

共享柔性资源约束下的多机并行作业 排程调度问题研究

林 聪, 汪达钦

东华大学旭日工商管理学院, 上海

收稿日期: 2022年8月25日; 录用日期: 2022年9月5日; 发布日期: 2022年9月19日

摘 要

随着C2B定制化生产模式的兴起, 催生了多品种小批量的生产方式。这种方式会导致各类生产任务在机器上频繁切换, 容易造成产能浪费和机器损耗。同时, 在多机并行生产的过程中, 经常需要共享一些柔性资源, 如人力、库存缓冲区、特定生产设备等, 因此需要精巧的生产排程计划来合理协调多种共享资源的利用, 减少产能浪费。本文以并行机器调度问题为研究对象, 研究了以最小化最大完工时间以及切换时间为目标的多机并行调度问题。在满足经典并行机调度问题的约束条件下, 同时考虑了多种共享柔性资源约束。针对该NP-hard问题, 本文设计了基于禁忌搜索的超启发式算法进行求解优化, 采用禁忌搜索算法作为高层次启发式策略, 并结合并行机作业的调度特点设计了7种LLH来构成低层次启发式算法池, 通过算例分析验证了算法的有效性。最后通过不同规模的数值分析对多种共享柔性资源的影响进行分析, 结果表明人力资源和共享产线容量的增加都会改善排程的目标, 然而这两种资源的边际效应是随之递减的。由于资源的使用以及扩容都需要一定成本, 因此科学选择合适的人力资源水平以及共享产线的容量对提高整体效率非常重要。

关键词

共享柔性资源, 不相关并行机调度, 禁忌搜索, 超启发式算法

Unrelated Parallel Machines Scheduling with Shared Flexible Resources

Cong Lin, Daqin Wang

Glorious Sun School of Management, Donghua University, Shanghai

Received: Aug. 25th, 2022; accepted: Sep. 5th, 2022; published: Sep. 19th, 2022

文章引用: 林聪, 汪达钦. 共享柔性资源约束下的多机并行作业排程调度问题研究[J]. 管理科学与工程, 2022, 11(3): 341-355. DOI: 10.12677/mse.2022.113042

Abstract

With the rise of C2B customized production mode, multi-variety and small-batch production mode has been spawned. This way will lead to frequent switching of various production tasks on the machine, which is easy to cause capacity waste and machine loss. At the same time, in the process of multi-machine parallel production, it is often necessary to share some flexible resources, such as manpower, inventory buffer, specific production equipment, etc. Therefore, exquisite production scheduling is needed to rationally coordinate the utilization of various shared resources and reduce capacity waste. In this paper, the unrelated parallel machine scheduling problem is studied to minimize the maximum completion time and switch time. Under the constraints of the classical parallel machine scheduling problem, multiple shared flexible resource constraints are considered at the same time. To solve the NP-hard problem, this paper designs a hyper-heuristic algorithm based on tabu search for optimization. Tabu search algorithm is used as a high-level heuristic strategy. Combined with the scheduling characteristics of parallel machine jobs, seven low-level heuristics are designed to form a low-level heuristic algorithm pool. The effectiveness of the algorithm is verified by an example. Finally, the impact of multiple kinds of shared flexible resources is analyzed by numerical analysis of different scales. The results show that the increase of worker resources and shared production line capacity will improve the scheduling goals, but the marginal effect of these two kinds of resources decreases accordingly. Due to the cost of resource utilization and expansion, it is important to scientifically choose the right level of worker resources and share the capacity of production line to improve the overall efficiency.

Keywords

Shared Flexible Resources, Unrelated Parallel Machine Scheduling, Tabu Search, Hyper-Heuristic Algorithm

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着工业 4.0 时代的到来, C2B 的商业模式逐渐兴起, 工业生产方式逐渐向多样性、灵活性以及定制化方向进行发展, 企业的订单类型逐渐以小订单多批次为主, 从而导致所需的生产设备种类以及人力需求量加大。因此, 企业在生产过程中需要考虑更多的因素, 比如多资源的共享、设备的调整时间等相关因素。此外, 在传统生产过程中, 设备的生产效率低下, 对快速变化的市场需求适应力较差, 生产柔性水平较低。随着生产技术的不断提高以及行业定制化的发展趋势, 共享柔性资源在生产过程中逐渐被重视, 有助于企业提高生产效率, 降低管理风险。资源的柔性可以体现在生产资源、加工路线、生产方案等方面的可替换性以及多样性[1]。在柔性的生产资源中生产线、生产设备以及人力资源尤为重要, 人力资源分配已经成为决定生产成本、工期、质量等生产系统指标的关键因素之一[2], 每个生产周期内人力资源是否均衡将会影响生产的稳定性及可靠性。此外, 由于共享产线是已知的固定资源, 若重新投资开设生产线将会影响生产效率造成一定的浪费, 且不同的生产设备需要通过特定的产线进行加工, 因此, 在同一时间内对生产设备的启用是否合理将会影响生产资源的利用率以及生产效率。综上所述, 生产管理者在进行生产决策的时候, 还需要考虑人力资源的均衡约束以及共享产线约束。

并行机器调度问题(PMSP)是为了优化多台机器之间的资源分配和作业顺序,它是实际生产过程中最典型的一种调度问题[3],在工程科学的各个领域都得到了广泛的考虑。根据并行机器是否具有同一性,并行机器调度问题可以分为不相关并行机器和同型并行机器。本文将对不相关并行机器的调度问题进行研究。

2. 文献综述

近年来,由于调度在车间和生产环境中的越来越重要且并行机器调度问题在现实世界中的普遍存在于制造业、半导体业等行业中,因此PMSP受到了广泛研究人员的关注。Bitar [4]等研究了具有辅助资源、序列相关和顺序相关准备时间的不相关并行机的调度问题,并提出了整数线性规划模型进行求解。Bektur 和 Saraç [5]以总加权厌恶最小化为目标,研究了具有顺序相关准备时间以及机器合格性约束的不相关并行机调度问题,并将改进的ATCS调度规则与TS和SA算法结合进行求解。雷德明和杨海[6]针对具有预防性维修(PM)和顺序相关准备时间(SDST)的不相关并行机调度问题,提出了一种多群体人工蜂群算法(MABC)以同时最小化完工时间和总延迟时间。齐玉欣[7]等以最小化最大完工时间以及能源消耗为优化目标,并提出一种改进的遗传算法进行求解。王东军[8]等研究了一类具有工件释放时间机器可用时间和机器适用限制等约束的并行同速机调度问题,并提出了基于优先规则的调度算法。Vallada 和 Villa [9]等以最小化最大完工时间为目标,研究了考虑一个辅助附加资源的不相关并行机问题,并提出了三种改进的搜索算法对该问题进行求解。张寒等[10]同时以最小化最大完工时间以及总加权拖期目标,建立了带恶化和学习效应的不相关并行机调度模型,并对模拟退火算法进行改进从而得到该问题更好的近似解。Fanjul-Peyro [11]对具有设置和资源约束的不相关并行机问题进行研究,建立了混合整数线性规划模型,并利用三阶段算法对该问题进行求解。Lei 和 Yuan [12]等以最小化最大完工时间和总延迟为目标,利用改进后的人工蜂群对分布式不相关并行机问题进行求解。

上述文献所研究的并行机器调度问题目标通常为单一的目标约束,比如最小化最大加工完成时间、最小化总加工完成时间以及当存在任务输送约束时,通常还使任务的总延迟最小化。这些研究大多运用了遗传算法、蜂群算法等进化类智能启发式算法进行问题求解。然而在复杂的优化问题中,单一智能算法存在一定的局限性使得求出的解不够好。比如启发式算法在解决问题时往往根据某些规则,从而导致无法比较其求出的近似解与最优解的近似程度,可以结合遗传算法、禁忌搜索算法、模拟退火算法进行改进。基于个体启发的禁忌搜索算法在求解过程中容易受到其禁忌长度以及特赦规则的影响,从而导致其容易陷入局部最优。因此,国内外许多学者开始将多个智能启发式算法进行结合改进来弥补算法间的不足。

受上述启发,本文以设备最大完工时间以及切换时间最短为多优化目标,建立了共享柔性资源下并行多机作业调度问题模型,运用基于禁忌搜索的超启发式智能优化算法进行求解,通过实例验证该算法的有效性并制定出与企业实际生产情况更加贴近、具有实际应用价值的并行多机作业调度方案。

3. 问题描述及模型建立

3.1. 问题描述

本文所研究的并行多机调度问题可描述为有 n 个独立的任务,每一个任务可以在 m 个不同质的设备上任意选择一个进行加工作业。在整个加工过程中,每个任务只有一道加工工序,任务 i 在设备 k 上的加工时间 w_{ik} , w_{ikt} 为生产周期 t 中每个任务 i 在设备 k 上所需要的人力资源,每个生产周期内的人力资源需要均衡分配。此外,在生产过程中,不同设备需要通过指定的产线进行加工,而共享产线 m 允许多个同类型的加工机器同时进行加工,但对不同类型加工机器需要进行限制来确保任务的生产质量。因此,

该问题的调度优化决策包括了两个部分, 一是任务如何分配给不同的并行机进行加工, 二是根据人力资源以及共享产线约束对分配到不同并行机的任务的加工顺序进行优化, 以求最小化最大完工时间以及切换时间。

并行多机调度问题具有以下的任务和机器约束:

- 1) 每个任务只需加工一次;
- 2) 每个设备同一时间只能进行一个任务的加工;
- 3) 每个任务可以任意选择某个设备进行加工作业;
- 4) 每个任务在不同的机器上的加工时间是已知的;
- 5) 每个设备的切换时间不依赖于加工顺序。

本文以设备完工总时长以及切换时间最短为优化目标建立数学模型。

3.2. 变量定义

根据以上的目标以及假设, 定义了如下变量(表 1)。

Table 1. Symbol description

表 1. 符号说明表

符号	符号说明
i, j	表示任务 i 和任务 j , $i \in \Omega$, $j \in \Omega$
K	所有设备集合
Ω	所有任务集合
T	所有生产阶段集合
S	所有产线集合
W	人力资源波动水平
M	表示无限大的正数
0~1 决策变量	含义
x_{ijk}	设备 k 上任务 i 是任务 j 的紧前任务取值为 1, 否则为 0
y_{ik}	任务 i 在设备 k 上生产取值为 1, 否则为 0
u_{ik}	任务 i 在设备 k 上是第一个任务取值为 1, 否则为 0
v_{ik}	任务 i 在设备 k 上是最后一个任务取值为 1, 否则为 0
z_{km}	设备 k 生产并通过产线 m 取值为 1, 否则为 0
其他变量	含义
$makespan$	所有任务中最晚结束的时间
$wcmin_t$	在 t 生产阶段里面小于所有人力资源的数量
$wcmax_t$	在 t 生产阶段里面大于所有人力资源的数量
wt_{ik}	任务 i 在设备 k 上所需生产时间
wf_{ikt}	在班次 t 内任务 i 在设备 k 上所需人力资源
swt_{ijk}	任务 i 和任务 j 在设备 k 上的切换时间
et_i	任务 i 最早开始时间

Continued

lt_i	任务 i 最迟完成时间
C_{\max}	最长生产时间
s_m	产线 m 通过指定生产设备数量的上限
st_i	任务 i 开始时间
ft_i	任务 i 结束时间

3.3. 模型建立

基于以上的问题描述, 本文建立了以下的不相关并行多机作业调度模型, 以设备最大完工时间最小化以及切换时间最短为多优化目标。

$$\min \text{makespan} \quad (1)$$

$$\min \sum_{k \in K} \sum_{i \in \Omega} \sum_{j \in \Omega} x_{ijk} * swt_{ijk} \quad (2)$$

$$\sum_{k \in K} \left(\sum_{i \in \Omega} x_{ijk} + u_{jk} \right) = 1, \quad \forall j \in \Omega, i \neq j \quad (3)$$

$$\sum_{k \in K} \left(\sum_{j \in \Omega} x_{ijk} + v_{jk} \right) = 1, \quad \forall i \in \Omega, i \neq j \quad (4)$$

$$\sum_{i \in \Omega} u_{ik} \leq 1, \quad \forall k \in K \quad (5)$$

$$\sum_{i \in \Omega} v_{ik} \leq 1, \quad \forall k \in K \quad (6)$$

$$y_{ik} = \sum_{j \in \Omega} x_{ijk} + v_{ik}, \quad \forall i \in \Omega, \forall k \in K \quad (7)$$

$$st_i + \sum_{k \in K} (wt_{ik} \cdot y_{ik}) - \left(1 - \sum_{k \in K} y_{ik} \right) \cdot M \leq ft_i, \quad \forall i, j \in \Omega, i \neq j \quad (8)$$

$$ft_i + \sum_{k \in K} x_{ijk} \cdot st_{ijk} - \left(1 - \sum_{k \in K} x_{ijk} \right) \cdot M \leq st_j, \quad \forall i, j \in \Omega, i \neq j \quad (9)$$

$$st_i \geq et_i, \quad \forall i \in \Omega \quad (10)$$

$$ft_i \geq lt_i, \quad \forall i \in \Omega \quad (11)$$

$$\text{makspan} \geq ft_i, \quad \forall i \in \Omega \quad (12)$$

$$\text{makspan} \leq C_{\max} \quad (13)$$

$$w\text{cmax}_t \geq \sum_{k \in K} (wf_{ikt} \cdot y_{ik}), \quad \forall i \in \Omega, \forall t \in T \quad (14)$$

$$w\text{cmin}_t \leq \sum_{k \in K} (wf_{ikt} \cdot y_{ik}), \quad \forall i \in \Omega, \forall t \in T \quad (15)$$

$$\max(w\text{cmax}_t - w\text{cmin}_t) \leq W, \quad \forall t \in T \quad (16)$$

$$\sum_{k \in K} z_{km} \leq s_m, \quad \forall m \in S \quad (17)$$

其中: 表达式(1)、(2)为本文两个调度优化的目标: 最小化最大完工时间、最小化切换时间; 表达式(3)~(7)表示每个任务都有唯一的紧前和紧后任务并完成加工, 且每个加工设备最多只有一个任务作为

首要任务和尾部任务；表达式(10)、(11)表示生产任务最早开始时间和最晚结束时间的限制；表达式(12)、(13)表示完工时间的定义及约束。

表达式(8)~(9)、表达式(14)~(17)为本文的特殊约束。其中：表达式(8)表示每个加工任务的结束时间需要大于前置加工任务的开始时间以及本任务的加工时长；表达式(9)表示每个加工任务的开始时间需要大于等于前置任务的结束时间与切换时间之和；表达式(14)、(15)表示所有生产阶段内的最大人力资源数以及最少人力资源数；表达式(16)表示控制所有生产阶段内人力资源的最大值与最小值的差值来达到人力资源均衡的目标；表达式(17)表示共享产线有限问题，共享产线只能同时通过一定数量的不同类型加工设备。

4. 基于禁忌搜索算法的超启发式算法设计

不相关并行机器调度问题(Unrelated parallel machine scheduling problem, UPMSP)是一个已被证实的NP-hard问题[9]，目前大多数都采用传统的启发式算法或者最优化算法进行求解，其中最优化算法往往需要耗费大量的时间进行求解，当求解问题规模加大时则无法进行求解。在几个常规的启发式算法中：遗传算法效率较低且容易陷入局部最优解，而禁忌搜索算法可以通过标记已经得到的局部最优解，在再次搜索的过程中避开这些带有标记的解从而可以大概率避免陷入局部，直至找到全局最优解。此外，最近提出的超启发式算法具有较好的寻优能力，该算法主要由高层次启发式策略(High-level heuristic, HLH)以及低层次启发方法(Low-level heuristic, LLH)构成，由HLH对LLH进行一系列的操作管理，通过自动选择、组合或生成几个简单的低级启发式方法来处理复杂的NP-hard问题。

因此，本文设计了一种有效的基于禁忌搜索算法的超启发式算法来实现共享柔性资源约束下的并行多机作业调度问题的优化。本文采用禁忌搜索算法作为超启发式算法的高层次启发式策略，并构造7种简单的启发式规则来组成低层次启发式方法池。该算法在超启发式算法的框架下，将禁忌搜索算法的记忆能力强的特点与LLH的灵活性强进行结合，根据实际情况对算法进行调整并以较快的速度对NP-hard问题进行求解。

4.1. 低层次启发式方法

低层次启发式方法是超启发式算法的重要组成部分，过于复杂的LLH规则将会提高超启发式算法的复杂度从而影响算法的整体运行效率。因此，本文采用7种简单的启发式规则来构成低层次启发式方法池，使得算法能够以较快的速度以及丰富的策略寻找最优解。现有加工任务编码为 $\{W_1, W_2, W_3, \dots, W_n\}$ ，加工设备的编码为 $\{M_1, M_2, M_3, \dots, M_n\}$ ，本文的LLH方法池构造如下：

随机交换任务(LLH1)：随机选择两个加工任务 W_a 和 W_b ，将这两个任务进行交换。

随机交换组任务(LLH2)：随机选择两个加工任务组 $\{W_a, W_{a+1}, \dots, W_{a+k}\}$ 和 $\{W_b, W_{b+1}, \dots, W_{b+k}\}$ ，将这两组加工任务进行交换。

随机交换加工设备(LLH3)：随机选择两个加工设备 M_a 和 M_b ，将这两个加工设备上的所有加工任务按照固定的加工顺序进行交换，交换以后加工设备 M_a 的加工任务顺序为 $\{W_b, W_{b+1}, \dots, W_n\}$ ，加工设备 M_b 的加工任务顺序为 $\{W_a, W_{a+1}, \dots, W_n\}$ 。

随机切换任务至不同设备(LLH4)：随机选择一台加工设备 M_a 上的任一加工任务 W_a ，将该任务 W_a 转移到另外一台加工设备 M_b 的加工任务 $\{W_b, W_{b+1}, \dots, W_n\}$ 的尾链后，因此加工设备 M_b 的加工任务变成 $\{W_b, W_{b+1}, \dots, W_n, W_a\}$ 。

随机交换任务并同时交换加工设备(LLH5)：随机选择加工设备 M_a 上的任一加工任务 W_a 和加工设备 M_b 上的任一加工任务组 W_b ，将这两个加工任务进行交换。

随机交换组任务并同时交换加工设备(LLH6): 随机选择加工设备 M_a 上的任一加工组 $\{W_a, W_{a+1}, \dots, W_{a+k}\}$ 和加工设备 M_b 上的任一加工任务组 $\{W_b, W_{b+1}, \dots, W_{b+k}\}$, 将这两组加工任务进行交换加工。

随机改变任务尾链并改变加工设备(LLH7): 随机选择加工设备 M_a 上加工任务的尾链 $\{W_a, W_{a+1}, \dots, W_n\}$ 并转移到加工设备 M_b 的加工任务 $\{W_b, W_{b+1}, \dots, W_n\}$ 的尾链后, 因此加工设备 M_b 的加工任务变成 $\{W_b, W_{b+1}, \dots, W_n, W_a, W_{a+1}, \dots, W_n\}$ 。

根据以上的 LLH 方法池, 可以发现 LLH1 和 LLH2 只对加工任务的加工顺序进行改变, LLH3 只对加工设备进行改变, LLH4-7 则是对加工任务和加工设备同时进行一定的改变, 对不同的主体设置了不同的 LLH 规则, 保证了方法池的多样性以及灵活性, 有利于超启发式算法的寻优能力。

4.2. 高层次启发式策略

文选用禁忌搜索算法(TS)作为超启发式算法的高层次启发式策略, 禁忌搜索算法是一种基于个体的元启发式算法, 它可以通过标记已经得到的局部最优解, 在再次搜索的过程中避开这些带有标记的解从而可以在大概率的情况下避免陷入局部, 直至找到全局最优解。

禁忌搜索算法主要由初始解、邻域生成方法、解的评价函数、禁忌表、特赦准则和终止条件组成。

1) 获得初始解, 初始解对禁忌搜索算法的工作效率有很大的影响, 本文采用贪婪算法构造初始解。其过程为: a) 选择加工时间最长的任务并随机选择加工机器进行加工; b) 选择加工时间次长的任务并选择完工时间最短的设备进行加工; c) 不断重复步骤直至所有任务都完成。

2) 邻域生成方法, 即适用于当前解的一组可能移动, 在当前解的基础上按照特定的移动策略产生一定数量的新的解。通过邻域方法产生的解成为邻域解, 由此产生的新解的数目则称为邻域解的规模。本文主要采用随机交换的邻域生成方法, 对任意两个元素进行交换因此可产生 Cn^2 个邻域解, 从这些邻域解中选择质量最好的解进行禁忌判断以及是否优于之前最好解的结果, 通过则成为下一次迭代的解。

3) 解的评价函数, 通过解的评价函数来判断解的优劣性, 从而决定是否需要进行替换。本文采用目标函数作为解的评价函数即设备的最大完工时间以及最短切换时间加权之和, 因此解的评价值越小时, 解就越好。然而, 两个目标的单位存在一定差别, 需要对最大完工时间以及最大切换时间进行无量纲化处理, 其中 $\overline{C_{\max}}$ 为已知最大完工时间解的平均值, 令 $S_{\max} = \sum_{k \in K} \sum_{i \in \Omega} \sum_{j \in \Omega} swt_{ijk}$, $\overline{S_{\max}}$ 为已知切换时间解的平均值。因此, 解的评价函数如下所示:

$$F = \min \left(0.5 \frac{C_{\max} - \overline{C_{\max}}}{\overline{C_{\max}}} + 0.5 \frac{S_{\max} - \overline{S_{\max}}}{\overline{S_{\max}}} \right)$$

4) 禁忌表, 禁忌表用来存放禁忌对象的容器, 放入禁忌表以后的对象则不进行搜索, 禁忌表的长度是每个禁忌对象在禁忌表中的生存时间。本文采用 FIFO 的方式对禁忌对象进行替换, 禁忌表的长度 L 为 $0.2 \times$ 数据规模, 当禁忌表满时, 有新的禁忌对象进入时则释放最先进入的解。

5) 特赦准则和终止条件, 特赦准则保证了搜索过程中存在当前最优解被禁时, 能够释放特定解, 从而实现全局最优。

4.3. 基于禁忌搜索的超启发式算法模型

在基于禁忌搜索的超启发式算法中, 禁忌搜索算法是作为高层次启发式策略对低层次启发式方法池进行管理和操纵, 禁忌搜索算法的搜索空间为 LLH 方法池中所有的组合。将 LLH 的方法池与得出的解进行组合, 从而达到更新解的目的并找到更好质量的解。因此基于禁忌搜索的超启发式算法的设计流程图如图 1 所示。

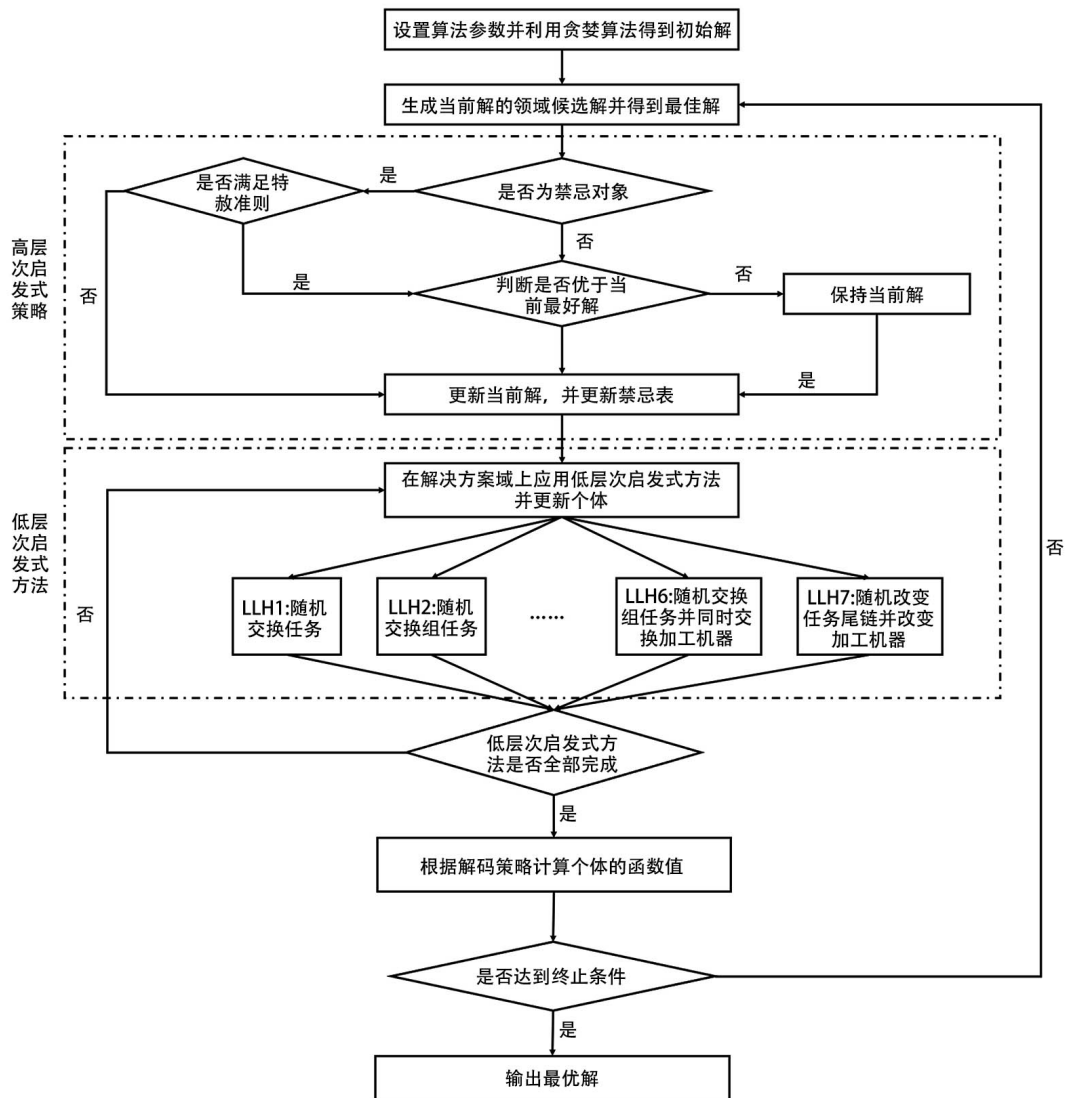


Figure 1. Algorithm flow diagram
图 1. 算法流程图

该算法的其具体流程描述如下:

- 1) 在基于禁忌搜索超启发式算法中, 采用贪婪算法构造初始解, 选择加工时间最长的任务并随机选择加工机器进行加工, 选择加工时间次长的任务并选择完工时间最短的设备进行加工。运用随机交换的方式产生当前解的领域解, 并选择最佳解。
- 2) 判断选中的最佳解是否为禁忌对象, 若是则继续判断是否满足特赦准则, 若满足则根据评价函数进行判断, 解的评价值越小说明解的质量越好。当进入解的评价值优于当前解的评价值, 则更新当前解。
- 3) 根据低层次启发式方法池中的方法顺序, 对解进行重组构造, 并计算个体的适应度函数值, 选择个体函数值最大的进入下一个环节。

5. 数值试验

为了验证基于禁忌搜索算法的超启发式算法的有效性以及分析共享柔性资源对并行多机调度的影响, 本文主要对共享柔性资源中的人力资源以及产线容量进行相关分析, 分析过程如下:

1) 利用本文设计的超启发式算法对不同的案例进行求解, 并与求解器 *gurobi* 得出的结果进行对比分析。

2) 考虑人力资源均衡约束下并行多机调度求解结果, 计算不同的 $wc_{max} - wc_{min}$ 差值下不同案例的最大完工时间以及切换时间, 并进行对比分析。

3) 考虑产线容量约束下并行多机调度求解结果, 计算不同产线容量下不同案例的最大完工时间以及切换时间, 并进行对比。

本文以某个冷饮加工工厂为例, 利用基于禁忌搜索算法的超启发式算法求解分析, 算法终止条件为当前算法迭代次数超过 1000 次时。该加工工厂主要的生产工艺为混料、冷凝以及罐装, 而罐装主要分为普通包装线和蛋筒线, 其实际需求为减少总切换时间, 降低最大完工时间从而提高资源的利用率以及应对风险的能力。

根据工厂生产的实际情况, 本文设置了待加工任务在不同可用生产设备上的生产速度以及每个加工任务所需的人力资源, 不同加工任务之间的切换时间以及加工任务间的优先级类别及顺序。每个任务的需求量以及在不同设备上所需要的加工速度将在附录里面进行展示。

基于以上的任务数据资料, 本文利用基于禁忌搜索算法的超启发式算法, 运用 *java* 开发算法对多机并行作业调度问题进行求解分析。

5.1. 算法有效性验证

本文利用基于禁忌搜索算法的超启发式算法, 选取某个冷饮加工工厂的实际生产案例进行验证。本文选用了 4 种加工设备种类数目(4/6/8/12)、3 种加工任务数(20/30/40), 以最小化最大完工时间以及切换时间为目标, 并对每一个案例测试 10 遍选取目标的平均值, 并将其与求解器 *Gurobi*、*NSGA-II* 算法得出的结果进行对比, 从而来验证算法的有效性和模型的实用性, 其对比结果如下表 2 所示。

Table 2. Effectiveness comparison
表 2. 有效性对比表

设备种类	任务数量	本文算法		精确算法		NSGA-II	
		C_{max}	swt_{ijk}	C_{max}	swt_{ijk}	C_{max}	swt_{ijk}
4	20	13395	960	12947	960	13573	960
	30	16649	1140	16372	1140	16972	1140
	40	21509	1260	21028	1260	23784	1260
6	20	12647	720	12530	720	12771	720
	30	13498	1140	13105	1140	13875	1140
	40	16764	1320	16737	1320	17214	1320
8	20	11435	660	11055	660	11975	660
	30	12851	1140	12782	1140	12853	1140
	40	15404	1560	15031	1560	16579	1560
12	20	11013	600	10676	600	11016	600
	30	12036	960	11704	960	12715	960
	40	13614	1380	13260	1380	14731	1380

本文采用基于禁忌搜索算法的超启发式算法所得的结果与 Gurobi 得出的精确解进行对比分析。从上表可以发现基于禁忌搜索算法的超启发式算法的求解结果与精确算法的求解结果比较相近, 最大完工时间的总体误差在 4% 以内, 平均误差为 2.16%, 总切换时间的误差为 0%。此外, 本文算法在运行过程中求解速度较快, 能够保证解的质量情况下提高求解速度。为了进一步验证算法的有效性, 将本文算法与 NSGA-II 算法进行比较, 通过上表可以发现本文求解的算法所得到的最大完工时间均优于 NSGA-II 算法求得结果, 从而验证了本文算法的有效性。

5.2. 人力资源平衡对多机并行作业调度的影响

在实际的生产情况中, 不同的加工任务所需的人力也不相同, 从而导致不同生产周期内人力资源较大的波动, 使得生产人员的稳定性较差。因此, 对不同生产周期内的人力资源波动进行合理控制可以帮助企业有效地降低人力生产成本, 从而提高企业的核心竞争力。在多机并行作业调度中, 控制每个周期人力资源平衡都非常重要, 因此本文对同个生产周期内对不同的人力波动差值 $wc_{max} - wc_{min}$ 进行枚举分析, 分析不同人力波动下各个案例的最大完工时间以及切换时间, 并进行对比, 其结果如下表 3 所示。

Table 3. Comparison of results under different human resource fluctuations

表 3. 不同人力资源波动下结果对比表

设备种类	任务数量	差值 10		差值 20		差值 30	
		C_{max}	swt_{ijk}	C_{max}	swt_{ijk}	C_{max}	swt_{ijk}
4	20	8804	840	8455	780	8215	780
	30	13558	1200	12419	1080	11659	1080
	40	21569	1500	20513	1500	19805	1500
6	20	8625	720	7925	720	7585	720
	30	10713	1120	9502	1080	9013	1020
	40	14059	1560	13151	1480	12954	1480
8	20	8371	720	7613	780	7411	780
	30	10447	1120	9447	1080	8947	1020
	40	13584	1440	12234	1440	11632	1440
12	20	8065	720	7315	720	6931	720
	30	10031	1200	9042	1200	8705	1200
	40	12803	1320	11425	1320	11062	1320

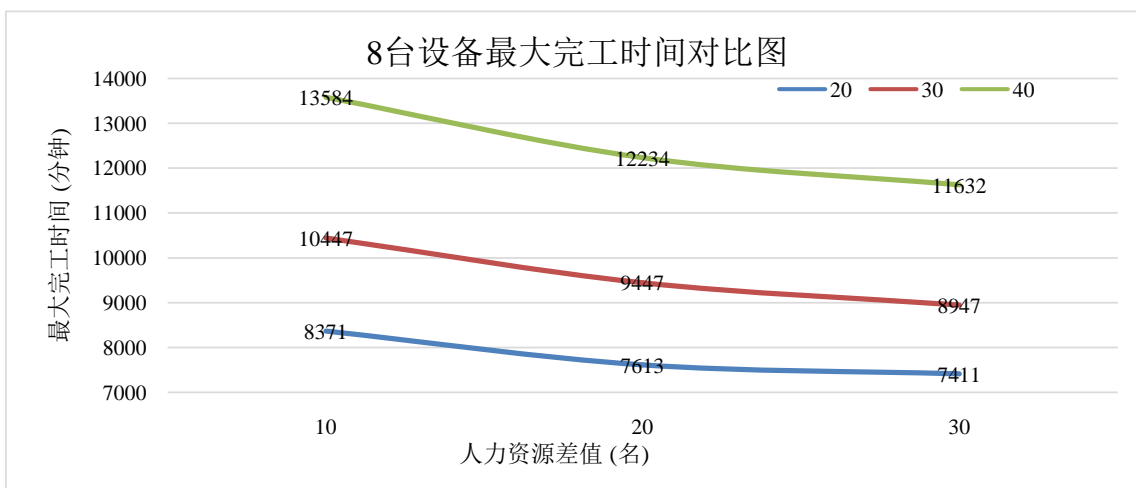
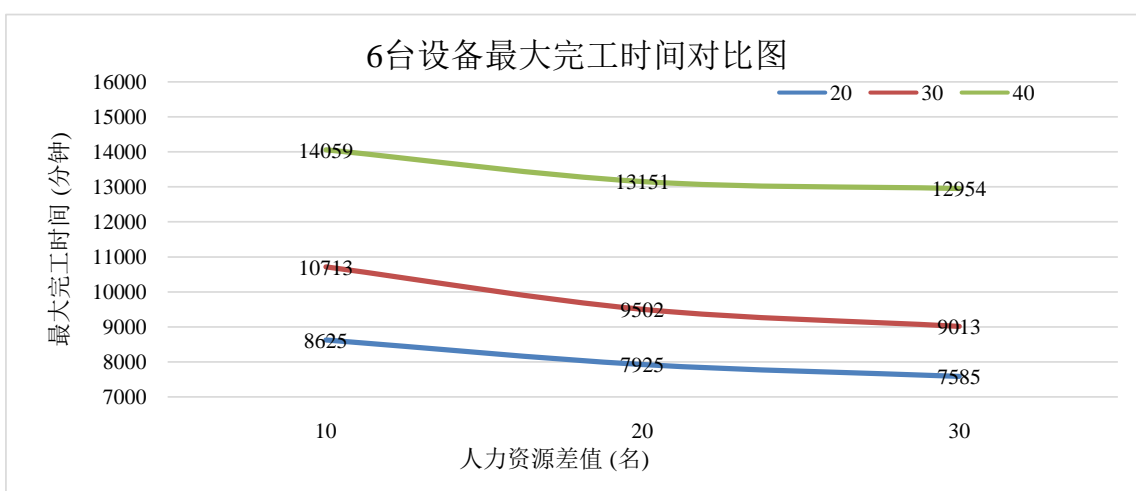
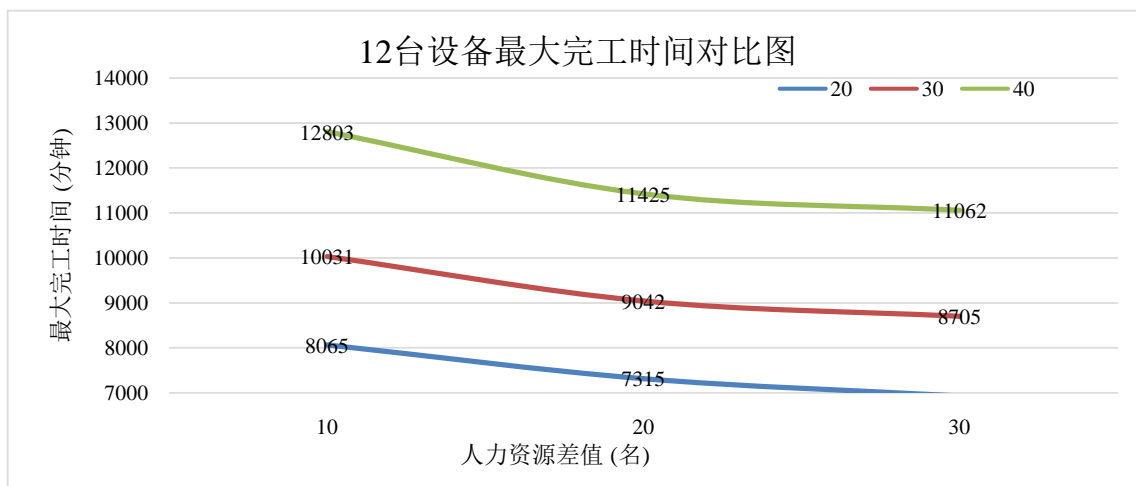
不同人力资源差值下的最大完工时间的变化如图 2 所示。

通过上述图表, 可以发现在相同的加工任务数量以及加工设备数量水平下, 多机并行作业调度的最大完工时间随着生产周期内人力资源波动的加大而缩短, 并且减少的幅度随着人力资源波动的增大而减小。当所有生产周期内人力资源波动的差值足够大时, 调度的最大完工时间将达到稳定的水平即不会影响调度的安排, 所有的任务会同时在最早的时间进行加工作业, 因此整个系统的最大完工时间将缩短。

此外, 本文对不同人力波动情况之间的增速进行比较, 对比结果如表 4 所示。

通过该表可以发现, 随着加工设备数量以及人力资源波动程度的增加, 同等数量下的加工任务完工时间减少的幅度也随之增加。这是由于当只有人力资源波动增加时, 而设备数量存在限制, 尽管每台设

备都以最大的产能进行加工, 但最大完工时间减少的速度也会减慢。由于一些任务无法满足当前的人力资源约束, 需要将调整加工顺序, 从而导致整体切换时间的增加。此外, 当加工设备具有的生产能力没有达到本身的阈值时, 其用来加工的产能随着产线的增长而增长。



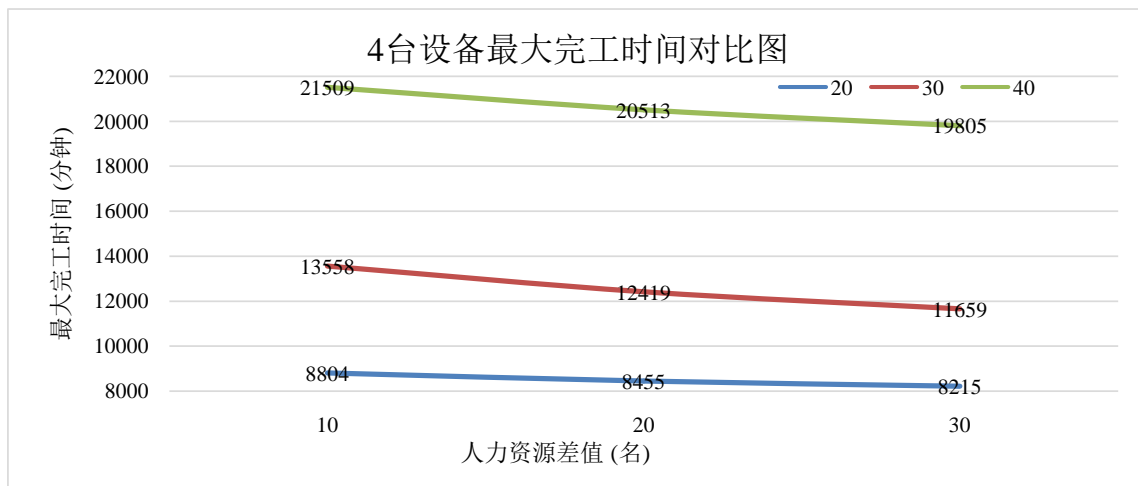


Figure 2. Comparison of the maximum completion time of different equipment numbers in three cases
图 2. 三种情况下不同设备数最大完工时间对比图

Table 4. Comparison of maximum completion time growth rate under different manpower difference
表 4. 不同人力差值下的最大完工时间增速对比表

差值对比	任务数量	设备种类			
		4	6	8	12
$\Delta = 10$ (10~20)	20	3.96%	8.12%	9.06%	9.30%
	30	8.40%	11.30%	9.57%	9.86%
	40	4.63%	6.46%	9.94%	10.76%
$\Delta = 10$ (20~30)	20	2.84%	4.29%	2.65%	5.25%
	30	6.12%	5.15%	5.29%	3.73%
	40	3.45%	1.50%	4.92%	3.18%
$\Delta = 20$ (10~30)	20	6.69%	12.06%	11.47%	14.06%
	30	14.01%	15.87%	14.36%	13.22%
	40	7.92%	7.86%	14.37%	13.60%

5.3. 共享产线容量对多机并行作业调度的影响

在实际生产情况中, 很多企业存在一定的生产设备约束, 部分生产设备只能通过指定的生产线后才能进行生产, 而为了控制生产质量, 需要控制同一时间内通过产线加工设备种类。因此, 共享产线容量的大小将会影响企业的生产效率以及生产成本。在多机并行作业调度中, 产线容量的限制不容忽视, 因此本文对不同的产线容量进行分析, 分析不同容量约束下各个案例的最大完工时间以及切换时间, 并进行对比, 其结果如下表 5 所示。

不同产线容量下的最大完工时间的变化如下图 3 所示。

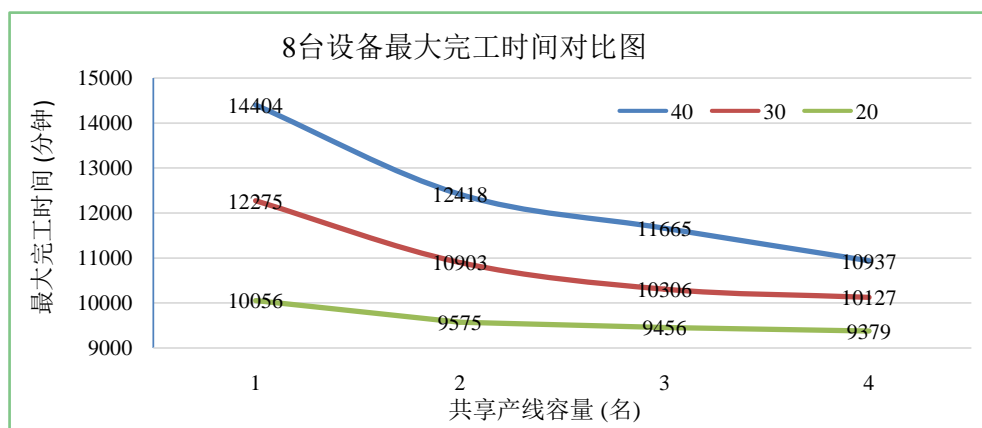
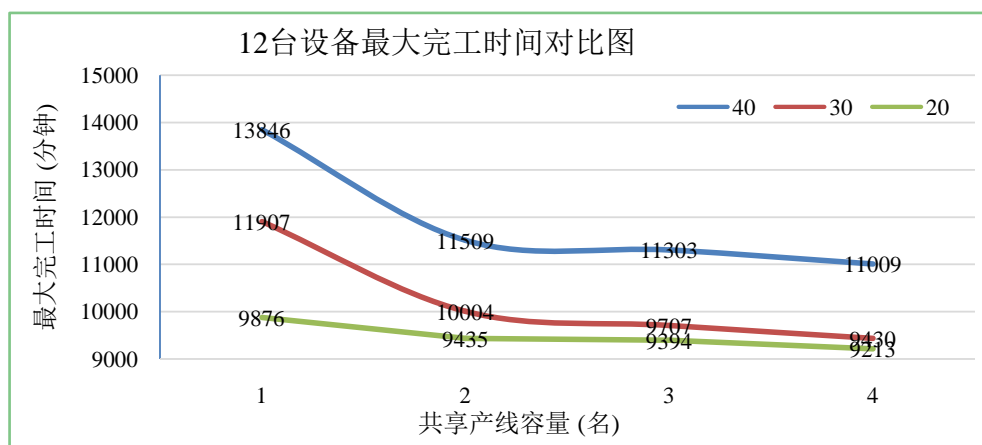
通过上述图表, 我们可以发现在相同的加工任务数量以及加工设备种类数目水平下, 多机并行作业调度的最大完工时间随着产线容量的加大而缩短, 并且减少的幅度随着产线容量的增大而减小。当产线容量足够大时, 调度的最大完工时间将达到稳定的水平即不会影响调度的安排。在相同的加工任务数量下, 为了减少最大完工时间, 一些任务将被安排在可生产且不同产线的其他生产设备上生产, 因此随着加工设备种类的增多, 调度的最大完工时间也会随之减少。此外, 我们可以发现, 在相同的加工设

备种类情况下, 调度的切换时间是相同, 但随着加工设备种类的增加而不断减少, 这是因为一些加工任务被安排到其他设备上加工而减少了同一台设备上的切换时间。

Table 5. Comparison of results under different shared production line capacity

表 5. 不同共享产线容量下结果对比表

设备种类	任务数量	容量 1		容量 2		容量 3		容量 4	
		C_{max}	swt_{ijk}	C_{max}	swt_{ijk}	C_{max}	swt_{ijk}	C_{max}	swt_{ijk}
4	20	13515	960	11455	960	10150	960	9985	960
	30	14649	1140	12659	1140	11638	1140	11530	1140
	40	21509	1440	18219	1440	17302	1440	16935	1440
6	20	11647	780	9947	780	9683	780	9572	780
	30	13595	1140	11498	1140	10963	1140	10691	1140
	40	15764	1380	12954	1380	12364	1380	11925	1380
8	20	10056	660	9575	660	9456	660	9379	660
	30	12275	1080	10903	1080	10306	1080	10127	1080
	40	14404	1260	12418	1260	11665	1260	10937	1260
12	20	9876	600	9435	600	9394	600	9213	600
	30	11907	1020	10004	1020	9707	1020	9430	1020
	40	13846	1200	11509	1200	11303	1200	11009	1200



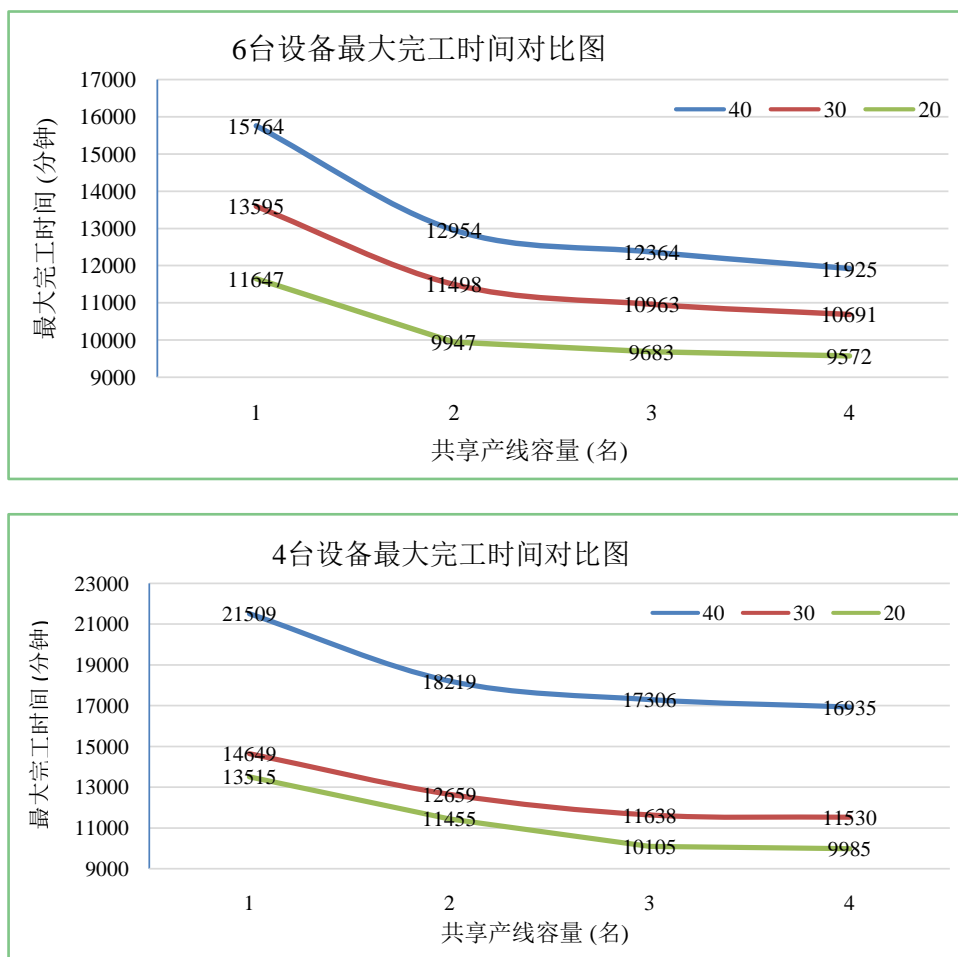


Figure 3. Comparison of the maximum completion time of different equipment numbers in four cases

图 3. 四种情况下不同设备数最大完工时间对比图

6. 总结

本文对共享柔性资源约束下的并行多机作业排产调度问题进行研究, 以最小化最大完工时间以及切换时间最短为多目标, 建立了更加符合实际生产情况的并行多机作业调度问题模型, 从而制定出与企业实际生产情况更加贴近、具有实际应用价值的调度方案。

与之前的研究相比, 本文主要的贡献有以下几点: 1) 根据并行多机调度问题的复杂性以及需要对多目标进行优化, 本文设计了基于禁忌搜索的超启发式算法对该 NP-hard 问题进行优化求解, 其中将禁忌搜索算法作为高层次启发式策略, 设计了 7 种低层次启发式方法。2) 结合某工厂的实际生产数据对模型及方法进行验证, 并将结果与 gurobi 精确方法、NSGA-II 算法进行对比, 从而验证模型的正确性以及算法的有效性和高运行效率。3) 在调度问题中引入了共享柔性资源这一约束, 并在不同人力资源以及共享产线容量的情况下, 对整个调度系统进行对比分析, 从而使得模型更加符合生产中的实际情况, 所制定的并行多机调度方案更具有科学性。

参考文献

- [1] 鲁陈璐. 制造系统柔性资源调度问题及其优化算法研究[D]: [硕士学位论文]. 杭州: 浙江大学, 2016.

-
- [2] 王一帆, 刘士新, 陈迪. 求解多技能人力资源约束的项目调度问题的两阶段算法[J]. 东北大学学报(自然科学版), 2014, 35(2): 184-189.
- [3] 王凌. 车间调度及其遗传算法[M]. 北京: 清华大学出版社, 2003: 146-148.
- [4] Bitar, A., Dauzère-Pérès, S. and Yugma, G.C. (2021) Unrelated Parallel Machine Scheduling with New Criteria: Complexity and Models. *Computers & Operations Research*, **132**, Article No. 105291. <https://doi.org/10.1016/j.cor.2021.105291>
- [5] Bektur, G. and Saraç, T. (2019) A Mathematical Model and Heuristic Algorithms for an Unrelated Parallel Machine Scheduling Problem with Sequence-Dependent Setup Times, Machine Eligibility Restrictions and a Common Server. *Computers & Operations Research*, **103**, 46-63. <https://doi.org/10.1016/j.cor.2018.10.010>
- [6] 雷德明, 杨海. 求解多目标不相关并行机调度问题的多群体人工蜂群算法[J]. 控制与决策, 2022, 37(5): 1174-1182.
- [7] 齐玉欣, 付亚平, 孙翠华. 考虑能耗优化和学习效应的随机多目标并行机调度问题研究[J]. 青岛大学学报(自然科学版), 2021, 34(1): 82-86.
- [8] 王东军, 刘翱, 刘克, 刘波. 基于优先规则的复杂并行机调度问题研究[J]. 系统工程理论与实践, 2016, 36(3): 779-786.
- [9] Vallada, E., Villa, F. and Fanjul-Peyro, L. (2019) Enriched Metaheuristics for the Resource Constrained Unrelated Parallel Machine Scheduling Problem. *Computers & Operations Research*, **111**, 415-424. <https://doi.org/10.1016/j.cor.2019.07.016>
- [10] 轩华, 张寒, 王薛苑. 带恶化和学习效应的不相关并行机调度优化[J/OL]. 控制工程. https://kns.cnki.net/kcms/detail/detail.aspx?dbcode=CAPJ&dbname=CAPJLAST&filename=JZDF20210802000&uniplatform=NZKPT&v=ydIOHjjzTr7Th2NRRAHJr5hMJcbdIGXWquvUhACCN-tswc234O6_n13pjXrpMi4z, 2020-1-10.
- [11] Fanjul-Peyro, L. (2020) Models and an Exact Method for the Unrelated Parallel Machine Scheduling Problem with Setups and Resources. *Expert Systems with Applications X*, **5**, Article ID: 100022. <https://doi.org/10.1016/j.eswax.2020.100022>
- [12] Lei, D., Yuan, Y. and Cai, J. (2020) An Improved Artificial Bee Colony for Multi-Objective Distributed Unrelated Parallel Machine Scheduling. *International Journal of Production Research*, **2020**, 1-13. <https://doi.org/10.1080/00207543.2020.1775911>