运筹学与控制论

DOI:10.11721/cqnuj20250405

异构并行车间多目标排产优化粒子群算法研究*

葛建军,陈 鑫,温玉玺
(辽宁工业大学 电子与信息工程学院,辽宁 锦州 121001)

摘要:针对异构并行车间排产优化问题,提出了一种改进的多目标粒子群算法,以提高生产效率,进而促进企业利润最大化。建立了基于生产加工成本与工件拖期时间的多目标优化模型,在算法中针对惯性权重、全局最优解、外部存档等方面进行了优化设计。通过动态调整惯性权重增强粒子群在解空间中的搜索能力,通过全局比例排序和计算循环距离寻找全局学习样本以提高算法效率。实验结果表明,算法在研究问题上实现了 2 个目标之间的均衡,可生成一定数量的 Pareto 最优解,供决策者选择。

关键词:异构并行车间;排产优化;多目标;粒子群算法

中图分类号:TP18 **文献标志码**:A **文章编号**:1672-6693(2025)04-0024-10

随着信息化的发展,传统工厂向智能工厂的转变已经成为新的趋势。与传统工厂人工排产不同,智能工厂利用先进的制造系统和信息系统寻求更优的排产。排产的核心在于如何在合理的时间内将有限的资源分配给各个任务,以满足 1 个或多个优化目标[1]。在生产过程中,优秀的排产计划可以缩短完工时间、减少成本、提升生产效率和机器使用率,因此研究生产制造业的排产问题具有重要的现实意义。

在车间生产过程中,通常会有多台机器同时运行,这些机器能够并行处理多个不同的任务。如何合理地为这些机器分配任务以最大化整体生产效率,即为并行机调度问题。一直以来,并行机调度问题受到了广大学者的关注与研究[2],且已被证明为 NP-困难问题。根据机器的特性,并行机调度可细分为等同并行机调度与非等同并行机调度。在等同并行机调度方面,Anghinolfi 等人[3]提出了一种多目标优化算法,旨在同时最小化完工时间和总能耗,实验证明该方法在不同电价政策下表现优异;Kim 等人[4]针对具有作业分割和顺序相关设置时间的问题,提出了一种元启发式算法,结果显示该方法在解的质量和计算效率上优于商业求解器;Loffredo 等人[5]通过能效控制策略优化了多阶段等同并行机生产线,以在满足生产约束的同时最小化能耗,实验验证了该方法的有效性。非等同并行多机调度方面,Tan 等人[6]分别用单种群和多种群遗传算法优化非等同并行机的绿色批调度,目标为减少总电费,实验证明多种群算法效果更佳;Diamantidis 等人[7]提出了一种高效分解技术,用于分析具有有限缓冲区、非等同和非可靠并行机的串行生产线,实验结果表明该方法在处理大规模系统时,较其他方法具有更高的计算效率和准确性;Noman 等人[8]提出了一种基于下界和精确算法的改进禁忌搜索算法,用于最小化非等同并行机调度问题中的完工时间。结果显示,该算法在求解速度和解的质量上优于文献中的方法;Beldar 等人[9]针对非等同并行机批处理问题,提出了基于模拟退火和可变邻域搜索的元启发式方法,目标为最小化总延迟时间,实验结果表明所提算法在大多数情况下优于其他算法;Bok 等人[10]提出了一个多目标模型,优化非等同并行机在高混合、低批量生产中的调度问题,目标为降低生产成本和碳排放,并通过实验验证了该模型,显示了该模型在绿色制造中的应用潜力。在实际生产环境中,由于设备的老化与更新,车间内存在非等同并行机的现象变得愈加普遍。这些由非等同并行机组成的车间通常被称为异构并行车间。现有研究中,以车间为基本单位进行调度的文献相对较少。本文将采用一种改进的多目标粒子群算法(multi-objective particle swarm optimization,MOPSO)对异构并行车间进行排产优化,并通过数值实验验证该算法的较优 Pareto 解。

* **收稿日期**:2024-09-01 **修回日期**:2025-01-17 **网络出版时间**:2025-05-18T07:01
资助项目:国家自然科学基金面上项目(No. 12371363);辽宁省“兴辽英才计划”项目(No. XLYC2403125)
第一作者简介:葛建军,男,研究方向为组合优化问题,E-mail:gjj201869@163.com;**通信作者简介**:陈鑫,男,教授,博士,E-mail:chenxin.lut@hotmail.com
网络出版地址:https://link.cnki.net/urlid/50.1165.N.20250507.1635.017

1　问题描述与建模

本问题中,加工任务需要根据合同号和订单号进行编号。每个合同包含若干个订单,每个订单由若干同种产品组成,订单可以按产品数量拆分成若干份生产,每个订单或者产品由不同的物料按比例生产,订单分为 A、B 这 2 种类型。加工机器存在于 2 种类型的车间:混流车间和非混流车间。区别在于混流车间允许同时加工不同合同的订单,而非混流车间则不允许。本文研究的对象为混流车间,调度以车间为单位,每个车间有若干个机器,机器之间功能一致,但根据加工速度的不同分为 A、B 这 2 种类型。

本问题的约束如下:

1) 同一合同的不同订单需按照先后顺序生产,且尽量安排在同一机器上进行;

2) 加工时需按物料消耗系数与加工单位数量的乘积消耗,物料不足时无法生产;

3) 机器的能力按消耗系数与加工数量的乘积进行消耗,能力不足时无法生产,每次合同号切换,都会对当天机器的能力造成扣减,扣减量为当天能力总数乘以能力损失百分比;

4) 任意时刻,车间可同时加工 N 个订单(N 为车间参数中设置的最大并行加工订单数);

5) 前 6 周的最大提前生产天数为 10 d,其余时间的最大提前生产天数为 14 d;

6) 任务在同一台机器上需连续生产至少 γ h,γ 为超参数。

表 1 为模型参数,表 2 为模型的决策变量。

表 1　模型参数

Tab. 1　Model parameters

模型参数	含义
j	任务集合,$j=\{1,2,3,\cdots,n\}$,其中 n 为任务数
M	机器集合,$M=\{1,2,3,\cdots,m\}$,其中 m 为机器数
P_{ij}	任务 j 在机器 i 上处理时间
G	无穷大正数

表 2　模型决策变量

Tab. 2　Decision variables of the model

模型决策变量	含义
S_{ij}	任务 j 在机器 i 上的开始处理时间
E_{ij}	任务 j 在机器 i 上的完成时间
C_j	任务 j 的加工总成本
D_i	机器 i 上的合同切换成本
T_j	任务 j 的拖期时间
X_{hj}	若任务 h 为任务 j 的紧前任务,则 $X_{hj}=1$;否则 $X_{hj}=0$

本问题的数学模型如下。

目标函数:

$$\min\left(\sum_{j=1}^{n}C_j+\sum_{i=1}^{m}D_i\right), \tag{1}$$

$$\min\sum_{j=1}^{n}T_j。 \tag{2}$$

约束函数:

$$S_{ih}+P_{ih}\leqslant S_{ij}+G\cdot(1-X_{hj}), \tag{3}$$

$$X_{hj}+X_{jh}=1, \tag{4}$$

$$C_j\geqslant 0, \tag{5}$$

$$S_{ij}\geqslant 0,\ E_{ij}\geqslant 0, \tag{6}$$

$$X_{hj}\in\{0,1\}。\tag{7}$$

其中:$i=1,2,3,\cdots,m$; $h=1,2,3,\cdots,n$; $j=1,2,3,\cdots,n$;式(1)和(2)表示目标函数为最小化总生产成本和最小化总拖期时间,总生产成本和总拖期时间是生产过程中的关键指标,它们相互影响且需权衡优化以实现高效生产和客户满意度的平衡;式(3)表示同一时间每台机器只能处理1个任务;式(4)表示机器上任务的顺序关系;式(5)、(6)及(7)定义了变量的取值范围。C_j 成本构成根据5种约束及对应的惩罚成本单位计算后汇总,计算方式为:1) 跨类型成本:加工决策中A类任务在B类机器上生产或B类任务在A类机器上生产时会产生跨类型生产成本;2) 过早开工成本:加工决策中有加工时间小于加工任务的要求完工时间减去最大提前加工天数时即会产生过早开工成本;3) 缺货成本:加工决策中实际加工数量小于加工任务要求的加工数量即会产生缺货成本;4) 粗排产结果偏差成本:加工决策中加工日期小于粗排产加工时间开始的部分或完工日期大于粗排产加工结束时间即会产生粗排产偏差成本。D_j 成本为跨合同切换成本,即同一机器在连续时间上生产不同合同号的任务时会产生跨合同切换(需同时考虑机器的最后加工合同号)成本。T_j 为该任务的完工时间与规定交货期的差值。

2 MOPSO

粒子群算法(particle swarm optimization,PSO)是一种基于群体智能的优化算法,由Kennedy等人在1995年提出[11-12]。由于参数少、易于理解及较强的鲁棒性等特点,PSO在许多单目标优化问题中得到成功应用[13-15]。

MOPSO是一种基于PSO的进化计算技术,旨在同时优化多个相互冲突的目标函数[16]。MOPSO在处理多目标优化问题时具备高效的全局搜索能力和保持解集多样性的优势[17],通过粒子群的协同进化,能够有效地平衡生产成本与拖期时间等相互冲突的目标,并适应动态环境的变化。因此,该算法被选用于解决异构并行车间的排产优化问题。

2.1 算法流程

本算法流程如图1所示。输入数据后,首先生成一定比例的精英个体(剩余个体随机编码),随后计算每个初始解的适应度。接着,通过排序和循环距离的策略选取全局学习样本。随后,遍历当前种群中的每个粒子,基于全局学习样本和粒子的历史最优位置更新位置和速度。最后,将更新后的粒子加入到下一代种群中,并更新每个粒子的适应度和历史最优位置。结合全局最优和个体历史最优的信息,算法能够逐步逼近多目标优化问题的最优解。

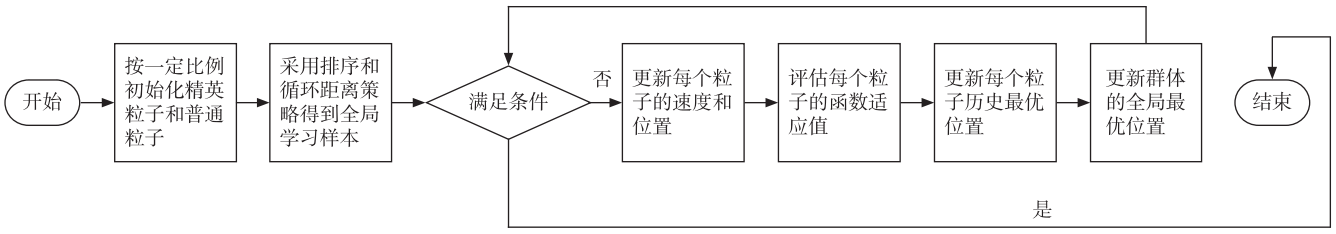

图1 改进的多目标粒子群算法流程图

Fig. 1 Flow chart of improved multi-objective particle swarm optimization

2.2 编码设计

在本算法中,为解决同一合同的不同订单尽量在同一机器上生产的问题,采用合同号进行编码(表3)。初始化种群时进行以下操作:1) 根据完工时间对待加工任务的合同号进行权重赋值。完工时间越晚权重越大,因此更可能在安排生产时加工顺序靠后。2) 随机生成1个小数 α 作为生产顺序,$\alpha\in(0,1)$,且 α 与权重相乘生成新 α,再根据新值的大小进行生产排序。在区间$(1,m)$中随机生成1个整数 β,该值代表任务被安排了几台机器进行生产,即 m 为此时刻机器总数。3) 初始种群中精英个体的 α 值等于权重,其他保持不变。

表3 编码方案示例

Tab. 3 Coding scheme example

合同号	con01	con02	con03	con04	con5
编码	2.56	3.13	4.74	2.65	3.24

合同 con01 的编码为 2.56，代表安排 2 台机器进行生产，同时生产的顺序权重为 0.56；con02 的编码为 3.13，代表安排 3 台机器生产，生产顺序权重为 0.13。因此，可知 con02 应当安排在 con01 前面生产。以此类推，可知合同的生产顺序为 con02、con05、con01、con04、con03。

2.3　解码设计

本算法中，解码部分等效于安排生产操作。为可行解安排生产的流程见图 2，具体步骤如下。

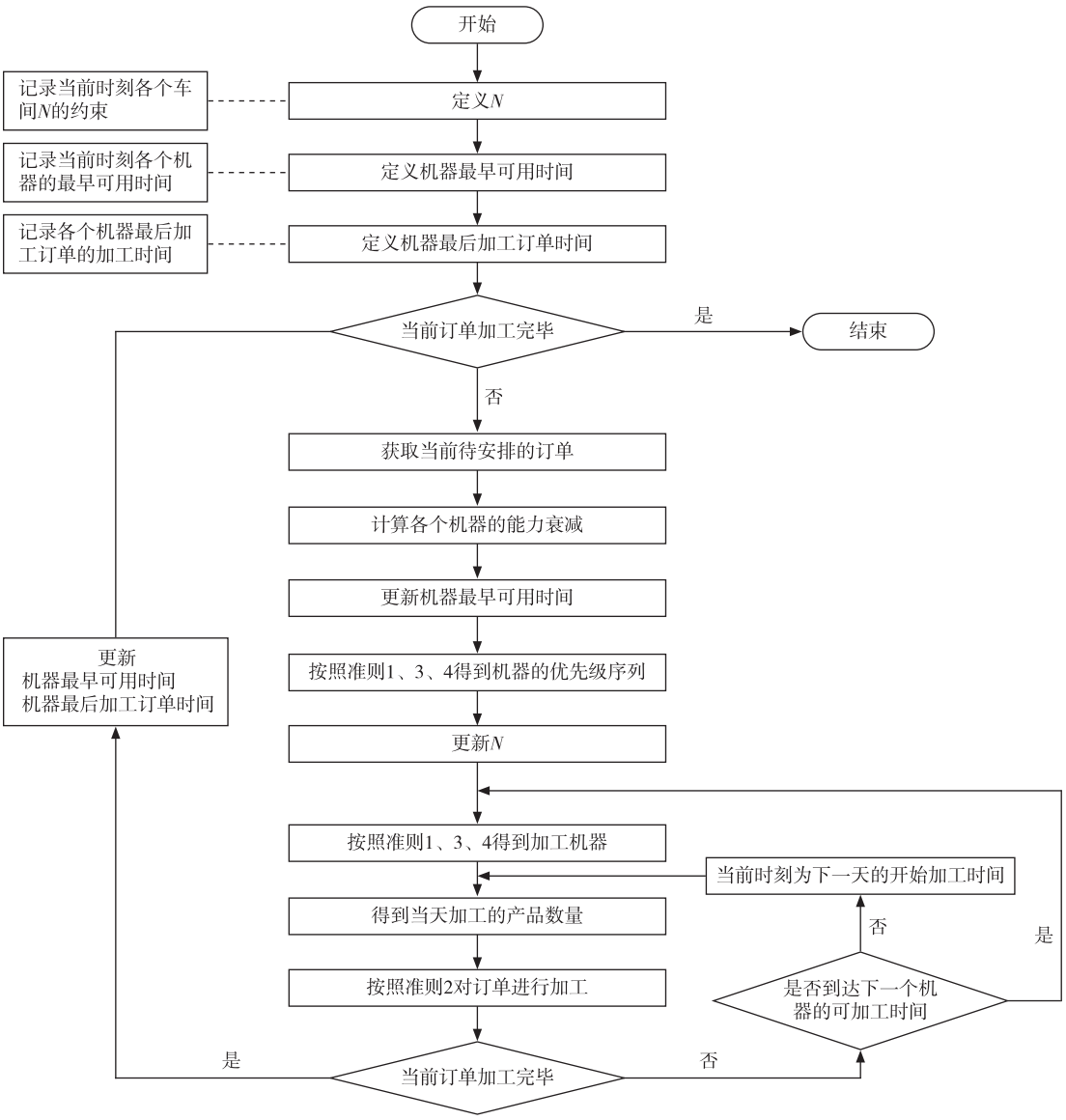

图 2　安排生产流程图

Fig. 2　Flow chart of arrange produciton

步骤 1，记录当前时刻各个车间 N 的约束、各个机器的最早可用时间、各个机器最后加工订单的加工时间。

步骤 2，检查所有订单是否已安排完毕。如果全部安排完毕，则结束求解过程。如果尚未安排完毕，则获取当前待安排的订单，计算各台机器的能力衰减情况，更新当前时刻各台机器的最早可用时间。接着，根据以下准则确定机器的优先级序列，并更新车间的约束条件 N：选择机器时，相同合同的工件优先；如果机器的可用时间相差 1 个单位时间，则视为相同；对 $N=1$ 的同车间机器进行统一调度。

步骤 3，按照优先级序列得到优先加工机器的编号。

步骤 4，根据物料的存量计算当天可加工的产品数量，对订单进行加工，尽量保证合同号中各个订单加工的结束时间一致。

步骤 5，判断当前订单是否加工完毕。若加工完毕，更新当前时刻各个机器的最早可用时间、各个机器最后

加工订单的加工时间,回到步骤2。若未安排完毕,继续下一步骤。

步骤6,判断是否达到下一个机器的可加工时间,若达到时间,回到步骤3;若未达到时间,则令当前时刻为下一天的开始加工时间,回到步骤4。

2.4 动态调整惯性权重

在MOPSO中,惯性权重在不同阶段有不同的需求:前期需要较大的惯性权重以增强全局搜索能力,后期则需要较小的惯性权重以提高局部搜索能力。在大多数PSO的相关文献中,惯性权重的调整是改进MOPSO的首选方向。例如,Shi等人[18]通过改变惯性权重提出了如下改进公式:

$$\omega=\omega_{\max}-\frac{(\omega_{\max}-\omega_{\min})\times t}{T_{\max}}。$$

本文提出一种对惯性权重动态调整的改进策略[19]。具体的改进描述如下:采用指数函数控制惯性权重 w 的变化,随着迭代次数的增加,e^{-x} 非线性减小,在算法的迭代后期可以增加算法的全局搜索能力,降低算法陷入局部最优的可能性,改进的惯性权重表达式为:

$$\omega=\omega_{\min}+(\omega_{\max}-\omega_{\min})\times e^{1-\frac{t}{T_{\max}}}\times\left(1-\frac{t}{T_{\max}}\right),$$

其中:t 为当前迭代次数,$T_{\max}$ 是最大迭代次数,$W_{\max}$ 为初始惯性权重,取0.9;$W_{\min}$ 为算法最大迭代次数时的惯性权重,取0.4。

2.5 适应度计算

在MOPSO中,初始化种群后需要计算每个粒子的适应度,在迭代过程中每当粒子位置更新时,必须重新计算适应度。适应度能够评估当前解的优劣,并据此更新粒子的个体最佳位置和全局最佳位置,从而引导种群逐步向最优解收敛。因此,适应度求解的设计对于算法的收敛性和最终解的质量有着关键影响。

本文的适应度计算步骤如下:首先,对每一个可行解进行生产调度,并记录该解的总生产成本和总拖期时间。随后,为便于在适应度比较过程中综合考虑,这2个值被存储在一维数组中:

$$f=\{f_1,f_2\},$$

其中:f_1 表示最小化总生产成本的目标函数,根据5种约束及对应的惩罚成本单位计算后汇总。f_2 表示最小化总拖期时间的目标函数,为所有任务的完工时间与规定交货期的差值的总和。

2.6 更新种群

2.6.1 基于全局比例排序和循环距离的全局学习样本选取策略

选择合适的全局学习样本不仅对粒子飞行速度具有重要影响,还会明显影响种群的进化方向。因此,优化全局学习样本的选择对于提高算法的运行效率至关重要。本文采用全局比例排序和循环距离结合的策略选取全局学习样本[20]。

全局比例排序是粒子的目标值与所有粒子目标值的和之比。计算公式如下:

$$G(X_i)=\frac{\sum_{j=1}^{M}f_j(X_i)}{\sum_{i=1}^{N}\sum_{j=1}^{M}f_j(X_i)},$$

其中:N 代表种群中粒子数量,M 表示目标个数,$f_j(X_i)$ 表示粒子 X_i 的第 j 个目标值,$G(X_i)$ 即为粒子 X_i 的全局比例排序值。通过全局比例排序值可以比较所有粒子的适应度信息,如表4所示。

表4 全局比例排序示例

Tab.4 Global scale sorting example

粒子	X_1	X_2	X_3	X_4
全局比例排序值	0.17	0.10	0.32	0.24

这4个粒子按照全局比例排序:$G(X_2)<G(X_1)<G(X_4)<G(X_3)$。由于粒子 X_2 的全局比例排序值小于其他粒子的值,应该选取粒子 X_2 作为全局学习样本。根据公式可对所有粒子进行排序,根据实际问题需求选取排序值最大或者最小的粒子,使算法更快地收敛。该种方法的局限性是:适应度相同的粒子也可能排序值相同,

此时如果随机选择 1 个粒子作为全局学习样本,可能会导致种群的多样性缺失。为此,本文引入了循环距离策略对粒子的拥挤度进行评估,目的是选择不在密集区域的解,以避免种群中的解过于集中的在某些区域,从而保持解的多样性。

循环距离的定义如下:

$$D(X_i)=\begin{cases}\sum_{j=1}^{M}|f_j(X_{i+1})-f_j(X_{i-1})|+\sqrt{\sum_{j=1}^{M}(f_j(X_{i+1})-f_j(X_{i-1}))^2}, 1<i<n;\\ \sum_{j=1}^{M}|f_j(X_i)-f_j(X_{i+1})|+\sqrt{\sum_{j=1}^{M}(f_j(X_i)-f_j(X_{i+1}))^2}, i=1;\\ \sum_{j=1}^{M}|f_j(X_{i-1})-f_j(X_i)|+\sqrt{\sum_{j=1}^{M}(f_j(X_{i-1})-f_j(X_i))^2}, i=n。\end{cases}$$

其中:$D(X_i)$表示粒子 X_i 的循环距离,n 表示外部存档中的粒子数,m 表示目标个数,$f_j(X_i)$表示粒子 X_i 的第 j 个目标值。从公式可以看出,循环距离的计算是在传统拥挤距离的基础上加入欧氏距离,这样能够更好地评估粒子的拥挤度。如图 3 所示,粒子 1 和 2 的拥挤距离相同,如果采用普通的拥挤距离计算,则粒子 1 和 2 会被同时删除或者保留。但采用循环距离策略计算,则可看出粒子 2 的循环距离大于粒子 1,因此保留粒子 1,同时淘汰粒子 2。保留较大循环距离的粒子有利于提升种群的多样性。

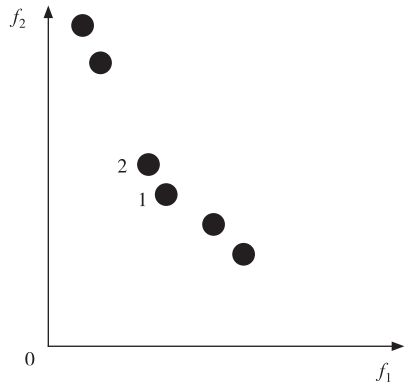

图 3　循环距离与拥挤距离的比较

Fig. 3　Comparison of circulation distance and crowding distance

综上,对于全局学习样本的选择,首先计算出粒子的全局比例排序值,然后选择排序值最小的粒子作为全局选择样本,如若遇到多个粒子的全局比例值相等,则计算这些粒子的循环距离并选择循环距离最大的粒子作为全局选择样本。

2.6.2　基于综合排序指标的外部存档维护策略

在 MOPSO 中,外部存档是一种重要的机制,用于存储每次迭代中生成的优势解。控制外部存档的策略不仅影响种群的多样性,还影响算法的精度。综合排序指标(composite ranking index,CRI)通过计算粒子的全局比例排序值与循环距离值之比来定义,旨在评估粒子的相对优劣[18]。CRI 的计算公式定义如下:

$$R(X_i)=\frac{G(X_i)}{D(X_i)}。$$

其中:$R(X_i)$表示粒子 X_i 的 CRI,$G(X_i)$表示粒子 X_i 的全局排序值,$D(X_i)$表示粒子 X_i 的循环距离值。$R(X_i)$值越小,表明该粒子在目标空间中的支配性较好,且在种群中的分布较为稀疏。随着算法的迭代,当外部存档的规模超过预定的最大值时,需要对存档中的粒子进行删除和更新。与其他维护策略相比,这种方法的优势在于它同时考虑了粒子的支配性和多样性,使得优秀的粒子更容易在后续的迭代中被保留和进一步发展。这为决策者提供了一个性能更优的候选解集。

3　实验及结果分析

实验使用 Java 语言编程,开发平台为 Visual Studio Code 1.92.2,测试机器为 Intel(R) Core(TM) i5-

9300H 2.40 GHz,16 G RAM的笔记本电脑。

3.1 实验条件与参数设置

MOPSO找到优质解的概率取决于参数选择,如种群数量、惯性权重和学习因子等。本文通过多次参数替换实验,最终确定了最佳参数设置(表5),除此之外,其他参数为:前6周订单最大生产天数为10 d,其余时间订单最大提前生产天数为14 d;设置虚拟订单大小为5;控制再生产参数为1.0;控制再添加机器参数为1.2。

表5 实验参数设置

Tab.3 Experimental parameter setting

参数	参数值	参数	参数值
粒子数	200	外部档案大小	50
迭代次数	30	空间位置纬度值	[0,0.999 9]
惯性权重	[0.4,0.9]	速度纬度值	[−0.199 9,0.199 9]
学习因子	0.4	精英种群比例	0.15

3.2 算法输入数据

本研究基于2023年8月某企业的实际生产数据。该时间段内,企业共签订了20份生产合同。车间内机器的类型、切换成本和加工能力等详细信息见表6。具体而言,A类型的机器编码为05L1～05L5、06L1～06L5;B类型的机器编码为07L1～07L4以及08L1～08L2。机器编码的前2位表示所属车间,后2位为机器的具体编号。例如,07L2表示7号车间中的第2台机器。表7展示了合同中订单类型、加工数量、物料消耗系数、机器能耗系数以及订单规定的完成时间等详细信息。合同订单编号分别为con01order01、con02order01、……、con19order01以及合同con20的2个订单:con20order01和con20order02。其中,合同con20的2个订单反映了实际生产中订单分配的灵活性。所有订单均被归类为A类型任务,并被分配至4个车间加工,以确保生产效率。

表6 机器详细信息

Tab.6 Machine details

机器编码	机器类型	可用数量	可用工时/min	机器编码	机器类型	可用数量	可用工时/min
05L1	A	3 900	1 440	06L4	A	3 900	1 440
05L2	A	3 900	1 440	06L5	A	3 900	1 440
05L3	A	3 900	1 440	07L1	B	4 100	1 440
05L4	A	3 900	1 440	07L2	B	4 100	1 440
05L5	A	3 900	1 440	07L3	B	4 100	1 440
06L1	A	3 900	1 440	07L4	B	4 100	1 440
06L2	A	3 900	1 440	08L1	B	4 100	1 440
06L3	A	3 900	1 440	08L2	B	4 100	1 440

3.3 算法停止条件

当算法满足以下任一条件时,实验终止并输出找到的Pareto解集:运行时间达到优化器的最大限制(本实验设置为1 h);算法达到最大迭代次数(本实验设置为30次);连续多次迭代无变化(本实验设置为5次)。

3.4 图示与分析

图4是Pareto前沿用于评估总生产成本与总拖期时间之间的权衡关系。通过分析帕累托前沿上的各个解,可以识别出在不同成本水平下的最优生产策略。这些策略在降低总生产成本的同时,最大限度地减少了拖期时间,为决策者提供了有效的参考依据。

表 7　合同详细信息

Tab. 7　Contract details

任务编码	订单类型	要求完工时间	加工数量	粗排产加工开始时间	粗排产加工结束时间
con01order01	A	2023-08-13	14 567	2023-08-07	2023-08-13
con02order01	A	2023-08-13	16 454	2023-08-07	2023-08-13
con03order01	A	2023-08-13	21 349	2023-08-07	2023-08-13
con04order01	A	2023-08-13	25 520	2023-08-07	2023-08-13
con05order01	A	2023-08-13	28 541	2023-08-07	2023-08-13
con06order01	A	2023-08-13	29 129	2023-08-07	2023-08-13
con07order01	A	2023-08-13	31 173	2023-08-07	2023-08-13
con08order01	A	2023-08-13	40 136	2023-08-07	2023-08-13
con09order01	A	2023-08-13	65 491	2023-08-07	2023-08-13
con10order01	A	2023-08-13	43 774	2023-08-07	2023-08-13
con11order01	A	2023-08-13	50 570	2023-08-07	2023-08-13
con12order01	A	2023-08-13	72 382	2023-08-07	2023-08-13
con13order01	A	2023-08-20	17 504	2023-08-07	2023-08-20
con14order01	A	2023-08-20	23 329	2023-08-14	2023-08-20
con15order01	A	2023-08-20	26 392	2023-08-14	2023-08-20
con16order01	A	2023-08-20	33 682	2023-08-14	2023-08-20
con17order01	A	2023-08-20	43 774	2023-08-14	2023-08-20
con18order01	A	2023-08-20	50 570	2023-08-14	2023-08-20
con19order01	A	2023-08-20	56 098	2023-08-14	2023-08-20
con20order01	A	2023-08-20	42 382	2023-08-14	2023-08-20
con20order02	A	2023-08-20	35 437	2023-08-14	2023-08-20

图 5 为排产甘特图，展示了各订单的加工时间和机器分配情况。通过该图可以直观地观察到订单的加工顺序及分配到的机器。例如，机器 05L1 上依次加工了订单 con09order01、con02order01 和 con19order01，这 3 个订单的加工时间为 2023 年 8 月 7 日—8 月 19 日。此外，图中还清楚显示了其他订单的生产顺序和机器分配情况，为优化生产提供了可视化参考。

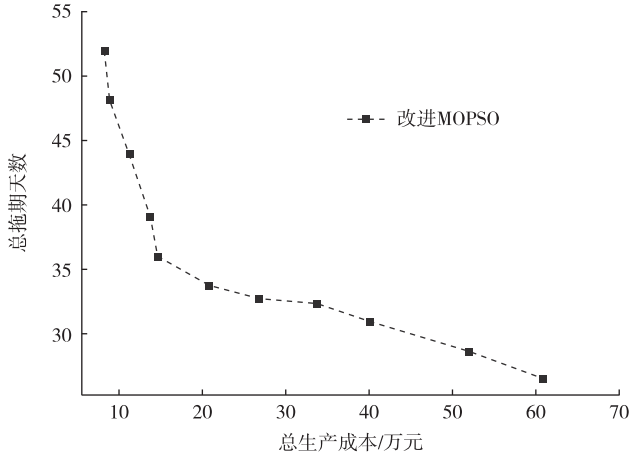

图 4　Pareto 前沿

Fig. 4　Pareto front

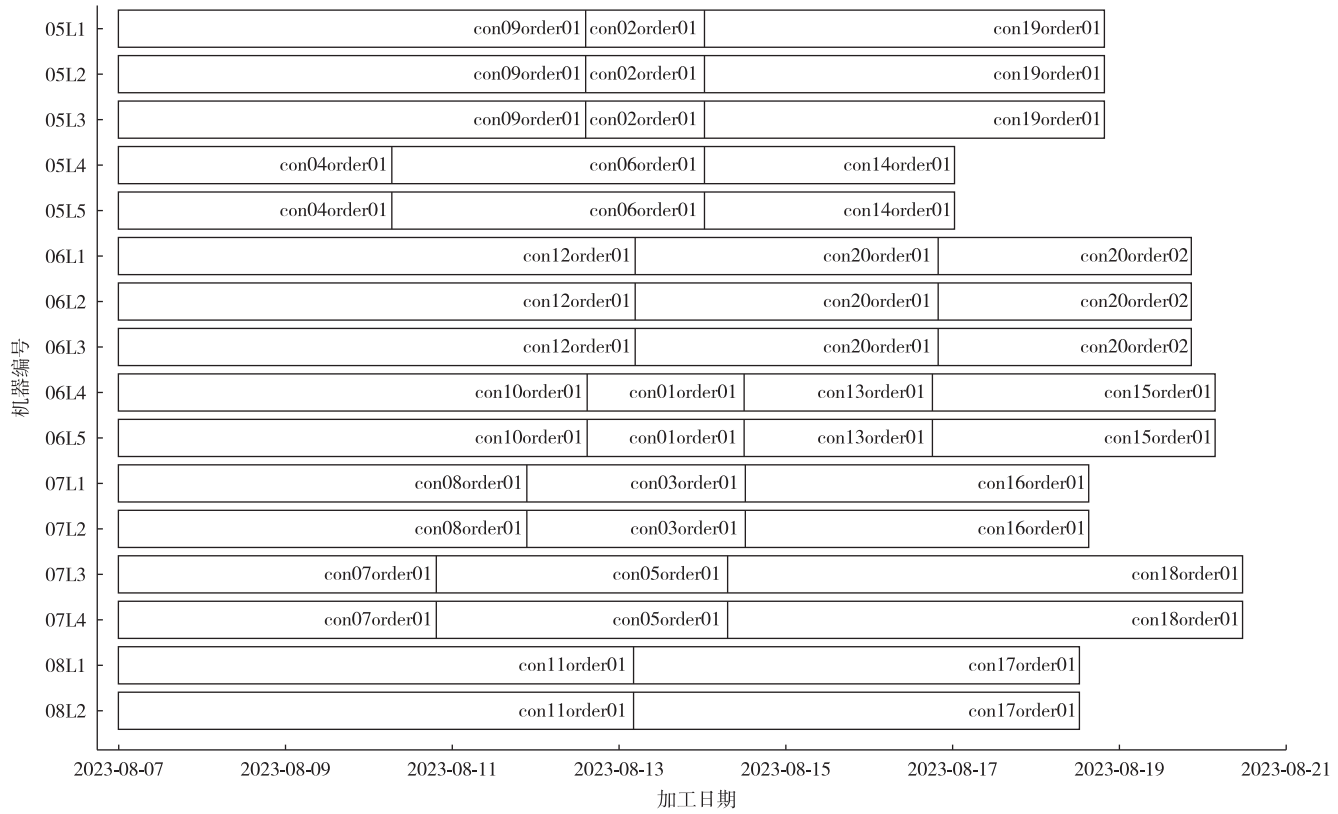

图5 排产甘特图

Fig. 5 Gantt chart for production scheduling

4 结束语

本文针对异构并行车间多目标排产优化问题,构建了以最小化总生产成本和总拖期时间为目标的数学模型,设计了一种改进的MOPSO并进行求解,最终得到了质量较高的Pareto解集,生产者可根据实际需求选择相对应的解。从排产甘特图中可以清晰地了解各加工任务的生产顺序、时间及成本。这有助于合理分配机器资源,优化生产序列,减少整体完工时间,并通过替换机器类型获得更合理的加工路径。本文提出的方法可为异构并行车间排产优化提供一定的参考价值。

参考文献:

[1] PINEDO M. 调度:原理、算法和系统[M]. 张智海译. 2版. 北京:清华大学出版社,2007.
PINEDO M. Scheduling:theory,algorithms and systems[M]. ZHANG Z H translation. 2nd edition. Beijing:Tsinghua University Press,2007.

[2] PINEDO M,HADAVI K. Scheduling:theory,algorithms and systems development[M]//Operations research proceedings 1991. Berlin:Springer,1992:35-42.

[3] ANGHINOLFI D,PAOLUCCI M,RONCO R. A bi-objective heuristic approach for green identical parallel machine scheduling [J]. European Journal of Operational Research,2021,289(2):416-434.

[4] KIM J G,SONG S,JEONG B. Minimising total tardiness for the identical parallel machine scheduling problem with splitting jobs and sequence-dependent setup times[J]. International Journal of Production Research,2020,58(6):1628-1643.

[5] LOFFREDO A,FRIGERIO N,LANZARONE E,et al. Energy-efficient control in multi-stage production lines with parallel machine workstations and production constraints[J]. IISE Transactions,2024,56(1):69-83.

[6] TAN M,YANG H L,SU Y X. Genetic algorithms with greedy strategy for green batch scheduling on non-identical parallel machines[J]. Memetic Computing,2019,11(4):439-452.

[7] DIAMANTIDIS A,LEE J H,PAPADOPOULOS C T,et al. Performance evaluation of flow lines with non-identical and unreliable parallel machines and finite buffers[J]. International Journal of Production Research,2020,58(13):3881-3904.

[8] NOMAN M A,ALATEFI M,AL-AHMARI A M,et al. Tabu search algorithm based on lower bound and exact algorithm

solutions for minimizing the makespan in non-identical parallel machines scheduling[J]. Mathematical Problems in Engineering, 2021,2021(1):1856734.

[9] BELDAR P,MOGHTADER M,GIRET A,et al. Non-identical parallel machines batch processing problem with release dates, due dates and variable maintenance activity to minimize total tardiness[J]. Computers & Industrial Engineering,2022,168:108135.

[10] BOK Y,LEE N K,JO S,et al. The production scheduling problem employing non-identical parallel machines with due dates considering carbon emissions and multiple types of energy sources[J]. Expert Systems with Applications,2024,238:121990.

[11] EBERHART R, KENNEDY J. A new optimizer using particle swarm theory[C]//MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science. Piscatway:IEEE,1995:39-43.

[12] KENNEDY J,EBERHART R. Particle swarm optimization[C]//Proceedings of ICNN'95-International Conference on Neural Networks. Piscatway:IEEE,2002:1942-1948.

[13] 杨维,李歧强. 粒子群优化算法综述[J]. 中国工程科学,2004,6(5):87-94.
YANG W,LI Q Q. Survey on particle swarm optimization algorithm[J]. Engineering Science,2004,6(5):87-94.

[14] 陈鑫,李昕昀,谢鹏宇. 平行机最小化误工损失调度问题的粒子群算法[J]. 重庆师范大学学报(自然科学版),2018,35(5):10-16.
CHEN X,LI X Y,XIE P Y. A particle swarm optimization algorithm for scheduling on unrelated machines with late work criterion[J]. Journal of Chongqing Normal University (Natural Science),2018,35(5):10-16.

[15] 许可,宫华,刘芳,等. 基于离散粒子群算法的应急物资选址与调度[J]. 重庆师范大学学报(自然科学版),2018,35(6):15-21.
XU K,GONG H,LIU F,et al. Research on emergency resources storage facilities location and scheduling based on binary particle swarm optimization algorithm[J]. Journal of Chongqing Normal University (Natural Science),2018,35(6):15-21.

[16] COELLO COELLO C A,LECHUGA M S. MOPSO:a proposal for multiple objective particle swarm optimization[C]//Proceedings of the 2002 Congress on Evolutionary Computation. Piscatway:IEEE,2002:1051-1056.

[17] COELLO COELLO C A,REYES-SIERRA M. Multi-objective particle swarm optimizers:a survey of the state-of-the-art[J]. International Journal of Computational Intelligence Research,2006,2(3):287-308.

[18] SHI Y, EBERHART R. A modified particle swarm optimizer[C]//1998 IEEE International Conference on Evolutionary Computation Proceedings IEEE World Congress on Computational Intelligence. Piscatway:IEEE,1998:69-73.

[19] 胡堂清,张旭秀,曹晓月. 一种动态调整惯性权重的混合粒子群算法[J]. 电光与控制,2020,27(6):16-21.
HU T Q,ZHANG X X,CAO X Y. A hybrid particle swarm optimization with dynamic adjustment of inertial weight[J]. Electronics Optics & Control,2020,27(6):16-21.

[20] LIU Y M,WANG S H,SONG X,et al. Novel multiobjective particle swarm optimization based on ranking and cyclic distance strategy[J]. International Journal of Intelligent Systems,2022,37(10):7379-7418.

Operations Research and Cybernetics

Multi-Objective Scheduling for Heterogeneous Parallel Workshops with a Particle Swarm Optimization Algorithm

GE Jianjun, CHEN Xin, WEN Yuxi

(School of Electronics and Information Engineering, Liaoning University of Technology, Jinzhou Liaoning 121001, China)

Abstract: With the development of intelligent manufacturing, the heterogeneous parallel workshop scheduling problem has increasingly attracted attention. In such workshops, machines vary in number and processing capabilities, requiring coordinated optimization to enhance production efficiency. It addresses the optimization problem of scheduling in heterogeneous parallel workshops by establishing a mathematical model with the objectives of minimizing total production costs and minimizing total tardiness. An improved multi-objective particle swarm optimization (MOPSO) algorithm is proposed to solve this problem. The algorithm dynamically adjusts the inertia weight to enhance the global search capability of the particle swarm in the solution space. It also improves algorithm efficiency by ranking and calculating cycle distances to identify global learning samples. Experimental results demonstrate that the proposed algorithm effectively balances the two objectives and generates Pareto-optimal solutions, providing decision-makers with various options.

Keywords: heterogeneous parallel workshop;scheduling optimization;multi-objective;particle swarm optimization

(责任编辑 黄 颖)