

# Dynamic Schedule Strategy of RGV Based on the Weighted Directed Graph

Fan Yang, Hao Zhang, Xiaojin Guo, Kang Zhao

School of Mathematics and Statistics, Changsha University of Science and Technology, Changsha Hunan  
Email: zhaokangmath@126.com, yf3092030@foxmail.com

Received: Feb. 27<sup>th</sup>, 2019; accepted: Mar. 13<sup>th</sup>, 2019; published: Mar. 20<sup>th</sup>, 2019

---

## Abstract

To solve the RGV scheduling problem, a RGV dynamic scheduling model based on graph theory is established. We transform distance into time in the problem so that the practical problem can be transformed into a mathematical model represented by a weighted directed graph, and virtual nodes are set up to solve the situation that the time is zero. In the one process, we schedule dynamically based on Greedy Algorithm. In each step, Floyd algorithm is used to search the virtual node with the smallest total distance from current of RGV position weight to determine the next moving node of RGV. In the two processes, we improve the traditional Simulated Annealing Algorithm (SA) based on Particle Swarm Optimization (PSO), and combine the scheduling algorithm in the one process. Aiming at maximizing the output, the optimal allocation of CNCs is provided, and then the optimal scheduling strategy of RGV is obtained within the specified working time.

## Keywords

Weighted Directed Graph, Greedy Algorithm, Objective Optimization, Schedule Strategy of RGV

---

# 基于有向赋权图的RGV动态调度策略研究

杨帆, 张昊, 郭肖晋, 赵康

长沙理工大学数学与统计学院, 湖南 长沙  
Email: zhaokangmath@126.com, yf3092030@foxmail.com

收稿日期: 2019年2月27日; 录用日期: 2019年3月13日; 发布日期: 2019年3月20日

---

## 摘要

本文针对RGV调度问题, 建立了基于有向赋权图的RGV动态调度模型。我们将问题中所提及的距离时间

化，以时间作为权值，将实际问题转化为一个有向赋权图的数学模型，并设置虚拟节点以解决时间为零时的情况。对于一道工序问题，我们基于贪心算法进行动态调度，每一步都通过Floyd算法搜索距离RGV当前位置权值总和最小的虚拟节点，用于确定RGV的下一步移动节点。针对两道工序问题，我们基于粒子群算法(PSO)改进了传统的模拟退火算法(SA)，并结合一道工序的调度算法，以产量最大作为目标，搜索得到CNC的最佳分配。从而得出在规定的工作时间内，RGV最佳的调度策略。

### 关键词

有向赋权图，贪心算法，目标优化，RGV的调度策略

Copyright © 2019 by author(s) and Hans Publishers Inc.  
 This work is licensed under the Creative Commons Attribution International License (CC BY).  
<http://creativecommons.org/licenses/by/4.0/>



## 1. 引言

随着控制工程、机电一体化、人工智能等多个领域的高速发展，物流系统开始向自动化、无人化、智能化发展，RGV (Rail Guide Vehicle)有轨制导车辆系统也随之出现。RGV 是一种无人驾驶、在轨道自由运行的智能车，能根据指令自动控制移动方向和距离。它既可作为立体仓库的辅助设备，也有自己独立运行的系统，被广泛应用于现代化物流体系。因此，如何通过合理的调度规则提高RGV的工作效率是急需解决的问题。

本文研究智能加工系统(见图 1)的RGV的最佳动态调度策略[1]，该系统由8台计算机数控机床(Computer Number Controller, CNC)、1辆RGV、1条RGV的直线轨道、1条上料传送带、1条下料传送带等设备组成。在设备正常工作时，若有CNC处于空闲状态，则会立即向RGV发出上料需求信号；否则，CNC处于加工状态。当RGV在接到上料需求信号时，RGV需确定最佳的上下料及移动策略。RGV在下料完毕后，需先对熟料进行清洗才能将熟料放置下料传送带，并且在清洗熟料时，RGV不能移动。

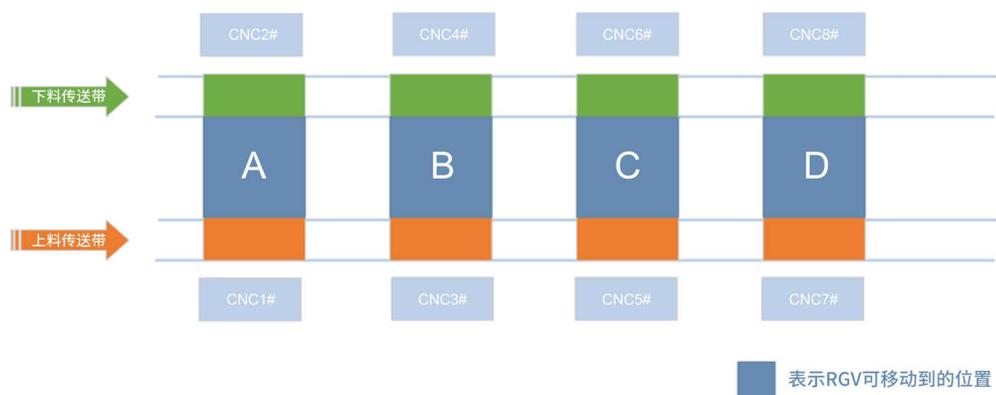


Figure 1. Schematic diagram of RGV working principle  
 图 1. RGV 工作原理示意图

## 2. 模型假设与符号说明(表 1)

我们做以下假设：

- 1) 假设上下料传送带都是连续不断的,能及时为 RGV 提供生料和运输熟料;
- 2) 假设 RGV 在完成上下料作业后即刻进入清洗作业阶段;
- 3) 假设每班连续工作八小时;
- 4) 作业开始时, RGV 的初始位置位于 1 号 CNC 前;
- 5) 假设该班次结束时仍处于加工状态的物料不计入生产件数。

Table 1. Symbolic explanation

表 1. 符号说明

符号	说明
$v_A, v_B, v_C, v_D$	表示 RGV 位置节点
$v_1, \dots, v_8$	表示 CNC 位置节点
$v'_1, \dots, v'_8$	表示虚拟节点
$t_{ij}$	表示由 $i$ 节点至 $j$ 节点路径的权值, 及相应的时间
$s_j$	表示物料到 $j$ 节点开始工作所需时间
$X_i(t)$	表示 $t$ 时刻编号为 $i$ 的 CNC 的工作状态
$Y_i(t)$	表示 $t$ 时刻编号为 $i$ 的 CNC 的剩余加工时间
$Z_i(t)$	表示 $t$ 时刻编号为 $i$ 的 CNC 正在加工材料的编号
$RGVP(t)$	表示 $t$ 时刻 RGV 所处的位置
$RGVF(t)$	表示 $t$ 时刻 RGV 清洁槽内是否有熟料

### 3. RGV 动态调度问题的数学模型

我们以系统工作时间最少为目标建立模型。显然,单位作业时间的长短是由 CNC 加工时间和 RGV 移动作业时间共同决定的。因此,我们考虑利用 CNC 加工、RGV 移动、清洗等工作,制定相应的动态调度模型,以尽量减少系统的空闲时间为目标进行调度。我们将距离时间化,以 RGV 可移动到的四个位置和八个 CNC 的位置作为节点,以相应过程的时间作为权值,建立基于有向图的优化调度模型。这样时间最短问题转换为图论问题中的最短路径问题。注意到,我们的目标应是使物料尽快进入工作状态,而不是使物料尽快移动到指定地点。例如,时刻  $t$  时 RGV 的初始位置在  $A$  处,若  $t_{AB} = 10$  s,  $Y_1(t) = 8$  s,  $Y_3(t) = 0$  s, 则 RGV 不移动,等待 1 号 CNC 加工完毕后进行上下料更为。因此,当 RGV 进入空闲状态时,为了使每步调度都达到当前最优,我们为每个 CNC 引入一个虚拟节点,权值为  $t_{ij'} = \max\{0, Y_i(t) - t_{ij}\}$ , 则当 RGV 进入空闲状态时,可令 RGV 进行一次搜索,找出距离 RGV 当前位置最近的虚拟节点,其对应的 CNC 即为 RGV 下一步作业的目标。

#### 3.1. 节点说明

节点集合  $V$  包含三种节点: RGV 可移动到的位置节点,每台 CNC 所处位置的节点、虚拟节点(见图 1)。用  $(v_A, v_B, v_C, v_D)$  表示 RGV 位置节点,  $(v_1, \dots, v_8)$  表示 8 台 CNC 位置节点,  $(v'_1, \dots, v'_8)$  表示虚拟节点位置。

#### 3.2. 权值说明

用  $\langle v_i, v_j \rangle$  表示从  $v_i$  到  $v_j$  的一条边,  $t_{ij}$  为对应权值,同时也表示相应的工作时间或移动时间,其中  $i, j$  表示任一节点。图的边集  $E$  包含以下三个部分:

1) RGV 位置节点  $v_A, v_B, v_C, v_D$  之间的权值。

2) RGV 位置节点(用  $i$  表示)与 CNC 位置节点(用  $j$  表示)间的权值。弧  $\langle v_i, v_j \rangle$  表示 RGV 为 CNC 上下料, 权值为上下料所需时间。弧  $\langle v_i, v_j \rangle$  表示 RGV 清洗熟料, 当 CNC 未加工处于空闲时, 无熟料产生, 下料为空, 不进行清洗操作, 权值为 0 s; 当 CNC 已加工完处于空闲时, 有熟料产生, 进入清洗作业阶段, 权值为清洗作业所需时间  $t_q$ , 如公式(1)所示。

$$t_{ij} = \begin{cases} 0 & \text{CNC未加工空闲} \\ t_q & \text{CNC加工完毕空闲} \end{cases} \quad (1)$$

3) 环  $\langle v_i, v_j \rangle$  表示 RGV 不进行任何操作, 权值为等待时间。因此, 图  $(V, E)$  的点集和边集[2]可表示为:

$$V = \{v_i | i = A, \dots, D, 1, \dots, 8, 1', \dots, 8'\}, E = \{\langle v_i, v_j \rangle | i, j = A, \dots, D, 1, \dots, 8, 1', \dots, 8'\} \quad (2)$$

由图 2 所示, 结合图论的 Floyd 算法, 我们以减少系统空闲时间为目标设立调度规则, 求出一个班次内加工物料的编号、上下料时间、CNC 编号以及最大加工物料数, 从而给出调度策略。

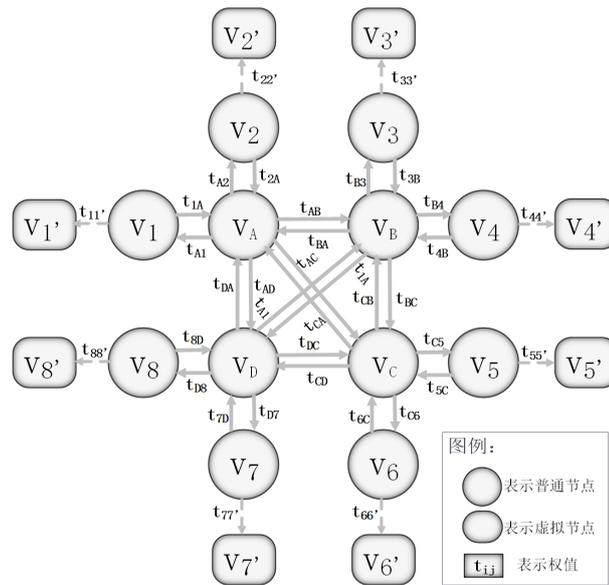


Figure 2. Scheduling optimization of network diagram  
图 2. 优化调度图

RGV 作业周期[3]: 当工作状态达到稳定后, RGV 访问 CNC 的次序不再改变, 将此时依次访问各 CNC 编号序列的最小重复单元个数  $n$  定义为周期长度, 则 RGV 连续访问  $n$  个 CNC 可定义为一个作业周期。

RGV 作业序列: 将一个周期内 RGV 依次访问各 CNC 的编号顺序定义为作业序列, 并表示为:

$$Sep = (\lambda_1 \rightarrow \lambda_2 \rightarrow \dots \rightarrow \lambda_n \rightarrow \lambda_1) \quad (3)$$

当 RGV 在一个作业周期内消耗时间达到最少时, 为最优作业序列, 记作  $Sep_{min}$ 。

#### 4. 基于图论的 RGV 动态调度算法

对于 RGV 动态调度问题, 两道工序与多道工序没有本质的区别。因此, 我们仅考虑一道工序和两道

工序两种情形。在一道工序物料加工作业时，每台 CNC 配置相同的刀具，物料可在任一台 CNC 上进行加工；在两道工序物料加工作业时，每个物料需在不同 CNC 上依次完成两道工序。我们先给出一道工序物料加工时 RGV 的动态调度算法，在此基础上给出两道工序物料加工时 RGV 的动态调度算法。

### 4.1. 一道工序物料加工作业——算法一：基于贪心算法的 RGV 动态调度算法

结合贪心算法在图论中的应用[4]，我们为该调度模型制定了一个规则，通过 Floyd 算法求得任意点到某虚拟节点最短距离，提出基于贪心算法的规则求解算法，最终得到一个较好的局部最优解，如图 3。

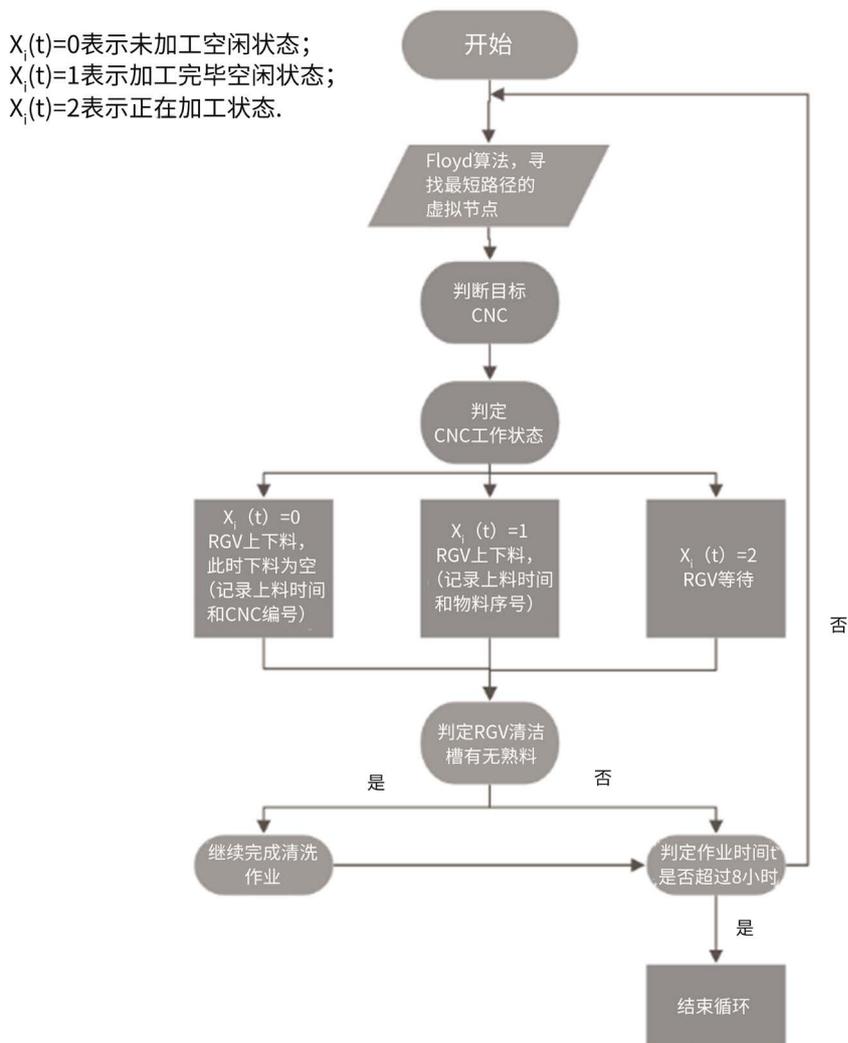


Figure 3. Solution flow based on Greedy Algorithm

图 3. 基于贪心算法的求解算法

用  $X_i(t)$  表示 CNC 工作状态：

$$X_i(t) = \begin{cases} 0 & \text{CNC未加工空闲状态} \\ 1 & \text{CNC加工完毕空闲状态} \\ 2 & \text{CNC加工完毕空闲状态} \end{cases} \quad (4)$$

用  $RGVF(t)$  表示 RGV 清洁槽内是否有熟料：

$$RGVF(t) = \begin{cases} 0 & \text{无熟料} \\ 1 & \text{有熟料} \end{cases} \quad (5)$$

**步骤一：**用  $t$  表示当前作业时间， $RGVF(t)$  表示  $t$  时刻 RGV 所处位置， $t$  的初值为 0，此时 RGV 处在  $A$  位置。

**步骤二：**使用 Floyd 算法寻找距 RGV 当前位置最近的虚拟节点，则该虚拟节点所对应的 CNC 即为 RGV 下一步的目标 CNC。

**步骤三：**RGV 移动到权值最小的 CNC 后，RGV 开始等待，判断 CNC 的工作状态并确定等待时间。

1) 当  $X_i(t) = 0$  时，CNC 处于空闲状态，等待 0 s，RGV 进行上下料，但下料为空并记录上料时间和 CNC 编号；

2) 当  $X_i(t) = 1$  时，CNC 已加工完毕，等待 0 s，RGV 进行上下料，记录上下料时间和物料序号  $Z_i(t)$ ；

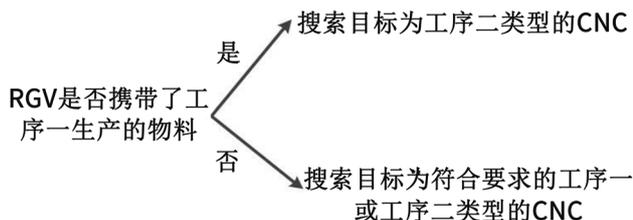
3) 当  $X_i(t) = 2$  时，CNC 正在加工，等待剩余加工时间为  $Y_i(t)$ 。

**步骤四：**判断 RGV 清洁槽内有无熟料，当  $RGVF(t) = 1$  时，进行清洗作业并判断总作业时间  $t$  是否超过一个班次的作业时间；当  $RGVF(t) = 0$  时，直接判断作业时间  $t$  是否超过一个班次的作业时间，若未超过，返回步骤二；若超过，结束循环，并输出一个班次内的物料编号、上下料时间以及最大加工物料数。

## 4.2. 两道工序物料加工作业——算法二：改进型模拟退火算法

### 4.2.1. 算法一的改进

由于 CNC 分为两种加工方式，因此在算法一中需增加一步判断：



### 4.2.2. 改进模拟退火算法

问题分析中已指出，模拟退火(SA)算法[5]具有局部搜索能力强，运行时间较短的优点，但其全局搜索能力较差，极易受到参数影响。因此我们使用粒子群(PSO)算法对其进行改进，使其参数可以进行动态调整从而克服算法的局限性。针对当前问题，我们借助改进后的算法一求得一个班次内的生产件数，然后通过改进的模拟退火算法搜索生产件数最大值，其对应的工序分配即为最优分配。算法流程见图 4。

#### 1) 算法二迭代过程

a) 生成  $m$  个粒子作为初始种群，赋给粒子的位置初值，给定初始接受概率  $p$ 、初始温度  $T$ 、终止温度  $T_e$ 、温度动态衰减系数  $k$ 、迭代次数  $n$ 。

b) 判断算法是否达到循环终止条件：

i) 循环次数  $r$  从 1 循环至  $n$ ；ii) 体系降温至终止温度的  $T_e$ 。

c) 计算每个粒子适应度  $f_i(r)$ 、 $f_{agv}(r)$

i) 若粒子的适应度优于原个体极值  $p_{best}$ ，则进行更新并选择最优个体值作为群体极值  $g_{best}$ 。

ii) 若粒子的适应度不能优于原个体极值  $p_{best}$ ，则更新  $p_0 = \exp(-1/p \cdot T_0)$ ，并以  $p_0$  为概率接受当前适应度作为  $p_{best}$ ；若未接受粒子的新适应度，则不做更新。

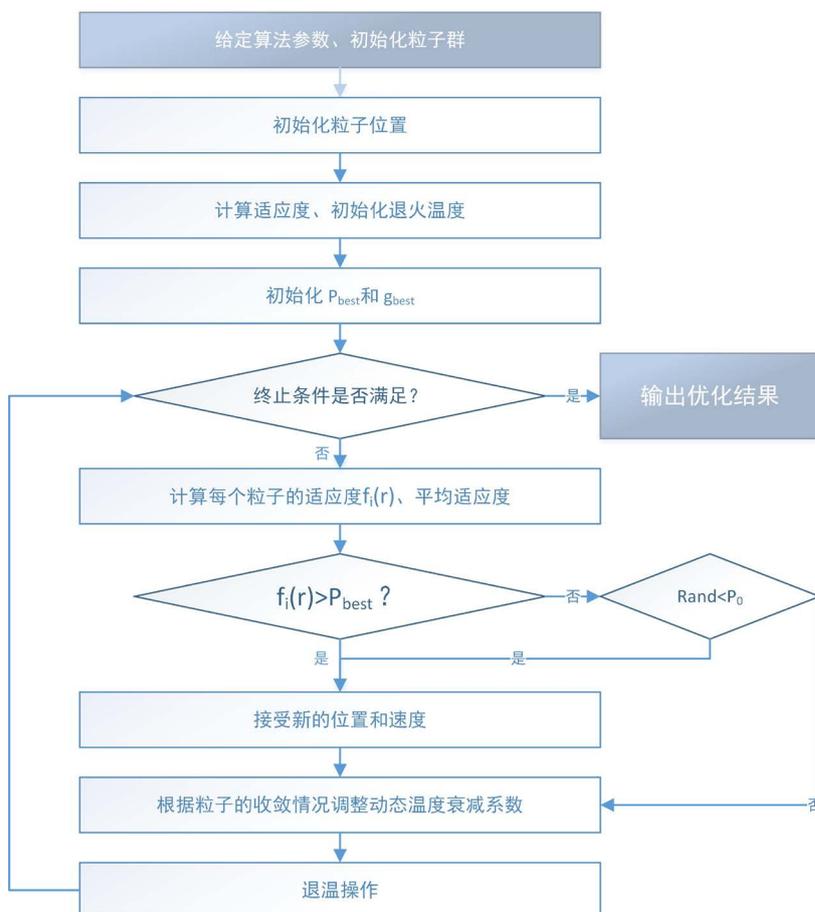


Figure 4. Improved simulated annealing algorithm

图 4. 改进的模拟退火算法

d) 在每个粒子当前位置中增加一个服从正态分布的随机值作为新的粒子位置。

e) 根据粒子个体适应度和平均适应度，计算温度自动衰减系数  $k$ 。

f) 更新当前环境温度： $T_0 = k \cdot T_e$ 。

迭代完成后得到  $g_{best}$  为函数的最优值。

## 2) 退火速度的改进

模拟退火算法的全局搜索能力取决于退火速度。一般固定温度衰变无法感知当前的收敛情况，因此我们引入温度动态衰减系数，使得算法可以根据当前粒子已知种群的适应度来感知局部和全局收敛情况，保证搜索过程不至于过早收敛陷入局部最优解。温度动态衰减系数为：

$$K = \mu + \left[ 1 - \exp(f_{pi} - f_{avg}) \right] \cdot N(0,1) / (2T_{r-1}) \quad (6)$$

其中， $f_{avg}$  为种群的平均适应度， $f_{pi}$  为当前粒子的适应度， $\mu$  为初始温度衰减系数， $N(0,1)$  表示标准正态分布， $T_{r-1}$  为上一次迭代中的粒子温度。

## 3) 最优分配的计算

a) 生成服从均匀分布  $U(0,256)$  的随机数并取整数后得到粒子初始位置，并转为八位二进制编码(不足八位在数字最前补零)。加工第一道工序的 CNC 用“1”表示，加工第二道工序的 CNC 用“0”表示，第  $i$  位数字表示编号为  $i$  的 CNC。

b) 将得到的二进制八位数代入基于贪心算法的求解算法中得到当前粒子位置对应的适应度  $f_i$ ，即八小时的生产件数。

c) 带入算法中迭代求得函数最优值  $g_{best}$ ，并将其对应的粒子位置转化为八位二进制数字，即得机器工序的最优分配。

### 5. 仿真结果分析

假设我们所研究的 RGV 智能加工流水线的五组参数如表 2 所示，单位为秒。将表 2 中的五组系统作业参数分别代入算法一，得出一道工序物料加工作业的产量和最优作业序列，具体结果见表 3。

针对两道工序物料，将表 2 中的五组系统作业参数分别代入算法二，可得出两道工序物料加工作业的产量、最优作业序列和最佳工序分配，具体结果见表 4。

Table 2. System operating parameters

表 2. 系统作业参数

参数组号	一	二	三	四	五
RGV 移动 1 个单位所需时间	20	23	18	21	22
RGV 移动 2 个单位所需时间	33	41	32	35	30
RGV 移动 3 个单位所需时间	46	59	46	50	42
CNC 加工一个一道工序的物料所需时间	560	580	545	600	520
CNC 加工一个两道工序物料的第一道工序所需时间	400	280	455	380	220
CNC 加工一个两道工序物料的第二道工序所需时间	378	500	182	500	480
RGV 为编号为奇数的 CNC 上下料所需时间	28	30	27	29	26
RGV 为偶数的 CNC 上下料所需时间	31	35	32	33	35
RGV 完成一个物料的清洗作业所需时间	25	30	25	30	25

Table 3. Production situation of material processing in single procedure

表 3. 一道工序物料加工作业的生产情况

参数组号	最优作业序列 $Sep_{min}$	生产件数
第一组	1→2→3→4→5→6→7→8→1	383
第二组	1→2→3→4→5→6→7→8→1	360
第三组	1→2→3→4→5→6→7→8→1	392
第四组	1→2→3→4→5→6→7→8→1	357
第五组	1→2→3→4→5→6→7→8→1	408

Table 4. Production situation of material processing in double procedure

表 4. 两道工序物料加工作业的生产情况

参数组号	最优作业序列 $Sep_{min}$	生产件数	各 CNC 的工序分配情况							
			1#	2#	3#	4#	5#	6#	7#	8#
第一组	1→2→3→4→5→6→7→8→1	253	—	二	—	二	—	二	—	二
第二组	1→2→3→4→5→6→7→8→1	211	二	—	二	—	二	—	二	—
第三组	1→2→5→3→1→6→5→7→8→4→1	240	二	—	—	—	二	—	—	二
第四组	1→2→3→4→5→6→7→8→1	210	—	二	—	二	—	二	—	二
第五组	1→5→7→8→4→3→2→5→6→3→1	229	二	二	—	二	—	二	二	—

## 6. 结论

本文基于图论, 给出了 RGV 作业时的动态调度策略, 并通过实验发现 RGV 作业时访问各 CNC 的次序会逐渐稳定, 从而得出 RGV 的最优作业序列。虽然我们所建的 RGV 智能调度模型基于贪心算法的思想, 但通过对一道工序物料加工作业结果的分析, 可验证出在本问题的背景下, 我们所求出的局部最优解即为全局最优解。

## 参考文献

- [1] 中国工业与应用数学协会. 全国大学生数学建模竞赛[Z]. B 题 2018.
- [2] 严蔚敏. 数据结构[M]. 北京: 清华大学出版社, 1992.
- [3] 潘全科, 朱剑英. 作业车间动态调度研究[J]. 南京航空航天大学学报, 2005, 37(2): 262-268.
- [4] 汪莹. 论贪心算法在图论中的应用[J]. 计算机光盘软件与应用, 2013(16): 309 + 311.
- [5] 刘爱军, 杨育, 李斐, 邢青松, 陆惠, 张煜东. 混沌模拟退火粒子群优化算法研究及应用[J]. 浙江大学学报(工学版), 2013, 47(10): 1722-1730.

### 知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>  
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2160-7583, 即可查询
2. 打开知网首页 <http://cnki.net/>  
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: [pm@hanspub.org](mailto:pm@hanspub.org)