

# 控制与决策

Control and Decision

## 预制构件并行生产线资源配置与生产调度集成优化

熊福力, 张杏, 曹劲松, 袁子阳, 储梦伶, 杜瑶

引用本文:

熊福力, 张杏, 曹劲松, 袁子阳, 储梦伶, 杜瑶. 预制构件并行生产线资源配置与生产调度集成优化[J]. *控制与决策*, 2022, 37(9): 2399–2406.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2021.0113>

---

## 您可能感兴趣的其他文章

### Articles you may be interested in

#### [基于混合元启发式算法的订单分批问题](#)

Hybrid metaheuristic algorithm for order batching problem

*控制与决策*. 2022, 37(8): 2110–2118 <https://doi.org/10.13195/j.kzyjc.2021.0364>

#### [基于双种群模糊引力搜索算法的舰载机甲板作业调度](#)

Flight deck operations scheduling based on dual population fuzzy gravitational search algorithm

*控制与决策*. 2021, 36(11): 2751–2759 <https://doi.org/10.13195/j.kzyjc.2020.0523>

#### [面向建材装备集团制造的分布式多项目资源调度](#)

Distributed multi-project resource scheduling oriented to manufacturing of building materials equipment group

*控制与决策*. 2021, 36(9): 2133–2142 <https://doi.org/10.13195/j.kzyjc.2019.1802>

#### [顺序依赖的调整时间和拖期的无缝钢管热轧批量调度算法](#)

Hot-rolled batch scheduling algorithm for seamless steel tube with sequence-dependent setup times and tardiness

*控制与决策*. 2021, 36(2): 505–512 <https://doi.org/10.13195/j.kzyjc.2019.0723>

#### [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>

# 预制构件并行生产线资源配置与生产调度集成优化

熊福力<sup>†</sup>, 张 杏, 曹劲松, 袁子阳, 储梦伶, 杜 瑶

(西安建筑科技大学 信息与控制工程学院, 西安 710055)

**摘 要:** 并行生产线和特定工序生产资源共享模式可以显著改善客户满意度并节约成本. 针对预制构件并行生产线资源配置与生产调度集成优化问题, 基于分解策略和交替迭代优化思想, 提出一种交替式混合果蝇-禁忌搜索算法(AHFOA\_TS)以最小化拖期惩罚费用. 首先, 通过快速启发式方法产生一较好初始解; 然后, 固定资源配置方案, 为提高算法局部搜索能力, 通过集成多种局部搜索方式, 设计一种离散果蝇优化算法优化订单指派及调度方案; 最后, 固定订单指派及调度方案, 为减少无效搜索次数, 设计一种基于双层变异算子和精英劣解交叉策略的混合禁忌搜索算法以优化资源配置方案, 如此两个阶段交替运行直至满足终止条件. 此外, 设计 4 种基于交替搜索框架的智能优化算法用于比较. 计算结果表明, AHFOA\_TS 算法能够更有效求解预制构件生产线资源配置和生产调度集成优化问题.

**关键词:** 预制构件并行生产线; 有限生产资源; 拖期惩罚; 果蝇优化算法; 禁忌搜索; 集成优化

中图分类号: TP8

文献标志码: A

DOI: 10.13195/j.kzyjc.2021.0113

引用格式: 熊福力, 张杏, 曹劲松, 等. 预制构件并行生产线资源配置与生产调度集成优化[J]. 控制与决策, 2022, 37(9): 2399-2406.

## Integrated resource allocation and production scheduling for parallel concrete precast production lines

XIONG Fu-li<sup>†</sup>, ZHANG Xing, CAO Jin-song, YUAN Zi-yang, CHU Meng-ling, DU Yao

(School of Information and Control Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, China)

**Abstract:** Customer satisfaction and production cost can be improved significantly by the parallel production and resources sharing mode. To deal with integrated resources configuration and production scheduling problem for parallel precast production lines, based on the decomposition strategy and the alternative iterative optimization framework, an alternative hybrid fruit fly and tabu search algorithm (AHFOA\_TS) is proposed to minimize total weighted tardiness. In the algorithm, a good initial solution is obtained using a fast heuristic method at first. Then, based on the characteristics of the problem, to enhance the local search ability, an effective discrete fruit fly optimization algorithm is designed to optimize order assignment and scheduling by integrating multiple local search methods. Subsequently, to reduce the number of invalid searches, a hybrid tabu search algorithm based on double-mutation operator and elite inferior solution crossover strategy is designed to optimize the allocation scheme of limited production resources in specific processes. The previous two phases run alternately until the termination criterion is met. For comparison, four hybrid intelligent algorithms are designed based on the alternative search framework. The calculation results verify that, the AHFOA\_TS performs better than the other four algorithms for dealing with the proposed integrated optimization problem.

**Keywords:** parallel precast production lines; limited production resources; tardiness penalty; fruit fly optimization; tabu search; integrated optimization

## 0 引 言

装配式建筑现已成为国家和行业大力发展的新型建筑模式, 具有施工时间短、生产效率高、环境污染小等特点. 预制构件制造作为装配式建筑中极为

重要的一环, 其生产效率直接决定了装配式建筑供应链系统的整体运作效率. 随着市场竞争的不断加剧, 很多预制构件制造商纷纷采用并行生产线模式组织生产以提高生产效率和改善客户满意度. 但由于特

收稿日期: 2021-01-21; 录用日期: 2021-07-05.

基金项目: 国家自然科学基金项目(61473216); 陕西省自然科学基金项目(2020JM-489); 陕西省教育厅自然科学基金项目(17JK0459); 西安建筑科技大学自然科学基金基础研究项目(ZR18049).

<sup>†</sup>通讯作者. E-mail: xiongfuli@xauat.edu.cn.

定工序生产资源昂贵(如浇筑机)和特定工种劳动力成本较高(如瑕疵修整)等问题,使得多条生产线往往只能共享有限的高成本资源,因此,在实际预制构件生产管理过程中,不但要给出合理的订单调度方案,还要给出有限资源配置方案.二者相互影响,相互制约.无论是生产调度问题还是资源配置问题均为典型的NP-难问题.由此可见,在并行生产线生产模式下,研究预制构件有限资源配置和生产调度集成优化问题是非常必要而又具有一定挑战性的课题.

在预制构件生产调度方面,Chan等<sup>[1]</sup>以最小化总提前和拖期惩罚为目标,建立预制流水车间调度模型并设计一种遗传算法对问题进行求解. Benjaoran等<sup>[2]</sup>以最小化机器空闲、拖期以及makespan为目标建立了多目标定制化预制构件生产调度模型. Ko等<sup>[3]</sup>考虑了有限缓冲区预制构件调度问题,提出一种多目标遗传算法同时优化makespan和拖期惩罚. Yang等<sup>[4]</sup>针对预制构件并行流水车间调度问题提出一种遗传算法用于解决该问题. Wang等<sup>[5]</sup>通过集成模具制造、预制构件生产和运输等重要环节建立了预制构件供应链模型. Wang等<sup>[6]</sup>构建了一种预制构件生产两层级仿真-GA混合调度模型. Ma等<sup>[7]</sup>研究了一类预制构件并行生产线重调度问题. 需要指出的是,由于问题的复杂性,上述绝大多数预制构件调度问题研究中均采用GA等智能优化算法进行求解.

近年来,许多智能方法应用于生产调度问题,如模拟退火<sup>[8]</sup>、禁忌搜索<sup>[9]</sup>、蚁群算法<sup>[10]</sup>、粒子群优化<sup>[11]</sup>、人工蜂群算法<sup>[12]</sup>、果蝇优化算法(fruit fly optimization algorithm, FOA)<sup>[13]</sup>等. 其中FOA是一种新颖的群体智能优化算法,由于其控制参数少、寻优速度快以及较好的收敛效果已成功应用于生产调度优化领域<sup>[14-16]</sup>. 郑晓龙等<sup>[14]</sup>提出离散混合FOA求解置换流水车间调度问题,获得了比模拟退火更高质量的解. 王凌等<sup>[16]</sup>设计了一种采用双种群协同搜索策略的混合离散FOA算法,能够有效求解区间数分布式置换流水线调度问题.

上述研究为预制构件生产调度和约束资源配置集成优化研究提供了良好的理论基础,而关于预制构件并行生产线资源配置和生产调度集成优化方面的研究目前还较少见到相关报道. 本文主要内容如下:从问题方面看,以最小化订单总拖期惩罚费用为目标,在充分考虑工序特殊性和有限共享资源等特点的基础上,研究了一种特定工序生产资源受限的预制并行生产线资源配置和生产调度集成优化问题;从算法方面看,本文针对问题特点,基于分解策略和交替

迭代优化思想,提出了一种交替式混合果蝇-禁忌搜索算法(alternative hybrid fruit fly optimization and tabu search algorithm, AHFOA\_TS)以最小化拖期惩罚费用.

## 1 问题描述和目标函数计算

### 1.1 问题描述

在预制构件生产工厂内,某一计划周期开始前收到来自客户的 $n$ 个预制构件订单,需要分配到 $l$ 条预制生产线上处理. 生产过程由6道工序组成:1)模具组装;2)钢筋预埋;3)混凝土浇筑;4)蒸汽养护;5)模具拆除;6)瑕疵修整. 根据生产工艺的特征分为并行工序和串行工序. 其中蒸汽养护为并行工序,该工序可同时多个预制构件进行养护. 其余为串行工序,每次只能按序处理1个订单. 根据生产资源数量,可分为特定工序和一般工序. 其中工序3和工序6为特定工序,混凝土浇筑需要浇筑机,瑕疵修整需要专业技术人员,其他工序则为一般工序,与传统流水线生产类似. 由于特定工序的生产资源购置成本或雇佣费用比较昂贵,这类资源往往是有限的,通常特定工序可用的机器或专业人员等生产资源数量 $M_k$ 少于生产线数量 $L$ ,不同生产线订单进行到特定工序处理时可能会出现资源抢占,需要对有限的生产资源进行合理分配以减少特定工序生产等待时间. 设每个订单 $j$ 均有1个对应的交货期 $d_j$ ,订单拖期完工会产生相应的合同罚款. 假设条件设定如下:

- 1)所有订单、机器在调度零时刻均为可用状态.
- 2)订单在相邻工序之间的安装和运输时间忽略不计.
- 3)除并行工序外,其余工序上的生产资源同一时刻只能处理1个订单.
- 4)不考虑机器损坏和维修等突发故障情况.
- 5)对于特定工序,被分配到同一生产资源处理的订单,按照上一工序完工时间先后排序进行特定工序的生产.
- 6)生产采用24小时轮班制全天生产模式.

所研究的集成优化问题包括两个子问题:1)订单的生产线分配和调度;2)特定工序生产资源分配. 两个子问题决策相互影响,共同决定目标函数值大小,需要同时进行决策.

### 1.2 目标函数计算

问题描述和目标函数计算过程中所用到的符号定义如下.

- $n$ : 订单数量;
- $F$ : 生产线数量;

$j \in \mathcal{N} := \{1, 2, \dots, n\}$ : 订单标号;  
 $l \in \mathcal{L} := \{1, 2, \dots, F\}$ : 生产线标号;  
 $k \in \mathcal{K} := \{1, 2, \dots, 6\}$ : 工序标号;  
 $m \in \mathcal{M}_k := \{1, 2, \dots, M_k\}$ : 生产资源编号;  
 $n_l$ : 分配到生产线  $l$  上的订单数量;  
 $d_j$ : 订单  $j$  的交货期;  
 $w_j$ : 订单  $j$  单位时间拖期惩罚费用;  
 $C_{j,k}$ : 订单  $j$  在工序  $k$  上的完成时间;  
 $p_{j,k}$ : 订单  $j$  在工序  $k$  上的处理时间;  
 $D_{j,k}$ : 订单  $j$  在工序  $k$  上的开始时间;  
 $n_{j,k}$ : 工序  $k$  被分配到生产资源  $m$  上处理的订单数量;

$z_l \in \mathcal{N}_l := \{1, 2, \dots, n_l\}$ : 订单被分配到生产线  $l$  上的处理位置;

$u_m \in \mathcal{N}_{k,m} := \{1, 2, \dots, n_{k,m}\}$ : 订单被分配到生产资源  $m$  上的排序位置;

$[l, z_l]$ : 分配到生产线  $l$  上第  $z_l$  个位置处理的订单编号;

$\alpha_m^k$ : 工序  $k$  上被分配到生产资源  $m$  上处理的订单集合;

$\beta_m^k$ : 工序  $k$  上被分配到生产资源  $m$  上处理的订单调度序列;

$\beta_m^k(u_m)$ : 进行工序  $k$  处理时被分配到生产资源  $m$  上第  $u_m$  个位置的订单编号.

已知特定工序  $k$  的生产资源分配方案  $\alpha_m^k$  和各订单进行到特定工序  $k$  上一道工序的完工时间  $C_{j,k-1}$ , 由假设条件 5) 可得特定工序  $k$  被分配到生产资源  $m$  上生产的订单调度排序  $\beta_m^k$ . 对于给定的 1 个解, 即生产线指派与调度方案和特定工序生产资源分配方案, 若要计算目标函数值(总拖期惩罚费用), 则需要先确定各订单的完工时间. 各订单的完工时间可通过下式递推计算得出:

$$C_{[l,z_l],1} = \sum_{t=1}^{z_l} p_{[l,t],1}, \forall l, z_l; \quad (1)$$

$$C_{[l,1],2} = C_{[l,1],1} + p_{[l,1],2}; \quad (2)$$

$$C_{[l,z_l],2} = \max\{C_{[l,z_l-1],2}, C_{[l,z_l],1} + p_{[l,z_l],2}\}, \forall l, z_l \in \mathcal{N}_l \setminus \{1\}; \quad (3)$$

$$C_{\beta_m^3(1),3} = C_{\beta_m^3(1),2} + p_{\beta_m^3(1),3}; \quad (4)$$

$$C_{\beta_m^3(u_m),3} = \max\{C_{\beta_m^3(u_m-1),3}, C_{\beta_m^3(u_m),2} + p_{\beta_m^3(u_m),3}\}, \forall m \in \mathcal{M}_k, u_m \in \mathcal{N}_{k,m} \setminus \{1\}; \quad (5)$$

$$C_{[l,z_l],4} = C_{[l,z_l],3} + p_{[l,z_l],4}, \forall l, z_l \in \mathcal{N}_l; \quad (6)$$

$$C_{[l,1],5} = C_{[l,1],4} + p_{[l,1],5}; \quad (7)$$

$$C_{[l,z_l],5} = \max\{C_{[l,z_l-1],5}, C_{[l,z_l],4} + p_{[l,z_l],5}\}, \forall l, z_l \in \mathcal{N}_l \setminus \{1\}; \quad (8)$$

$$C_{\beta_m^6(1),6} = C_{\beta_m^6(1),5} + p_{\beta_m^6(1),6}; \quad (9)$$

$$C_{\beta_m^6(u_m),6} = \max\{C_{\beta_m^6(u_m-1),6}, C_{\beta_m^6(u_m),5} + p_{\beta_m^6(u_m),6}\}, \forall m \in \mathcal{M}_k, u_m \in \mathcal{N}_{k,m} \setminus \{1\}. \quad (10)$$

目标函数计算如下:

$$\min f = \sum_{j=1}^n w_j \times \max\{C_{j,6} - d_j, 0\}. \quad (11)$$

式(1)表示订单  $j$  在工序 1 上的完成时间为订单  $j$  以及同生产线上排在订单  $j$  之前处理的订单工序 1 处理时间的累加和; 式(2)、(3)、(7)和(8)分别计算订单  $j$  在工序 2 和工序 5 上的完成时间, 工序 2 和工序 5 为一般工序, 因此, 对每条线上第 1 个处理的订单, 开工时间即为该订单上道工序完成时间, 其他订单  $j$  的开工时间是对订单  $j$  上道工序的完成时间和同生产线上于订单  $j$  前 1 个处理的订单在该道工序的完成时间取最大值; 式(4)、(5)、(9)和(10)为订单  $j$  在工序 3 和工序 6 上的完工时间, 工序 3 和工序 6 为特定工序, 因此, 在计算前先通过启发式规则得到分配至各生产资源  $m$  上的订单排序  $\beta_m^k$ , 而后该特定工序分配到每个生产资源的订单完成时间可以通过与工序 2 和工序 5 同理类似的计算方式得出; 式(11)为最小化拖期惩罚,  $w_j \times \max\{C_{j,6} - d_j, 0\}$  为订单  $j$  的拖期惩罚费用, 若订单在最后一道工序上的完成时间  $C_{j,6}$  小于或等于订单交货期  $d_j$ , 则该订单的拖期惩罚为 0, 否则将会产生拖期惩罚费用.

## 2 交替式混合果蝇-禁忌搜索算法

### 2.1 问题求解框架

传统流水车间调度问题已被证明是一个 NP-hard 问题<sup>[7]</sup>. 在预制并行生产线订单指派调度和有限资源配置的集成优化问题中, 两个决策子问题之间具有强耦合关系, 比传统流水车间调度更为复杂, 问题求解难度更大. 本文采用分解决策和交替迭代优化的思想, 设计了 AHFOA\_TS 算法对该问题进行求解. 首先, 根据启发式规则策略生成订单在不同生产线间的分配和调度解, 并随机产生 1 个初始约束资源分配解, 随后执行 FOA 用于优化订单分配和调度方案; 然后, 固定该方案, 执行 TS 算法用于处理有限资源配置层子问题. 交替优化资源配置方案和生产调度方案, 最终得到整体方案满意解. 在订单指派和调度层子问题采用 FOA 算法的原因在于, 与传统智能算法相比, FOA 算法具有控制参数少、收敛速度快和

局部搜索能力强的特点. 基于FOA算法, 结合问题特点提出了顶层互换、前插、整层互换和变异4种邻域动作方式, 生成果蝇种群. 在此基础上, 采用基于概率选择的自适应嗅觉搜索方式, 保证算法局部开发性的前提下, 增强全局探索能力. 而有限资源配置层子问题采用TS的原因在于, 有限资源配置编码方式中重复元素较多, 导致邻域搜索时易产生重复解, 为减少无效搜索次数, 将迭代中的局部最优解加入禁忌表, 避免在下次迭代时重复搜索, 并且提出了精英劣解交叉策略, 以降低算法陷入局部最优的风险<sup>[18]</sup>. AHFOA\_TS算法流程如图1所示.

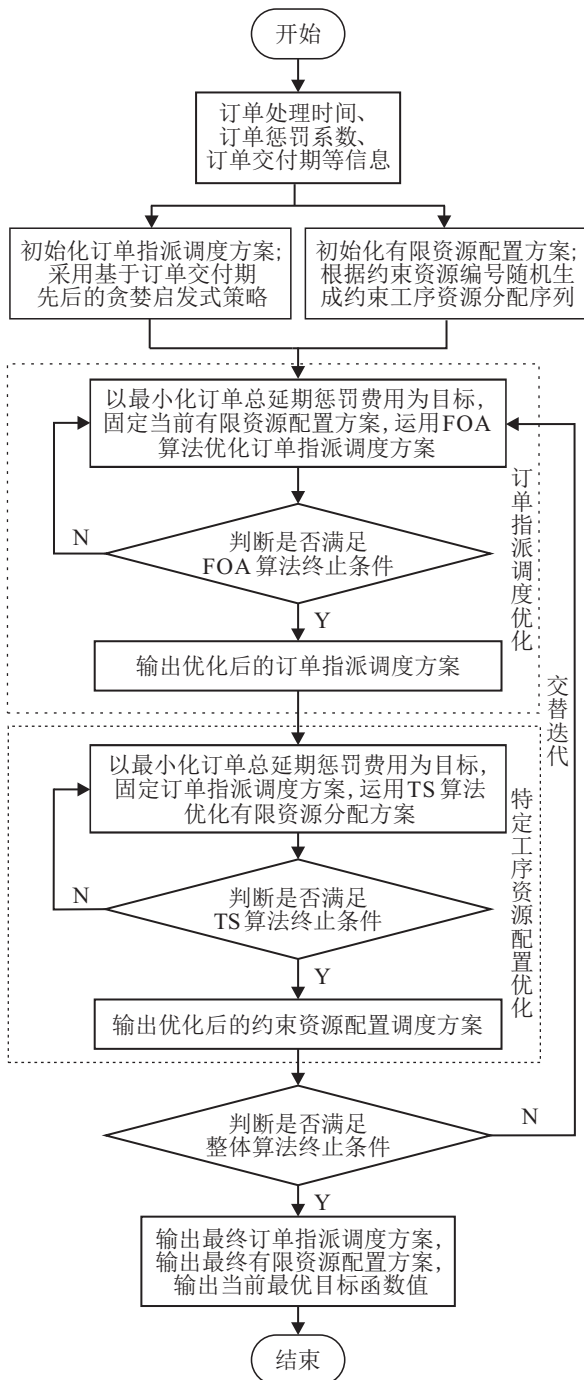


图1 AHFOA\_TS算法流程

## 2.2 编码和解码方式

按照订单指派调度层和有限资源配置层将问题分为两部分. 其中订单指派调度层的解由双层编码构成, 待处理预制构件订单有 $n$ 个, 现有 $l$ 条预制构件生产线, 从区间 $[1, l]$ 随机生成 $n$ 个数字序列作为解的第1层. 从区间 $[1, n]$ 生成 $n$ 个不重复的数字序列作为解的第2层. 订单分配和调度解的双层编码中, 每一列表示订单到生产线的分配, 各生产线订单按第2层编码中订单号由左至右的顺序进行处理, 进而确定订单分配和调度方案.

有限资源配置层解编码分为两层. 第1、2层分别表示工序3和工序6的有限资源分配方案, 从区间 $[1, M_3]$ 随机生成 $n$ 个整数作为解的第1层, 从区间 $[1, M_6]$ 随机生成 $n$ 个整数组成解的第2层. 有限生产资源分配解的解码是将订单分配和调度解排序后与有限生产资源分配解纵向对齐, 按生产资源编号分别取出订单, 即得到订单进行到特定工序时有限资源的分配方案.

## 2.3 FOA调度层算法

FOA的基本原理是初始化种群的中心位置, 利用嗅觉搜索, 即根据中心位置随机产生多个邻域解. 计算各可行解的味道浓度, 即评价值, 然后利用视觉搜索从中选择较好的解, 更新替换中心位置, 不断迭代寻优, 更好地靠近食物源. 本文在FOA算法框架下, 结合问题结构特点提出了顶层互换、前插、整层互换和变异4种邻域动作方式, 从多个不同方向进行邻域搜索生成果蝇种群, 提高种群的多样性, 从而增加获得全局最优解的可能性. 在此基础上采用基于概率选择的自适应嗅觉搜索方式, 根据每次迭代后各邻域中产生改良果蝇个体的比例, 更新下次迭代时各邻域的选择概率, 从而使得算法局部开发能力与全局探索性得到平衡.

### 2.3.1 基于双层整数编码的邻域构造

按各预制构件订单交货期由小至大排序, 将每个待加工订单依次分配至各生产线, 基于该启发式策略生成一个订单分配和调度解作为初始种群中心. 由于需要同时对订单指派方案和各生产线上订单的调度方案作出决策, 将生产线编号作为第1层, 订单编号作为第2层, 结合问题特点设计了4种特殊的邻域构造方式, 具体如下.

1) 顶层互换: 在产生的订单指派调度解内, 第2层订单调度不变, 在第1层随机选取两点交换位置, 目的是改变订单在生产线间的指派.

2) 前插: 从区间 $[1, n - 1]$ 随机产生1个位置点, 将

该位置点后面的所有部分与前面所有部分进行互换, 目的是改变订单在生产线上的排序。

3) 整层互换: 从区间  $[1, n]$  随机选取 2 个位置点, 将两位置点一、二层包含的元素整层互换, 从而改变订单在生产线上的处理顺序。

4) 变异: 在解的第 1 层随机选取若干个, 从  $[1, L]$  内随机产生和选取点不一样的值进行替换, 从而改变订单在生产线的分配。

### 2.3.2 基于概率选择的自适应嗅觉搜索

为增加解的分散性和算法的全局搜索能力, FOA 算法中的嗅觉搜索采用基于概率选择的自适应多邻域协同方式进行<sup>[16]</sup>。若以  $P_{v,g}$  表示第  $g$  代种群通过邻域结构  $v$  产生新果蝇个体的概率,  $B_{v,g}$  表示第  $g$  代种群通过邻域结构  $v$  得到使目标值改进的果蝇个体数, 则  $P_{1,g}$ 、 $P_{2,g}$ 、 $P_{3,g}$  和  $P_{4,g}$  分别为第  $g$  代种群通过顶层互换、前插、整层互换和变异方式产生新果蝇个体的概率,  $B_{1,g}$ 、 $B_{2,g}$ 、 $B_{3,g}$  和  $B_{4,g}$  分别为在第  $g$  代种群通过顶层互换、前插、整层互换和变异得到使目标值优化的果蝇个体数。公平起见, 初次迭代时,  $P_{1,1}$ 、 $P_{2,1}$ 、 $P_{3,1}$  和  $P_{4,1}$  均设定为 0.25。在以后迭代中,  $P_{v,g}$  计算如下:

$$P_{v,g} = \max \left\{ 0.1, B_{v,g} / \sum_{t=1}^4 B_{t,g-1} \right\}, \forall v = 1, 2, 3; \tag{12}$$

$$P_{4,g} = 1 - P_{1,g} - P_{2,g} - P_{3,g}. \tag{13}$$

为防止出现分子为 0 的情况, 设置各邻域操作的最低选择概率为 0.1。若出现分母为 0 的情况, 则令  $P_{1,g}$ 、 $P_{2,g}$ 、 $P_{3,g}$  和  $P_{4,g}$  均为 0.25。

### 2.3.3 视觉搜索

每个通过嗅觉搜索产生的果蝇个体代表 1 个订单分配和调度方案, 结合 TS 算法得到的当前最佳约束资源分配方案, 通过式 (1)~(11) 计算得到每个果蝇个体对应的目标值。本文研究的目的是最小化目标值, 因此, 将每个果蝇个体的适应度值定义为该果蝇个体对应目标值与该次迭代中产生所有果蝇个体目标值之和做比, 之后取倒数。由此可见, 适应度值越高的个体对应的目标值越优。将该次迭代中适应值最优的个体与当前种群中心个体的适应度值进行比较, 若更佳, 则替换为下次迭代的种群中心。

## 2.4 TS 资源配置层算法

资源配置层编码中重复元素较多, 容易在邻域动作时产生重复解, 导致算法无效搜索并降低算法性能, 因此, 采用 TS 算法将迭代中产生的局部最优解引入禁忌表, 避免循环搜索降低寻优效率。同时, 为克服

传统 TS 算法易陷入局部最优的缺陷, 设计了精英劣等群体交叉策略, 通过优劣解的信息交互扩大算法的搜索范围, 有利于跳出局部最优, 增加算法的全局探索能力。

在 TS 资源配置层算法中, 每个邻域解在当前最好解的基础上通过变异算子获得, 需经过如下两步操作:

1) 随机选择当前约束资源分配解中的若干个索引位置。若变异点 1 在第 3 道工序, 则第 3 道工序可使用的约束资源数量为  $M_3$ , 从区间  $[1, M_3]$  内随机产生 1 个与变异点 1 位置不同的值进行替换, 同理完成变异点 2 的操作。每个通过邻域动作产生的邻域解代表 1 个约束资源分配方案, 结合 FOA 算法得到当前最优订单分配调度方案, 再通过式 (11) 计算每个邻域解的目标值。

2) 将目标值由小至大进行排序, 抽取目标值排在前  $n/10$  的邻域解作为精英群体, 目标值排在后  $n/5$  的邻域解作为劣等群体, 每个劣个体随机匹配 1 个精英个体进行多段交叉最终产生  $2n/5$  新个体。

TS 配置层算法的特赦准则: 若通过交叉策略可以得到比当前最佳解更好的解, 则直接替换为当前解, 同时更新禁忌表, 将此解置于禁忌表的头部并将禁忌表尾端的解移除。

## 3 实验设计和结果分析

所有实验均通过 Matlab 2018b 编程实现, 计算机配置为 Microsoft Windows 10, 处理器为 Intel Core i5-9400 F CPU @ 2.9 GHz / 8 GB RAM。

### 3.1 实验设计

目前较少见到本文所研究问题的相关报道, 尚无参考的集成优化算法作为对比。但如引言中所述, 预制构件生产调度问题大多采用基于 GA 的智能优化方法, 因此, 调整预制构件生产调度中常用的 GA<sup>[1-7]</sup> 和传统生产调度问题常用的 VNS<sup>[19-20]</sup> 以适用于求解订单指派和调度层子问题, 在资源配置层则采用 TS、VNS 以及 FOA 等重要邻域搜索方法。进而基于交替搜索框架, 分别设计了 AHGA\_TS 算法、AHGA\_VNS 算法、AHVNS\_FOA 算法和 AHVNS\_TS 算法等 4 种算法作为对比算法。

根据对实际预制构件企业的调研, 设定生产线数量  $f = 3$ , 特定工序生产资源数量  $M_3$  和  $M_6$  均为 2, 在待处理订单数量  $n \in \{20, 30, 50, 70\}$  四种不同情形下进行实验测试。以文献 [5] 预制构件车间调度问题标准算例作为生产数据, 基于此数据生成各订单数量规模下的测试实例。订单  $j$  的交货期在区间  $[P_T/n, 3n]$

内随机产生,其中  $P_T$  为所有订单处理时间的累加和. 每种订单规模下测试10个不同实例,每个实例运行20次. 同时,为公平起见,对于同一算例,每种算法运行时间均设定为  $t_{\max} = 0.6n$ ,并记录20次运行取得目标值的最小值(min)和平均值(avg). 采用相对偏差率(relative percentage deviation, RPD)评估算法性能,有

$$RPD_i^A = (f_i^A - f_i^{\text{best}}) / f_i^{\text{best}} \times 100\%, \quad (14)$$

$$ARPD_n^A = \sum_{j=1}^{10} RPD_j^A / 10. \quad (15)$$

其中:  $f_i^{\text{best}}$  为问题实例  $i$  通过各算法求得目标函数的最佳值;  $f_i^A$  为问题实例  $i$  通过算法  $A$  求得的值;  $RPD_i^A$  为算法  $A$  求解问题实例  $i$  的偏差率;  $ARPD_n^A$  为算法  $A$  在计算的问题实例订单数量规模为  $n$  时,10个问题实例测试取得的平均偏差率. 对各算法采用田口方法进行调节,参数设置如下:

- 1) AHFOA\_TS 参数:  $PSF = 70, PST = 20, L = 10$ .
- 2) AHVNS\_FOA 参数:  $PSF = 50$ .
- 3) AHVNS\_TS 参数:  $PST = 20, L = 13$ .
- 4) AHGA\_VNS 参数:  $PSG = 60, p_c = 0.7, p_m = 0.2$ .

5) AHGA\_TS 参数:  $PSG = 80, p_c = 0.8, p_m = 0.1, PST = 20, L = 10$ . 其中:  $p_m$ 、 $p_c$ 、 $PSG$ 、 $PSF$  和  $PST$  分别为变异概率、交叉概率、遗传算法种群大小、果蝇算法种群规模和禁忌搜索候选集规模.

### 3.2 实验结果

为验证 AHFOA\_TS 算法在解决预制构件并行生产线资源配置和生产调度集成优化问题时的性能,对 AHFOA\_TS、AHVNS\_FOA、AHVNS\_TS、AHGA\_VNS 和 AHGA\_TS 五种算法进行实验对比分析. 计算结果表明,在订单数量  $n \in \{20, 30, 50, 70\}$  四种不同情形下的40个问题实例中, AHFOA\_TS 算法取得了其中31个实例最小 RPD 的最优值和33个算例平均 RPD 的最优值. 表明 AHFOA\_TS 算法在求解该问题时具有良好的求解质量和寻优性能. 同时, AHVNS\_TS 算法和 AHVNS\_FOA 算法在部分算例上也表现

出良好的求解性能. AHGA\_VNS 算法和 AHGA\_TS 算法在5种算法中求解性能表现较差,其主要原因是订单分配和调度层涉及生产工序较多且整体处理时间较长,对目标值影响更大. 而本文针对问题模型特点设计的4种特殊邻域构造方式应用于 VNS 内,在求解订单分配和调度层问题上具有良好的寻优效果.

图2为5种算法在求解不同订单规模实例取得的平均偏差率. 由图2可见, AHFOA\_TS 算法在不同订单规模下均取得了5种算法中最好的 ARPD 值,表明该算法比其他4种对比算法的综合求解质量和寻优能力更好. 其主要原因是,基于概率选择的嗅觉搜索方式和精英劣等种群交叉策略在保证算法局部开发性的前提下,增强了算法的全局探索能力,一定程度上避免了传统 VNS 算法、TS 算法和 FOA 算法易陷入局部最优的缺陷.

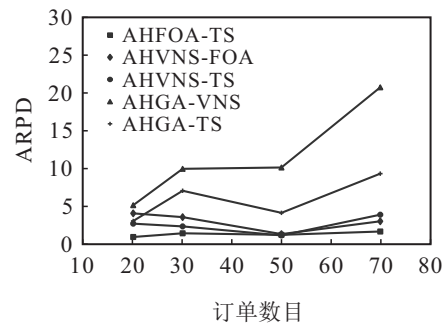


图2 不同订单情形下的 ARPD

传统预制构件制造企业通常采用规则启发式方法安排生产调度和资源配置. 表1为不同规模订单下5种交替迭代式智能优化算法相对规则启发式方法的求解质量平均改进率. 由表1可见,对于不同规模订单问题,5种智能优化算法相对于规则启发式方法均取得了较为显著的平均改进率,其中 AHFOA\_TS 算法取得最大平均改进率. 综合不同订单改进率, AHFOA\_TS 取得了8.42%的最大综合平均改进率. 需要指出的是,5种智能优化算法均在求解时间上比规则启发式方法略长,但最长时间不超过1 min. 在预制构件制造企业,用不到1 min 的计算时间换取净收益的较大提升完全是值得的.

表1 各智能优化算法相对于规则启发式方法的平均改进率

$n$	平均改进率/%				
	AHFOA_TS	AHVNS_FOA	AHVNS_TS	AHGA_VNS	AHGA_TS
20	10.58	7.83	8.62	8.62	8.45
30	10.57	8.90	9.45	9.13	5.56
50	4.11	3.71	4.03	4.09	1.33
70	8.40	6.15	6.43	4.18	1.59
平均值	8.42	6.65	7.13	6.51	4.23

为对比各算法的收敛性,以每种订单数量下的问题实例1为例,给出了5种算法的收敛曲线,如图3~图6所示.由图3~图6可见,在各种订单数量情形下,AHFOA\_TS算法均能够快速收敛,且表现出较其他4种算法更好的求解质量.

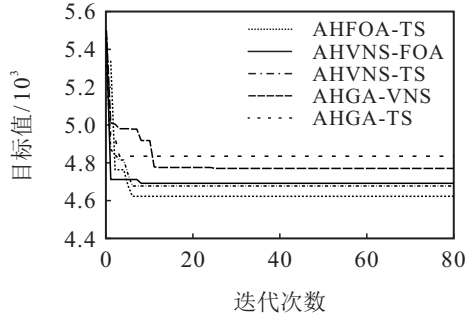


图3 n=20时各算法收敛曲线

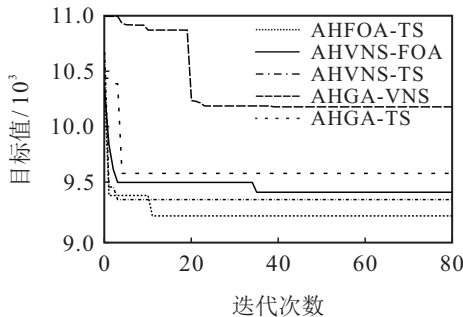


图4 n=30时各算法收敛曲线

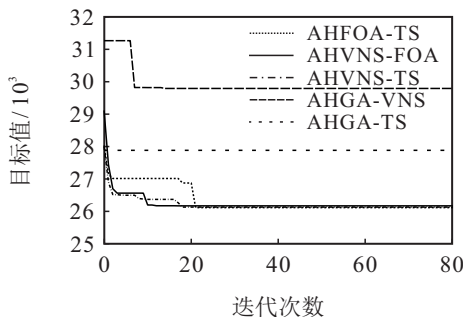


图5 n=50时各算法收敛曲线

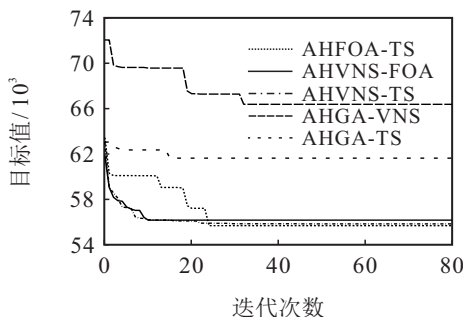


图6 n=70时各算法收敛曲线

### 3.3 算法时间复杂度分析

智能优化算法时间复杂度是程序中最深处基本操作的次数,包括并行结构和嵌套结构,因此有了加减乘除运算. AHFOA\_TS算法中,主要部分为调度层FOA算法、资源配置TS算法以及对于种群规模为

$n$ 、订单规模为 $n$ 和生产线数为 $F$ 的问题,AHFOA\_TS算法时间复杂度分析如下.

目标函数值的复杂度为 $O(6 \times n \times F) = O(n \times F)$ . 初始化种群复杂度为 $O(\text{PSF} \times n \times F)$ . 初始化种群后,对于调度层FOA算法,每一次迭代均需经过嗅觉搜索和视觉搜索阶段.其中嗅觉搜索阶段和视觉搜索阶段复杂度分别为 $O(\text{PSF} \times n)$ 和 $O(\text{PSF} \times n \times F)$ . 对于配置层TS算法,每次迭代产生PST个候选解的复杂度为 $O(\text{PST} \times n \times F)$ ,将目标函数值采用快速排序的时间复杂度为 $O(\text{PST} \times \log(\text{PST}))$ ,精英和劣等种群交叉的时间复杂度为 $O(\text{PST} \times n \times F)$ . 因此,算法每次迭代最高复杂度为 $O(\max\{\text{PSF}, \text{PST}\} \times n \times F)$ . 对于传统基于GA<sup>[1-7]</sup>的智能优化方法AHGA\_TS和AHGA\_VNS,采用类似的方法分析可得时间复杂度分别为 $O(\max\{\text{PSG}, \text{PST}\} \times n \times F)$ 、 $O(\text{PSG} \times n \times F)$ . 由此可见,所提出算法的时间复杂度与基于传统GA的智能优化算法接近.

## 4 结论

本文研究了预制构件并行线调度和特定工序受生产资源配置集成优化问题.以最小化订单拖期惩罚为目标,提出一种有效的交替式混合果蝇-禁忌搜索算法.结合问题特征设计了顶层互换、整层互换、前插和变异4种特殊有效的邻域构造方式,在此基础上,采用基于自适应概率选择嗅觉搜索方式的FOA算法以优化订单在不同生产线间分配和线上调度方案,以及基于双层变异算子和精英劣等群体交叉策略的TS算法以优化特定工序生产资源配置方案.通过不同规模实例测试各算法性能,计算结果验证了AHFOA\_TS算法求解所提出问题时的有效性和可行性.在有限的机器和人力资源下,可以为企业决策者选择合理的调度方案提供支持.未来研究将考虑机器故障和生产人员临时调整等不确定情形下的预制构件资源配置和生产调度集成优化问题,并根据问题特征设计更高效的求解算法,同时考虑将所提出算法推广至类似生产制造企业.

### 参考文献(References)

- [1] Chan W T, Hu H. An application of genetic algorithms to precast production scheduling[J]. Computers & Structures, 2001, 79(17): 1605-1616.
- [2] Benjaoran V, Dawood N, Hobbs B. Flowshop scheduling model for bespoke precast concrete production planning[J]. Construction Management and Economics, 2005, 23(1): 93-105.
- [3] Ko C H, Wang S F. Precast production scheduling using multi-objective genetic algorithms[J]. Expert Systems

- With Applications, 2011, 38(7): 8293-8302.
- [4] Yang Z T, Ma Z L, Wu S. Optimized flowshop scheduling of multiple production lines for precast production[J]. Automation in Construction, 2016, 72: 321-329.
- [5] Wang Z J, Hu H. Improved precast production-scheduling model considering the whole supply chain[J]. Journal of Computing in Civil Engineering, 2017, 31(4): 04017013.
- [6] Wang Z J, Hu H, Gong J. Framework for modeling operational uncertainty to optimize offsite production scheduling of precast components[J]. Automation in Construction, 2018, 86: 69-80.
- [7] Ma Z L, Yang Z T, Liu S L, et al. Optimized rescheduling of multiple production lines for flowshop production of reinforced precast concrete components[J]. Automation in Construction, 2018, 95: 86-97.
- [8] Low C. Simulated annealing heuristic for flow shop scheduling problems with unrelated parallel machines[J]. Computers & Operations Research, 2005, 32(8): 2013-2025.
- [9] Wang S J, Liu M. Two-stage hybrid flow shop scheduling with preventive maintenance using multi-objective tabu search method[J]. International Journal of Production Research, 2014, 52(5): 1495-1508.
- [10] Alaykýran K, Engin O, Döyen A. Using ant colony optimization to solve hybrid flow shop scheduling problems[J]. The International Journal of Advanced Manufacturing Technology, 2007, 35(5/6): 541-550.
- [11] Marichelvam M K, Geetha M, Tosun Ö. An improved particle swarm optimization algorithm to solve hybrid flowshop scheduling problems with the effect of human factors-A case study[J]. Computers & Operations Research, 2020, 114: 104812.
- [12] Li J Q, Pan Q K. Solving the large-scale hybrid flow shop scheduling problem with limited buffers by a hybrid artificial bee colony algorithm[J]. Information Sciences, 2015, 316: 487-502.
- [13] Pan W T. A new fruit fly optimization algorithm: Taking the financial distress model as an example[J]. Knowledge-Based Systems, 2012, 26: 69-74.
- [14] 郑晓龙, 王凌, 王圣尧. 求解置换流水线调度问题的混合离散果蝇算法[J]. 控制理论与应用, 2014, 31(2): 159-164.  
(Zheng X L, Wang L, Wang S Y. A hybrid discrete fruit fly optimization algorithm for solving permutation flow-shop scheduling problem[J]. Control Theory & Applications, 2014, 31(2): 159-164.)
- [15] Li J Q, Pan Q K, Mao K, et al. Solving the steelmaking casting problem using an effective fruit fly optimisation algorithm[J]. Knowledge-Based Systems, 2014, 72: 28-36.
- [16] 王凌, 郑洁, 王晶晶. 求解区间数分布式流水线调度的混合离散果蝇优化算法[J]. 控制与决策, 2020, 35(4): 930-936.  
(Wang L, Zheng J, Wang J J. A hybrid discrete fruit fly optimization algorithm for distributed permutation flowshop scheduling with interval data[J]. Control and Decision, 2020, 35(4): 930-936.)
- [17] 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.
- [18] Glover F. Future paths for integer programming and links to artificial intelligence[J]. Computers & Operations Research, 1986, 13(5): 533-549.
- [19] Mejía G, Yuraszeck F. A self-tuning variable neighborhood search algorithm and an effective decoding scheme for open shop scheduling problems with travel/setup times[J]. European Journal of Operational Research, 2020, 285(2): 484-496.
- [20] Rosa B F, Souza M J F, de Souza S R. Algorithms based on VNS for solving the single machine scheduling problem with earliness and tardiness penalties[J]. Electronic Notes in Discrete Mathematics, 2018, 66: 47-54.

### 作者简介

熊福力(1974—), 男, 副教授, 博士, 从事生产计划与调度、大规模系统优化等研究, E-mail: xiongfuli@xauat.edu.cn;  
张杏(1995—), 女, 硕士生, 从事预制构件生产调度、智能优化的研究, E-mail: zx@xauat.edu.cn;  
曹劲松(1993—), 男, 硕士生, 从事预制构件生产调度、智能优化的研究, E-mail: caojinsong@xauat.edu.cn;  
袁子阳(1996—), 男, 硕士生, 从事能效生产调度、智能优化的研究, E-mail: yuanziyang@xauat.edu.cn;  
储梦伶(1994—), 女, 硕士生, 从事预制构件生产调度、智能优化的研究, E-mail: chumengling0418@xauat.edu.cn;  
杜瑶(1996—), 男, 硕士生, 从事预制构件生产调度、智能优化的研究, E-mail: duyao@xauat.edu.cn.

(责任编辑: 魏冰)