

# 具有空闲时间的云制造作业车间调度方法

王 贞<sup>1,2</sup>, 张纪会<sup>1,2†</sup>, 齐元青<sup>3</sup>

(1. 青岛大学 复杂性科学研究所, 山东 青岛 266071; 2. 青岛大学 自动化与电气  
工程学院, 山东 青岛 266071; 3. 青岛大学 机电工程学院, 山东 青岛 266071)

**摘要:** 针对云制造企业的剩余能力利用问题, 研究具有空闲时间的车间调度方法. 构建云制造作业车间的调度框架, 以最小化工期为目标, 考虑设备空闲时间, 提出工序加工时间序列的确定方法和空闲时间段的更新策略, 采用基于工件的精简编码和改进的二阶粒子群算法求解工件最优调度顺序. 仿真结果验证了所提出调度方法的可行性. 与已有算法的对比分析表明, 改进算法具有良好的搜索性能.

**关键词:** 云制造; 空闲时间段; 作业车间调度; 粒子群算法

**中图分类号:** TP301      **文献标志码:** A

## Job shop scheduling method with idle time in cloud manufacturing

WANG Zhen<sup>1,2</sup>, ZHANG Ji-hui<sup>1,2†</sup>, QI Yuan-qing<sup>3</sup>

(1. Institute of Complexity Science, Qingdao University, Qingdao 266071, China; 2. College of Automation & Electrical Engineering, Qingdao University, Qingdao 266071, China; 3. College of Mechanical & Electronic Engineering, Qingdao University, Qingdao 266071, China)

**Abstract:** Aiming at the utilization of the surplus capacity of manufacturing enterprises in cloud manufacturing (CMfg), a job shop scheduling method with idle time is studied. The scheduling framework of job shops in CMfg is built. Including the idle time of processing units, the method of determining processing time series and update strategy of idle time are proposed with the objective to minimize the makespan. A simplify encoding based on jobs and an improved second order particle swarm optimization are adopted to solve jobs' optimal scheduling sequence. Simulation results show the feasibility of the proposed scheduling method. Compared with other algorithms, it is proved that the improved algorithm can get better searching performance.

**Keywords:** cloud manufacturing; idle time; job shop scheduling; particle swarm optimization algorithm

## 0 引言

云制造是一种基于网络、面向服务的智能化制造新模式. 它借用“云计算”的思想, 利用信息技术将制造资源和能力集中, 建成公共的资源服务平台, 为需求者提供租用服务. 2010年初, 科技部高新司先进制造及自动化处周平首次通过“云制造”阐述了未来制造业发展走势<sup>[1]</sup>. 同一时期, 李伯虎等<sup>[2]</sup>发表了第一篇关于“云制造”的学术论文. 从此, “云制造”在学术界和工业界掀起了研究热潮, 涌现了大量研究成果, 许多制造企业也纷纷参与云制造支撑平台的建设.

对于提供服务的云制造企业而言, 企业生产任务通常包括两类: 1) 正常安排自有的生产任务(简称自有任务); 2) 接受来自云平台的生产任务(简称云任

务). 如何在保证自有任务顺利完成的前提下, 充分利用剩余制造资源和能力, 合理调度随机到达的云任务便成为企业参与云制造后亟待解决的问题. 文献[3]针对现有异地分布多车间协同生产计划的关联协调问题, 研究了以车间自有生产任务连续性生产为基础的资源指定占用下作业排产技术. 文献[4]建立了面向云制造作业车间的机器能力界定模型, 基于聚类算法提出了非瓶颈能力界定方法, 给出了不同扰动水平下每台非瓶颈机器的生产能力、保护能力和云服务能力. 但已有云制造的研究成果大多集中在云制造的特征、体系架构、关键技术等方面<sup>[5-7]</sup>, 关于云制造背景下企业车间级生产调度的研究还非常少见.

实现自有任务和云任务协同生产是云制造企业

收稿日期: 2016-04-16; 修回日期: 2016-06-29.

基金项目: 国家自然科学基金项目(61673228); 山东省自然科学基金项目(ZR2010GM006).

作者简介: 王贞(1981—), 女, 博士生, 从事云制造系统、车间调度管理的研究; 张纪会(1969—), 男, 教授, 博士生导师, 从事复杂适应系统理论、物流与供应链管理等研究.

†通讯作者. E-mail: zhangjihui@qdu.edu.cn

车间调度的目的. 自有任务的作业计划可看作初始调度, 云任务作为新任务插入. 新插入任务的调度通常有3种方法: 1) 初始调度不变, 待所有新任务所需资源均可用时安排新任务; 2) 将初始调度中没有开始生产的部分任务和新任务一起进行重调度; 3) 在初始调度的机器空闲时间段内安排新任务. 其中: 1) 和 3) 属于静态调度, 且新任务调度不会改变初始调度; 2) 则属于动态调度, 通常要求对初始调度的变动不应过大. Kai等<sup>[8]</sup>针对有新工件插入的柔性车间调度问题, 比较了这3种调度方法, 给出了各自的应用场合.

根据生产计划与运作管理的要求, 通常生产计划投产前会有一个相对稳定的冻结期(如一周). 在此期间, 初始调度不允许改变, 称初始调度被锁定. 文献[9]研究了锁定初始调度、单机环境下紧急工作的重调度问题. 文献[10]研究了不改变初始调度, 有单个工件到达的柔性流水车间的在线调度问题. 针对类似问题, 文献[11]将其推广到有一批工件到达的情况. 然而, 以上文献均没有涉及作业车间调度问题(JSP). 实际上, 通常云任务具有小批量、多品种的特点, 参与云制造的也多为作业车间(Job Shop). 因此, 本文研究生产计划冻结期, 云任务到达的JSP. 由于JSP的零工型特点, 初始调度后, 车间的加工设备不可避免地存在空闲时间段. 在锁定初始自有任务调度的前提下, 充分利用加工设备的空闲时间段, 合理调度随时到来的小批量云任务是本文的研究目的.

本文首先构建云制造企业作业车间的调度框架, 以最小化工期为目标, 考虑设备空闲时间, 提出了工序加工时间序列的确定方法和空闲时间段的更新策略, 并设计改进的二阶粒子群算法用于求解工件最优调度顺序. 最后通过算例进行仿真验证, 并讨论进一步提高制造资源利用率的方法.

## 1 问题描述和模型建立

### 1.1 云任务调度

云任务的调度是以自有任务完成为基础, 以提高制造资源利用率为目标, 调度框架如图1所示.

**Step 1: 机器能力界定.** 企业车间按自有生产任务的要求进行资源调度优化, 形成初始调度方案, 同时经过机器能力界定, 明确车间的剩余生产能力, 以便为云任务提供制造服务.

**Step 2: 云任务调度.** 已知车间的剩余生产能力, 根据云任务的工艺要求进行云任务调度, 明确云任务的工件调度顺序和各工序开工/完工时刻, 最终决策选取最优的云任务调度方案.

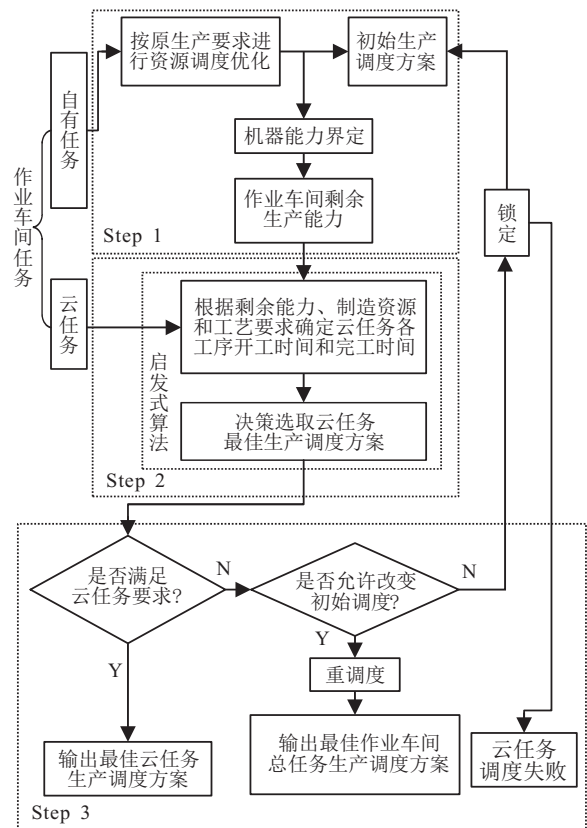


图1 云制造企业作业车间调度框架

**Step 3: 调度方案确定.** 若云任务调度满足云任务要求, 则输出此调度方案进行投产; 否则, 需要对初始调度进行适当调整, 重调度. 若初始调度已锁定, 则当前生产周期内云任务调度失败, 需要进入下一个生产周期进行调度.

### 1.2 问题描述

假设参与云制造企业的某车间内有  $M$  台加工设备, 按照自有任务要求已完成初始调度但尚未投产, 各加工设备的空闲时间段已明确. 现有一批包含  $N$  个工件的云任务进入车间, 考虑到实际云任务小批量多品种的特点, 云任务各工件的加工工序未必相同但是预先已知, 所有工件均在零时刻同时释放. 调度的目标是在加工设备的空闲时间段内合理调度, 使得云任务工期最短. 假设:

- 1) 各工件相互独立, 且无优先级;
- 2) 每台加工设备同时只能加工一道工序;
- 3) 每道工序一旦开始不允许中断;
- 4) 工件一旦开始加工, 不允许中途取消;
- 5) 物料搬运时间、作业准备时间和运输时均计入加工时间;
- 6) 不考虑机器故障, 加工过程是非抢占的.

### 1.3 模型建立

建立具有空闲时间段的作业车间调度模型如下:

$$\begin{aligned} \min F &= \max_i S_{i(O_i+1)}; & (1) \\ \text{s.t. } S_{i_2 k_2} \delta_{i_2 k_2 j} \mu_{i_1 i_2 j} + H(1 - \mu_{i_1 i_2 j}) &\geq \\ &(S_{i_1 k_1} \delta_{i_1 k_1 j} + P_{i_1 j}) \mu_{i_1 i_2 j}, & (2) \\ S_{i k_2} \delta_{i k_2 j_2} \omega_{i j_1 j_2} + H(1 - \omega_{i j_1 j_2}) &\geq \\ &(S_{i k_1} \delta_{i k_1 j_1} + P_{i j_1}) \omega_{i j_1 j_2}, & (3) \\ S_{i k} \delta_{i k j} &\geq \alpha_{j r_j}, & (4) \\ (S_{i k} + P_{i j}) \delta_{i k j} &\leq \beta_{j r_j}. & (5) \end{aligned}$$

其中:  $i, i_1, i_2 \in \{1, 2, \dots, N\}$ ;  $O_i$  为工件  $i$  的总工序数;  $k, k_1, k_2 \in \{1, 2, \dots, O_i\}$ ;  $S_{i k}$  为工件  $i$  第  $k$  道工序的开工时刻;  $S_{i(O_i+1)}$  为工件  $i$  最后一道工序的完工时刻;  $j, j_1, j_2 \in \{1, 2, \dots, M\}$ ;  $P_{i j}$  为工件  $i$  在加工设备  $j$  上的加工时间;  $Q_j$  为加工设备  $j$  的空闲时间段个数;  $r_j \in \{1, 2, \dots, Q_j\}$ ;  $\alpha_{j r_j}$  为加工设备  $j$  的第  $r_j$  个空闲时间段的上界;  $\beta_{j r_j}$  为加工设备  $j$  的第  $r_j$  个空闲时间段的下界. 且有

$$\begin{aligned} \delta_{i k j} &= \begin{cases} 1, & \text{工件 } i \text{ 的第 } k \text{ 道工序在加工设备 } j \text{ 上加工;} \\ 0, & \text{otherwise;} \end{cases} \\ \omega_{i j_1 j_2} &= \begin{cases} 1, & \text{加工设备 } j_1 \text{ 先于 } j_2 \text{ 加工工件 } i; \\ 0, & \text{otherwise;} \end{cases} \\ \mu_{i_1 i_2 j} &= \begin{cases} 1, & \text{工件 } i_1 \text{ 先于 } i_2 \text{ 在加工设备 } j \text{ 上加工;} \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

式(1)表示调度目标是云任务的工期最短;式(2)是资源约束,表示每台加工设备同时只能加工一道工序;式(3)是工序约束,表示同一工件的不同工序不能同时加工;式(4)和(5)是加工设备可用约束,表示工件必须在加工设备的空闲时间段内进行加工.

## 2 模型求解

上述问题的求解是确定各工件各工序的开工时刻,即寻找一个加工时间序列  $S = \{S_1, S_2, \dots, S_N\}$ , 其中  $S_i = \{S_{i1}, S_{i2}, \dots, S_{O_i}\}$  表示工件  $i$  各工序的开工时刻. 求解思路是首先结合加工设备的空闲时间段,确定工序开工时刻  $S_i$  的求解方法;然后按工期最短为目标求解各工件的最优调度顺序即加工序列  $X = \{x_1, x_2, \dots, x_N\}$ , 从而得到最优的加工时间序列  $S$ . 每道工序调度后要对空闲时间段进行更新,因此,模型求解包括工序开工时刻的求解、空闲时间段的更新和加工序列的求解.

### 2.1 工序开工时刻的求解方法

#### 2.1.1 方法描述

工序加工必须在加工设备的可用空闲时间段内进行. 求解某工序开工时刻时,首先要选择可用的空

闲时间段. 对于工件  $i$  的第  $k$  道工序,从对应加工设备的第1个空闲时间段开始寻找,若满足以下约束:

$$P_{i j} \delta_{i k j} \leq \beta_{j r_j} - \alpha_{j r_j}, \quad (6)$$

$$S_{i(k-1)} \delta_{i(k-1)j_1} + P_{i j_1} + P_{i j} \delta_{i k j} \leq \beta_{j r_j}, \quad (7)$$

则说明加工设备  $j$  的第  $r_j$  个空闲时间段可用. 式(6)保证工序  $k$  的加工时间小于空闲时间段,式(7)保证工序  $k$  在空闲时间段内能完成.

其次是确定工序开工时刻. 从第1道工序开始,对于工件  $i$  的第1道工序的开工时刻为可用空闲时间段的上界. 假设工件  $i$  的第1道工序选择加工设备  $a$  的第  $r_a$  个空闲时间段,第1道工序的开工时刻满足

$$S_{i1} \delta_{i1a} = \alpha_{a r_a}. \quad (8)$$

对于其他工序,还需考虑前一道工序的完工时刻,假设第  $k$  道工序选择加工设备  $j$  的第  $r_j$  个空闲时间段,第  $k$  道工序的开工时刻为

$$S_{i k} \delta_{i k j} = \max(\alpha_{j r_j}, S_{i(k-1)} \delta_{i(k-1)j_1} + P_{i j_1}). \quad (9)$$

#### 2.1.2 方法可行性

**定理1** 通过式(8)和(9)确定的  $S_i$  满足调度模型的约束(3)和(4),且为最小的可能值.

**证明** 由式(9)得到  $S_{i k} \delta_{i k j} \geq \alpha_{j r_j}$ , 即式(4)成立.

由式(9)得

$$S_{i k} \delta_{i k j} \geq S_{i(k-1)} \delta_{i(k-1)j_1} + P_{i j_1}. \quad (10)$$

令  $k = k_2, k - 1 = k_1, j_2 = j$ , 若  $\delta_{i k_1 j_1} = \delta_{i k_2 j_2} = 1$ , 则  $\omega_{i j_1 j_2} = 1$ . 那么,式(10)可写成

$$\begin{aligned} S_{i k_2} \delta_{i k_2 j_2} \omega_{i j_1 j_2} + H(1 - \omega_{i j_1 j_2}) &\geq \\ S_{i k_1} \delta_{i k_1 j_1} \omega_{i j_1 j_2} + P_{i j_1} \omega_{i j_1 j_2}, \end{aligned}$$

即式(3)成立.

由式(8)可知,工件  $i$  第1道工序开工时刻  $S_{i1} \delta_{i1a} = \alpha_{a r_a}$  是最小的可开工时刻. 这样,算法中的  $S_i$  取值不可能再小. 因此,此方法确定的开始加工时刻是最小的可能值. □

**定理2** 约束(6)和(7)是确定  $S_i$  的必要条件.

**证明** 由式(10)得

$$\begin{aligned} S_{i k} \delta_{i k j} + P_{i j} \delta_{i k j} &\geq \\ S_{i(k-1)} \delta_{i(k-1)j_1} + P_{i j_1} + P_{i j} \delta_{i k j}. \end{aligned} \quad (11)$$

由式(5)和(11)得

$$S_{i(k-1)} \delta_{i(k-1)j_1} + P_{i j_1} + P_{i j} \delta_{i k j} \leq \beta_{j r_j},$$

即式(7)成立.

由式(4)和(5)得

$$P_{ij}\delta_{ikj} \leq \beta_{jr_j} - S_{ik}\delta_{ikj} \leq \beta_{jr_j} - \alpha_{jr_j},$$

即式(6)成立. □

### 2.2 空闲时间段的更新策略

各加工设备的空闲时间段是随着生产推进实时变化的. 每道工序安排后需对相应加工设备的空闲时间段进行更新, 以正确安排下一道工序. 空闲时间段的更新分为4种情况. 为便于说明, 用  $C_{ik}$  表示工件  $i$  第  $k$  道工序的完工时刻. 根据  $S_{ik}$  和  $C_{i(k-1)}$  与空闲时间段上下界的关系, 选择合适的更新策略:

1) 若  $S_{ik} = \alpha_{jr_j}, C_{i(k-1)} < \beta_{jr_j}$ , 则更新后的空闲时间段缩短,  $\alpha_{jr_j} = C_{ik}, \beta_{jr_j} = \beta_{jr_j}$ , 总空闲时间段个数不变, 如图2(a)所示.

2) 若  $S_{ik} > \alpha_{jr_j}, C_{i(k-1)} = \beta_{jr_j}$ , 则更新后的空闲时间段缩短,  $\alpha_{jr_j} = \alpha_{jr_j}, \beta_{jr_j} = S_{ik}$ , 总空闲时间段个数不变, 如图2(b)所示.

3) 若  $S_{ik} > \alpha_{jr_j}, C_{i(k-1)} < \beta_{jr_j}$ , 则更新后的空闲时间段被分成两个部分,  $\alpha_{jr_j} = \alpha_{jr_j}, \beta_{jr_j} = S_{ik}, \alpha_{j(r_j+1)} = C_{ik}, \beta_{j(r_j+1)} = \beta_{jr_j}$ . 此时, 总空闲时间段个数增加一个, 如图2(c)所示.

4) 若  $S_{ik} = \alpha_{jr_j}, C_{i(k-1)} = \beta_{jr_j}$ , 则更新后的空闲时间段不再存在, 即被置空, 总空闲时间段个数减少一个, 如图2(d)所示.

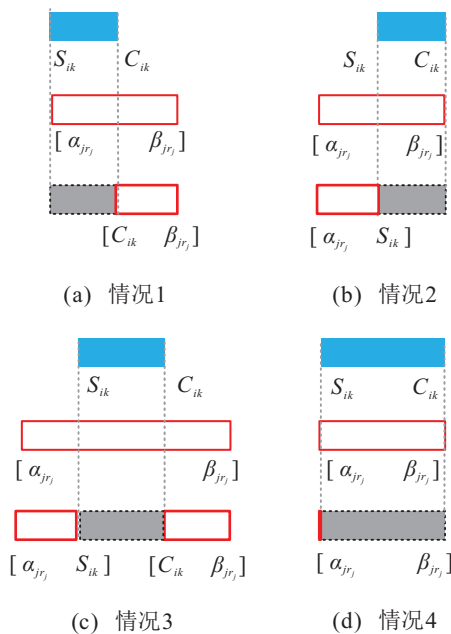


图2 空闲时间段更新策略图

### 2.3 加工序列的求解

本文设计了一种改进的二阶粒子群算法用于求解各工件的调度顺序. 粒子群算法是1995年由Eberhart等<sup>[12]</sup>提出的模拟鸟群觅食行为的启发式算法. 为提高搜索速度、增大获得最优解的几率, 改进的二阶粒子群算法采用基于工件的简易编码和并在

二阶粒子群算法中引入随机权重.

#### 2.3.1 种群编码

采用基于工件的整数编码, 编码方式极其简单, 长度仅为工件的个数, 表示工件的调度顺序. 如粒子 [2 3 1 4] 表示4个工件的调度顺序是  $2 \rightarrow 3 \rightarrow 1 \rightarrow 4$ .

#### 2.3.2 种群初始化

假设  $d$  维搜索空间中第  $i$  个粒子的位置和速度分别为  $X_i = (x_{i1}, x_{i2}, \dots, x_{id}), V_i = (v_{i1}, v_{i2}, \dots, v_{id})$ . 随机初始种群  $X_i, V_i$  的元素均为随机数. 作业车间调度属于组合优化问题, 每个种群粒子的位置对应一种工件调度顺序, 即种群元素应是和工件数  $N$  同维的互不相同整数序列. 为此, 本文采用基于粒子位置元素值排序的方法对初始随机种群进行变换. 具体方法是确定一个与各元件相对应的向量, 对向量元素进行升序排序. 元素越小, 其对应的工件越优先调度<sup>[11]</sup>. 由此得到变换后代表工件调度顺序的向量. 如表示5个工件的工件调度顺序, 可随机产生一个5维粒子向量 [2.54, 1.63, 0.05, 0.89, 3.45]. 可以看出, 5个随机数中工件3对应的值最小, 所以将工件3先安排调度. 依次类推, 变换后的粒子向量为 [3, 4, 2, 1, 5]. 向量中元素代表工件号, 工件号在向量中出现的顺序即为调度顺序.

取目标函数  $F$  为适应度函数

$$\text{fitness}(X) = F.$$

其中:  $X$  为工件调度顺序,  $\text{fitness}(X)$  为解  $X$  的适应度,  $F$  为按式(1)计算的目标函数值.

#### 2.3.3 粒子更新

标准粒子群算法中, 粒子的飞行速度仅仅是粒子当前位置的函数, 容易陷入局部最优解. 二阶粒子群算法对其进行了改进, 算法中粒子飞行速度的变化还与粒子的位置变化有关. 本文在二阶粒子群算法中引入随机权重, 将惯性权重  $w$  设定为服从正态分布的随机数, 对粒子进行更新. 第  $i$  个粒子速度更新公式为

$$v_{ij}(t+1) = \omega v_{ij}(t) + c_1 r_1 [p_{ij} - 2x_{ij}(t) + x_{ij}(t-1)] + c_2 r_2 [g_{\text{best}} - 2x_{ij}(t) + x_{ij}(t-1)]; \quad (12)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1), \quad \forall j \in \{1, 2, \dots, d\}; \quad (13)$$

$$\omega = \mu + \sigma \times N(0, 1); \quad (14)$$

$$\mu = \mu_{\min} + (\mu_{\max} - \mu_{\min}) \times \text{rand}(0, 1). \quad (15)$$

其中: $\omega$ 为惯性权重,决定了粒子当前速度的继承,服从标准正态分布; $\sigma$ 为 $\omega$ 的方差; $\mu_{\max}$ 和 $\mu_{\min}$ 为 $\omega$ 数学期望的最大值和最小值; $c_1$ 和 $c_2$ 为正的学习因子,分别表示粒子自我总结和向群体中优秀个体学习的能力; $r_1$ 和 $r_2$ 为区间(0,1)上均匀分布的随机数; $p_{ij}$ 为第*i*个粒子的历史最优值; $g_{\text{best}}$ 为种群的全局最优值.

### 2.4 求解步骤

采用改进的二阶粒子群算法求解工件加工序列的步骤如下.

Step 1: 随机产生初始种群,并进行整数化变换(详见第2.3.2节).

Step 2: 计算每个粒子的适应值.

Step 3: 寻找全局最优粒子,计算其适应值.

Step 4: 更新粒子(详见第2.3.3节),更新后重新进行整数化变换并计算适应值.

Step 5: 比较、更新全局最优粒子.

Step 6: 判断是否满足停止条件,若满足,则输出结果,否则转入Step 4继续更新.

### 3 算例分析

为验证本文算法的有效性,对以下算例进行仿真.表1给出了云任务的工艺和加工信息.以24小时为一个生产周期,作业车间的自有任务初始调度后,各加工设备的空闲时间段如表2所示.

表1 云任务的工艺及加工信息

工件	工艺要求	加工时间/h				
		$M_1$	$M_2$	$M_3$	$M_4$	$M_5$
1	$M_3 \rightarrow M_1 \rightarrow M_2$	2	1	1	1	1
2	$M_1 \rightarrow M_5$	1	1	2	1	1
3	$M_1 \rightarrow M_2 \rightarrow M_3 \rightarrow M_4$	2	1	1	2	2
4	$M_2 \rightarrow M_1 \rightarrow M_5 \rightarrow M_4$	1	1	2	1	1
5	$M_3 \rightarrow M_5$	1	2	1	1	1
6	$M_4 \rightarrow M_2 \rightarrow M_1$	1	1	1	2	1

表2 加工设备信息

加工设备号	一个生产周期内空闲时间段/h			
$M_1$	[0, 6]	[8, 16]	[20, 24]	
$M_2$	[8, 12]	[13, 18]	[22, 24]	
$M_3$	[7, 10]	[15, 20]		
$M_4$	[1, 5]	[8, 10]	[13, 15]	[21, 24]
$M_5$	[0, 5]	[7, 10]	[11, 15]	[22, 24]

在 Intel CoreI3 1.9GHz 处理器/4G 内存/win 7/ Matlab 2013 的环境下运行算法程序.粒子群算法参数设置如下:粒子数 20,迭代次数 100,  $c_1 = c_2 = 2$ ,  $\mu_{\max} = 0.9$ ,  $\mu_{\min} = 0.4$ ,  $\delta = 0.15$ .

图3为作业车间调度的加工设备甘特图,图中灰色表示初始调度.由图3可见,加工设备的剩余能力可完成一批包含6个工件的云任务,完工时间为15时,工件调度顺序为[2, 3, 6, 5, 1, 4].

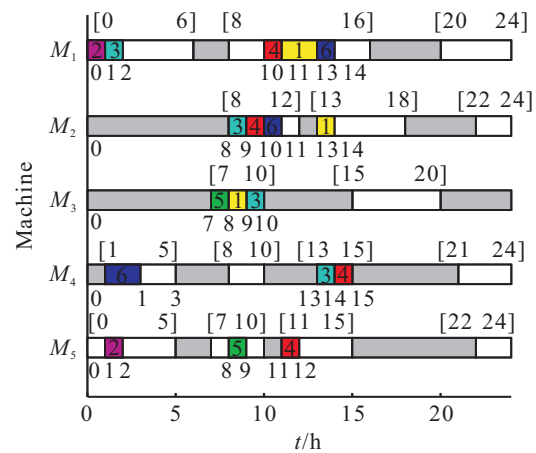


图3 作业车间调度的加式设备甘特图

车间加工设备剩余能力及其利用率的情况如表3所示.可以看出,加工设备和剩余能力的利用率都有所提高,但加工设备剩余能力的利用率尚不足40%,仍有较大提升空间.

表3 车间加工设备剩余能力及其利用率统计

任务	剩余能力及利用率	$M_1$	$M_2$	$M_3$	$M_4$	$M_5$
自有任务	剩余能力(时长)	18	11	8	11	14
	加工设备利用率/%	25	54.2	66.7	54.2	41.7
	剩余能力利用率/%	0	0	0	0	0
云任务	剩余能力(时长)	12	7	5	7	11
	加工设备利用率/%	50.0	70.8	79.2	70.8	54.2
	剩余能力利用率/%	33.3	36.4	37.5	36.4	21.4

结合云任务特征,下面讨论进一步提高剩余能力利用率的情况:

1) 云任务按批次交付,且剩余能力可完成多个整批云任务.可在剩余空闲时间段内继续安排整批云任务,这样,在一个生产周期可交付多批云任务,提高企业收益.

2) 云任务按批次交付,但剩余能力无法完成整批任务.可以剩余空闲时间段最小为目标安排云任务中的某些工件(或工序)生产,在下一生产周期内,再继续生产余下的工件(或工序),从而加快下一个生产周期云任务整批交付的进度.

3) 云任务允许按工件交付,则调度目标可由最小化完工时间变为最大化加工工件数量,在空闲时间段内以工件为单位进行调度.根据实际生产的具体情况,可进行合适的生产计划,以进一步提高车间加工设备的利用率.

为验证改进的二阶粒子群算法的优越性,将其分别与标准粒子群算法(PSO)、随机权重粒子群算法(RWPSO)和二阶粒子群算法(SOPSO)进行比较.以在相同时间内各算法达到最优解的次数作为评价指标,独立运行30次,设定相同的时间为25 s.各算法达到最优解的次数如图4所示.

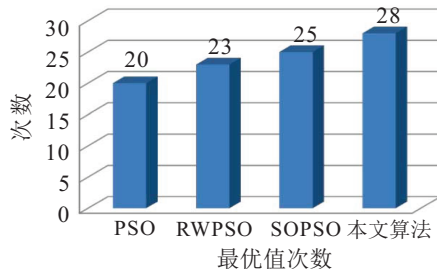


图4 算法搜索性能比较

由图4可见,本文提出的算法在相同时间内达到最优解的次数比其他算法多,表明改进的具有随机权重的二阶粒子群算法的搜索性能得到了增强.

## 4 结 论

本文针对作业车间剩余能力利用问题,研究了具有空闲时间段的云制造车间调度问题.给出了云制造作业车间调度框架,考虑加工设备的空闲时间,提出了工序加工时间序列的确定方法和空闲时间段的更新策略,证明了算法确定的工件各工序开工时刻是最早可能值.为提高搜索速度和性能,设计了具有随机权重的二阶粒子群算法,采用了基于工件的精简编码方式和基于粒子位置元素值排序的方法.仿真算例表明,具有空闲时间段的云制造作业车间调度方法是可行的,可为企业参与云制造提供一定的决策依据,具有随机权重的二阶粒子群算法具有较高的搜索性能和较快的收敛速度.

本文将在以下方面开展进一步的工作:初始调度允许改变,随机云任务到达的车间重调度问题;考虑其他性能指标;研究柔性作业车间的情况.

## 参考文献(References)

- [1] 侯闯. 云制造:摸不着[J]. 中国制造业信息化, 2010, 39(6): 14-17.  
(Hou C. Cloud manufacturing: Cannot touch[J]. Manufacture Information Engineering of China, 2010, 39(6): 14-17.)
- [2] 李伯虎,张霖,柴旭东. 云制造概论[J]. 中兴通讯技术, 2010, 16(4): 5-8.  
(Li B H, Zhang L, Chai X D. Introduction to cloud manufacturing[J]. ZTE Communications, 2010, 16(4): 5-8.)
- [3] 李京生,王爱民,唐承统,等. 基于动态资源能力服

务的分布式协同调度技术[J]. 计算机集成制造系统, 2012, 18(7): 1563-1574.

(Li J S, Wang A M, Tang C T, et al. Distributed coordination scheduling technology based on dynamic manufacturing ability service[J]. Computer Integrated Manufacturing Systems, 2012, 18(7): 1563-1574.)

- [4] 王军强,崔福东,张承武,等. 面向云制造作业车间的机器能力界定方法[J]. 计算机集成制造系统, 2014, 20(9): 2146-2163.  
(Wang J Q, Cui F D, Zhang C W, et al. Capacity partition method of machines in cloud manufacturing-oriented job shops[J]. Computer Integrated Manufacturing Systems, 2014, 20(9): 2146-2163.)
- [5] 李伯虎,张霖,任磊,等. 云制造典型特征、关键技术与应用[J]. 计算机集成制造系统, 2012, 18(7): 1345-1356.  
(Li B H, Zhang L, Ren L, et al. Typical characteristics, technologies and applications of cloud manufacturing[J]. Computer Integrated Manufacturing Systems, 2012, 18(7): 1345-1356.)
- [6] Xu X. From cloud computing to cloud manufacturing[J]. Robotics and Computer-Integrated Manufacturing, 2012, 28(1): 75-86.
- [7] Wu D Z, Matthew J G, Rosen D W, et al. Cloud manufacturing: Drivers, current status, and future trends[C]. Proc of the ASME 2013 Int Manufacturing Science and Engineering Conf. New York: ASME, 2013: 1-10.
- [8] Kai Z G, Ponnuthurai N S, Tay J C, et al. A two-stage artificial bee colony algorithm scheduling flexible job-shop scheduling problem with new job insertion[J]. Expert System with Applications, 2015, 42(21): 7652-7663.
- [9] 郭艳东,黄敏,王庆. 锁定初始调度的紧急工作单机重调度问题[J]. 东北大学学报:自然科学版, 2013, 34(5): 628-631.  
(Guo Y D, Huang M, Wang Q. Rescheduling for rush jobs on single machine with loads locked original scheduling[J]. J of Northeastern University: Natural Science, 2013, 34(5): 628-631.)
- [10] Fabrice C, Eugene L, Lenoid K M, et al. On-line scheduling in a surface treatment system[J]. European J of Operation Research, 2000, 120(2): 382-392.
- [11] 常桂娟,张纪会. 带运输时间的无等待供应链在线调度问题研究[J]. 控制与决策, 2008, 23(10): 1092-1097.  
(Chang G J, Zhang J H. No-wait on-line chain scheduling problems with transfer time[J]. Control and Decision, 2008, 23(10): 1092-1097.)
- [12] Kennedy J, Eberhart R C. Particle swarm optimization[C]. Proc of the IEEE Int Conf on Neural Networks. Piscataway: IEEE Press, 1995: 1942-1948.

(责任编辑:郑晓蕾)