

DOI:10.3969/j.issn.1001-4551.2021.02.005

# 基于混合多目标遗传算法的 柔性作业车间调度问题研究\*

宋昌兴, 阮景奎\*, 王 宸

(湖北汽车工业学院 机械工程学院, 湖北 十堰 442002)

**摘要:**针对多目标柔性作业车间调度问题,建立了以最大完工时间、机器总负荷、瓶颈机器负荷为目标的调度数学模型,提出了一种基于混合多目标遗传算法(HMO-NSGA-II)的求解方法。首先,采用了全局选择和快速选择相结合的初始化方式,得到分布均匀的初始种群;其次,对其交叉变异算子进行了自适应改进,以提高对种群的搜索能力;接着,针对精英策略在维持种群多样性上的局限性,设计了一种精英保留机制,并引入改进的和声搜索算法,提高了精英库中的个体质量;最后,采用基准算例 Kacem 测试集、BRdata 数据集和实际生产案例进行了测试。研究表明:采用 HMO-NSGA-II 求解多目标柔性作业车间调度问题,求解精度高、收敛速度快,可在实际生产中为决策者提供可行、有效的调度方案,具有很好的实用价值。

**关键词:**自适应算子;NSGA-II;混合优化算法;柔性作业车间调度

中图分类号:TH165

文献标识码:A

文章编号:1001-4551(2021)02-0169-08

## Flexible job shop scheduling problem based on hybrid multi-objective genetic algorithm

SONG Chang-xing, RUAN Jing-kui, WANG Chen

(School of Mechanical Engineering, Hubei University of Automotive Technology, Shiyan 442002, China)

**Abstract:** Aiming at the problem of multi-objective flexible job shop scheduling, a mathematical model of scheduling with maximum completion time, total machine load, and bottleneck machine load as targets was established, and a solution method based on hybrid multi-objective non-dominated sorting genetic algorithm II (HMO-NSGA-II) was proposed. Firstly, the initial population with uniform distribution was obtained by combining global selection with fast selection. Secondly, the cross-mutation operator was improved adaptively to improve the searching ability of the population. Then, in view of the limitations of the elite strategy in maintaining population diversity, an elite retention mechanism was designed, and the improved harmony search algorithm was introduced to improve the quality of individuals in the elite library. Finally, the benchmark case Kacem test set, BRdata data set and actual production cases were used for testing. The results indicate that HMO-NSGA-II is used to solve multi-objective flexible job shop scheduling problems with high solution accuracy and fast convergence. The method can provide decision-makers with feasible and effective scheduling schemes in actual production, and has good practical value.

**Key words:** adaptive operator; non-dominated sorting genetic algorithm II (NSGA-II); hybrid optimization algorithm; flexible job shop scheduling

收稿日期:2020-06-15

基金项目:国家科技重大专项项目(2018ZX04027001);教育部人文社科资助项目(20YJJCZH150);湖北省教育厅科学技术资助项目(Q20181801);汽车动力传动与电子控制湖北省重点实验室基金资助项目(ZDK1201703);湖北汽车工业学院博士基金资助项目(BK201905)

作者简介:宋昌兴(1996-),男,湖北十堰人,硕士研究生,主要从事智能制造技术与装备方面的研究。E-mail:201810004@huat.edu.cn

通信联系人:阮景奎,男,博士,教授,硕士生导师。E-mail:gerry\_r@tom.com

## 0 引 言

车间生产调度是一种有效的调度方法与优化技术,企业在生产加工制造过程中对生产计划的合理编排,对提升企业生产效率、资源利用率、降低成本等起着关键作用。柔性作业车间调度问题(flexible job scheduling problem, FJSP)相比传统的作业车间调度问题引入了新的决策内容,包括对两个子问题的求解:一是为每道工序选择合适的机器,二是确定工序在机器上的起止加工时间,是更复杂的 NP-hard 问题<sup>[1]</sup>。

经过近 30 年的探索, FJSP 的研究取得了丰硕的成果。以遗传算法<sup>[2]</sup>、粒子群算法<sup>[3]</sup>、禁忌搜索算法<sup>[4]</sup>、模拟退火算法<sup>[5]</sup>等为代表的元启发式算法,在实际调度问题中都得到了成功的应用。

随着研究的深入,更加符合生产实际的多目标柔性作业车间调度(multi-objective FJSP, MO-FJSP)成为目前亟待解决的问题,近年来许多新算法的不断涌现,也为解决 MO-FJSP 提供了更多的选择方案。孟冠军等<sup>[6]</sup>提出了混合人工蜂群算法,在雇佣蜂、跟随蜂、侦查蜂 3 个阶段分别采用了不同的搜索方式,并引入禁忌搜索算法提升了获得最优解的概率;王英彦等<sup>[7]</sup>提出了改进的人工免疫算法,采用了模拟退火算法的 Metropolis 准则,在保证种群多样性的同时加快搜索效率;王思涵等<sup>[8]</sup>提出了改进的鲸鱼群算法,采用了协同搜索机制扩大鲸鱼个体的搜索范围,同时引入了变邻域搜索算法,提高了种群全局及局部搜索能力;曹磊等<sup>[9]</sup>提出了变邻域杂草优化算法,建立了具有 De-jong 学习效应的调度模型,在迭代后期引入了 N1、N2、N3 3 种邻域结构的变邻域搜索算法,并通过实验对算法的有效性进行了验证。

近年来的研究表明,针对单一算法的不足,混合算法将两种及以上算法的特性进行互补,相比单一算法具有更优的适应性和鲁棒性,是研究和解决 MO-FJSP 的重要方向。

本文将在非支配排序遗传算法的基础上引入和声搜索算法(harmony search, HS),提出一种混合多目标遗传算法(HMO-NSGA-II),并通过实验证明所提算法的可行性、有效性和实用性。

## 1 多目标柔性作业车间调度问题

### 1.1 问题描述

FJSP 可描述为:有  $n$  个工序顺序已知的工件  $J = \{J_1, J_2, \dots, J_n\}$  和  $m$  台不同机器  $M = \{M_1, M_2, \dots,$

$M_m\}$ ,其中工件  $J_i$  的每道工序  $O_{ij}$  可在相应的可选机器集  $M_{ij}$  中任意选择 1 台机器进行加工,且选择不同机器对应的加工时间可能不同。

在实际生产调度过程中,常会对多个不同的目标进行优化,如最大完工时间、生产成本、总拖期时间、机器总负荷、瓶颈机器负荷等。MO-FJSP 的求解即在满足时间、资源等约束条件下,采用最优化方法合理解决机器分配和工序排列 2 个子问题,分配各工序在机器上加工的起止时间,并实现对多个给定性能指标的优化,最终得到 1 组非劣调度方案。

### 1.2 数学模型

本文选择以下 3 个具有实际应用价值的目标,以降低机器生产负荷、缩短加工产品完工时间,从而提高生产效率。

(1) 最大完工时间最短:

$$f_1 = \min(\max(C_i)) \quad (1)$$

(2) 最大瓶颈机器负荷最小:

$$f_2 = \min(\max(\sum_{i=1}^n \sum_{j=1}^{n_i} P_{ijk} X_{ijk})) \quad (2)$$

(3) 最大机器负荷最小:

$$f_3 = \min(\sum_{i=1}^n \sum_{j=1}^{n_i} \sum_{k=1}^m P_{ijk} X_{ijk}) \quad (3)$$

式中: $C_i$ —工件  $J_i$  的完工时间; $n$ —工件总数; $n_i$ —工件  $i$  的总工序数; $m$ —机器总数; $P_{ijk}$ —工件  $i$  的第  $j$  道工序在机器  $k$  上的加工时间; $X_{ijk}$ —判断工件  $i$  的第  $j$  道工序是否在机器  $k$  上加工,是则取 1,否则取 0。

### 1.3 约束条件

一般的,在调度过程中 MO-FJSP 模型应满足以下约束条件<sup>[10,11]</sup>:

(1) 所有设备在  $t_0$  时刻均可用;

(2) 不同工件的加工优先级相同,一旦开始加工便不可中断;

(3) 同工件的不同工序之间,需满足工艺先后顺序约束,即:

$$S_{ij} + P_{ijk} X_{ijk} \leq S_{i(j+1)} \quad (4)$$

(4) 同时刻,同一台机器只能加工某工件的某一道工序,即:

$$S_{ij} + P_{ijk} \leq S_{hl} + L(1 - y_{ihlk}) \quad (5)$$

$$C_{ij} \leq S_{i(j+1)} + L(1 - y_{ji(j+1)k}) \quad (6)$$

(5) 同时刻,同一个工件的同一工序能且只能被一台机器加工,即:

$$\sum_{k=1}^m X_{ijk} = 1 \quad (7)$$

式中: $S_{ij}$ —工件  $i$  的第  $j$  道工序开始加工的时间; $P_{ijk}$ —工件  $i$  的第  $j$  道工序在机器  $k$  上的加工时间; $X_{ijk}$ —判断工件  $i$  的第  $j$  道工序是否在机器  $k$  上加工,是则取 1,否则取 0; $S_{i(j+1)}$ —工件  $i$  的第  $j+1$  道工序开始加工的时间; $S_{hl}$ —工件  $h$  的第  $l$  道工序开始加工的时间; $L$ —足够大的正数; $y_{ijhk}$ —判断在同机器  $k$  上,工件  $i$  的第  $j$  道工序较工件  $h$  的第  $l$  道工序是否先加工,若是值为 1,否则为 0; $C_{ij}$  为工件  $i$  的第  $j$  道工序停止加工的时间; $y_{ij(j+1)k}$ —在同机器  $k$  上,工件  $i$  的第  $j$  道工序较工件  $i$  的第  $j+1$  道工序是否先加工,若是值为 1,否则为 0。

## 2 混合多目标遗传算法设计

NSGA-II 在求解 MO-FJSP 上应用广泛,但由于交叉、变异操作以及精英策略等自身的局限性,在求解收敛速度以及保持种群多样性等方面存在缺陷。和声搜索算法实现原理简单,易于与其他算法结合,在新解的产生上能够全面利用存在的解向量,有效保证了种群的多样性,但比较依赖好的初始解。

因此,本文以 NSGA-II 为主体,把每代经过自适应交叉变异以及改进的精英保留策略等操作所得到的种群精英库,作为和声搜索算法的和声记忆库,并通过改进的和声搜索算法提高精英库种群质量,加快引导种群向 pareto 最优前沿收敛。

### 2.1 编码与解码

针对求解 MO-FJSP,本文采用基于机器、工序双层编码的方式,以对应机器分配和工序排列 2 个子问题。以一个  $4 \times 5$  的部分柔性作业车间调度问题 (partial FJSP, P-FJSP) 为例,工件 1 有 3 道工序,工件 2-4 分别有 2 道工序,各工件每道工序对应的可选机器集  $M_{ij}$  为 5 台机器中的部分机器。编码的长度等于各加工工件工序总数之和,工序编码部分为所有工序的一种排列,数字代表工件号,出现的次数代表相应工序,接着为每道工序从可选机器集中分配加工机器组成机器编码。

基于机器、工序的双层编码过程如图 1 所示。

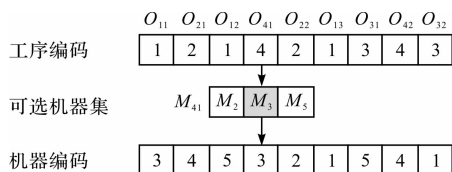


图 1 基于机器、工序的双层编码过程

解码采用插入式贪婪解码方法,将个体编码转换为调度甘特图。

加工起始时间计算如下式所示:

$$t_a = \max(C_{i(j-1)}, TS_{kr}) \quad (8)$$

$$t_a + P_{ijk} \leq TE_{kr} \quad (9)$$

$$t_b = \max(C_{i(j-1)}, LM_k) \quad (10)$$

式中: $t_a$ —工序的加工起始时间; $C_{i(j-1)}$ —工件  $i$  的第  $j-1$  道工序停止加工的时间; $TS_{kr}$ —机器  $k$  上第  $r$  个空闲时间段的开始时间; $P_{ijk}$ —工件  $i$  的第  $j$  道工序在机器  $k$  上的加工时间; $TE_{kr}$ —机器  $k$  上第  $r$  个空闲时间段的结束时间; $t_b$ —工序的加工起始时间; $LM_k$ —机器  $k$  上最后一道工序的结束时间。

具体步骤流程如下:

- (1) 找到当前同机器上所有加工空闲时间段  $[TS_{kr}, TE_{kr}]$ ;
- (2) 根据式(8) 计算各工序  $O_{ij}$  的加工起始时间  $t_a$ ;
- (3) 根据式(9) 判断当前工序能否在同机器上向前插入空闲间隙,若能则按起始时间  $t_a$  进行加工,若不能则根据式(10) 按  $t_b$  进行加工。

### 2.2 种群初始化

为提高种群初始解的质量,本文采用全局选择与快速选择相结合的方式,快速得到较好的初始解:

- (1) 通过全局选择将工序尽可能安排到当前累积加工负荷最小的加工机器上,以保证加工机器负荷平衡;
- (2) 通过快速选择,为各工序按概率随机分配或分配到加工该工序用时最短的机器上,以扩大搜索范围保证初始种群的多样性。

对应两种方式的种群产生比例分别设置为 0.4、0.6。

### 2.3 交叉与变异

#### 2.3.1 混合交叉方案

工序编码部分采用张超勇等<sup>[12]</sup> 提出的 IPOX 交叉。为适应 2.1 节中的编码策略,在交叉过程中机器编码也做出相应的变换。

基于工序编码的 IPOX 交叉如图 2 所示。

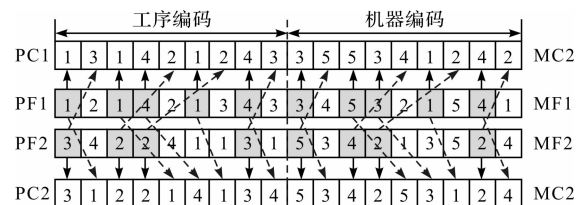


图 2 基于工序编码的 IPOX 交叉

机器编码部分采用改进的 MPX 交叉。首先从工件集  $J$  中随机选取若干工件如:  $\{J_2, J_3\}$  并记录其出现的位置;接着分别将父代机器编码  $MF_1, MF_2$  中非记录位置基因放入子代机器编码  $MC_1, MC_2$ ;最后交换

MF<sub>1</sub>、MF<sub>2</sub> 中记录位置的同工序机器基因并依次插入子代对应位置。

基于机器编码的改进 MPX 交叉如图 3 所示。

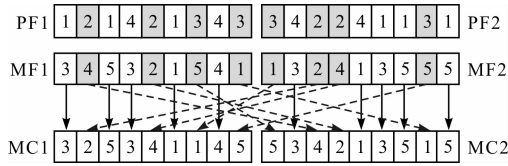


图 3 基于机器编码的改进 MPX 交叉

2.3.2 混合变异方案

工序编码部分采用插入变异。随机选取 2 个位置点 S<sub>1</sub>、S<sub>2</sub>, 假设 S<sub>1</sub> < S<sub>2</sub> 则将位于 S<sub>2</sub> 的工序、机器基因在 S<sub>1</sub> 处前插。

基于工序编码的插入变异如图 4 所示。

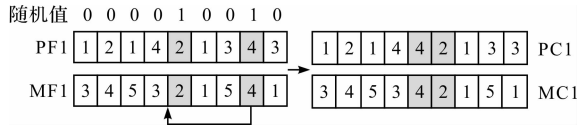


图 4 基于工序编码的插入变异

机器编码部分采用贪婪组合变异策略。随机产生一个位置点, 一是在该道工序的可选机器集中随机选出一台不重复的机器, 替换当前机器基因; 二是在其可选集中选择当前负荷最小的机器, 替换当前机器基因, 两种方式比例设置为 0.5、0.5。

基于机器编码的贪婪组合变异如图 5 所示。

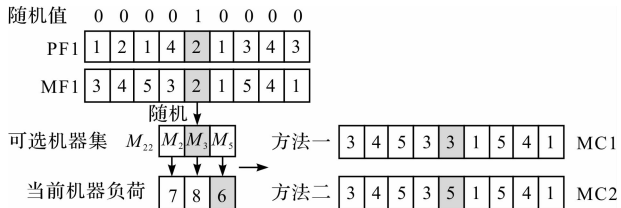


图 5 基于机器编码的贪婪组合变异

2.3.3 交叉、变异算子的自适应改进。

交叉、变异操作分别影响算法的全局、局部搜索性能。在种群进化前期, 优秀解距离 pareto 最优前沿较远, 为保证群体的参与性, 笔者采用较大的交叉概率 P<sub>c</sub>, 以提高全局搜索能力, 加快种群进化过程; 在种群进化中后期, 种群中优秀解的数量居多, 为防止算法陷入局部最优, 笔者采用较大的变异概率 P<sub>m</sub>, 以提高局部搜索能力。

交叉概率范围设置为 (0.4, 0.8), 变异概率范围设置为 (0.01, 0.2), 自适应算子计算公式如下:

$$P_c(i) = \min P_c + \frac{(\max P_c - \min P_c) \left(1 + \cos\left(\pi \frac{i}{gen}\right)\right)}{2} \quad (11)$$

$$P_m(i) = \min P_m + \frac{(\max P_m - \min P_m) \left(1 + \sin\left(\pi \frac{i}{gen} - \frac{\pi}{2}\right)\right)}{2} \quad (12)$$

式中: P<sub>c</sub>(i) — 第 i 代的交叉概率; P<sub>m</sub>(i) — 第 i 代的变异概率; gen — 总迭代数; max P<sub>c</sub> — 最大交叉概率; min P<sub>c</sub> — 最小交叉概率; max P<sub>m</sub> — 最大变异概率; min P<sub>m</sub> — 最小变异概率。

2.4 精英策略

针对传统精英策略保持种群多样性能力差、易于收敛于局部最优等问题, 本文对其进行了改进:

(1) 对各等级非支配层上的个体进行拥挤度排序;

(2) 为了保留尽可能多的精英解, 将位于 1 级非支配层上的所有个体直接选入子代中, 若其数量超过子代种群数量 N 的 0.15 倍, 则按照拥挤度排序选择前 0.15 N 个个体进入子代;

(3) 对其他等级非支配层上的个体, 分别按在种群大小为 N<sub>p</sub> 的合并种群中所占的比例, 选取 N<sub>i</sub> 个个体进入子代。

其中, 每级非支配层个体数量选择函数如下:

$$N_i = \begin{cases} |F_i|, & i = 1, |F_i| \leq 0.15N \\ 0.15N, & i = 1, |F_i| > 0.15N \\ |F_i| \frac{N}{N_p}, & i \geq 2 \end{cases} \quad (13)$$

式中: i — 非支配层等级; F<sub>i</sub> — 第 i 级非支配层; N<sub>i</sub> — 从第 i 级非支配层 F<sub>i</sub> 中选取的个体数量; |F<sub>i</sub>| — 第 i 级非支配层上的个体数; N<sub>p</sub> — 合并种群的大小; N — 子代种群的大小。

2.5 改进的和声搜索算法

标准和声搜索算法的核心在于新和声的产生以及和声记忆库的更新。为了更好地与 HMO-NSGA-II 相融合, 本文做出了以下改进:

(1) 将种群精英库与和声记忆库相结合, 以解决 HS 依赖较好初始解的弊端, 并采用改进的精英策略完成和声记忆库的更新;

(2) 在新和声的产生上采用优秀片段保留以及基于关键路径的变邻域搜索策略。

对应标准 HS 中对音调高低进行升降微调的 2 种方式, 具体方法如下:

(1) 从和声记忆库中随机抽出一个和声, 即:

$$l = l_{hs} \left(1 - \frac{index}{HMS}\right) \quad (14)$$

式中: l — 保留的优秀基因片段长度; l<sub>hs</sub> — 和声的长度, 即总工序数; index — 该和声在和声记忆库中的非支配

排序序号; $HMS$ —和声记忆库的大小。

按该和声的优秀度选择保留优秀基因片段的长度  $l$ ,对于剩余位置基因按 1.3 节中的约束条件随机补充。

(2) 变邻域搜索采用 2 种邻域结构。对于工序的调整采用张国辉等<sup>[13]</sup>提出的改进的 N5 邻域结构,即分别交换关键块首块、尾块中块尾、块首的两相邻工序,其他关键块只交换块首及块尾的两相邻工序,若两相邻工序为同工件则不交换;对于机器的调整,在关键路径中随机选择一道工序,将其当前使用机器替换为可选集中加工时间最短的机器。

### 2.6 步骤流程

(1) 设置迭代次数、交叉变异概率等初始化参数,采用 2.2 节中的方法生成包含  $N$  个个体的初始种群  $P_0$  作为第一代父种群  $P_i$ ,并对其进行非支配排序及拥挤度计算;

(2) 通过二元锦标赛机制,随机选择  $P_i$  中的个体进行 2.3 节中的自适应混合交叉、变异操作,得到子代种群  $Q_i$ ;

(3) 合并父代种群  $P_i$  和子代种群  $Q_i$  得到新种群  $R_i$ ,大小为  $N_p$ 。然后对  $R_i$  进行非支配排序并计算每个个体拥挤度大小,得到各级 pareto 非支配层  $F_1, F_2, \dots, F_n$ ;

(4) 采用 2.4 节中改进的精英策略保留各级非支配层中的优秀个体,形成大小为  $N$  的新父代种群  $P'_i$ ;

(5) 将种群精英库作为 HS 的和声记忆库,用精英库父代种群  $P'_i$  初始化和声记忆库 HM,和声记忆库大小  $HMS$  为  $N$ ;

(6) 产生一组随机数  $r_1, r_2$ ,其中  $r_1 \in [1, HMS], r_2 \in (0, 1)$ ;

(7) 从和声记忆库 HM 中选择第  $r_1$  条和声,并判断  $r_2$  是否小于和声微调概率  $PAR$ ,若成立则采用 2.5 节(1) 中的和声优秀片段保留策略对选择的和声进行微调,否则采用 2.5 节(2) 中的基于关键路径的变邻域搜索策略进行微调,得到新和声  $X_{new}$ ;

(8) 对和声库 HM 进行非支配排序和拥挤度计算,并采用 2.4 节中改进的精英策略更新和声记忆库 HM,形成大小为  $HMS$  的新父代种群  $P''_i$ ,从而改善精英库中的个体质量;

(9) 判断迭代次数  $j$  是否达到最大,若是则跳转至下一步,否则迭代次数加一跳转至(6) 进行下一轮循环;

(10) 判断迭代次数  $i$  是否达到最大,若是则结束迭代得到一组非劣调度方案,否则迭代次数加一跳转至(2) 进行下一轮循环。

HMO-NSGA-II 流程框架如图 6 所示。

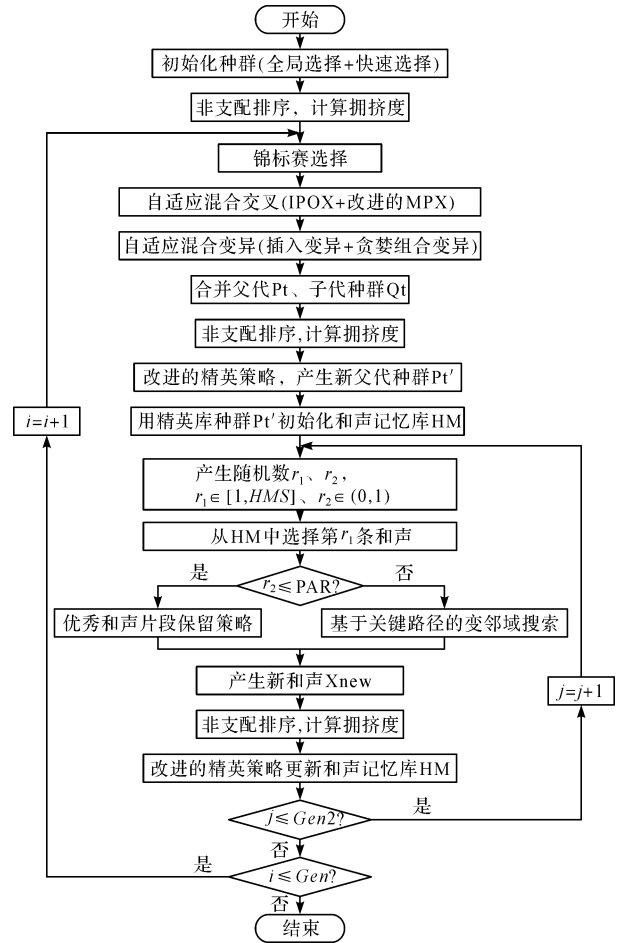


图 6 HMO-NSGA-II 流程框架

### 3 实验及结果分析

为验证本文提出的 HMO-NSGA-II 在 MO-FJSP 求解上的可行性、有效性和实用性,笔者选择通用国际标准算例和实际生产案例进行测试,其中,标准算例包括 Kacem 测试集<sup>[14]</sup> (8x8 (P-FJSP)、10x10 (T-FJSP)、15x10 (T-FJSP)) 和 BRdata 数据集<sup>[15]</sup> (MK01-MK10)。

算法采用 MATLAB R2018a 编程,运行环境为: CPU Intel i7-4558U 主频 2.80 GHz,内存 4 GB, Windows 7 操作系统的个人笔记本电脑。

测试目标函数选用 1.2 节中的式(1~3), HMO-NSGA-II 参数设置为:总迭代次数,种群大小,交叉概率范围(0.4,0.8),变异概率范围(0.01,0.2),和声搜索算法迭代次数,和声记忆库微调概率。

具体步骤为:

(1) 在 BRdata 数据集下,本文将 NSGA-II 与 HMO-NSGA-II 各运行 20 次后的非劣解按最大完工时间最小进行对比。

HMO-NSGA-II 与 NSGA-II 对比结果如表 1 所示。

表 1 HMO-NSGA-II 与 NSGA-II 对比结果

| 问题<br>类型 | $n \times m$ | $T$ | Brandimarte 方法 |        |        | NSGA-II |        |        |              |           | HMO - NSGA-II |        |        |              |           |
|----------|--------------|-----|----------------|--------|--------|---------|--------|--------|--------------|-----------|---------------|--------|--------|--------------|-----------|
|          |              |     | $f_1/$         | $f_2/$ | $f_3/$ | $f_1/$  | $f_2/$ | $f_3/$ | $\bar{f}_1/$ | $\sigma/$ | $f_1/$        | $f_2/$ | $f_3/$ | $\bar{f}_1/$ | $\sigma/$ |
|          |              |     | min            | min    | min    | min     | min    | min    | min          | min       | min           | min    | min    | min          | min       |
| MK01     | 10 × 6       | 55  | 42             | N/A    | N/A    | 36      | 167    | 36     | 36.6         | 0.5       | 36            | 167    | 36     | 36.3         | 0.4       |
| MK02     | 10 × 6       | 58  | 32             | N/A    | N/A    | 30      | 142    | 30     | 30.3         | 1.5       | 27            | 147    | 27     | 27.6         | 0.9       |
| MK03     | 15 × 8       | 150 | 211            | N/A    | N/A    | 213     | 844    | 213    | 214.5        | 2.6       | 204           | 850    | 204    | 204.9        | 1.7       |
| MK04     | 15 × 8       | 90  | 81             | N/A    | N/A    | 67      | 357    | 62     | 67.8         | 1.5       | 63            | 360    | 60     | 63.5         | 1.1       |
| MK05     | 15 × 4       | 106 | 186            | N/A    | N/A    | 185     | 676    | 185    | 186.5        | 2.7       | 174           | 683    | 174    | 175.1        | 1.7       |
| MK06     | 10 × 15      | 150 | 86             | N/A    | N/A    | 70      | 407    | 59     | 71.7         | 1.7       | 66            | 398    | 62     | 66.7         | 1.2       |
| MK07     | 20 × 5       | 100 | 157            | N/A    | N/A    | 161     | 660    | 161    | 162          | 1.8       | 148           | 673    | 144    | 148.9        | 0.9       |
| MK08     | 20 × 10      | 225 | 523            | N/A    | N/A    | 542     | 2 509  | 542    | 543.8        | 3.6       | 524           | 2 519  | 524    | 524.9        | 2.8       |
| MK09     | 20 × 10      | 240 | 369            | N/A    | N/A    | 335     | 2 244  | 331    | 336.4        | 1.7       | 315           | 2 300  | 294    | 315.6        | 1.3       |
| MK10     | 20 × 15      | 240 | 296            | N/A    | N/A    | 254     | 1 916  | 207    | 254.5        | 0.7       | 229           | 1 974  | 209    | 229.7        | 0.8       |

$T$ —问题的总工序数; $\bar{f}_1$ —目标的平均值; $\sigma$ —目标 $f_1$ 的标准差;“N/A”—无数据

从表 1 可以看出:相对于 Brandimarte 方法而言,两种算法都取得了较好的测试结果,但随着问题规模的增大,NSGA-II 较 HMO-NSGA-II 求得非劣解的质量有所下降;在 MK01 问题上两者所得的值相同,但 HMO-NSGA-II 的均值及标准差均低于 NSGA-II;在 MK02-MK10 问题上 HMO-NSGA-II 所得的非劣解均优

于 NSGA-II,且 20 次运算结果波动不大,在求解质量上表现出很好的稳定性。其中,以 MK01 问题为例,两算法某次运行结果的迭代过程及非劣解分布对比分析如下。

HMO-NSGA-II 与 NSGA-II 求解 MK01 问题过程对比如图 7 所示。

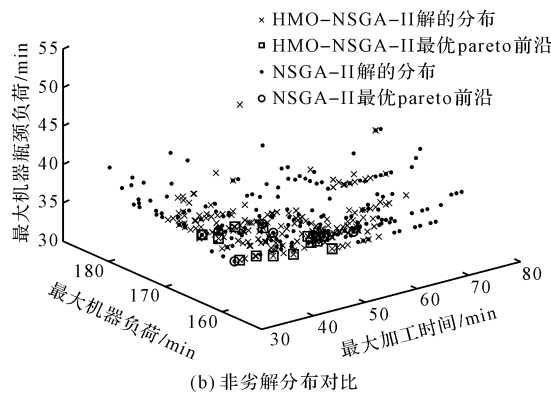
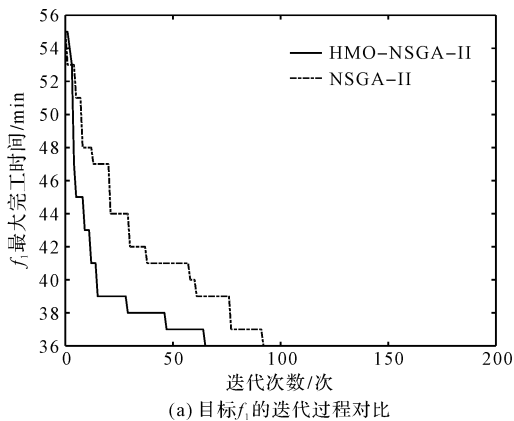


图 7 HMO-NSGA-II 与 NSGA-II 求解 MK01 问题过程对比

从图 7 可以看出:在得到相同解的情况下 HMO-NSGA-II 能够稳定、快速的找到最优解,具有很好的收敛性;HMO-NSGA-II 最终得到 13 个非支配解,NSGA-II 最终得到 8 个非支配解,HMO-NSGA-II 得到的非支配解个数多、分布范围广且均匀,体现出较好的全局、局部搜索性能。

(2)在 Kacem 测试集下,本文将 HMO-NSGA-II 与孟冠军提出的混合人工蜂群算法(ABC + TS)、曹磊提出的变邻域杂草优化算法(IWO + VNS)、Kacem 提出

的局部逼近与控制遗传算法的混合算法(AL + CGA)、喻明让提出的离散粒子群算法(PDPSO)<sup>[16]</sup>的运行结果进行了对比。

各算法运行结果对比如表 2 所示。

从表 2 可以看出:在 Kacem8 × 8 问题上曹磊提出的 IWO + VNS 得到了更多的非劣解,但均未达到最优值;对于其余问题,与其他算法相比 HMO-NSGA-II 在求解精度上都可以达到最优值标准,在求得的非支配解个数上均不劣于其他算法。

表 2 各算法运行结果对比

| 问题类型    | T  | 解编号 | AL + CGA |        |        | ABC + TS |        |        | IWO + VNS |        |        | PDPSO  |        |        | HMO-NSGA-II |        |        |
|---------|----|-----|----------|--------|--------|----------|--------|--------|-----------|--------|--------|--------|--------|--------|-------------|--------|--------|
|         |    |     | $f_1/$   | $f_2/$ | $f_3/$ | $f_1/$   | $f_2/$ | $f_3/$ | $f_1/$    | $f_2/$ | $f_3/$ | $f_1/$ | $f_2/$ | $f_3/$ | $f_1/$      | $f_2/$ | $f_3/$ |
|         |    |     | min      | min    | min    | min      | min    | min    | min       | min    | min    | min    | min    | min    | min         | min    | min    |
| 8 × 8   | 27 | 1   | 15       | 79     | N/A    | 14       | 77     | 12     | 15        | 75     | 13     | 14     | 77     | 12     | 14          | 72     | 11     |
|         |    | 2   | 16       | 75     | N/A    | 15       | 75     | 15     | 16        | 73     | 13     | 15     | 75     | 12     | 14          | 72     | 12     |
|         |    | 3   |          |        |        | 15       | 75     | 12     | 17        | 78     | 11     |        |        |        | 15          | 70     | 11     |
|         |    | 4   |          |        |        |          |        |        | 17        | 76     | 12     |        |        |        |             |        |        |
|         |    | 5   |          |        |        |          |        |        | 18        | 75     | 12     |        |        |        |             |        |        |
| 10 × 10 | 30 | 1   | 7        | 45     | 5      | 7        | 41     | 6      | 9         | 41     | 7      | 7      | 43     | 5      | 7           | 42     | 6      |
|         |    | 2   |          |        |        | 7        | 41     | 7      | 8         | 43     | 5      | 8      | 42     | 5      | 7           | 43     | 5      |
|         |    | 3   |          |        |        | 7        | 42     | 6      | 7         | 42     | 6      | 7      | 42     | 6      | 8           | 41     | 7      |
| 15 × 10 | 56 | 1   | 11       | 91     | 11     | 11       | 93     | 11     | 11        | 91     | 11     | 11     | 91     | 11     | 11          | 91     | 11     |
|         |    | 2   |          |        |        | 12       | 91     | 11     | 12        | 93     | 10     | 11     | 93     | 10     | 11          | 93     | 10     |
|         |    | 3   |          |        |        | 12       | 96     | 12     |           |        |        |        |        |        | 12          | 94     | 10     |

T—问题的总工序数;“N/A”—无数据

Kacem15 × 10 问题调度甘特图如图 8 所示。

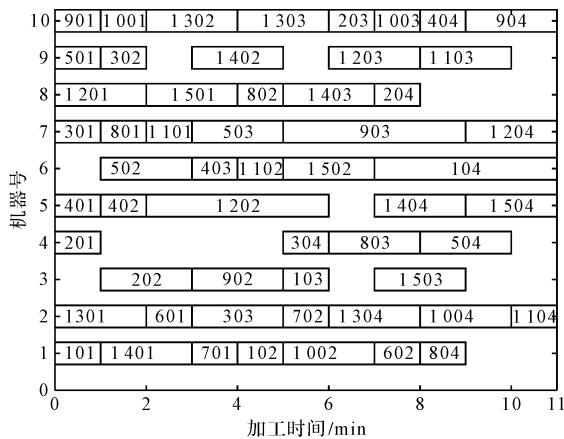


图 8 Kacem15 × 10 问题调度甘特图

(3)为进一步验证 HMO-NSGA-II 算法求解实际问题的实用性,本文选择某汽车线束生产企业的实际调度案例进行测试。汽车线束生产工艺流程包括下线、压接、内联、分装、总装等,以某个具有 6 条支路的线束产品为例,对应工艺流程中 6 条分装支路相当于 6 个工作件,其中工件  $J_1$ 、 $J_3$ 、 $J_4$  分别包含工序序号为 1、2、3、4 的 4 道工序,工件  $J_2$ 、 $J_5$ 、 $J_6$  分别包含工序序号为 1、2、4 的 3 道工序,该产品产量为 50 根。

线束产品加工生产数据如表 3 所示。

表 3 线束产品加工生产数据

| 工序序号 | 工序名称 | 可选设备       | 对应加工时间/(min · 次 <sup>-1</sup> ) |
|------|------|------------|---------------------------------|
| 1    | 下线   | $M_1, M_2$ | 0.03, 0.05                      |
| 2    | 压接   | $M_3, M_4$ | 0.02, 0.03                      |
| 3    | 内联   | $M_5, M_6$ | 0.06, 0.08                      |
| 4    | 分装   | $M_7, M_8$ | 0.04, 0.05                      |

根据表 3 中的实际生产数据,本文采用 NSGA-II 和 HMO-NSGA-II 进行调度求解,并将求解结果与传统的人工调度方法进行对比。

各调度方法求解实际问题的结果对比如表 4 所示。

表 4 各调度方法求解实际问题的结果对比

| 评价指标             | 人工调度 | NSGA-II | HMO-NSGA-II |
|------------------|------|---------|-------------|
| $f_1/\text{min}$ | 72   | 42      | 39          |
| 非劣解个数/个          | N/A  | 9       | 14          |
| 调度用时/s           | 540  | 16      | 24          |
| 对人依赖程度           | 大    | 小       | 小           |

“N/A”—无数据

从表 4 可以看出:人工调度方法较两种智能调度算法对人的依赖程度大,并且调度与生产加工所用时间长,调度效果比较差;HMO-NSGA-II 相对于 NSGA-II 虽调度用时稍长,但在生产加工用时和非劣解个数的性能指标上优势明显,该汽车线束企业可根据实际情况,从所得非劣解集中选择合适的解用于指导生产。

汽车线束实际调度方案甘特图如图 9 所示。

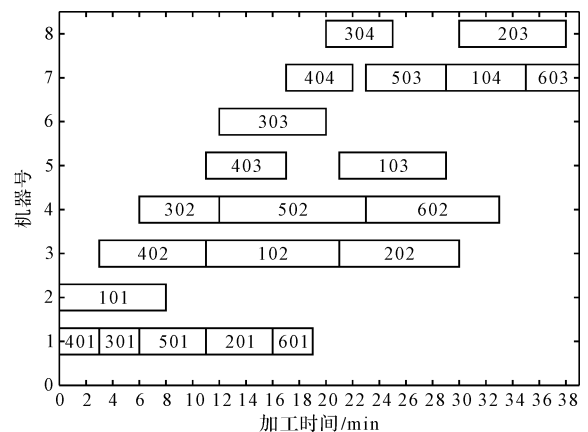


图 9 汽车线束实际调度方案甘特图

综上所述,本文提出的 HMO-NSGA-II 与单一算法相比求解精度高、收敛速度快,具有很好的全局、局部搜索能力;在测试选取的不同规模基准实例与实际生产案例上都取得了最优的结果,具有很好的适应性和

鲁棒性。由此可以证明, HMO-NSGA-II 用于求解 MO-FJSP 是可行、有效的, 具有很好的实用价值。

## 4 结束语

(1) 本文对 NSGA-II 的种群初始化、交叉变异方式、精英保留策略等进行了改进, 有效保证了种群多样性, 提高了算法的全局、局部搜索能力;

(2) 以 NSGA-II 为主和声搜索算法为辅, 充分结合两种算法的优点提出了混合多目标遗传算法 HMO-NSGA-II, 提高了求解质量并通过测试国际标准算例 Kacem 测试集、BRdata 数据集和实际生产案例, 验证了算法求解 MO-FJSP 的可行性和有效性, 具有很好的实用价值;

(3) 接下来, 笔者将深入研究 MO-FJSP 自身特征领域, 在不同目标需求下继续探索混合多目标遗传算法的改进策略, 以不断完善、提高算法的求解质量和效率。

### 参考文献 (References):

- [1] 彭建刚, 刘明周, 张铭鑫, 等. 多目标柔性作业车间调度算法研究综述[J]. 中国机械工程, 2014, 25(23): 3244-3254.
- [2] 刘 胜, 于海强. 基于改进遗传算法的多目标 FJSP 问题研究[J]. 控制工程, 2016, 23(6): 816-822.
- [3] SONG H, NA T, YAN W, et al. Multi-objective flexible job-shop scheduling problem using modified discrete particle swarm optimization [J]. **Springer International Publishing**, 2016, 5(1): 1432.
- [4] LI X Y, GAO L. An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem[J]. **International Journal of Production Economics**, 2016, (174): 93-110.
- [5] CRUZ-CHÁVEZ M A, MARTÍNEZ-RANGEL M G, CRUZ-ROSALES M H. Accelerated simulated annealing algorithm applied to the flexible job shop scheduling problem[J]. **In-**

**ternational Transactions in Operational Research**, 2017, 24(5): 1119-1137.

- [6] 孟冠军, 杨大春, 陶细佩. 基于混合人工蜂群算法的多目标柔性作业车间调度问题研究[J]. 计算机应用研究, 2019, 36(4): 972-974, 979.
- [7] 王思涵, 黎 阳, 李新宇. 基于鲸鱼群算法的柔性作业车间调度方法[J]. 重庆大学学报, 2020, 43(1): 1-11.
- [8] 王英彦, 曾 瑞, 方锦明, 等. 求解柔性车间作业调度的改进人工免疫算法[J]. 西南师范大学学报: 自然科学版, 2016, 41(11): 62-67.
- [9] 曹 磊, 叶春明, 黄 霞. 基于员工学习行为的多目标柔性车间调度[J]. 计算机集成制造系统, 2018, 24(8): 2023-2034.
- [10] 赵 敏, 殷 欢, 孙棣华, 等. 基于改进人工鱼群算法的柔性作业车间调度[J]. 中国机械工程, 2016, 27(8): 1059-1065.
- [11] 徐本柱, 吉 靖, 费晓璐. 柔性作业车间中基于工序分批的调度问题与求解[J]. 中国机械工程, 2016, 27(23): 3221-3229.
- [12] 张超勇, 董 星, 王晓娟, 等. 基于改进非支配排序遗传算法的多目标柔性作业车间调度[J]. 机械工程学报, 2010, 46(11): 156-164.
- [13] 张国辉. 柔性作业车间调度方法研究[D]. 华中科技大学, 2009.
- [14] KACEM I, HAMMADI S, BORNE P. Approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems[J]. **IEEE Transactions on Systems Man and Cybernetics Part C-Applications and Reviews**, 2002, 32(1): 1-13.
- [15] BRANDIMARTE P. Routing and scheduling in a flexible job shop by tabu search[J]. **Annals of Operations Research**, 1993, 41(3): 157-183.
- [16] 喻明让, 陈 云, 张志刚. 离散粒子群优化算法求解多目标柔性作业车间调度问题[J]. 制造技术与机床, 2019(1): 159-165.

[编辑: 杨骏泽]

### 本文引用格式:

宋昌兴, 阮景奎, 王 宸. 基于混合多目标遗传算法的柔性作业车间调度问题研究[J]. 机电工程, 2021, 38(2): 169-176.

SONG Chang-xing, RUAN Jing-kui, WANG Chen. Flexible job shop scheduling problem based on hybrid multi-objective genetic algorithm[J]. *Journal of Mechanical & Electrical Engineering*, 2021, 38(2): 169-176.

《机电工程》杂志: <http://www.meem.com.cn>