

控制与决策

Control and Decision

基于混合学习策略的可变速AGV与机器绿色集成调度

陈仁胜, 吴斌, 闫飞一

引用本文:

陈仁胜, 吴斌, 闫飞一. 基于混合学习策略的可变速AGV与机器绿色集成调度[J]. *控制与决策*, 2024, 39(12): 3955-3963.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2023.1708>

您可能感兴趣的其他文章

Articles you may be interested in

[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>

[基于复杂昂贵仿真的体系效能多目标优化](#)

Complex and expensive simulation based multi-objective optimization to system-of-system effectiveness
控制与决策. 2021, 36(3): 589-598 <https://doi.org/10.13195/j.kzyjc.2019.0844>

[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>

[基于迁移学习灰支持向量回归机的交互式进化计算](#)

Interactive evolutionary computation based on transfer learning grey support vector regression
控制与决策. 2021, 36(10): 2399-2408 <https://doi.org/10.13195/j.kzyjc.2020.0420>

[基于改进多目标优化算法的分布式数据中心负载调度](#)

Multi-objective optimization of energy and performance management in distributed data centers
控制与决策. 2021, 36(1): 159-165 <https://doi.org/10.13195/j.kzyjc.2019.0702>

基于混合学习策略的可变速 AGV 与机器绿色集成调度

陈仁胜, 吴斌[†], 闫飞一

(南京工业大学 经济与管理学院, 南京 211816)

摘要: 传统制造业正逐渐向智能化、绿色化制造模式转型. 为实现柔性制造车间的增效减排, 以最小化最大完工时间和总能耗为目标, 构建充电约束下可变速 AGV 与机器绿色集成调度模型, 并设计一种基于混合学习策略的改进 NSGA-II 算法进行优化求解. 采用基于工序、机器、AGV 及其速度的四段式染色体编码方案, 对各编码段分别设计不同的交叉变异算子; 采用基于反向学习的精英保留策略, 以提高算法的种群多样性; 提出针对问题特征的邻域搜索算子, 基于 Q -learning 强化学习算法, 动态调整迭代过程中的邻域结构, 增强算法的局部搜索能力. 最后通过仿真实验表明, 改进 NSGA-II 算法是求解该问题的有效方法.

关键词: 绿色集成调度; 多目标优化; 混合学习策略; 可变速 AGV; NSGA-II; 充电约束

中图分类号: TP18; TP23 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2023.1708

引用格式: 陈仁胜, 吴斌, 闫飞一. 基于混合学习策略的可变速 AGV 与机器绿色集成调度 [J]. 控制与决策, 2024, 39(12): 3955-3963.

Hybrid learning strategy for green integrated scheduling with variable speed AGV

CHEN Ren-sheng, WU Bin[†], YAN Fei-yi

(School of Economics and Management, Nanjing Tech University, Nanjing 211816, China)

Abstract: The traditional manufacturing industry is gradually transitioning toward intelligent and environmentally friendly production modes. To achieve efficiency improvements and emissions reduction in flexible manufacturing workshops, this study aims to minimize makespan and total energy consumption. It constructs an integrated scheduling model for a variable-speed AGV and machine under charging constraint. An improved NSGA-II optimization algorithm is designed based on a hybrid learning strategy. This algorithm adopts a four-segment chromosome encoding scheme based on process, machines, AGV and AGV speed, with different crossover and mutation operators for each encoding segment. Additionally, an elite preservation strategy based on opposition-based learning is employed to enhance the algorithm's population diversity. Furthermore, a neighborhood search operator tailored to problem characteristics is proposed, utilizing the Q -learning reinforcement learning algorithm to dynamically adjust the neighborhood structure during the iteration process, thereby enhancing the algorithm's local search capabilities. Finally, the effectiveness of the improved NSGA-II in solving this problem is verified through simulation tests.

Keywords: green integrated scheduling; multi-objective optimization; hybrid learning strategy; variable speed AGV; NSGA-II; charging constraint

0 引言

随着我国“双碳”战略的提出, 节能减排已成为制造业发展的主要目标之一. 柔性制造车间作为生产系统的基础, 其调度计划的合理性很大程度上影响绿色制造的进程, 基于此, 一些学者已着手开展绿色调度^[1]问题的相关研究. AGV 作为重要的智能车间

物流设备, 考虑 AGV 运行与机器加工之间的协同优化是打造智能化工厂^[2]的重要举措, AGV 和机器的集成调度问题日益受到重视^[3-5]. 可变速 AGV 能够通过合理调整运输速度, 在提高机器利用效率的同时控制运输过程中产生的能耗. 在实际生产车间中, 还需要考虑充电约束对 AGV 运输的影响, 保证 AGV 自身

收稿日期: 2023-12-09; 录用日期: 2024-02-14.

基金项目: 国家社科基金一般项目(20BGL025); 国家重点研发项目子课题项目(2021YFB3301302); 江苏省研究生科研与实践创新计划项目(KYCX23_1522).

责任编辑: 王凌.

[†]通讯作者. E-mail: wubin@njtech.edu.cn.

电量消耗与集成调度之间的协调性. 因此,综合考虑充电约束下可变速AGV资源与机器资源的集成,构建一个节能高效的智能化生产系统,对促进制造业绿色化、智能化转型具有重要的理论价值和实际意义.

AGV在实际应用中可以表现出不同的速度,其运行速度与实际能耗、完工时间都有着紧密的联系. Yang等^[6]研究了自动集装箱码头AGV调度,以最小碳排放成本与延迟成本为目标,提出了一种考虑不同运输环境下的AGV速度调整策略. 郭兴海等^[7]提出了一种多目标与速度控制法,通过设定AGV速度控制规则,确保AGV在静态与动态环境下的转向过程,提升工作效率. Liu等^[8]在仓库分拣系统的AGV路径规划问题中假设AGV速度可以被调度,每一个运输任务对应一个独立AGV速度,提出了一种自适应遗传算法. 目前的研究仅在部分AGV调度问题中考虑到AGV的可变速度,尚缺乏在集成调度中可变速AGV的研究.

对于AGV充电约束问题,部分学者分析了不同充电策略与充电阈值对调度目标的影响^[9-10]. Dang等^[11]使用自适应大领域搜索算法,研究了具有电池约束的异构多负载AGV调度问题,允许AGV在临近电池临界阈值时进行部分充电. Han等^[12]提出了一种基于数字孪生的动态AGV调度方法,借助虚拟现实数据交互共生的特点,解决AGV调度系统中的AGV充电问题. Zhan等^[13]提出AGV双阶段充电策略,开发了两个基于启发式算法的路由决策,根据电池特性调整充电时长,并通过实例验证了该方法的有效性. 然而,上述研究对象均为恒定速度AGV,对于充电约束下的可变速AGV考虑较少. 可变速AGV在不同速度状态下会消耗不同的电量,对于充电时刻的安排具有更强的柔性,导致问题的求解难度更大.

柔性制造车间中AGV与机器集成调度作为NP难问题,可以由许多智能优化算法^[14-16]进行求解. 其中,遗传算法作为一种基于自然进化的智能优化算法,具有结构简单、易于实现等优点,广泛应用于组

合优化问题. Lyu等^[17]考虑集成调度问题中的AGV路径冲突和AGV数量,提出了一种结合时间窗与Dijkstra算法的改进遗传算法. Yuan等^[18]使用非支配排序遗传算法(NSGA-II)求解车间调度问题,通过调整迭代过程中父代个体比例,防止算法陷入局部最优. Amelian等^[19]结合NSGA-II和仿真方法,对随机故障易发生的作业车间调度问题进行了多目标优化.

综合分析相关文献,目前在集成调度问题中尚缺乏考虑充电约束下可变速AGV的相关研究. 因此,本文以最小化最大完工时间和总能耗为目标,考虑AGV充电约束,构建可变速AGV与机器绿色集成调度(green integrated scheduling of variable speed AGV and machine, GIS-VSAM)模型,设计一种四段式染色体编码,对各编码段采用不同的交叉变异算子,提出一种基于混合学习策略改进NSGA-II求解该问题. 分别使用反向学习策略与强化学习中的Q-learning算法,改进精英保留策略与邻域搜索. 最后通过仿真实验验证了所提出算法的优越性.

1 问题描述

GIS-VSAM问题如图1所示,其描述为:作业车间中有 M 台机器、 N 个工件、 K 个AGV,每个工件 i 有 O_i 道工序, O_{ij} 表示工件 i 的第 j 道工序. 工件与AGV从装载区出发,工序 O_{ij} 由AGV运输到该工序可选机器中的一台上进行加工,工件加工完成后由AGV运往卸载区. AGV的运输任务可以分为空载运行与负载运行两个部分,分别表示AGV从当前位置运行到工件所在位置,以及AGV将工件运送到本道工序 O_{ij} 所使用的加工机器旁. 在单独一段空/负载运行任务中,AGV保持匀速运行,不同任务间可以切换AGV速度. AGV在不同速度以及空/负载运行情况下存在不同的电量消耗,当AGV低于最低电量阈值时,需要前往充电站充电. 要求以最小化最大完工时间和总能耗为目标,在满足约束条件的情况下,合理安排AGV及其速度,将工件送往合适的机器上进行加工.

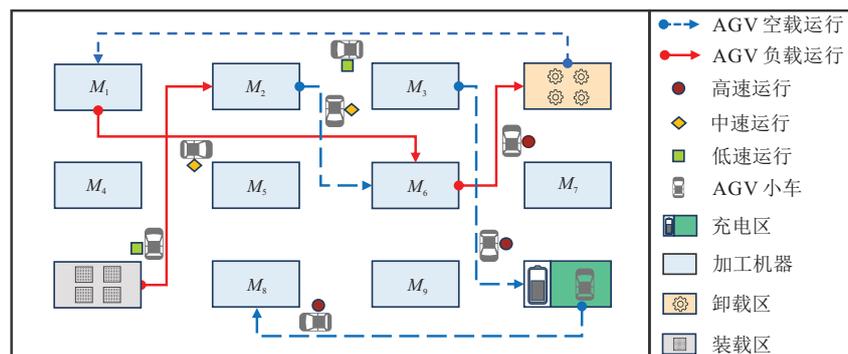


图1 GIS-VSAM问题示意图

1.1 条件假设

1) 所有AGV与工件的初始位置均为装载区, 充电站与卸载区处于同一位置。

2) AGV在空/负载状态下有多种运行速度, 相同速度情况下, AGV负载能耗高于空载能耗, 不考虑AGV等待时的能耗。

3) AGV运输速度越快单位时间能耗越高, AGV空/负载运行任务过程中速度恒定, 不同运输任务之间可以变换AGV速度。

4) AGV的最低电量阈值能够满足AGV以最高能耗完成一个最远距离任务并前往充电站。

5) AGV负载时不允许返回充电站, 充电过程不可中断, 直至电量达到最高阈值为止。

6) 不考虑机器故障、路径冲突等问题, 忽略工件的装卸载时间。

1.2 符号变量说明

模型中所使用的符号变量含义如下: 工件索引号 $i \in \{1, 2, \dots, N\}$; 工序索引号 $j \in \{1, 2, \dots, O_i\}$; 机器索引号 $m \in \{1, 2, \dots, M\}$; AGV索引号 $k \in \{1, 2, \dots, K\}$; AGV速度索引号 $r \in \{1, 2, \dots, R\}$; p' 和 p 分别表示AGV某次运行的起点、终点; E_P 和 E_S 分别表示机器加工与待机总能耗; E_U 和 E_L 分别表示AGV空/负载总能耗; d_U^{ijk} 和 d_L^{ijk} 分别表示AGV k 执行 O_{ij} 的空/负载运行距离; d_i^n 、 d_k^c 、 d_k^i 分别表示已完工的工件 i 与卸载区 n 的距离、AGV k 与充电站 c 的距离、AGV k 与已完工的工件 i 的距离; v_r 表示AGV的第 r 种速度; $O_{i,j'}$ 表示机器 m 在加工 O_{ij} 之前的一道工序; O_{ab} 表示AGV k 在完成工序 O_{ij} 后的下一道工序; P_P^m 和 P_S^m 分别表示机器 m 的加工与待机功率; P_U^r 和 P_L^r 分别表示AGV k 使用第 r 种速度时的空/负载功率; P_C 表示AGV的充电速率; B 表示AGV的最高充电阈值; t_C^k 表示AGV k 的充电时长; C_i 表示工件 i 到达卸载区的时刻; C_m 表示机器 m 完成所有任务的时刻; S_{ijm} 和 C_{ijm} 分别表示工序 O_{ij} 在机器 m 上的开始与结束加工时刻; t_{ijm} 表示加工时长; UST_{ijk} 、 UET_{ijk} 、 LST_{ijk} 、 LET_{ijk} 分别表示AGV k 执行工序 O_{ij} 空载开始与结束时间、负载开始与结束时间; CST_{ijk} 和 CET_{ijk} 分别表示AGV k 完成工序 O_{ij} 并到达充电站后的充电开始与结束时间; USB_{ijk} 和 LEB_{ijk} 分别表示AGV k 执行工序 O_{ij} 的空载开始与负载结束时的电量。

定义如下决策变量: x_{ijm} , 若工序 O_{ij} 在机器 m 上加工则为1, 否则为0; y_{ijk} , 若工序 O_{ij} 由AGV k 运输则为1, 否则为0; α_{ijk} , 若AGV k 完成工序 O_{ij} 后前

往装取、运送某个完成所有工序的工件则为1, 否则为0; ρ_{ijk} , 若AGV k 完成工序 O_{ij} 后需要充电则为1, 否则为0; $\sigma_{p'pkr}$, 若AGV k 使用第 r 种速度从 p' 到 p 则为1, 否则为0。

1.3 GIS-VSAM数学模型

目标函数为

$$f_1 = \min(\max(C_i)), \forall i; \quad (1)$$

$$f_2 = \min(E_P + E_S + E_L + E_U). \quad (2)$$

约束条件为

$$E_P = \sum_{m=1}^M \sum_{i=1}^N \sum_{j=1}^{O_i} t_{ijm} \cdot x_{ijm} \cdot P_P^m; \quad (3)$$

$$E_S = \sum_{m=1}^M \left(C_m - \sum_{i=1}^N \sum_{j=1}^{O_i} t_{ijm} \cdot x_{ijm} \right) \cdot P_S^m; \quad (4)$$

$$E_L = \sum_{i=1}^N \sum_{j=1}^{O_i} \sum_{k=1}^K \sum_{r=1}^R \left(\frac{d_L^{ijk}}{v_r} + \frac{d_i^n}{v_r} \cdot \alpha_{ijk} \right) \cdot P_L^r; \quad (5)$$

$$E_U = \sum_{i=1}^N \sum_{j=1}^{O_i} \sum_{k=1}^K \sum_{r=1}^R \left(\frac{d_U^{ijk}}{v_r} + \frac{d_k^c}{v_r} \cdot \rho_{ijk} + \frac{d_k^i}{v_r} \cdot \alpha_{ijk} \right) \cdot P_U^r; \quad (6)$$

$$\sum_{m=1}^M x_{ijm} \leq 1, \forall i, j; \quad (7)$$

$$\sum_{k=1}^K y_{ijk} \leq 1, \forall i, j; \quad (8)$$

$$\sum_{r=1}^R \sigma_{p'pkr} = 1, p', p \in \{1, 2, \dots, M+2\}, \forall k; \quad (9)$$

$$S_{ijm} \geq \max\{\text{LET}_{ijk}, C_{i'j'm}\}, \forall i, i', j, j', m, k; \quad (10)$$

$$\text{LST}_{ijk} \geq \max\{\text{UET}_{ijk}, C_{i(j-1)m}\}, \forall i, j, m, k; \quad (11)$$

$$\text{UST}_{abk} \geq \rho_{ijk} \cdot \left(\text{LET}_{ijk} + \frac{d_k^c}{v_r} + t_C^k \right) + (1 - \rho_{ijk}) \cdot \text{LET}_{ijk}, \forall i, j, m, k, r; \quad (12)$$

$$t_C^k = \left(B - \text{LEB}_{ijk} + \frac{d_k^c}{v_r} \cdot P_U^r \right) / P_C, \forall i, j, k, r; \quad (13)$$

$$\text{USB}_{abk} = (1 - \rho_{ijk}) \cdot \text{LEB}_{ijk} + \rho_{ijk} \cdot B, \forall i, j, k; \quad (14)$$

$$\text{CST}_{ijk} = \text{LET}_{ijk} + \frac{d_k^c}{v_r}, \forall i, j, k, r; \quad (15)$$

$$\text{CET}_{ijk} = \text{CST}_{ijk} + t_C^k, \forall i, j, k. \quad (16)$$

式(1)和(2)分别表示最小化最大完工时间和总能耗的目标函数。式(3)和(4)分别表示机器加工能耗、机器待机能耗。式(5)表示AGV负载总能耗, 包括执行工序 O_{ij} 的负载运行能耗、运输已完工的工

件前往卸载区的负载运行能耗. 式(6)表示AGV空载总能耗,包括执行工序 O_{ij} 的空载运行能耗、前往充电站的空载运行能耗、前往运输已完工工件的空载运行能耗. 式(7)和(8)分别表示一个工件只能同时被一个AGV运输且只能同时被一个机器加工. 式(9)表示AGV在两节点间运行时,只能保持一种速度. 式(10)和(11)分别表示工件加工时间约束、工件运输时间约束. 式(12)表示一个AGV完成前一个运送任务后,若不需要充电,则AGV在原位置等待下一个任务;若需要充电,则AGV需要前往充电站充电至电量最高阈值后才能开始下一个任务. 式(13)表示充电时长. 式(14)表示AGV的电量更新. 式(15)和(16)分别表示AGV到达充电站后的充电开始时间与充电结束时间.

2 算法设计

NSGA-II算法具有强大的优化能力,特别是对于多目标、大规模的NP难问题. 然而,NSGA-II算法也存在解集多样性较差、局部搜索能力不足等缺点. 本文针对NSGA-II算法的不足之处,设计一种基于混合学习策略的改进NSGA-II(NSGA-II-HLS)算法进行求解.

2.1 编码与解码

GIS-VSAM问题包含工序、机器、AGV及其速度4个子问题,并考虑了AGV的充电约束,问题模型复杂,因此本文采用四段式整数编码. 相比基于工序和机器的二段式编码,四段式编码更为直观,且简化了后续的解码步骤,染色体编码如图2所示.

OS	2	1	2	3	1	3	3	2								
MS	1	2	1	3	2	3	1	3								
AS	2	1	3	1	2	1	3	2								
VS	1	2	2	3	3	2	1	2	3	1	2	3	3	2	1	1

图2 四段式编码

图2中第1段为工序编码段OS,每个数字代表工件的编号,出现的次数代表该工件的第几道工序. 第2段为机器编码段MS,分别对应 $O_{11}, O_{12}, \dots, O_{33}$ 的加工机器. 第3段为AGV编码段AS,表示运输各道工件工序所使用的AGV. 第4段为AGV速度编码段VS,表示AGV执行每道工序的空载运行与负载运行速度,其长度为其他编码段的两倍.

本文对GIS-VSAM问题解码规则设置如下.

step 1: 初始化工件、机器、AGV及其速度信息.

step 2: 对OS中从左至右的每一道工序,获取其工件索引号 i 和工序索引号 j ,在MS、AS、VS中查找该工序对应的机器索引号 m 、AGV索引号 k 、AGV速度索引号 r .

step 3: AGV每道运输任务结束后,对自身电量进行检查. 如果AGV电量不足,则AGV以最高速度前往充电站进行充电,计算AGV的到达时间与能耗.

step 4: 找出AGV当前需要运输工件上一道工序所在的机器 m' ,若AGV在机器 m' 旁,则转至step 6,否则转至step 5.

step 5: AGV k 以空载运行速度 v' 移动到机器 m' 旁,计算AGV k 的到达时间与能耗.

step 6: 根据式(11)确定AGV k 的负载运行开始时间,如果 O_{ij} 为工件 i 的第1道工序,则AGV k 的负载开始时间即为AGV k 到达装载区的时间.

step 7: AGV k 以负载运行速度 v 将工件 i 从机器 m' 运往机器 m ,计算AGV k 的到达时间与能耗.

step 8: 使用基于贪婪策略的插入式解码方法,遍历机器 m 当前的空闲时间段,查找最早可插入 O_{ij} 的位置,并计算完工时间与能耗.

step 9: 若 O_{ij} 为工件 i 的最后一道工序,则选择最早可用的AGV以最高速度前往运输工件 i 并搬运至卸载区,计算AGV的到达时间与能耗.

step 10: 判断OS所有基因位是否结束,若未结束则转至step 2,否则终止,输出最终结果.

2.2 交叉变异操作

为保证子代染色体中OS的合法性,使用改进优先工序编码交叉算子(improved precedence operation crossover, IPOX)如图3所示,具体步骤如下.

step 1: 将所有工件随机分到工件集 S_1 和 S_2 中.

step 2: 将Parent 1中包含在 S_1 中的工件复制到Child 1相同的位置上,将Parent 2中包含在 S_2 中的工件复制到Child 2相同的位置上.

step 3: 复制Parent 2中包含在 S_2 的工件到Child 1,Parent 1中包含在 S_1 的工件到Child 2,保持它们相对顺序不变.

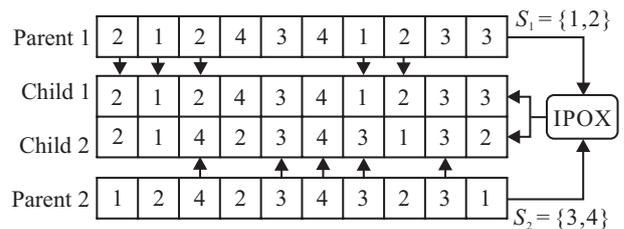


图3 IPOX交叉操作

针对MS使用多点交叉(multi-point crossover,

MPX), 由于MPX算子只针对同一工序所使用的加工机器进行交叉, 生成的子代染色体为可行解. 同样, 对于AS和VS也采用MPX算子, 如图4所示, 具体步骤如下.

step 1: 随机生成一个与编码段相同长度的0/1数组Random.

step 2: 随机选择出两条父代染色体的编码段Parent 1和Parent 2, 交换Parent 1和Parent 2中与数组Random中数字1对应的基因, 保持其他基因不变, 生成子代编码段.

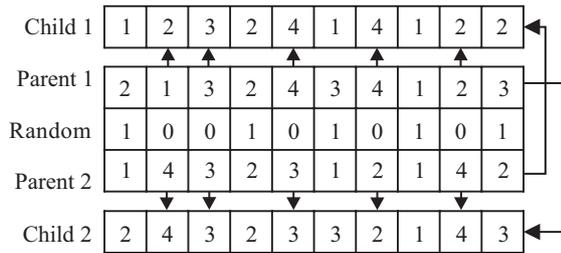


图4 MPX交叉操作

本文分别针对各基因编码段采用不同的变异方法. 在OS中, 任意交换两道不同工件的工序位置, 例如将图2中的OS“2, 1, 2, 3, 1, 3, 3, 2”变为“2, 2, 1, 3, 1, 3, 3, 2”, 通过该方法始终保证OS的可行性. 在MS中, 任选一道工序, 从可选设备集中另选一台设备替换当前设备. 针对AS与VS, 更改某道工序所使用的AGV及其速度, 作为对应编码段的变异操作.

2.3 基于反向学习的精英保留策略

反向学习 (opposition-based learning, OBL)^[20] 的基本思想是生成原解的反向解, 相较于随机解, 其能够以更高的概率到达全局最优. 该策略可以有效增加种群的多样性和质量, 防止算法进入“早熟”状态. 本文将反向学习机制应用于改进NSGA-II算法中的精英保留策略, 对精英种群中的部分个体生成反向解, 通过生成更好的个体作为父代种群, 以提高算法的多样性和收敛性.

假设 $\pi = (x_1, x_2, \dots, x_n)$ 是 N 维解空间上的一个解向量, 对每一维上的 x_i 均有 $x_i \in [l_i, u_i]$, 则反向解 $\pi' = (x'_1, x'_2, \dots, x'_n)$ 中第 i 维通过下式定义:

$$x'_i = (l_i + u_i) - x_i. \quad (17)$$

考虑到算法后期种群中的大部分个体趋向于最优解, 使用OBL进行全局搜索的效益不大且影响算法运行速度, 根据算法当前所处的迭代代数, 以一定的比例 $\text{Pr}(i)$ 选择精英种群 P 中前 N_{opp} 个个体, $\text{Pr}(i)$ 随着迭代次数的增加而递减, 即

$$\text{Pr}(i) = P_l + (P_u - P_l) \cdot (\text{GEN} - i) / \text{GEN}. \quad (18)$$

其中: GEN为最大迭代次数, i 为当前迭代次数, P_u 和 P_l 分别为 $\text{Pr}(i)$ 的最大值和最小值. 通过对比原个体与所生成的反向个体, 选择最优的 N 个个体进入下一次迭代过程, 具体步骤如下.

step 1: 读取当前迭代次数, 根据式(18)获取需要生成的反向个体个数 N_{opp} .

step 2: 根据式(17), 生成精英种群 P 中前 N_{opp} 个个体 π_i 的反向个体 π'_i .

step 3: 若生成的反向个体为不可行解, 则将 π'_i 修复为可行解.

step 4: 若原个体 π_i 不支配反向个体 π'_i , 则将反向解 π'_i 加入反向种群 H .

step 5: 结合种群 H 与 P , 进行快速非支配排序与拥挤度计算, 选取前 N 个个体作为新种群.

2.4 基于强化学习的邻域搜索策略

强化学习^[21](reinforcement learning, RL) 的思想是智能体 Agent 在与环境互动中使用学习策略选择合适的行动, 从而获得最大化的回报. Q -learning^[22] 是一种典型的RL算法, 它通过建立一个状态-动作对表格 (Q 表), 将状态与动作的不断改变产生新的 Q 值作为学习经验保存到 Q 表中.

邻域搜索是求解组合优化问题的一种简单有效的元启发式算法^[23]. 为提高NSGA-II算法后期的收敛性, 本文设计了一种基于 Q -learning 的邻域搜索策略. 由于种群中非支配解的质量会随着迭代过程不断改变, 将非支配解集的评价指标作为强化学习的环境状态. 超体积 (hypervolume, HV)^[24] 度量目标空间中 Pareto 解集覆盖的支配解区域, 能够同时反映出解集的收敛性和多样性, HV 值越大算法综合性能越好. Spacing 度量每个解到其他解的最小距离的标准差, Spacing 值越小解集越均匀. 计算两个指标当前代与上一代的差值 ΔHV 和 $\Delta\text{Spacing}$, 以如下方式将状态空间分为4个部分. $S_1 : \Delta\text{HV} > 0, \Delta\text{Spacing} < 0$; $S_2 : \Delta\text{HV} > 0, \Delta\text{Spacing} \geq 0$; $S_3 : \Delta\text{HV} \leq 0, \Delta\text{Spacing} < 0$; $S_4 : \Delta\text{HV} \leq 0, \Delta\text{Spacing} \geq 0$.

针对多目标优化下的GIS-VSAM问题, 对工件、机器和AGV三部分各设计2种邻域搜索算子, 如图5所示, 其中 $1 \leq n \leq N$ (工件数). 通过对各部分任意选择一种邻域搜索算子, 组成 $2^3 = 8$ 种邻域结构的方式构成 Q -learning 中的动作集.

O_1 : 随机选取 n 个相邻位置工序, 倒序后插入任意位置; O_2 : 随机删除 n 个工序, 然后按删除的顺序依次插入到工序编码段中; M_1 : 将任意 n 个工序使用的机器替换为可选机器中加工时间最少的机器;

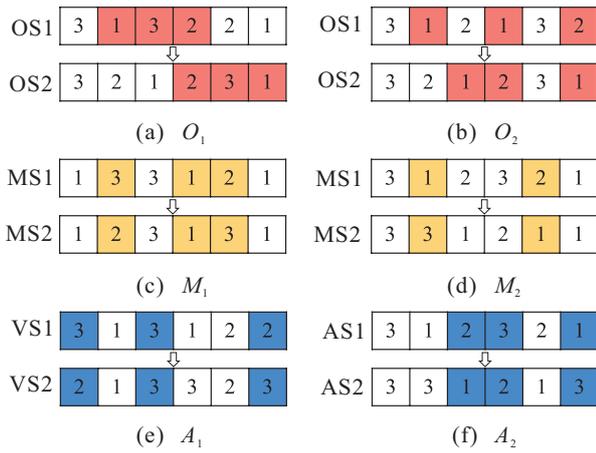


图5 不同邻域结构

M_2 : 将任意 n 个工序使用的机器替换为可选机器中能耗最小的机器; A_1 : 选择 VS 上的 n 个速度进行随机互换; A_2 : 选择充电次数最多的一个 AGV, 随机选取其负责的 n 道工序, 使用其他可选 AGV 运输。

使用 ϵ -greedy 策略选择一个动作 a_t , 状态 S_t 执行动作 a_t 后, Q -learning 对动作 a_t 的表现进行评估并给出一个奖赏值 R . 本文通过判断动作 a_t 执行后所转移的状态设置奖赏函数, 当所转移状态为 (S_1, S_2, S_3, S_4) 时, 奖赏值 R 分别为 $(1, 0, 0, -1)$. 再根据式 (20) 更新 $Q(S_t, a_t)$ 并加入到 Q 表中, 有

$$Q(S_t, a_t) = \alpha[R + \gamma \max_a Q(S_{t+1}, a)] + (1 - \alpha) \cdot Q(S_t, a_t). \quad (19)$$

其中: $Q(S_t, a_t)$ 为当前 S_t 状态下采取动作 a_t 产生的 Q 值, α 为学习率, γ 为折扣因子, $\max_a Q(S_{t+1}, a)$ 为下一个状态 S_{t+1} 采取所有动作 a 的最高的 Q 值。

2.5 算法描述

基于以上主要流程, 对于 GIS-VSAM 问题提出的 NSGA-II-HLS 算法描述总结如下。

- step 1: 四段式编码, 初始化种群、算法参数。
- step 2: 解码得到对应两个目标函数值, 进行快速非支配排序和拥挤距离计算。
- step 3: 对种群 NP 使用二元锦标赛选择, 对不同编码段进行不同类型的交叉操作, 得到交叉种群。
- step 4: 对种群交叉后产生的子代个体, 在不同编码段上执行对应的变异操作, 得到种群 P 。
- Step 5: 从精英种群 P 中选择最优的 N_{opp} 个个体执行 OBL 策略, 得到种群 R , 合并 P 和 R , 选出 N 个最优个体作为新的种群 NP。

step 6: 若算法在连续 3 次迭代内未产生优化效果, 则从当前代开始对种群 NP 使用基于 Q -learning 的邻域搜索策略, 否则转至 step 8。

step 7: 使用当前代的邻域结构进行邻域搜索并

更新 Q 表, 选出 N 个最优个体作为新的种群 NP。

step 8: 如果满足终止条件 (最大迭代次数), 则输出最终解, 否则, 返回种群 NP 到 step 3。

3 仿真实验

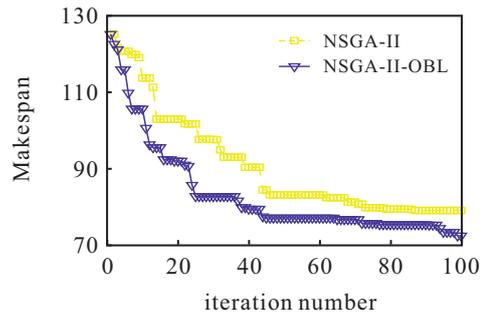
本文在 Brandimarte^[25] 标准算例的基础上进行扩展, 用以测试算法的效率和可靠性. 特别地, 在表 1 中对 AGV 充电与可变速度约束等数据进行说明. 问题规模用 $i \times j \times k$ 表示, 其中 i 代表工件数, j 代表机器数, v 代表 AGV 数量. 算法在 Matlab R2021b 软件上进行编程实现, 实验在平台 Intel (R) Core (TM) i5-8265U CPU @ 1.80 GHz, 内存 8.0 GB 上进行, 每个算例运行 20 次, 对优化结果进行统计分析. 通过田口实验方法确定的 NSGA-II-HLS 算法参数如下: 交叉概率 $P_c = 0.8$, 变异概率 $P_m = 0.1$, 反向生成概率下限 $P_l = 0.1$, 反向生成概率上限 $P_u = 0.4$, 探索率 $\epsilon = 0.7$, 学习率 $\alpha = 0.1$, 折扣因子 $\gamma = 0.7$.

表 1 AGV 相关数据说明

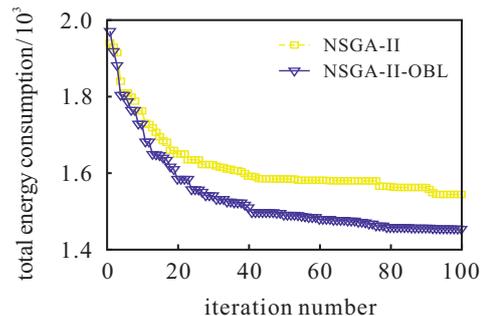
运输速度	空载能耗	负载能耗	充电速率	额定电量
0.5	0.5	1.5		
0.75	0.85	2.55	20	100
1	1.2	3.6		

3.1 改进策略有效性测试

为验证反向学习的有效性, 分别使用 NSGA-II 与 NSGA-II-OBL (基于反向学习改进精英保留策略的 NSGA-II) 求解 $10 \times 6 \times 3$ 算例. 从图 6 可以看出, NSGA-II-OBL 在求解最大完工时间与总能耗方面均



(a) Makespan 收敛曲线对比



(b) 总能耗收敛曲线对比

图 6 两种算法收敛曲线对比

优于NSGA-II,验证了该改进策略的有效性.

为验证 Q -learning 算法对 NSGA-II-HLS 中邻域搜索的改进效果,将 NSGA-II-HLS 与未使用 Q -learning 的算法 NSGA-II-LS 进行实验对比.在 NSGA-II-LS 中,算法使用随机的方式选择邻域结构,在参数设置以及其他改进策略方面均与 NSGA-II-HLS 无异.从表2可以看出,NSGA-II-HLS 在求解3种不同规模算例上的两个指标 HV 和 Spacing 都优于未使用 Q -learning 的 NSGA-II-LS.实验结果验证了 Q -learning 算法的有效性.

表2 两种算法 HV 和 Spacing 指标对比

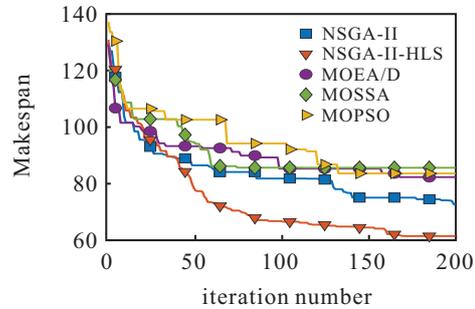
算法	HV		Spacing	
	NSGA-II-HLS	NSGA-II-LS	NSGA-II-HLS	NSGA-II-LS
10×6×3	0.8331	0.5844	0.0230	0.0525
15×8×4	0.9532	0.4730	0.1510	0.2843
20×10×5	0.9982	0.6228	0.0557	0.0698

3.2 GIS-VSAM算例测试

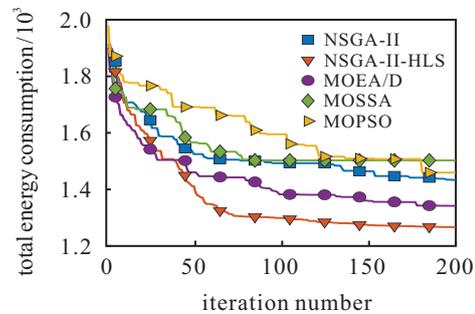
为进一步验证 NSGA-II-HLS 的有效性,将其与 NSGA-II、MOEA/D、MOSSA 和 MOPSO 进行比较.算法共同参数为:种群规模 100,迭代次数 200.算法各自参数设置如下: NSGA-II 交叉概率为 0.9,变异概率为 0.1; MOEA/D 邻域大小设置为 5; MOSSA 参考文献 [26] 设置自适应参数; MOPSO 惯性因子为 0.4,个体学习因子为 2,全局学习因子为 2.采用 HV 和反转世代距离 (inverted generational distance, IGD) 衡量算法的收敛性和多样性.其中,IGD 计算真实 Pareto 前沿上的最优解集到已知非支配解集之间的最小平均距离,IGD 的值越小算法的综合性能越好.本文选取 5 个算法产生的解集合中的非支配解作为近似 Pareto 前沿 (approximate Pareto frontier, APF) 以代替真实 PF,结果如表3所示.

表3展示了不同算法在不同规模算例中的 HV

和 IGD 结果.如表3所示,NSGA-II-HLS 在测试中均明显优于其他4种比较算法,验证了 NSGA-II-HLS 的有效性.图7、图8分别为5种算法求解 $10 \times 6 \times 3$ 算例两个目标值的收敛曲线与 Pareto 前沿分布.可以看出,NSGA-II-HLS 得到的非支配解的结果在收敛性、多样性和分布性上均明显优于其他算法.



(a) 不同算法Makespan收敛曲线



(b) 不同算法总能耗收敛曲线

图7 5种算法收敛曲线对比

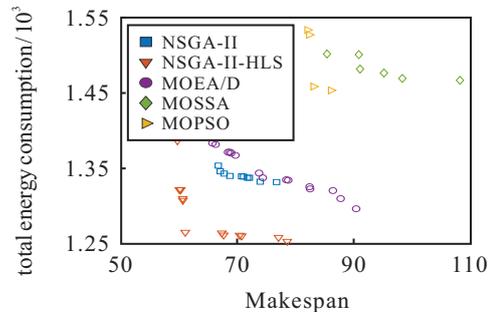


图8 5种算法Pareto前沿分布

表3 不同规模算例下各算法性能对比

算法	HV					IGD				
	NSGA-II-HLS	NSGA-II	MOEA/D	MOSSA	MOPSO	NSGA-II-HLS	NSGA-II	MOEA/D	MOSSA	MOPSO
10×6×3	1.1727	0.9009	0.7351	0.2690	0.1527	0.0276	0.1733	0.2122	0.8055	0.9554
10×6×4	1.2001	0.7769	0.5687	0.1206	0.1595	0	0.2702	0.4646	1.0318	0.9562
10×6×5	1.1816	0.8118	0.5994	0.0631	0.1340	0	0.2256	0.3565	1.0877	0.9412
15×8×3	1.2096	0.9030	0.8139	0.2428	0.0788	0	0.2319	0.2834	0.8956	1.1852
15×8×4	1.1481	0.8942	0.7698	0.2411	0.1073	0	0.1806	0.2719	0.8886	1.0684
15×8×5	1.1721	0.8368	0.5300	0.1580	0.0338	0.0028	0.1868	0.3848	0.8612	1.1615
20×10×3	1.1213	0.7841	0.8142	0.2566	0.1452	0	0.3159	0.2766	0.8873	1.0386
20×10×4	1.1875	0.5646	0.5127	0.2541	0.0828	0	0.2419	0.2814	0.8465	1.1248
20×10×5	1.1776	0.7921	0.5066	0.0791	0.0375	0.0161	0.3093	0.5301	1.1278	1.2479
20×15×3	1.0836	0.7456	0.8855	0.2156	0.0828	0.0281	0.3156	0.2445	1.0217	1.2005
20×15×4	1.1866	0.6498	0.7877	0.1525	0.0992	0	0.3214	0.2410	1.0073	1.1139
20×15×5	1.2095	0.6577	0.4800	0.1551	0.0863	0	0.3845	0.5553	0.9814	1.1465

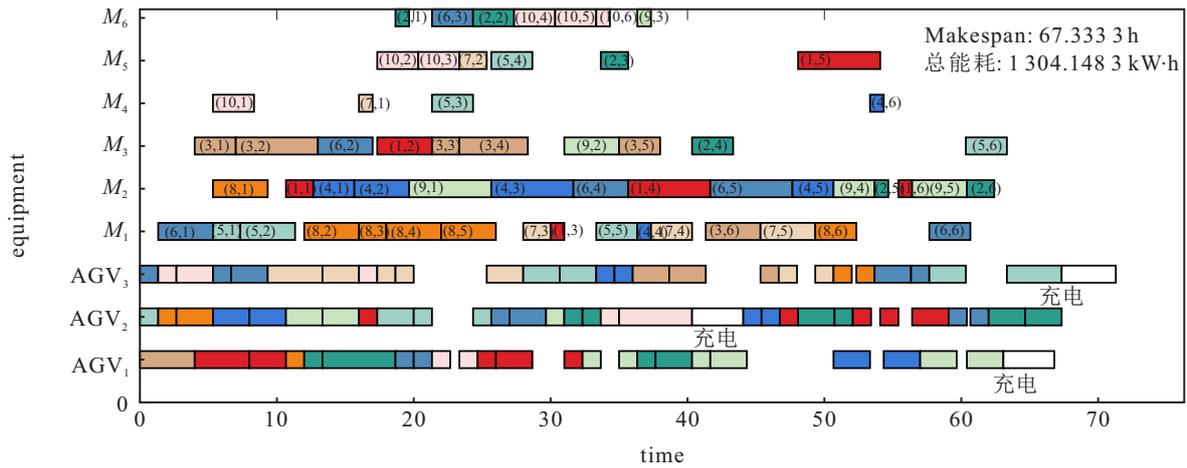


图9 不考虑可变速AGV的GIS问题甘特图

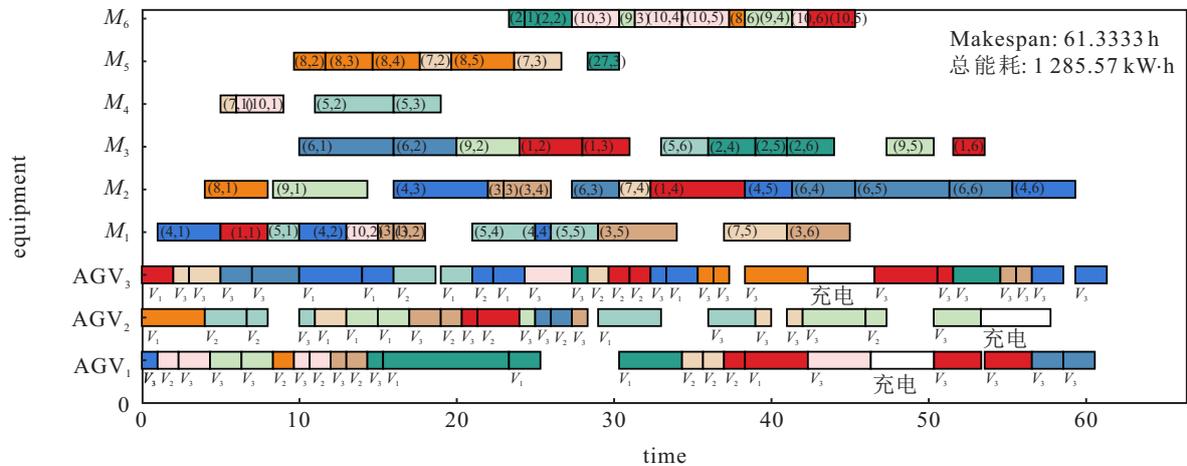


图10 GIS-VSAM问题甘特图

以 $10 \times 6 \times 3$ 算例为例,分别对不考虑可变速AGV的GIS问题以及GIS-VSAM问题进行求解.在GIS问题中,AGV以 V_2 速度恒定运行,其余参数相同.通过对比图9和图10可以看出,GIS-VSAM问题中的Makespan和总能耗均小于GIS问题,同时GIS-VSAM问题中的AGV空闲时间更少、运输任务更加紧凑.这是因为在GIS-VSAM问题中,AGV可以通过合理控制速度,在保证工件尽快开始加工的前提下,尽量使用更低能耗的速度.该结果表明了考虑可变速AGV对集成调度问题的有效性.

4 结论

本文研究了可变速AGV与机器的绿色集成调度问题,考虑了AGV的充电约束,以最小化最大完工时间与总能耗为目标,提出了NSGA-II-HLS算法.设计了基于工序、机器、AGV及其速度的四段式编码策略;对不同编码段设计了交叉变异算子;基于反向学习机制改进了精英保留策略,提升了算法的种群多样性;设计了邻域搜索算子组合,将Q-learning与邻域搜索相结合,提高了算法的局部搜索能力.最后通过

实验验证了算法的有效性和优越性.然而,本文所建立的集成调度模型未考虑AGV差异性以及路径冲突问题,与实际的生产系统之间仍存在差异,在后续研究工作中将着重考虑这些因素,并进一步改进求解算法,提高算法的效率和寻优能力.

参考文献(References)

[1] 王凌,王晶晶,吴楚格.绿色车间调度优化研究进展[J].控制与决策,2018,33(3):385-391.
(Wang L, Wang J J, Wu C G. Advances in green shop scheduling and optimization[J]. Control and Decision, 2018, 33(3): 385-391.)

[2] 李新宇,李昭甫,高亮.离散制造行业数字化转型与智能化升级路径研究[J].中国工程科学,2022,24(2):64-74.
(Li X Y, Li Z F, Gao L. Paths for the digital transformation and intelligent upgrade of China's discrete manufacturing industry[J]. Strategic Study of CAE, 2022, 24(2): 64-74.)

[3] Reddy B S P, Rao C S P. A hybrid multi-objective GA for simultaneous scheduling of machines and AGVs in FMS[J]. The International Journal of Advanced Manufacturing Technology, 2006, 31(5): 602-613.

- [4] Baldea M, Harjunoski I. Integrated production scheduling and process control: A systematic review[J]. *Computers & Chemical Engineering*, 2014, 71: 377-390.
- [5] Xue T F, Zeng P, Yu H B. A reinforcement learning method for multi-AGV scheduling in manufacturing[C]. 2018 IEEE International Conference on Industrial Technology. Lyon, 2018: 1557-1561.
- [6] Yang X R, Hu H T, Cheng C, et al. Automated guided vehicle (AGV) scheduling in automated container terminals (ACTs) focusing on battery swapping and speed control[J]. *Journal of Marine Science and Engineering*, 2023, 11(10): 1852.
- [7] 郭兴海, 计明军, 张卫丹. 融合多目标与速度控制的AGV全局路径规划[J]. *控制与决策*, 2020, 35(6): 1369-1376.
(Guo X H, Ji M J, Zhang W D. AGV global path planning integrating with the control of multi-objectives and speed[J]. *Control and Decision*, 2020, 35(6): 1369-1376.)
- [8] Liu Y, Ji S, Su Z, et al. Multi-objective AGV scheduling in an automatic sorting system of an unmanned (intelligent) warehouse by using two adaptive genetic algorithms and a multi-adaptive genetic algorithm[J]. *PLoS One*, 2019, 14(12): e0226161.
- [9] 李峥峰, 刘阳阳. 考虑充电的多AGV的作业车间调度问题[J]. *计算机集成制造系统*, 2021, 27(10): 2872-2879.
(Li Z F, Liu Y Y. Job shop scheduling considering multiple AGVs with charging[J]. *Computer Integrated Manufacturing Systems*, 2021, 27(10): 2872-2879.)
- [10] Li J X, Cheng W J, Lai K K, et al. Multi-AGV flexible manufacturing cell scheduling considering charging[J]. *Mathematics*, 2022, 10(19): 3417.
- [11] Dang Q V, Singh N, Adan I, et al. Scheduling heterogeneous multi-load AGVs with battery constraints[J]. *Computers & Operations Research*, 2021, 136: 105517.
- [12] Han W J, Xu J, Sun Z, et al. Digital twin-based automated guided vehicle scheduling: A solution for its charging problems[J]. *Applied Sciences*, 2022, 12(7): 3354.
- [13] Zhan X N, Xu L Y, Zhang J, et al. Study on AGVs battery charging strategy for improving utilization[J]. *Procedia CIRP*, 2019, 81: 558-563.
- [14] Li W, He L, Cao Y. Many-objective evolutionary algorithm with reference point-based fuzzy correlation entropy for energy-efficient job shop scheduling with limited workers[J]. *IEEE Trans Cybern*, 2022, 52(10): 10721-10734.
- [15] 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*, DOI: 10.13195/j.kzyjc.2023.0597.
(Wang Z S, Wu Y H. An improved differential evolution algorithm for simultaneous scheduling of machines and multi-load AGVs in an FMS[J]. *Control and Decision*, DOI: 10.13195/j.kzyjc.2023.0597.)
- [16] 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.)
- [17] 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.
- [18] Yuan M H, Li Y D, Zhang L Z, et al. Research on intelligent workshop resource scheduling method based on improved NSGA-II algorithm[J]. *Robotics and Computer-Integrated Manufacturing*, 2021, 71: 102141.
- [19] Amelian S S, Sajadi S M, Navabakhsh M, et al. Multi-objective optimization for stochastic failure-prone job shop scheduling problem via hybrid of NSGA-II and simulation method[J]. *Expert Systems*, 2022, 39(2): e12455.
- [20] Tizhoosh H R. Opposition-based learning: A new scheme for machine intelligence[C]. *International Conference on Computational Intelligence for Modelling, Control and Automation*. Vienna, 2005: 695-701.
- [21] Kaelbling L P, Littman M L, Moore A W. Reinforcement learning: A survey[J]. *Journal of Artificial Intelligence Research*, 1996, 4: 237-285.
- [22] Jang B, Kim M, Harerimana G, et al. Q-learning algorithms: A comprehensive classification and applications[J]. *IEEE Access*, 2019, 7: 133653-133667.
- [23] Zheng X C, Gong W Y. An improved artificial bee colony algorithm for fuzzy flexible job-shop scheduling problem[J]. *Control Theory & Applications*, 2020, 37(6): 1284-1292.
- [24] Li R, Gong W Y. An improved multi-objective evolutionary algorithm based on decomposition for bi-objective fuzzy flexible job-shop scheduling problem[J]. *Control Theory & Applications*, 2022, 39(1): 31-40.
- [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] Li B, Wang H L. Multi-objective sparrow search algorithm: A novel algorithm for solving complex multi-objective optimisation problems[J]. *Expert Systems with Applications*, 2022, 210: 118414.

作者简介

陈仁胜(1999—), 男, 硕士生, 主要研究方向为生产调度、智能优化算法, E-mail: chenrs99@163.com;

吴斌(1979—), 男, 教授, 硕士生导师, 主要研究方向为智能优化算法、系统建模与优化, E-mail: wubin@njtech.edu.cn;

闫飞一(1992—), 女, 博士, 讲师, 主要研究方向为生产系统建模与优化, E-mail: yanfy0727@163.com.