性,分析了AGV数量、AGV载位容量以及AGV电池可用时间对算法求解结果和改进效果的影响.在下一步的研究中,将考虑更多影响因素和约束条件,如AGV冲突的问题、缓存问题以及多目标问题.

### 参考文献(References)

- [1] 孟磊磊,张彪,任亚平,等.求解分布式柔性作业车间调度的混合蛙跳算法[J].机械工程学报,2021,57(17): 263-272.  
(Meng L L, Zhang B, Ren Y P, et al. Hybrid shuffled frog-leaping algorithm for distributed flexible job shop scheduling[J]. Journal of Mechanical Engineering, 2021, 57(17): 263-272.)
- [2] 孙爱红,雷琦,宋豫川,等.基于深度强化学习求解作业车间机器与AGV联合调度问题[J].控制与决策,2024,39(1): 253-262.  
(Sun A H, Lei Q, Song Y C, et al. Deep reinforcement learning for solving the joint scheduling problem of machines and AGVs in job shop[J]. Control and Decision, 2024, 39(1): 253-262.)
- [3] 武星,翟晶晶,楼佩煌,等.考虑任务行程时间的多载量自动导引车系统防死锁任务调度[J].中国机械工程,2021,32(23): 2840-2849.  
(Wu X, Zhai J J, Lou P H, et al. Deadlock-free task scheduling with task traveling time for a multi-load AGV system[J]. China Mechanical Engineering, 2021, 32(23): 2840-2849.)
- [4] Bilge Ü, Ulusoy G. A time window approach to simultaneous scheduling of machines and material handling system in an FMS[J]. Operations Research, 1995, 43(6): 1058-1070.
- [5] 胡晓阳,姚锡凡,黄鹏,等.改进迭代局部搜索算法求解多AGV柔性作业车间调度问题[J].计算机集成制造系统,2022,28(7): 2198-2212.  
(Hu X Y, Yao X F, Huang P, et al. Improved iterative local search algorithm for solving multi-AGV flexible job shop scheduling problem[J]. Computer Integrated Manufacturing Systems, 2022, 28(7): 2198-2212.)
- [6] Satish Kumar M V, Janardhana R, Rao C S P. Simultaneous scheduling of machines and vehicles in an FMS environment with alternative routing[J]. The International Journal of Advanced Manufacturing Technology, 2011, 53(1): 339-351.
- [7] Ho Y C, Chien S H. A simulation study on

- the performance of task-determination rules and delivery-dispatching rules for multiple-load AGVs[J]. *International Journal of Production Research*, 2006, 44(20): 4193-4222.
- [8] Ho Y C, Liu H C. The performance of load-selection rules and pickup-dispatching rules for multiple-load AGVs[J]. *Journal of Manufacturing Systems*, 2009, 28(1): 1-10.
- [9] Ho Y C, Liu H C, Yih Y. A multiple-attribute method for concurrently solving the pickup-dispatching problem and the load-selection problem of multiple-load AGVs[J]. *Journal of Manufacturing Systems*, 2012, 31(3): 288-300.
- [10] Du L Z, Ke S F, Wang Z, et al. Research on multi-load AGV path planning of weaving workshop based on time priority[J]. *Mathematical Biosciences and Engineering*, 2019, 16(4): 2277-2292.
- [11] Zou W Q, Pan Q K, Tasgetiren M F. An effective iterated greedy algorithm for solving a multi-compartment AGV scheduling problem in a matrix manufacturing workshop[J]. *Applied Soft Computing*, 2021, 99: 106945.
- [12] Zhang L X, Hu Y G, Guan Y. Research on hybrid-load AGV dispatching problem for mixed-model automobile assembly line[J]. *Procedia CIRP*, 2019, 81: 1059-1064.
- [13] 崔琪, 吴秀丽, 余建军. 变邻域改进遗传算法求解混合流水车间调度问题[J]. *计算机集成制造系统*, 2017, 23(9): 1917-1927.  
(Cui Q, Wu X L, Yu J J. Improved genetic algorithm variable neighborhood search for solving hybrid flow shop scheduling problem[J]. *Computer Integrated Manufacturing Systems*, 2017, 23(9): 1917-1927.)
- [14] 李佳磊, 顾幸生. 双种群混合遗传算法求解具有预防性维护的分布式柔性作业车间调度问题[J]. *控制与决策*, 2023, 38(2): 475-482.  
(Li J L, Gu X S. Two-population hybrid genetic algorithm for distributed flexible job-shop scheduling problem with preventive maintenance[J]. *Control and Decision*, 2023, 38(2): 475-482.)
- [15] 耿凯峰, 叶春明. 带工序跳跃的绿色混合流水车间机器与AGV联合调度[J]. *控制与决策*, 2022, 37(10): 2723-2732.  
(Geng K F, Ye C M. Joint scheduling of machines and AGVs in green hybrid flow shop with missing operations[J]. *Control and Decision*, 2022, 37(10): 2723-2732.)
- [16] 吴秀丽, 袁琦. 差分进化算法求解带批处理机的机器人制造单元调度问题[J]. *控制与决策*, 2020, 35(1): 74-82.  
(Wu X L, Yuan Q. Differential evolution algorithm for solving robotic cell scheduling problem with batch-processing machines[J]. *Control and Decision*, 2020, 35(1): 74-82.)
- [17] 王开, 龚文引. 求解非线性方程组系统的改进差分进化算法[J]. *控制与决策*, 2020, 35(9): 2121-2128.  
(Wang K, Gong W Y. Solving nonlinear equations system with an improved differential evolution[J]. *Control and Decision*, 2020, 35(9): 2121-2128.)
- [18] Pan Q K, Wang L, Qian B. A novel differential evolution algorithm for bi-criteria no-wait flow shop scheduling problems[J]. *Computers & Operations Research*, 2009, 36(8): 2498-2511.
- [19] 伍乐, 宋豫川, 吕向飞, 等. 基于差分进化算法的FMS中机器与AGV同时调度方法[J]. *重庆大学学报*, 2021, 45(12): 116-129.  
(Wu Y, Song Y C, Lyu X F, et al. An improved differential evolution algorithm for simultaneous scheduling of machines and AGVs in an FMS[J]. *Journal of Chongqing University: Natural Science Edition*, 2021, 45(12): 116-129.)
- [20] Li J Q, Pan Q K, Wang Y T, et al. A block-neighboring-structure based tabu search algorithm for job shop scheduling problems[J]. *Machine Tool & Hydraulics*, 2009, 37(12): 173-175.
- [21] 潘全科, 王凌, 高亮, 等. 基于差分进化与块结构邻域的作业车间调度优化[J]. *机械工程学报*, 2010, 46(22): 182-188.  
(Pan Q K, Wang L, Gao L, et al. Differential evolution algorithm based on blocks on critical path for job shop scheduling problems[J]. *Journal of Mechanical Engineering*, 2010, 46(22): 182-188.)
- [22] 刘刚, 王瑛, 张发. 基于关键路径求解作业车间调度问题的收敛性分析[J]. *计算机集成制造系统*, 2014, 20(5): 1078-1087.  
(Liu G, Wang Y, Zhang F. Convergence analysis of job shop scheduling problem based on critical path[J]. *Computer Integrated Manufacturing Systems*, 2014, 20(5): 1078-1087.)
- [23] 薛玲玲. 作业车间调度的块结构邻域搜索遗传算法[J]. *计算机集成制造系统*, 2021, 27(10): 2848-2857.  
(Xue L L. Block structure neighborhood search genetic algorithm for job-shop scheduling[J]. *Computer Integrated Manufacturing Systems*, 2021, 27(10): 2848-2857.)
- [24] 吴锐, 郭顺生, 李益兵, 等. 改进人工蜂群算法求解分布式柔性作业车间调度问题[J]. *控制与决策*, 2019, 34(12): 2527-2536.  
(Wu R, Guo S S, Li Y B, et al. Improved artificial bee colony algorithm for distributed and flexible job-shop scheduling problem[J]. *Control and Decision*, 2019, 34(12): 2527-2536.)
- [25] Brandimarte P. Routing and scheduling in a flexible job shop by tabu search[J]. *Annals of Operations Research*, 1993, 41(3): 157-183.
- [26] Ruiz R, Stützle T. A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem[J]. *European Journal of Operational Research*, 2007, 177(3): 2033-2049.

#### 作者简介

王子实(1999—), 男, 硕士生, 从事生产调度、物流系统仿真和优化等研究, E-mail: 202114820@mail.sdu.edu.cn;  
吴耀华(1963—), 男, 教授, 博士生导师, 从事自动拣选、仓储系统设计和优化等研究, E-mail: yaohua.wu@sdu.edu.cn.