

订单可分解多柔性作业车间协同鲁棒调度

李营力¹, 王家海¹, 刘银冬¹, 刘江山²

(1. 同济大学 机械与能源工程学院, 上海 201804; 2. 上海交通大学 密西根学院, 上海 200240)

摘要: 针对企业多车间协同生产需求, 对多柔性作业车间协同调度问题展开研究, 提出了一种订单可分解多柔性作业车间协同鲁棒调度方法。给出多车间协同调度的求解方案, 设计了二级决策树生产订单分配模型, 其中, 第一级决策树进行分配策略选择, 第二级决策树进行生产订单拆分。在第一级决策树中设计了基于遗传算法的拖期检测方法, 提高车间产能评估的准确性, 同时, 将新订单加入视为动态调度过程, 通过染色体的层次选择增强调度的鲁棒性。在第二级决策树中基于线性规划提出了生产订单分解的上下界准则。最后, 通过一个基准案例以及部分生产实例对订单可分解多柔性作业车间协同鲁棒调度方法进行测试。测试结果表明, 该方法具备可行有效性, 可供企业实际应用参考。

关键词: 订单分配; 多车间; 协同生产; 供应链

中图分类号: TH166

文献标志码: A

Order Decomposable Collaborative Robust Scheduling for Multiple Flexible Workshops

LI Yingli¹, WANG Jiahai¹, LIU Yindong¹, LIU Jiangshan²

(1. School of Mechanical Engineering, Tongji University, Shanghai 201804, China; 2. University of Michigan - Shanghai Jiao Tong University Joint Institute, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract: In response to the needs of collaborative production of multiple workshops, the collaborative scheduling problem of multiple flexible workshops was studied, and an order decomposable collaborative robust scheduling method was proposed. First, collaborative scheduling solution was given. Then, a two-level decision tree production order distribution model was designed, in which the first level was used to select the distribution strategy, and the second one was used to split production

order. In the first level, a genetic algorithm-based tardiness detection method was designed to improve the accuracy of workshop capacity evaluation, meanwhile, the new order insertion was regarded as a dynamic scheduling process, and scheduling robustness was enhanced through chromosome level selection. In the second level, a upper and lower bound criteria for production order decomposition was proposed based on linear programming. Finally, a benchmark case and some production cases were used to test the method proposed. The test results show that the method is feasible and effective, which can be used as a reference for practical application in enterprises.

Key words: order allocation; multiple workshops; collaborative production; supply chain

在智能制造和工业 4.0 的大背景下, 企业追求智能化精益生产。生产调度作为一种生产优化控制技术方法, 可在现有资源约束的情况下, 通过智能算法优化生产流程、高效利用生产资源、缩短制造周期以及节约生产成本。

根据研究对象的不同, 生产调度可分为流水车间调度、作业车间调度、柔性作业车间调度等, 其中柔性作业车间由于其高度的生产柔性, 以及更加符合当今社会小批量个性化定制的产品生产发展趋势, 一直是研究的热点。

现有针对柔性作业车间调度的研究主要关注单车间, Li 等^[1]针对单柔性作业车间提出了一种带有两阶段局部搜索的进化算法; Li 等^[2]设计了一种基于多智能体的单柔性作业车间分布式调度方法; 顾幸生等^[3]改进博弈粒子群算法求解单柔性作业车间调度问题。随着市场全球化, 很多企业为快速响应

收稿日期: 2021-07-16

基金项目: 国家重点研发计划(2017YFE0101400)

第一作者: 李营力(1990—), 男, 博士生, 主要研究方向为组合优化、生产调度、智能制造。

E-mail: liyingli@tongji.edu.cn

通信作者: 王家海(1964—), 男, 副教授, 博士生导师, 工学博士, 主要研究方向为数字化制造及相关技术。

E-mail: jhwang@tongji.edu.cn



论文
拓展
介绍

市场需求而采用多车间生产模式(即:在不同的地理位置建立多个生产车间),因此,多车间调度优化成为科研和生产管理人员不得不面对的一个现实问题。

理论上单柔性作业车间调度是 NP-hard 难题^[4]。由于多柔性作业车间调度除了包含单车间调度的所有复杂性外还要额外考虑订单分配问题,其求解比单车间调度更为复杂,也是一类 NP-hard 难题。

国内外已有少量多车间调度的研究文献,具有代表性的研究成果综述如下:

Behnamian 等^[5]以及 Gharaei 等^[6]从宏观角度研究多车间调度(即:不区分车间类型),分别提出一个集成线性规划启发式算法和基于多智能体的分布式调度模型。仅考虑车间层调度而忽视车间内部调度会导致订单分配不合理,无助于生产优化,甚至降低综合生产效率,例如:将一个订单分配给加工质量最优的车间,然而该车间负荷超载,无法按期完成,这增加了延期成本,降低了供应能力。因此,开展车间层调度和车间内部调度同时优化的协同调度方法研究是必要的。

Mao 等^[7]针对分布式流水车间设计了一种迭代贪心算法求解多车间协同调度问题;单汨源等^[8]针对多车间协同调度问题提出了一种基于多智能体的生产运作模型,但是未给出具体的控制方法;蒋大奎等^[9]研究大规模订单分配和排序问题,设计了禁忌搜索与动态规划相结合的集成优化算法;杨红红等^[10]针对多车间协同调度问题提出了一个基于两级遗传算法的优化求解策略;王静等^[11]针对云平台下多订单分配问题,设计了一个自适应模拟退火遗传算法;王旭亮等^[12]为解决跨企业生产协同生产调度问题,提出一种适用于云制造环境的混合集成规划方法;沈广亚等^[13]研究了航空制造业多车间协同生产调度问题,提出一个生产与运输综合计划模型;陈友玲等^[14]针对制造资源生产能力约束导致的任务分配不合理问题,建立了一种任务分配优化模型来求解多车间协同调度。这些多车间协同调度研究都存在不足,协同调度时都只提及要考虑每个车间的产能,产能评估方法并未给出,仅仅将机器非工作时间总和视作剩余产能的做法并不精确。此外,优化时仅考虑综合目标,比如:最大完工时间、平均通过时间等,忽视了算法的鲁棒性(即:新订单的分配不能无视车间原有的生产计划)。

多柔性作业车间调度作为多车间调度的一个分支,与其具有共性,既要解决车间层和车间内部的协

同调度问题,同时还要解决多车间协同调度中产能评估不准确、忽视算法鲁棒性的问题,解决这些问题才能提高协同调度的性能,供企业实际参考应用。然而,目前尚未发现能够同时解决这些问题的研究文献。

本文基于某国有企业多车间协同生产实际需求,对多柔性作业车间协同调度展开研究。首先,给出多车间协同调度求解方案;然后,建立了两级决策树模型用于生产订单分配,为了更好实现多车间协同,模型中提出了一个基于有效剩余产能和线性规划的订单分解上下界限定准则;最后,通过案例验证了所提方法的可行性和有效性。

1 问题描述

单柔性作业车间通常描述为^[4]:一个车间有多个工件在多台机器上加工;每个工件有多道工序;同一工件的不同工序之间有顺序约束;每道工序的可选加工设备有多个;同一时刻,一道工序只能在一个选定的机器上进行加工且不能中断;同一时刻,每台机器也只能加工一道工序;不同工件之间以及不同工件的工序之间无优先级约束;加工准备时间包含在工序加工时间内。

多柔性作业车间协同调度问题可描述为:一个企业有多个地理位置不同的柔性作业车间;每个车间的设备类型、设备数量以及可加工工件类型独立;每个车间可加工相同的工件,也可加工不同的工件。假设 i 是一个新订单,其要生产的产品数量为 Q_i ,交货期为 T_i ,总售价为 Z_i 。协同调度的任务就是将 i 分配到合适的车间,并优化关联车间内的生产计划,以实现某些目标。

2 求解方案

市场部门将订单信息发送至订单分配模型,分配模型接收到订单信息后首先进行产品检索,因为一个订单涉及的产品不一定所有的车间都有能力生产,因此这一步对于缩小判断决策区间是必要的。随后,将订单信息分别发送给符合条件的车间进行成本计算和拖期检测,并将计算和检测结果返回。订单分配功能模块根据返回的结果通过两级决策树进行订单分配决策。如果分配结果达到预期目标,例如:按时交货的同时成本最低,则订单分配结束,执行生产;如果分配结果达不到预期,例如:单个车

间都无法按时完成订单,需要多车间协同生产,则将订单进行拆分处理。具体流程如图1所示,图中,

ERP为企业资源管理系统,MES为制造执行系统, u 为车间号索引。

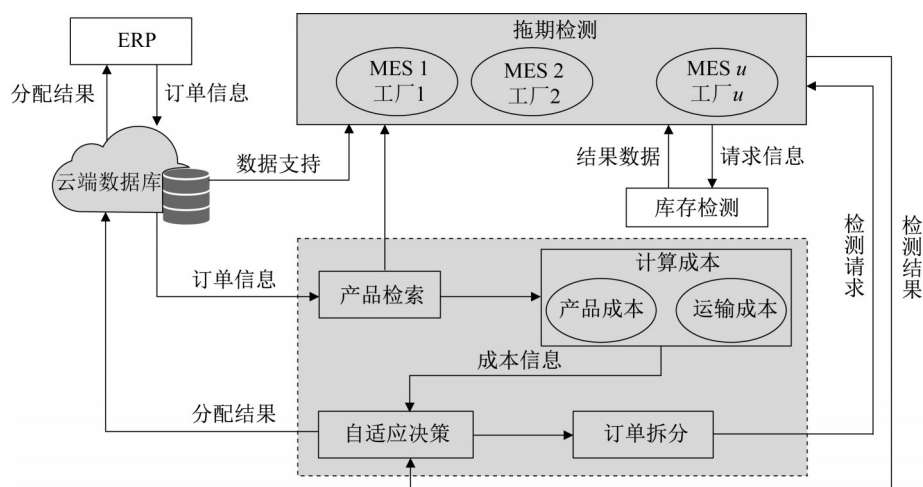


图1 订单分配方案

Fig. 1 Scheme of order allocation

3 两级决策树订单分配模型

对于多条件判断决策问题,决策树已被证实是一种有效的解决方法^[15]。本文通过建立两级决策树订单分配模型来解决多车间协同调度问题,第一级决策树用于决策选择,第二级决策树用于订单分解。

3.1 第一级决策树:决策选择

第一步,进行产品搜索,将所有车间集合 A 分为两个集合 A_1 和 A_2 , A_1 代表有能力生产该类型产品的车间集合, A_2 为没有能力生产该类型产品的车间集合。

第二步,对 A_1 进行收益评估,将其分为 $A_{1.1}$ 和 $A_{1.2}$ 两个集合,其中, $A_{1.1}$ 代表收益率符合要求的车间集合, $A_{1.2}$ 为收益率不符合要求的车间集合。进行收益评估是不可或缺的一步,因为企业生产一个订单通常会设定一个最低收益率 R 以保证可观的盈利,收益评估表达式如式(1)和式(2)所示:

$$C(S_{ij}, D_{ij}) = d_{ij}Q_i + w_{ij}l_{ij}\varphi \quad (1)$$

$$\frac{[Z_i - C(S_{ij}, D_{ij})]}{C(S_{ij}, D_{ij})} \cdot 100\% \geq R \quad (2)$$

式(1)~(2)中: $C(S_{ij}, D_{ij})$ 代表总成本; S_{ij} 和 D_{ij} 分别为车间 j 生产订单 i 的生产成本和运输成本; d_{ij} 为单个产品生产成本; w_{ij} 和 l_{ij} 为货物重量和运输距离; φ 为与重量和距离有关的运输单价。

第三步,对 $A_{1.1}$ 包含的各车间进行拖期检测,得

到能够按时交货车间数 K 的值。

决策选择示意图如图2所示。

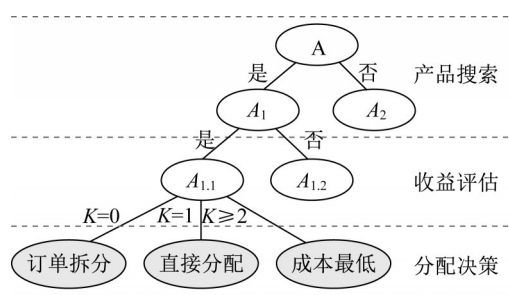


图2 第一级决策树:决策选择

Fig. 2 First level decision tree: decision selection

采用遗传算法进行拖期检测以获得 K 值。遗传算法是一种启发式算法,常被应用于柔性作业车间调度^[16]。本文在前人研究的基础上,对遗传算法进行改进,并将其应用于生产订单分配模型,实现订单在各车间的拖期检测功能,进行拖期检测可以解决现有仅靠非工作时间累加计算产能的产能评估不准确问题。拖期检测前需要进行库存检查,以修正实际需要生产的产品数量,如式(3)所示:

$$Q_{ij} = Q_i - q_{ij} \quad (3)$$

式中: Q_{ij} 为修正后所需生产的产品数量; q_{ij} 为 j 车间 i 订单产品库存数量。

采用Gao等^[17]所提的进化算法框架,不同之处在于选择策略和邻域搜索方法不同。

选择策略:为了控制多车间协同调度的鲁棒性,

将新订单加入视为动态调度过程,设定了一个优选准则,即:订单分配前车间按时完成的订单依旧按时完成;拖期完成的,订单分配后的完工时间不能超过原来的完工时间,其数学描述如式(4)所示:

$$\begin{cases} T_n^H \leq T_n, T_n^Y \leq T_n \\ T_n^H \leq T_n^Y, T_n^Y > T_n \end{cases} \quad (4)$$

式中: n 为车间中正在加工订单索引; T_n 为 n 订单交货期; T_n^Y 为新订单加入前 n 订单计划完工时间; T_n^H 为新订单加入后 n 订单计划完工时间。

因此,染色体选择采用层次选择法,首先满足公式(4)的染色体将优先进入下一次迭代;随后,最大完成时间(makespan)作为第二选择目标,makespan值小的染色体将被优先选择。

考虑到在实际生产中存在双休日 G 和假期 T_q ,且交货期 T_i 通常以天为单位,而遗传算法在求解最优值时是以小时为单位,不考虑非工作时间,因此,需要对 T_i 进行转换,以确保时间上的统一,转换表达式如式(5)所示:

$$T_i^* = \begin{cases} (T_i - 2G - T_q)t_m, \xi = 0 \\ (T_i - 2G - T_q + T_b)(t_m + t_b), \xi = 1 \end{cases} \quad (5)$$

式中: T_i^* 为转换后的交货期; ξ 代表是否加班,1为是,0为否; t_m 为每天法定工作时间; t_b 为每天法定工作时间外的加班时间; T_b 为法定节假日和双休日内需要加班的天数。

邻域搜索:基于关键路径的邻域搜索方法已被许多学者证实,是一种高效的局部最优解搜索算法,因此,本文采用赵诗奎^[18]所设计的邻域结构。与其不同之处在于,关键工序移动后是否产生最优解不再仅以最大完工时间减小为判断指标,同时还要满足公式(4)的限制。

当 $K=0$ 时,代表任何车间都无法独立完成新订单生产任务,选择订单拆分策略,进行多车间协同生产;当 $K=1$ 时,代表只有一个车间能够按时交货,选择该车间生产新订单,此时总成本不再仅仅包括生产成本和运输成本,还包括订单提前完工惩罚金 P_{ij} ,修正后的总成本 $C(S_{ij}, D_{ij}, P_{ij})$ 表达式如式(6)和式(7)所示:

$$t_{ij} = \begin{cases} L + 1 + 2G + T_q, L < \left| \frac{T_{ij}^* - T_i^*}{t_m + t_b} \right| < L + 1 \\ L + 2G + T_q, \left| \frac{T_{ij}^* - T_i^*}{t_m + t_b} \right| = L \end{cases} \quad (6)$$

$$C(S_{ij}, D_{ij}, P_{ij}) = d_{ij}Q_i + w_{ij}l_{ij}\varphi + \alpha_i t_{ij}Q_i \quad (7)$$

式(6)—(7)中: t_{ij} 为 j 车间生产 i 订单的提前天数,包括提前期内双休日和假期,不足一天按一天计算; T_{ij}^* 为转换后 j 车间生产 i 订单的计划完成时间; L 为计算所得整数; α_i 为 i 订单提前惩罚单价。

当 $K \geq 2$ 时,表示至少有两个车间可以按时交货,此时选择总成本最低的车间进行生产,其目标函数如式(8)所示:

$$f = \min C(S_{ij}, D_{ij}, P_{ij}) \quad (8)$$

3.2 第二级决策树:订单拆分

订单拆分的目的在于确定各车间能够完成的产品数量,实现多车间协同生产,从而确保按时交货。订单拆分遵循最小成本优先原则,即:总成本 $C(S_{ij}, D_{ij})$ 最小的车间优先确定生产数量,剩余数量在其他车间依次进行拆分,如图3所示。

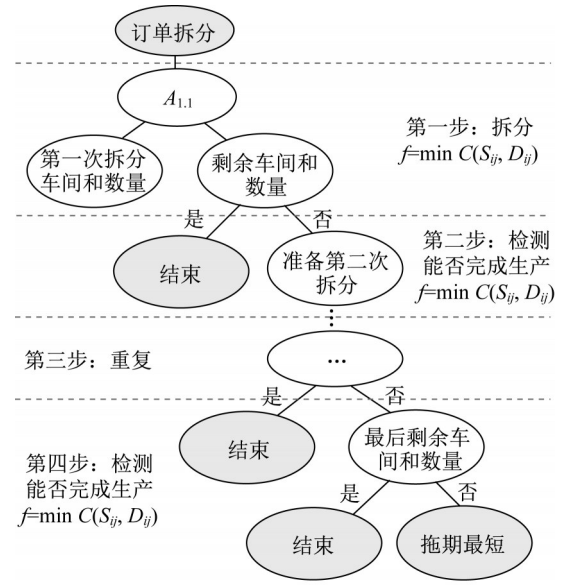


图3 第二级决策树:订单拆分

Fig. 3 Second level of decision tree: order split

如果订单拆分到最后一个车间之前,能够按时完成订单生产任务,或者拆分到最后一个车间的上一层,且剩余订单数量最后一个车间能够按期完成,则拆分成功,按拆分结果执行生产;如果拆分到最后一个车间,仍不能按时完成生产任务,则订单拆分失败,优化目标转变为拖期时间最短,其数学描述如式(9)所示:

$$\begin{cases} l, \delta = 1 \\ l = l + 1, \delta = 0 \\ f = \min(T_{ij}^* - T_i^*), \delta = -1 \end{cases} \quad (9)$$

式中: δ 代表订单拆分决策标签,当 $\delta=1$ 时,拆分成功,此时的拆分次数为 l ;当 $\delta=0$ 时,代表继续拆分,

此时拆分次数自增1;当 $\delta = -1$,代表拆分失败,目标函数转变为拖期时间最短。

与迭代求解算法相比,线性规划具有运算速度快的特点,考虑到企业对实时响应能力的需求,本文基于线性规划理论提出一个边界限定准则来快速确定订单拆分数目 S ,包括上界 S_a 和下界 S_d ,其中,上界为理想状态下能够完成的最大数量,下界为当前状态下能够完成的最小数量。假设待分配订单 i 有 x 道工序,每道工序对应的加工设备和加工时间分别为 B_x 和 t_x 。上界用设备的空闲时间与工序 x 在该设备上的加工时间 t_x 的比值来确定,表达式如式(10)所示:

$$S_a = \min \left[\frac{T_i^* - \sum T^*(B_x, \Gamma)}{t_x} \right] \quad (10)$$

式中: $\sum T^*(B_x, \Gamma)$ 为设备 B_x 的工作时间段; Γ 代表设备工作状态。

下界通过订单 i 规定交期时间占计划加工时间的比例来确定,表达式如式(11)所示:

$$S_d = \frac{T_i^*}{T_{ij}^*} Q_i \quad (11)$$

拆分数目优选下界值,为了追求数量的最大化,可以适当提高数量值,但是不能达到上界值,其表达式如式(12)所示:

$$S_d \leq S + \beta(S_a - S_d) < S_a, \beta \in [0, 1) \quad (12)$$

式中: β 为订单拆分量控制系数。

当订单拆分失败时,由拖期时间最短的车间生产该订单,此时,总成本不再是 $C(S_{ij}, D_{ij})$,还包括拖期惩罚 H_{ij} ,因此,修正后的总成本 $C(S_{ij}, D_{ij}, H_{ij})$,表达式如式(13)和式(14)所示:

$$v_{ij} = \begin{cases} L + 1 + 2G + T_q, & L < \frac{T_{ij}^* - T_i^*}{t_m + t_b} < L + 1 \\ L + 2G + T_q, & \frac{T_{ij}^* - T_i^*}{t_m + t_b} = L \end{cases} \quad (13)$$

$$C(S_{ij}, D_{ij}, H_{ij}) = d_{ij} Q_i + w_{ij} l_{ij} \varphi + \mu_i v_{ij} Q_i \quad (14)$$

式(13)—(14)中: μ_i 为订单 i 的拖期惩罚单价; v_{ij} 为订单 i 在车间 j 生产时的拖期时间,包括拖期内双休日和假期,不足一天按一天算。

式(1)—(2)、式(5)—(7)以及式(13)—(14)均为根据研究对象工况设定,实际应用时可根据企业工况特征进行替换或者更改相应数据的计算方法。

4 案例验证

利用Python编程语言实现所提生产订单分配模型,将拖期检测算法封装成多个函数,每个函数代表一个车间,每个车间的当前数据提前录入函数中,通过函数调用来模拟生产订单分配的决策过程。利用基准案例和实际生产数据对模型进行测试,测试所用计算机为Intel i5-6400处理器,主频2.70 GHz,内存8.00 G。参数设置为:种群规模300,交叉率0.76,变异率0.4,迭代次数200。本节中 N 表示待分配订单。

4.1 测试拖期检测算法

以Zhang等^[19]所提的基准案例为基础进行测试,订单信息如表1所示。表中, $J_1 \sim J_4$ 代表待分配订单加入前已有的订单, $M_1 \sim M_5$ 为生产机器, $O_{11} \sim O_{N4}$ 为工序,测试通过对比拖期检测算法在有约束下和无约束下求得的各订单完工时间,来证明所提拖期检测算法的可行性。

表1 订单信息表

Tab. 1 Information of orders

订单	工序	加工时长/h					交货期/h
		M_1	M_2	M_3	M_4	M_5	
J_1	O_{11}	2	6	5	3	4	14
	O_{12}		8		4		
J_2	O_{21}	3		6		5	17
	O_{22}	4	6	5			
	O_{23}		7	11	5	8	
	O_{31}	5			3	7	
J_3	O_{32}		8		6		12
	O_{33}	3	7	2			
	O_{34}	6		8	5		
	O_{41}		5		2	4	
J_4	O_{42}	8	6			10	15
	O_{N1}	9	11	7		8	
N	O_{N2}	6		5		10	25
	O_{N3}	9	5		5	7	
	O_{N4}		6	8	9		

待分配订单加入前车间生产计划如图4所示,由图可知,订单 $\{J_1, J_2, J_3, J_4\}$ 所对应的完工时间分别为 $\{13 \text{ h}, 17 \text{ h}, 17 \text{ h}, 10 \text{ h}\}$,其中订单 $\{J_1, J_2, J_4\}$ 能够按时交货, J_3 延期交货。待分配订单加入后,如果不添加约束,则拖期检测算法将按照生产完工时间最短进行排程,结果如图5所示,由图可知,订单 $\{J_1, J_2, J_3, J_4\}$ 所对应的完工时间分别变为 $\{9 \text{ h}, 17 \text{ h}, 22 \text{ h}, 17 \text{ h}\}$,这种转变的不足之处在于:虽然最大

完工时间小,然而,原来能够按时交货的订单有可能会延期交货,如订单 J_4 ;原来延期交货的订单延期时间有可能变长,如订单 J_3 ,延期时间由5 h增加到10 h。因此,为了避免新订单的加入导致原有订单生产计划优化结果失效现象的发生,加入约束条件至关重要。

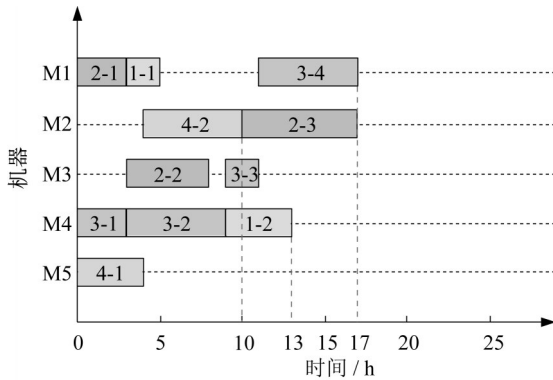


图4 待分配订单加入前的生产计划

Fig. 4 Production plan before adding the order to be allocated

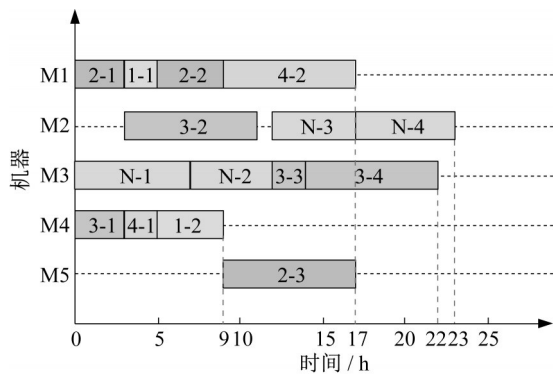


图5 待分配订单加入后的生产计划(无约束)

Fig. 5 Production plan after adding the order to be allocated (unconstrained)

根据拖期检测算法约束条件的限制,待分配订单加入后,原有订单的交货期应满足: $T_{J1} \leq 14$ h, $T_{J2} \leq 17$ h, $T_{J3} \leq 17$ h, $T_{J4} \leq 15$ h。使用带约束的拖期检测算法进行排程,结果如图6所示,由图可知,订单 $\{J_1, J_2, J_3, J_4\}$ 所对应的完工时间分别为 $\{13$ h, 16 h, 16 h, 13 h $\}$,满足约束条件。因此,得出结论:约束在求解过程中起到了作用,增强了多车间协同调度的鲁棒性。另外,绘制了目标值随迭代次数增加而变化的曲线,如图7所示,由图可以看出,该算法能够快速收敛。

4.2 测试生产订单分配模型

由于本文研究的重点在于如何确保生产计划的

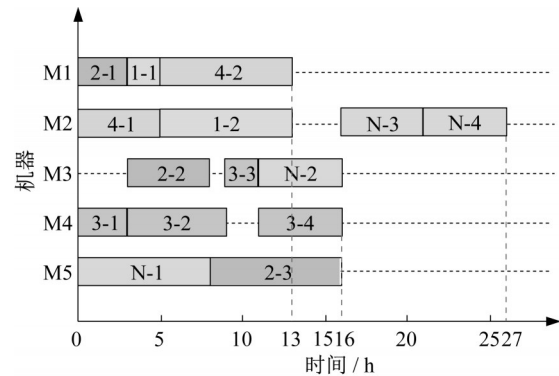


图6 待分配订单加入后的生产计划(有约束)

Fig. 6 Production plan after adding the order to be allocated (with constraints)

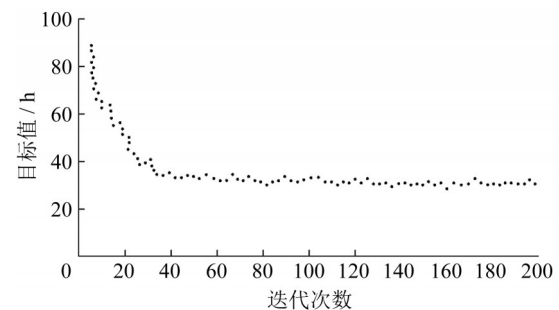


图7 目标函数值与迭代次数变化趋势图

Fig. 7 Objective value versus number of iterations

鲁棒性,考虑的是每个订单的完工情况,无法与其他优秀算法直接进行结果上的对比。因此,本文简化某国企一部分实例数据进行算法性能测试,该测试基于一条假设,即:车间具有完整的供应链,能够有效保证生产过程中的原材料供给。

测试案例中待分配订单N的销售金额为16 000元,要求生产的产品数量为125个,最低收益率为25%。表2为待分配订单N在各车间的基础生产信息,其中 $\{A, B, C, D, E\}$ 代表5个处于不同地理位置的车间,D不具备生产该订单的能力。表3为车间 $\{B, E\}$ 的订单及其工序信息表,其中 N_B 和 N_E 分别为B车间和E车间生产待分配订单N的工艺信息, $\{J_{B1}, J_{B2}, J_{B3}\}$ 和 $\{J_{E1}, J_{E2}, J_{E3}, J_{E4}\}$ 分别为待分配订单加入前B车间和E车间原有订单及其工序信息。待分配订单的单件产品提前完成惩罚金 α_i 和拖期惩罚金 μ_i 分别为0.8元和0.5元。

生产订单分配第一步是产品搜索,第二步是效益评估,由表4可知:参与效益评估的车间集合为 $\{A, B, C, E\}$,其中,B和E车间的收益率分别为31.73%和48.15%,满足最低收益率R的要求,因此,生产订单分配在车间 $\{B, E\}$ 中进行。

表2 待分配订单信息表

Tab. 2 Information of order to-be-allocated

项目	车间				
	A	B	C	D	E
单品生产成本/元	110	98	125		89
运输单价/元	1.2	1.7	1.5		1.4
运输距离/km	217	86	152		42
货物重量/kg	20	22	19		17
库存数量/个	0	5	10		5

生产订单分配第三步:分配决策,由表5知,B和E都无法独立完成待分配订单N的生产任务,拖期时间分别为58 h和176 h,因此,做出判断:执行订单拆分策略。根据订单拆分的原则,总成本最小的车间优先拆分,由表4可知,B和E车间生产待分配订单N的总成本分别为12 146元和10 800元,车间E优先进行订单拆分,由于该产品在车间E中有库存,库存数量值由表2可知为5个,因此实际需要生产的总量为120个。依据公式(10)和(11)可以计算出待分配订单N在车间E中拆分的下界值为95,上界值为161,根据下界优先原则,确定E车间生产的产品数量为95个。剩余25个产品除去B车间的5个库存后进行拖期检测,结果显示能够按期完成,订单拆分成功。待分配订单拆分后{B, E}车间的生产计划如表6所示,此时,该订单提前完成时间分别为8 d和2 d。因此,各车间生产该订单的成本应该还包括提前完成惩罚金,B和E修正后的总成本分别为2 690.41元和9 159.96元,一个企业完成该订单的总成本为2 690.41+9 159.96=11 886.37元。修正总成本金额可以为企业统计生产信息提供更准确的成本数据,使成本核算更加贴近实际。

测试结果表明,本文所提订单可分解多柔性作业车间协同鲁棒调度方法可行有效。

有时企业为了追求利益最大化,会选择在成本最低的车间生产尽可能多的产品,不采用下界优先原则,针对这种情况,本文对公式(12)中的 β 值进行了探讨,绘制了订单提前完工时间以及订单拆分数量与 β 值的变化曲线,如图8所示。可以看出,随着 β 值的增加,订单拆分的数量呈线性增长,但是提前完工时间会逐渐降低,当 β 超过一定值时,提前完工时间变为负值,意味着拖期交货。因此,在追求利益最大化的过程中要合理选取 β 值,以保证客户的最佳体验。

表3 车间生产订单及其工序信息表

Tab. 3 Information of workshop production order and its operation

车间	订单	工序	加工时间/h							数量/个	交货期/d
			M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₇		
B	J _{B1}	O ₁₁			1.2	0.8		0.6			
		O ₁₂	2.1		1.9		2.4				
		O ₁₃						0.9	0.6	50	28
		O ₁₄	1.1	0.7		1.5					
		O ₁₅	0.5	0.8	0.6				0.5		
	J _{B2}	O ₂₁	0.4		0.6		0.6				
		O ₂₂	1.7	2.4		1.4		2.2		70	34
		O ₂₃			0.9		0.7		1.2		
		O ₂₄	0.2				0.3	0.2	0.5		
	J _{B3}	O ₃₁		0.4	0.4	0.6					
		O ₃₂			0.5	0.8		1.1			
		O ₃₃	0.2	0.3					0.5	80	37
		O ₃₄	1.1	1.4		0.9	0.7		1.2		
		O ₃₅				0.9		1.3	0.7		
	N _B	O _{N1}		0.6		0.9	1.1				
		O _{N2}	0.5		0.3			0.6	0.5		
		O _{N3}	1.1			0.9				125	28
		O _{N4}					0.2	0.3	0.2		
		O _{N5}		0.7	1.3						
		O _{N6}	0.3		0.3		0.4				
E	J _{E1}	O ₁₁		0.4		0.7					
		O ₁₂	0.3	0.6			0.4		0.3		
		O ₁₃			0.8	1.1		1.3		60	25
		O ₁₄	0.5				0.2	0.3	0.2		
		O ₁₅		1.1		1.3					
		O ₁₆			0.2		0.3				
	J _{E2}	O ₂₁	1.2			0.9	1.5		0.9		
		O ₂₂	0.3	0.1							
		O ₂₃			0.5		0.7	0.5	1.1	50	20
		O ₂₄		0.6	1.2	0.7					
	J _{E3}	O ₃₁	0.5			0.6			0.3		
		O ₃₂		0.5	0.4		0.5				
		O ₃₃				1.1		0.9		40	20
		O ₃₄	0.3				0.7	0.5	0.7		
		O ₃₅		0.3	0.5				0.3		
	J _{E4}	O ₄₁	1.6				0.9	1.2			
		O ₄₂	0.5					0.7	1.2	70	23
		O ₄₃		0.8		0.7	1.1				
	N _E	O _{N1}	0.3	0.2	0.5				0.3		
		O _{N2}					0.9	1.3	0.8		
		O _{N3}		0.3			0.5	0.2		125	28
		O _{N4}	0.6		0.5			0.3			
		O _{N5}	0.2			0.4		0.5	0.4		

表4 收益评估信息表

Tab. 4 Information of income evaluation

项目	车间				
	A	B	C	D	E
总成本/元	13 825	12 146	15 520		10 800
收益率/%	15.73	31.73	3.09		48.15

表5 待分配订单拆分前车间调度结果

Tab. 5 Workshop scheduling results before splitting the order to be allocated

车间	订单	加工过程/h					
		O_{n1}	O_{n2}	O_{n3}	O_{n4}	O_{n5}	O_{n6}
B	J _{B1}	{M ₆ : 0—30}	{M ₃ : 30—125}	{M ₇ : 125—155}	{M ₂ : 155—190}	{M ₁ : 190—215}	
	J _{B2}	{M ₅ : 0—42}	{M ₄ : 42—140}	{M ₅ : 192—241}	{M ₁ : 241—255}		
	J _{B3}	{M ₂ : 0—32}	{M ₆ : 32—120}	{M ₁ : 120—136}	{M ₅ : 136—192}	{M ₄ : 192—264}	
B(N)	J _{B1}	{M ₆ : 0—30}	{M ₃ : 32—127}	{M ₇ : 132—162}	{M ₂ : 162—197}	{M ₁ : 197—222}	
	J _{B2}	{M ₁ : 0—28}	{M ₄ : 28—126}	{M ₅ : 126—175}	{M ₆ : 175—189}		
	J _{B3}	{M ₃ : 0—32}	{M ₆ : 32—120}	{M ₁ : 120—136}	{M ₅ : 175—231}	{M ₇ : 231—287}	
	N _B	{M ₂ : 0—72}	{M ₇ : 72—132}	{M ₄ : 132—240}	{M ₅ : 240—264}	{M ₂ : 264—348}	{M ₅ : 348—396}
E	J _{E1}	{M ₂ : 0—24}	{M ₇ : 24—42}	{M ₃ : 42—90}	{M ₅ : 90—102}	{M ₂ : 102—168}	{M ₃ : 168—180}
	J _{E2}	{M ₄ : 0—45}	{M ₂ : 45—50}	{M ₆ : 50—75}	{M ₃ : 90—150}	{M ₇ : 150—165}	
	J _{E3}	{M ₇ : 0—12}	{M ₃ : 12—24}	{M ₆ : 75—111}	{M ₁ : 111—123}	{M ₂ : 168—180}	
	J _{E4}	{M ₅ : 0—63}	{M ₁ : 63—98}	{M ₄ : 98—147}			
E(N)	J _{E1}	{M ₂ : 0—24}	{M ₁ : 24—42}	{M ₃ : 42—90}	{M ₅ : 90—102}	{M ₄ : 102—180}	{M ₅ : 180—198}
	J _{E2}	{M ₄ : 0—45}	{M ₂ : 45—50}	{M ₃ : 90—115}	{M ₂ : 115—145}	{M ₆ : 145—155}	
	J _{E3}	{M ₇ : 0—12}	{M ₃ : 12—24}	{M ₆ : 24—60}	{M ₁ : 98—110}	{M ₂ : 145—157}	
	J _{E4}	{M ₅ : 0—63}	{M ₁ : 63—98}	{M ₅ : 102—179}			
	N _E	{M ₂ : 50—74}	{M ₇ : 74—170}	{M ₆ : 170—194}	{M ₃ : 194—254}	{M ₁ : 254—278}	

表6 待分配订单拆分后车间调度结果

Tab. 6 Workshop scheduling results after splitting the order to be allocated

车间	订单	加工过程/h					
		O_{n1}	O_{n2}	O_{n3}	O_{n4}	O_{n5}	O_{n6}
E (N: 95)	J _{E1}	{M ₂ : 0—24}	{M ₁ : 24—42}	{M ₃ : 42—90}	{M ₅ : 90—102}	{M ₄ : 102—180}	{M ₃ : 185.5—197.5}
	J _{E2}	{M ₄ : 0—45}	{M ₂ : 45—50}	{M ₃ : 90—115}	{M ₂ : 115—145}	{M ₆ : 145—155}	
	J _{E3}	{M ₇ : 0—12}	{M ₃ : 12—24}	{M ₆ : 24—60}	{M ₁ : 60—72}	{M ₇ : 119—131}	
	J _{E4}	{M ₅ : 0—63}	{M ₁ : 72—107}	{M ₅ : 107—184}			
	N _E	{M ₂ : 24—43}	{M ₇ : 43—119}	{M ₆ : 119—138}	{M ₃ : 138—185.5}	{M ₁ : 185.5—204.5}	
B (N: 20)	J _{B1}	{M ₆ : 0—30}	{M ₃ : 30—125}	{M ₇ : 125—155}	{M ₂ : 155—190}	{M ₁ : 190—215}	
	J _{B2}	{M ₁ : 0—28}	{M ₄ : 56—154}	{M ₅ : 204—253}	{M ₆ : 253—267}		
	J _{B3}	{M ₂ : 12—44}	{M ₆ : 44—132}	{M ₁ : 132—148}	{M ₅ : 148—204}	{M ₇ : 204—260}	
	N _B	{M ₂ : 0—12}	{M ₁ : 28—38}	{M ₄ : 38—56}	{M ₆ : 132—138}	{M ₂ : 138—152}	{M ₁ : 152—158}

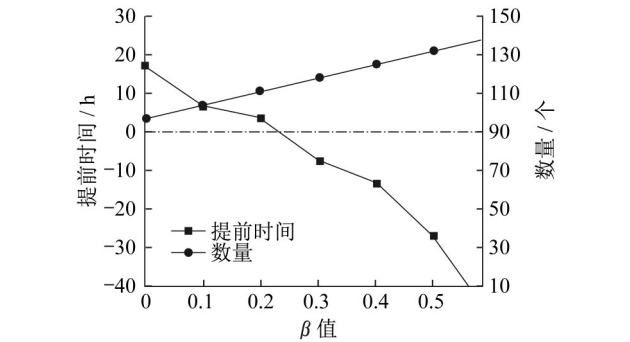


图8 订单提前时间和数量与 β 值关系图

Fig. 8 β value versus order advance time and quantity

5 结束语

本文根据某国有企业多车间协同生产的实际需求,对订单可分解多柔性作业车间协同鲁棒调度问题展开研究。给出了多车间协同调度的求解方案。通过两级决策树模型实现了生产订单分配。通过遗

传算法进行拖期检测,提高了产能评估的准确性。将新订单加入视为动态调度,在遗传算法中使用染色层次选择法,提高了调度的鲁棒性。基于线性规划思想提出了订单分解的上下界准则。最后通过一个基准案例和一部分车间生产实例对模型性能进行测试,测试结果表明该模型可行有效。

作者贡献声明:

- 李营力:完成实验并撰写文章。
- 王家海:提出研究方向,给予建设性建议。
- 刘银冬:协助实验,分析数据。
- 刘江山:给出论文修改建议和格式检查。

参考文献:

[1] LI Y, WANG J, LIU Z. An adaptive multi-objective evolutionary algorithm with two-stage local search for flexible job-shop scheduling[J]. International Journal of Computational Intelligence Systems, 2021, 14(1): 54.

- [2] LI Y, WANG J, LIU Z. A simple two-agent system for multi-objective flexible job-shop scheduling [J]. *Journal of Combinatorial Optimization*, 2022, 43(1): 42.
- [3] 顾幸生, 丁豪杰. 面向柔性作业车间调度问题的改进博弈粒子群算法[J]. *同济大学学报(自然科学版)*, 2020, 48(12): 1782.
GU Xingsheng, DING Haojie. An improved gaming particle swarm optimization algorithm for flexible job-shop scheduling problems[J]. *Journal of Tongji University (Natural Science)*, 2020, 48(12): 1782.
- [4] 王家海, 李营力, 刘铮玮, 等. 柔性作业车间调度的精确邻域结构混合进化算法[J]. *同济大学学报(自然科学版)*, 2021, 49(3): 440.
WANG Jiahai, LI Yingli, LIU Zhengwei, *et al.* Evolutionary algorithm with precise neighborhood structure for flexible workshop scheduling[J]. *Journal of Tongji University (Natural Science)*, 2021, 49(3): 440.
- [5] BEHNAMIAN J, GHOMI S. Multi-objective multi-factory scheduling[J]. *Rairo Operations Research*, 2021, 55: S1447.
- [6] GHARAEI A, JOLAI F. A multi-agent approach to the integrated production scheduling and distribution problem in multi-factory supply chain[J]. *Applied Soft Computing*, 2018, 65: 577.
- [7] MAO J, PAN Q, MIAO Z, *et al.* An effective multi-start iterated greedy algorithm to minimize makespan for the distributed permutation flowshop scheduling problem with preventive maintenance[J]. *Expert Systems with Applications*, 2021, 169: 114495.
- [8] 单汨源, 蔡自兴, 高阳. 一种多级分布式制造系统的多智能体协同生产机制[J]. *控制与决策*, 2001, 16(4): 410.
SHAN Miyuan, CAI Zixing, GAO Yang. The synergic production mechanism based on multi-agent for a multi-stage distributed manufacturing system [J]. *Control and Decision*, 2001, 16(4): 410.
- [9] 蒋大奎, 李波, 谭佳音. 一类求解订单分配和排序问题的集成优化算法[J]. *控制与决策*, 2013, 28(2): 217.
JIANG Dakui, LI Bo, TAN Jiayin. Integrated optimization approach for order assignment and scheduling problem [J]. *Control and Decision*, 2013, 28(2): 217.
- [10] 杨红红, 吴智铭. 基于两级遗传算法的多工厂供应链批量计划问题[J]. *上海交通大学学报(自然科学版)*, 2003, 37(4): 473.
YANG Honghong, WU Zhiming. Two level GA-based approach to the capacitated lot sizing problem for multi-plants supply chain [J]. *Journal of Shanghai Jiaotong University (Natural Science)*, 2003, 37(4): 473.
- [11] 王静, 潘开灵, 刘翔, 等. 云制造平台下订单可分解的协同生产计划模型及求解[J]. *上海交通大学学报(自然科学版)*, 2018, 52(12): 1655.
WANG Jing, PAN Kailing, LIU Ao, *et al.* The model and solution for collaborative production planning with order splitting in cloud manufacturing platform [J]. *Journal of Shanghai Jiaotong University (Natural Science)*, 2018, 52(12): 1655.
- [12] 王旭亮, 柴旭东, 张程, 等. 云制造环境下跨企业协同生产调度算法[J]. *计算机集成制造系统*, 2019, 25(2): 412.
WANG Xuliang, CHAI Xudong, ZHANG Cheng, *et al.* Collaborative production planning algorithm for cross-enterprises in cloud manufacturing [J]. *Computer Integrated Manufacturing System*, 2019, 25(2): 412.
- [13] 沈广亚, 李立恒, 张宁. 面向订单的航空制造业多车间生产与运输综合计划模型[J]. *北京航空航天大学学报*, 2020, 46(7): 1422.
SHEN Guangya, LI Liheng, ZHANG Ning. An integrated planning model for production and transportation in make-to-order multi-site aviation manufacturing industry [J]. *Journal of Beijing University of Aeronautics and Astronautics*, 2020, 46(7): 1422.
- [14] 陈友玲, 牛禹霏, 刘舰, 等. 面向云制造的多供应商协同生产任务分配优化[J]. *计算机集成制造系统*, 2019, 25(7): 1806.
CHEN Youling, NIU Yufei, LIU Jian, *et al.* Task distribution optimization for multi-supplier collaborative production in cloud manufacturing [J]. *Computer Integrated Manufacturing System*, 2019, 25(7): 1806.
- [15] VENS C, STRUYF J, SCHIETGAT L, *et al.* Decision trees for hierarchical multi-label classification [J]. *Machine Learning*, 2008, 73(2): 185.
- [16] TEEKENG W, THAMMANO A. Modified genetic algorithm for flexible job-shop scheduling problems [J]. *Procedia Computer Science*, 2012, 12: 122.
- [17] GAO J, SUN L, GEN M. A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems [J]. *Computers & Operations Research*, 2008, 35(9): 2892.
- [18] 赵诗奎. 柔性作业车间调度的改进邻域结构混合算法[J]. *计算机集成制造系统*, 2018, 24(12): 144.
ZHAO Shikui. Hybrid algorithm based on improved neighborhood structure for flexible job shop scheduling [J]. *Computer Integrated Manufacturing Systems*, 2018, 24(12): 144.
- [19] ZHANG G, GAO L, SHI Y. An effective genetic algorithm for the flexible job-shop scheduling problem [J]. *Expert Systems with Applications*, 2011, 38(4): 3563.