# 智能RGV的动态调度策略

# 摘 要

本文根据题目给定的智能加工系统及系统作业参数，针对一道工序物料加工作业、两道工序物料加工作业、作业中故障处理等三种情况，建立数学模型，分别给出了相应的RGV最佳调度策略。

**针对一道工序物料加工作业的情况，**本文设计当RGV完成当前指令后若未接收到任何CNC的上料需求信号，RGV将会根据调度模型立即判别执行一次移动指令，移动到下一步发出上料需求信号的CNC前。并将作业效率最佳问题转换为一班次8小时内CNC处于工作状态总时间最长，并假设RGV具有短时间的记忆储存功能，能够记录与匹配RGV与各CNC进行最后一次交互的时间，为RGV设计“八步一走”调度模型，在RGV进行移动指令之前都会遍历搜索选择未来八次移动过后八台CNC的总等待时间最小的路径的第一步移动指令作为当前的移动指令。遍历所有可能的初始八台CNC的上料情况，依据RGV“八步一走”调度模型取成料数最多的初始CNC上料顺序，完成任务1。将题目给定的针对一道工序的三组数据带入模型计算，得出第一组最大物料加工数量为382，第二组为359，第三组为392；推算了不考虑RGV运动时间的理想状态下，三组数据的最大加工数量分别为384、368、392；得到三组数据下加工系统的作业效率分别为99.48%、97.55%、100%，完成任务2。

**针对两道工序物料加工作业的情况，**在不可更换刀具的前提下，由第一道工序与第二道工序的比值，兼容考虑第二道工序之后的清洗时间，按比例分别为CNC安装4:4、3:5、5:3的刀具配比，并在对称性原则基础上调试具体安装方案；为RGV设计三步捆绑（或四步捆绑加工调度模型）：RGV遍历三步，取捆绑加工后的完成时间最前的走法。遍历所有的初始可能路径，依据捆绑调度模型取成料数最多的初始CNC上料顺序，完成任务1。将给定的针对两道工序的三组数据带入模型计算，得出三组最大物料加工数量分别为253、209、236；选择的两类CNC数量配比分别为4:4、3:5、5:3；通过与理想状态下最大物料加工数量268、216、236进行比较，得到三组数据下加工系统的作业效率分别为94.40%、96.76%、100%，完成任务2。

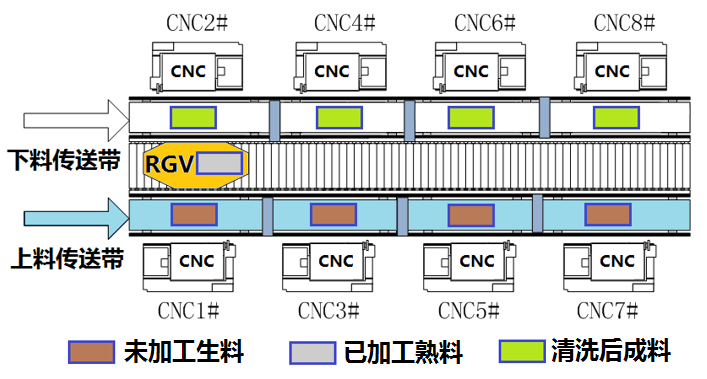
**针对作业中故障处理的情况，**本文将每一道工序加工的故障概率设为1%，在判定故障的CNC的加工时间内，以均匀分布随机一个时间点作为故障发生时间点，并从600~1200秒之间均匀随机生成一个整数作为修复时间，在一道工序与二道工序的模型中作出以下调整：在故障发生的那一刻起，在CNC未修复并发出上料需求信号之前，将该CNC从系统中暂时抹去，RGV在执行完当前指令后，不再进行有关该CNC的指令操作，直至CNC修复发出上料需求信号。考虑到故障发生的不确定性，以及人工修复时间的可操作性，在完成任务的基础下，再分别取修复时间为600~1200秒随机，600秒，900秒，1200秒做20组的随机试验探究成料数规律，进行均值和方差计算如下：一道工序的情况下，第一组数据关于4类修复时间的成料数方差分别为12.20，9.55，11.95，9.82；第二组数据方差分别为15.57,18.68,19.55,14.68；第三组数据方差分别为10.03,13.41,8.92,13.73；两道工序的情况下，第一组数据关于4类修复时间的成料数方差分别为9.66,7.38,7.12,13.17；第二组数据方差分别为7.85,3.39,5.87,9.69；第三组数据方差分别为7.66，4.58，7.72，10.13。由此可知，实际修复时，提升技工技术，将人工修复时间尽量控制在10~15分钟左右，可以较好增加结果稳定性。

**关键词：**RGV**智能动态调度、“八步一走”多步捆绑联动、故障排查概率分析**

# 一、问题重述

## 1.1 问题背景

RGV是一种无人驾驶、能在固定轨道上自由运行的智能车。在智能加工系统中，轨道式自动引导车RGV的调度方案对系统的加工效率存在着决定性的影响。RGV在智能加工系统中，因面临的工作环境各不相同，因此目前仍没有一个理想的算法可以对RGV的调度路径进行最佳优化，因此针对特定的系统工作环境，RGV的动态调度仍存在着很大的研究空间。本文研究的工作环境由8台CNC、1辆RGV、1条RGV直线轨道、1条上料传送带、1条下料传送带等设备组成。



**图1.1.1：智能加工系统示意图**

## 1.2 系统作业参数

**表1.2.1：智能加工系统作业参数的3组数据表 时间单位：秒**

|  |  |  |  |
| --- | --- | --- | --- |
| **系统作业参数** | **第**1**组** | **第**2**组** | **第**3**组** |
| RGV移动1个单位所需时间 | 20 | 23 | 18 |
| RGV移动2个单位所需时间 | 33 | 41 | 32 |
| RGV移动3个单位所需时间 | 46 | 59 | 46 |
| CNC加工完成一个一道工序的物料所需时间 | 560 | 580 | 545 |
| CNC加工完成一个两道工序物料的第一道工序所需时间 | 400 | 280 | 455 |
| CNC加工完成一个两道工序物料的第二道工序所需时间 | 378 | 500 | 182 |
| RGV为CNC1#，3#，5#，7#一次上下料所需时间 | 28 | 30 | 27 |
| RGV为CNC2#，4#，6#，8#一次上下料所需时间 | 31 | 35 | 32 |
| RGV完成一个物料的清洗作业所需时间 | 25 | 30 | 25 |

**注：每班次连续作业8小时。**

## 1.3 三种具体情况—工作环境

1. 一道工序的物料加工作业情况，每台CNC安装同样的刀具，物料可以在任一台CNC上加工完成；
2. 两道工序的物料加工作业情况，每个物料的第一和第二道工序分别由两台不同的CNC依次加工完成；
3. CNC在加工过程中可能发生故障（故障发生概率约为1%），每次人工处理故障（未完成的物料报废）需要耗时10~20分钟，故障排除后CNC即刻加入作业序列。要求分别考虑一道工序和两道工序的物料加工作业情况。

## 1.4 两个任务—问题要求

任务1：对一般问题进行研究，给出RGV动态调度模型和相应的求解算法；

任务2：利用表1中系统作业参数的3组数据分别检验模型的实用性和算法的有效性，给出RGV的调度策略和系统的作业效率，并将结果分别填入相应EXCEL表中。

# 二、问题分析

## 2.1 一道工序物料加工作业的问题分析

在该工作环境下，每台CNC都安装有相同的刀具，进行同一种加工工序，每一个生料均需在任意八台CNC之一上完成一道工序的加工并由RGV清洗后方可成料。本文为提高系统加工效率，考虑到当所有CNC处于加工状态时，RGV完成当前指令后，将会有长时间CNC不会对RGV发出上料需求信号，当CNC再次对RGV发出上料需求信号时，RGV将从上一次指令完成位置进行移动指令，因此考虑若在无谓的等原地待之前RGV提前执行移动指令将会更好地节约时间。因此在为RGV设计调度模型时，设计当RGV完成当前指令后若未接收到任何CNC的上料需求信号，RGV将会根据调度模型立即判别执行移动指令。为增大作业效率，即尽可能增加在一班次8小时内成料数量，即使得CNC尽可能多时间处于工作状态，换言之，即为CNC的总等待（闲置）时间尽可能小。综合以上考量，假设RGV具有短时间的记忆储存功能，能够记录与匹配RGV与各CNC进行最后一次交互的时间，由此时间为RGV设计“八步一走”调度模型，即RGV在判别执行移动指令之前，要遍历计算未来移动（或原地停留）八次，并要求每一次移动或停留过后必须间隔一道指令以上方可再次进行移动或停留指令，遍历搜索未来移动八次后八台CNC的总等待时间最小的路径的第一步移动指令（或原地停留）作为RGV的下一道指令，并要求实施检测判别完成当前指令后是否留有足够的时间在一班次8小时末回到初始点，若不能则拒绝接受有关移动回到初始点之外的任何指令。将表1系统作业参数的3组数据带入模型，给出RGV的调度策略，并估计理想状态下RGV运转速度足够快，忽略RGV的移动时间和清洗熟料的时间，进行计算此时情景下理想能加工完成的最大成料数，与模型结果进行比较估计系统的作业效率，并说明模型的实用性和算法的有效性。

## 2.2 两道工序物料加工作业的问题分析

在该工作环境下，CNC可选择安装两种刀具中的一种，对应进行两种加工工序，每一个生料均需在任意八台CNC之一上完成一道工序的加工并由RGV清洗后方可成料。仍然考虑当为RGV设计调度模型时，当RGV完成当前指令后若未接收到任何CNC的上料需求信号，RGV将会根据调度模型立即判别执行移动指令。此时需要考虑CNC上安装的刀具可以在初始固定后不再改变，也可以在CNC未进行加工时进行更换，这使得问题分为更换刀具与不更换刀具两种情况考虑。对于不更换刀具的情况，需要考虑安装不同刀具的CNC的配比及放置位置。就放置位置而言，可以根据简单的推算得出一个最佳方案；就配比而言，我们需要先筛选出可能出现最佳调度方案的几种情况，再对这几种情况分别建立模型完成调度。对每一种配比情况，由于该问题有两道工序，以CNC等待时间最短为目标会使计算更加复杂，因此直接以物料最大加工数量为目标。针对每一配比情况，会有一种最佳的分组方案，即以固定个数的物料完成加工的过程为一组，按照组序进行逐步调度。对每一组，有若干可能的顺序完成组内所有步骤，取其中时间最短的顺序方案，将每一组的方案组合成最终的调度方案，即为所求的模型。对于更换刀具的情况，需要考虑其更换刀具的频率、提高的加工效率，以此判断更换刀具的模型是否有意义，若无意义，则忽略此情形。将表1系统作业参数的3组数据带入模型，给出RGV的调度策略，并估计理想状态下RGV运转速度足够快，忽略RGV的移动时间和清洗熟料的时间，进行计算此时情景下理想能加工完成的最大成料数，与模型结果进行比较估计系统的作业效率，并说明模型的实用性和算法的有效性。

## 2.3 作业中故障处理的问题分析

在该工作环境下，CNC在加工过程中约有1%的概率发生故障，故障发生时间出现在CNC加工过程中的任一时刻的概率均相同，对故障进行人工排查修复处理需要耗时为10~20分钟，故障排除后CNC即刻加入作业序列。假设一道工序物料加工过程和两道工序的各步物料加工过程中故障发生的概率相同，皆视为1%。每一道工序加工均有1%的概率发生故障，在CNC的加工时间内，以均匀分布随机一个时间点作为故障发生时间点，并以10~20分钟，即600~1200秒之间均匀随机生成一个整数作为修复时间。对一道工序物料加工作业时间，两道工序物料加工作业的模型作出以下调整：在故障发生的那一刻起，在CNC未修复并发出上料需求信号之前，将该CNC从系统中暂时抹去，RGV在执行完当前指令后，不再进行有关该CNC的指令操作，直至CNC修复发出上料需求信号。考虑到故障发生的不确定性，模型结果的稳定性与能够人工干预的故障修复时间息息相关，因此本文在建模求解600~1200秒之间均匀随机生成一个整数作为修复时间后，还将分别取修复时间为600秒，900秒，1200秒做20组的随机试验，查验当将修复时间尽量控制在哪一个范围内，模型结果的稳定性将会更好。

# 三、模型假设

1. 假设对于确定的某一工序，CNC加工时间恒定且一致。
2. 假设传送带会将物料及时运送到上料处，不产生等待时间。
3. 不考虑两道工序间废渣的处理，即假设第一道工序加工产生的废渣不会对第二道工序的加工产生影响。
4. 假设RGV在接收到CNC发出的上料需求指令之后，当RGV移动到该CNC位置前时，RGV才会对其进行上下料。
5. 假设RGV具有短时间的记忆储存功能，能够记录与匹配与各CNC进行最后一次交互的时间。
6. 假设一道工序物料加工过程和两道工序的各步物料加工过程中故障发生的概率相同，皆视为1%。
7. 假设RGV性能良好，严格按照各组参数时间按时执行指令。

# 四、符号说明

|  |  |
| --- | --- |
| **关键符号**  **符号说明** | **关键符号**  **符号说明** |
| *Ai* | 编号为*i*的CNC的状态 | |
| *H* | 所有CNC的总等待时间 | |
| *ti* | 移动*i*个单位所需要的时间 | |
| *u*1*k* | 上料开始时间点 | |
| *u*2*k* | 下料开始时间点 | |
| *hik* | 表示第*i*台CNC对第*k*个物料的等待时间 | |
| *tt* | CNC完成一个物料的加工需要时间 | |
| *Hv* | 第*v*个初始情况对应的全过程CNC总等待时间 | |
| *Ni* | 理想情况一个班次能生产的成品数 | |
| *ƞi* | 系统的作业效率 | |
| *Ci* | 第*i*台CNC的故障状态 | |

# 五、模型的建立与求解

## 5.1 一道工序加工模型建立与求解

### 5.1.1 任务一

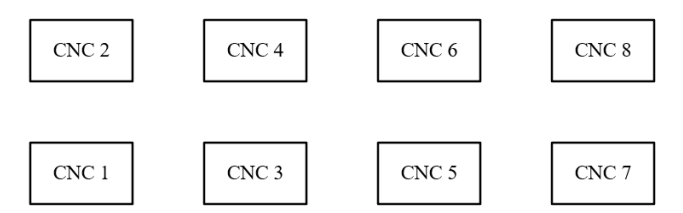
任务1要求给出一般情况下RGV的动态调度模型与相应的求解算法。一般情况表示适应可选范围内的任意情况，即不能仅仅建立适应于某一组或几组参数值的模型，而要给出对任何可选参数值均适应的一般化模型。因此，模型的建立不能基于任何特定参数值、依赖任何特殊情形；同理，相应的求解算法也不能针对任何一组特定参数，应当在给出任何一组可取参数时，都能给出相应的理想解。

* 智能加工系统的工作原理

问题展示的智能加工系统主要由三部分构成，即8台计算机数控机床（CNC）、1辆轨道式自动引导车（RGV）与1条RGV直线轨道、1条上料传送带与1条下料传送带，每个物料完成加工都需要经过三个部分的联动工作。因此，首先需要明确每个部分的工作原理，并对每个部分的工作过程准确进行定量描述。

* CNC的工作原理

在整个加工过程中，CNC共有空闲状态、上下料状态、加工状态、需求状态等4个状态。其中，空闲状态表示CNC上没有物料的状态；上下料状态表示RGV正在给CNC上下料的状态；加工状态表示CNC正在对物料进行加工的状态；需求状态表示CNC完成加工后等待RGV下料的状态。问题给出的加工系统中有8台已编号的CNC，每台CNC的具体位置如图所示：



**图5.1.1：CNC位置示意图**

由图可知，横向上相邻两台CNC的间隔相等；纵向上4组CNC两两位于同一竖直线。

设编号为*i*的CNC的状态为*Ai*，并对各种状态赋值：



其中，0、3表示的状态CNC未进行工作，1、2表示的状态CNC正在进行工作；但在实际情况中，我们期望CNC进行工作的时间尽可能长。

* RGV的工作原理

在整个加工过程中，RGV始终位于RGV直线轨道上，共有等待状态、运动状态、上下料状态、清洗状态等4个状态。其中，等待状态表示RGV静止在轨道上不进行任何工作的状态；运动状态表示RGV由一个位置驶向另一个位置时的状态，此状态不进行工作；上下料状态表示RGV正在给CNC进行上下料的状态；清洗状态表示RGV给物料进行清洗并将其放置到下料传送带时的状态[1]。

设RGV的状态为*B*，并对各状态进行赋值：



其中，0、2、3状态时，RGV只能处于轨道给定4个位置中的任意一个；1状态时，RGV正在由其中一个位置运动至另一个位置。当RGV处于某一位置时，只能与该位置的上下两台CNC进行联动。

设*j*为RGV所处的位置，则：



其中，*i*表示CNC的编号。

当RGV处于1状态时，设其运动状态为：



表示RGV此时RGV正由位置*j*1运动到位置*j*2。

* 传送带的工作原理

传送带在加工系统启用时仅有一个状态，即运动状态。上料传送带将负责运送未加工的生料，下料传送带负责运送已加工的熟料。已知传送带既可连动，也可独立运动；在实际中，传送带的运动速度也能达到数米每秒。基于上述两点，不妨假设当RGV需要时，传送带总能在相应位置给出生料或熟料空位，无需等待。

* 加工系统的调度时间

对于一道工序的加工系统而言，首要的目标是尽可能提高加工效率。最能直接反应这一目标的即为相同时间内加工的物料个数，个数越大，加工效率越高。然而物料的加工是一个复杂动态过程，物料在不同时间可能处于CNC中，也可能处于RGV中，我们需要进一步将目标进行转化。

容易得到，当物料处于加工状态时，相应的CNC必定处于工作状态；当物料处于非加工状态时，不对应工作时间的CNC。因此，不妨将目标转化为尽可能减少CNC的非工作时间，即需要给出一个调度模型，使得CNC的非工作总时间最小。

* 总时间的确定

加工系统的连续工作时间为8小时，不妨设8小时为一次完整的加工过程，目标为8小时内CNC的非工作总时间最小。由于上下料，位置移动等指令在实际中可以在几十秒内完成，因此将时间单位转化到秒，则一次加工过程的总时间为：



则对于依一次完整的加工过程而言，第0*s*加工系统启动，RGV由初始位置1开始执行指令；第28800*s*加工系统终止，RGV回到初始位置1。

* 时间参数的确定

对于整个加工系统，需要相应的时间参数反应调度过程。

首先，RGV需要在4个不同位置间来回移动，可能连续移动的位置单位有1、2、3。因此，设RGV移动一个单位的时间为𝑡1，移动两个单位的时间为𝑡2，移动3个单位的时间为𝑡3。除此以外，RGV自身拥有清洗物料的工作，故设完成一次清洗所需的时间为𝑡4。

其次，CNC需要时间单独对物料进行加工，设CNC完成一个物料的加工需要时间*tt*。

最后，对于RGV与CNC的联动，由于实际中RGV与两边CNC的位置非完全对称，我们设RGV对1、3、5、7号CNC完成一次上下料所需时间为𝑠1，对2、4、6、8号CNC完成一次上下料所需时间为𝑠2。

* 时间变量的确定

为了给出最佳调度方案，需要对未知、不确定的时间变量进行确定。首先考虑CNC的总等待时间，不妨设为*H*，则有：



其中，h𝑖𝑘表示第*i*台CNC对第*k*个物料的等待时间，物料编号*k*根据RGV抓取生料的顺序来计。

进一步考虑物料*k*，设其上料开始时间点为𝑢1𝑘，下料开始时间点为𝑢2𝑘，则有：



其中*s*表示𝑠1或𝑠2，具体取值根据*k*情况而定。

* 加工系统的优化模型
* 目标函数的确定

对于加工系统的一次8小时运行而言，需要给出调度方案使得CNC总等待时间最少，则建立目标函数如下：



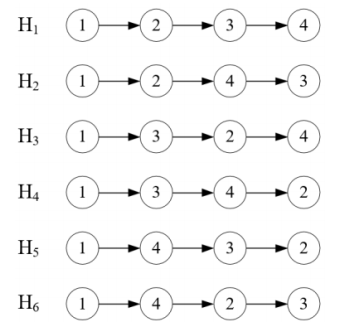
*H*表示8小时内的CNC总等待时间。

* 初始方案的规划

Step1：首先考虑初始情况。已知RGV的初始位置为左端位置1，CNC均属于空闲状态，而实际情况下，一次上料或移动的时间又远小于CNC完成一次加工的时间。综上，初始的指令确定为按照一定顺序完成对8台CNC的上料。

Step2：当RGV在某一位置时，对该位置的两台CNC依次完成下料期间无需移动，而对其它位置的CNC完成下料需要额外的移动时间。因此获得结论：当RGV对某一位置的一台CNC进行上下料时，下一个上下料的目标优先选择该位置的另一台CNC；RGV首先对初始位置1的两台CNC进行上料。

Step3：由上述两个结论，RGV完成初始对8台CNC的上料共有6种情况：



**图5.1.2：初始上料情况图**

故目标函数转化为：

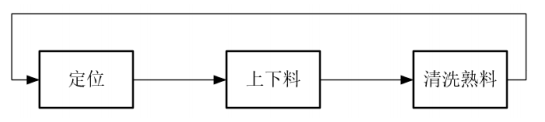


其中*Hv*表示第*v*个初始情况对应调度方案中CNC的总等待时间。

* 调度方案的优化模型

Step1：对任一初始方案，完成8台CNC的上料后处于位置𝑗。考虑下一步RGV的动向，虽然在后续加工中多出清洗的步骤，但实际中过长的加工时间仍会使RGV优先进行完一轮对8台CNC的上下料。因此考虑RGV8次连续完整作业的动态规划。

Step2：对于整个加工过程，RGV的工作流程如下：



**图5.1.3：RGV一次完整作业示意图**

考虑8次如图所示的RGV一次完整作业，当RGV对某台CNC进行上下料时，其它CNC的加工可能已经完成并进行需求提示，则在RGV到达前，这些CNC都将产生等待时间；此外，根据之前的结论，RGV会优先对同一位置的两台CNC依次作业。综上两点，对接下来的8次工作，共有48种调度方案。

Step3：对任意一种调度方案，设RGV每进行一次作业操作，都会使第*i*个CNC产生等待时间h𝑖𝑚，其中*m*表示8次连续作业中的第*m*次。则这种调度方案8个步骤产生的CNC等待总时间为：



特别地，这里*n*=1，表示进行全过程中的第一个8次连续作业。

Step4：求出每一种调度方案下的总等待时间，共有48个结果，取最小总等待时间对应的方案，对应目标函数如下：



该最优方案对应的8次连续作业中，第一步中第*i*个CNC的最小等待时间为h𝑖1。

Step5：选取8次连续作业中的第一次，令RGV执行该指令，完成调度方案中的一步。则该步骤中8个CNC产生的总等待时间为：



以第一个8八次连续作业中确定的第一次结束作为开始，重复Step3、4的步骤，进行若干个8次连续作业，每个8次连续作业确定实际的一次作业，直到28800秒的工作流程结束。对结束前的状态进行讨论，若最后一个8次连续作业的产生实际的一次作业无法在第28800秒前完成并令RGV返回初始位置，则放弃该次作业，令RGV直接回到初始位置。则对第v个初始情况对应的完整调度方案，有：



其中，𝐻𝑣表示第*v*个初始情况对应的全过程CNC总等待时间。

Step6：综合Step1-5，得到完整的优化模型如下：



* 调度模型的求解算法

模型求解的前提为给定一组可靠的参数数据，关于参数的具体内容，已经在先前给

出详细说明。

* 优化模型求解

对于48种调度方案，在MATLAB中使用遍历搜索算法进行求解，得到总等待时间最小的方案；对于其余过程，均可利用MATLAB进行直接计算。

* 初始目标求解

但此时我们必须考虑到，CNC总等待时间最小值是我们为方便求解而转化得到的目标。利用上述模型求解到最优方案后，需将目标转化回最初状态，求解得到最多的物料加工数，并给出相应的上下料开始时间点。

已知物料编号*k*根据RGV抓取生料的顺序来计，其上料开始时间点为𝑢1𝑘，下料开始时间点为𝑢2𝑘。利用MATLAB对整个最优调度方案进行遍历搜索，得到第*k*个物料编号，第*k*个上料开始时间点𝑢1𝑘，第*k*个下料开始时间点𝑢2𝑘，并记录所有*k*对应的结果。

* 结果总结

经过上述求解步骤，我们最终可以得到最优调度方案、最小总等到时间、最大物料加工数、每个被加工物料对应的上下料开始时间。

### 5.1.2任务二

* 一道工序加工模型的结果

将题目给出的三组参数分别带入模型，利用MATLAB编程求解(代码文件Ques1.m详见附录五)。得到的第一组结果如下表：

**表5.1.1：一道工序调度模型的第一组数据结果**

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 加工CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 588 |
| 2 | 2 | 28 | 641 |
| 3 | 3 | 79 | 717 |
| …… | | | |
| 380 | 4 | 27956 | 28547 |
| 381 | 5 | 28032 | 28623 |
| 382 | 6 | 28085 | 28676 |

第二组最优结果如下表（两个最优序列）：

**表5.1.2：一道工序调度模型的第一组数据结果 表5.1.3：一道工序调度模型的第一组数据结果**

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 加工CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 610 |
| 2 | 2 | 30 | 670 |
| 3 | 3 | 88 | 758 |
| 4 | 4 | 118 | 818 |
| 5 | 7 | 194 | 924 |
| 6 | 8 | 224 | 984 |
| 7 | 5 | 282 | 1072 |
| 8 | 6 | 312 | 1132 |
| …… | | | |
| 359 | 5 | 28076 | 28704 |

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 加工CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 610 |
| 2 | 2 | 30 | 670 |
| 3 | 3 | 88 | 758 |
| 4 | 4 | 118 | 818 |
| 5 | 5 | 176 | 906 |
| 6 | 6 | 206 | 966 |
| 7 | 7 | 264 | 1054 |
| 8 | 8 | 294 | 1114 |
| …… | | | |
| 359 | 7 | 28058 | 28686 |

第三组最优结果如下表：

**表5.1.4：一道工序调度模型**

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 加工CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 572 |
| 2 | 2 | 27 | 624 |
| 3 | 3 | 77 | 699 |
| …… | | | |
| 390 | 6 | 27997 | 28574 |
| 391 | 7 | 28072 | 28649 |
| 392 | 8 | 28124 | 28701 |

* 一道工序模型的实用性

从结果来看，模型针对3组参数的求解情况均符合实际，既没有出现异常的上下料时间、也没有异常的两个相邻编号的物料上下料开始时间间隔。而这3组数据各有特点，反应了不同工序的不同耗时配比，故3组结果均合理足以证明模型具有较强的实用性，可针对任何符合实际的一组参数进行求解，得到合理的最佳效率方案。

分析最优的结果序列，发现恰好为1-8循环作业的自然序列。这与实际的加工过程中，RGV的运动通常存在某种规律、不会做无规则的复杂运动相符合，进一步证明了模型的实用性。

* 一道工序模型系统的作业效率

首先计算最理想情况下即仅考虑物料加工时间和上下料时间，8小时能制造出多少个成品[2]。

理想初始情况下，对8台CNC进行上料操作，忽略RGV移动所消耗的时间，记第*i*组初始上料的总时间为*t*0i：



理想情况下第*i*组加工的工件数量*Ni*为：



*tti*表示第*i*组单个物料的加工时间，T表示一个班次的总时间。[]表示对括号内的式子向下取整。

将3组数据的参数分别带入上式得到：*N*1=384，*N*2=368，*N*3=392。需注意，这是理想情况下求得的8小时最多能加工的零件，并没有考虑一开始小车路上移动的时间和因RGV未能及时到达发出需要上下料的CNC位置而延误的时间。所谓理想情况即RGV总能及时达到需要上下料的CNC位置。

第*i*组的作用效率如下：



*nij*表示利用情况*i*的模型求得的第*j*组的成品数量。

得到：*ƞ*11=99.48%，*ƞ*12=97.55%，*ƞ*13=100%，

## 5.2 两道工序加工模型的建立与求解

### 5.2.1 任务一

对两道工序的情况而言，加工系统大部分的工作原理都与一道工序相同，仅少数发 生了改变。因此，对没有改变的工作原理、参数、变量，均延用情况一中的定义。

* 智能加工系统的工作原理
* CNC的工作原理

在两道工序中，CNC的位置、工作状态仍与一道工序相同，不同之处在于原先8台CNC都进行同一道工序，而此时的CNC可以通过更换刀具改变可加工工序。我们设第*i*台CNC对应的可加工工序为𝐺*𝑖*，则：



* RGV的工作原理

对任意一块物料，当完成第一道工序时，RGV需要将物料转移到能进行第二道工序的CNC中。在此过程中，第一道工序产生的碎屑可能需要进行处理，否则无法直接进行第二道工序。但由于此过程实际耗费的时间极小，考虑在模型中忽略这个时间。

考虑RGV的工作状态，在两道工序中，上下料状态显然会出现两种情况，即给加工第一道工序的CNC上下料和给加工第二道工序的CNC上下料。因此，设RGV的状态为𝐵1，并对各状态进行赋值：



* 加工系统的调度时间

对于两道工序的加工系统而言，首要的目标仍然是尽可能提高加工效率。最能直接反应这一目标的即为相同时间内加工的物料个数，个数越大，加工效率越高。然而区别于一道工序，两道工序的目标是不能盲目转换为CNC总等待时间最小的。对每一个物料，需要完成两道工序才能完成加工，而能加工每到工序的CNC编号是不确定的，故CNC总等待时间最小不能直接反应加工物料数最多，可能会出现更多半成品的情况。因此，两道工序的目标转化需要根据CNC具体调配情况而确定。

* 时间参数的确定、

不同加工工序的CNC加工物料的时间不同，因此设加工第一道工序的CNC单次加工时间为𝑡𝑡1，加工第二道工序的CNC单次加工时间为𝑡𝑡2。

* 时间变量的确定

根据RGV抓取生料的顺序，将第*k*个被抓取的生料记为物料*k*。则对每个物料*k*，设其第一道工序上料开始时间点为𝑢3𝑘，下料开始时间点为𝑢4𝑘；第二道工序上料开始时间点为𝑢5𝑘，下料开始时间点为𝑢6𝑘。

已知RGV对1、3、5、7号CNC完成一次上下料所需时间为𝑠1，对2、4、6、8号CNC完成一次上下料所需时间为𝑠2。则对第一道工序，有：



对第二道工序有：



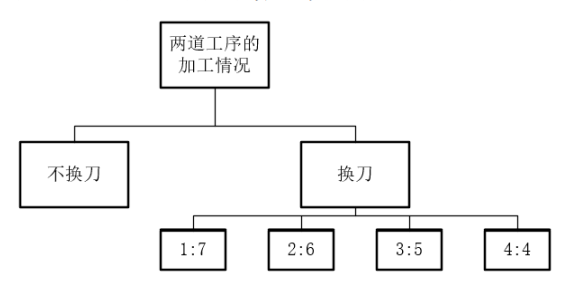
其中s表示𝑠1、𝑠2中对应*k*的情况。

* 加工系统的优化模型

对于两道工序的加工而言，问题给出两条准则：CNC在加工的过程中不能更换刀具；同一块物料的两道工序必须在不同的两台CNC上进行加工。分析这两条准则，我们不难发现准则在给出限定的同时也透漏了一条关键信息：CNC在非加工时间可以更换刀具。因此，两道工序的加工过程中，CNC可以持续的进行动态变化。

针对这一变化，我们不妨进行分类讨论，考虑可更换刀具与不可更换刀具两种情况，给出不同的模型使加工系统达到效率最高。进一步考虑不可更换刀具的情况，此情况下初始的CNC可加工工序设定即为整个流程的设定。首先排除8台CNC可加工工序相同的情况，此状态无法进行完整加工。于是可能出现的两道加工工序的CNC台数比为1:7、2:6、3:5、4:4。

综上，得到两道工序的加工系统的考虑分类情形如图所示：



**图5.2.1：两道工序加工情况的分类图**

* 不更换刀具
* CNC的配比方案

先前已经分析得出可能出现的两道加工工序的CNC台数比为3:5、4:4，而具体应该使用怎样的配比，主要取决于两道工序的加工时间[3]。例如，当两道工序的加工时间相近时，适合使用4:4的CNC配比；当一道工序的加工时间是另一道工序的3倍时，使用4:4会导致产生过长的CNC闲置等待时间，使用2:6的配比较为合理。但根据同一位置优先的法则，当比例为2:6或7:1时，模型会因法则失效程度过大而使可行性性降低，且实际中两道工序的加工CNC比例不会严重失衡，故我们不考虑2:6、1:7的配比情况。因此，我们给出如下判断法则：对第一道工序的CNC单次加工时间𝑡𝑡1，加工第二道工序的CNC单次加工时间𝑡𝑡2，有比值：



3种配比比例共有7个比值，分别为3/5、1、5/3，将比值记作*Tt*。寻找与加工时间比值最接近的配比，相应公式如下：



将该最小值对应的配比*Tt*确定为当前参数对应选择的CNC配比。然而我们必须考虑到，配比的分子分母越接近，系统的周转能力会更加灵活，不排除比分子分母更接近的配比会产生更优方案的可能。因此，做如下规定：

当配比*Tt*确定时，将参数分别带入*Tt*配比与分子分母比其更接近的配比中求解最优结果，取这其中的最优结果作为最终结果。例如，求得*Tt*为3/5，则取3/5、4/4、5/3这3中配比情况的模型分别求解，取其中最优解的配比作为最终方案。

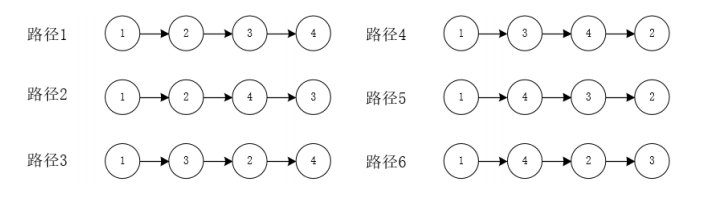
利用上述法则，可以保证给出不同的加工时间参数时，可以选择更合理的初始CNC配比设置，使得在该配比下得到的最佳调度方案优于其它配比下的方案。

* 4:4型的调度方案

Step1：首先需确定不同刀具类型的CNC应摆放的位置。考虑与一道工序中相似的问题，当RGV处于某一位置并完成对该位置一台第一道工序类型CNC的下料时，需要寻找第二道工序类型的CNC进行上料；此时，最佳的选择是该位置的另一台CNC。因此，我们得出将两类CNC分别置于轨道两次为最佳的分配方式。



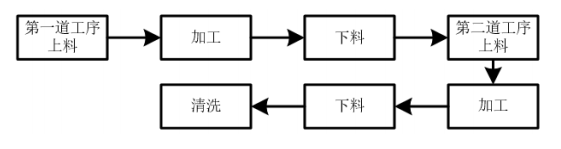
Step2：考虑初始情况。RGV从位置1出发，由于位置1-4各有个加工第一道工序的CNC，故初始仅需要对4台CNC进行上料，则上料的顺序共有以下6种情况：



**图5.2.2：初始上料的顺序图**

故模型将基于6种初始状态给出6种最佳调度方案，取6种方案中的最优作为结果。

Step3：对于每一块物料，经历的加工流程如图所示：



**图5.2.3：物料的加工流程图**

对任意一种初始状态，RGV下一步开始对目前CNC上的4块物料进行操作，直到4块物料都完成如图所示的流程，RGV会开始对其它物料进行作业[4]。因此，我们将每4块物料绑定为一组；对第一组，从初始放置结束为起始，到第4块物料完成清洗为结束；对接下来的第*x*组，以*x*−1组结束为起始，第4*x*个物料清洗完成为结束；循环这个流程，直到8小时工作结束。

Step4：对于每一组而言，第1块物料完成第一道工序后有4个可选择的第二道工序CNC进行放置；第2块物料完成第一道工序后，有3个可选择的第二道工序CNC进行放置。以此类推，得到总的选择数为4×3×2×1=24。

同样，对于每一组而言，在4块进行第一道工序的物料中选择第一块加入第二道工序的物料，共有4个可能的选择；选择第二块加入第二道工序的物料，共有3个可能的选择。依次类推，得到总的可选择数为4×3×2×1=24。

因此，得到每一组4块物料加工过程的可选调度方案数为24×24×=576。

设第x组的开始时间点为𝑓1*𝑥*，结束时间为点𝑓2*𝑥*，则对每一组的576种可选方案，取最小时间的方案为该组最终调度方案。公式表示为：



其中，𝑓0表示完成一组调度需要的总时间。

Step5：考虑最后一组的情况。当RGV完成对最后一组第*m*块物料的清洗后返回初始位置时，整个流程时间恰好已超出28800秒，则放弃对第*m*块物料的加工，直接返回初始位置。设最后一组加工完成的物料数为*d*，则：



Step6：确定目标函数。设每种初始状态对应的最大加工物料数分别为𝐷1、𝐷2、𝐷3、𝐷4、𝐷5、𝐷6。则对任意一种初始状态，最大加工物料数表示为：



其中，*n*为28800秒内完整完成的组数。

故目标函数表示为：



Step7：根据RGV抓取生料的顺序，将第*k*个被抓取的生料记为物料*k*。记录每个*k*，并记录每个*k*对应的第一道工序上料开始时间点𝑢3𝑘，下料开始时间点𝑢4𝑘；第二道工序上料开始时间点𝑢5𝑘，下料开始时间点𝑢6𝑘。

* 3:5型的调度方案

Step1：首先需确定不同刀具类型的CNC应摆放的位置。

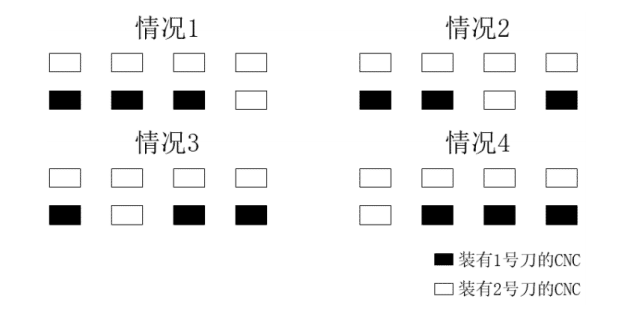
第一，根据之前同一位置优先的原则，尽量使不同类型的CNC在同一RGV位置点的两侧。由此确定3个加工第一道工序的CNC应当不处于同一位置的两侧。

第二，由一道工序的加工系统，我们已知轨道两侧的CNC上下料时间不同，因此具体哪一类CNC放置在哪一侧，需要根据给出的加工时间比值*TT*，而具体目标为尽量减少RGV的等待时间。综上给出判定方式如下：



这里需要注意，由于第一道工序的CNC仅有3台，因此第一道工序CNC的一侧必定会出现1台第二道工序的CNC。

第三，基于前两点，可行的放置情况共有4种（假设在轨道哪一侧已经选定），如图所示：



**图5.2.4：两种刀具的CNC的分布图**

接下来从4种情况中确定最佳的放置方案。已知RGV的初始位置在位置1，故优先选择在位置1放置第一道工序的CNC；由对称关系，当RGV进行工作时，在位置1、4的情况近似，在位置2、3的情况近似，故在已经选择了位置1的情况下，在位置4也放置第一道工序的CNC；同样根据对称性，在2、3位置放置的结果会近似相同，不妨将最后一个CNC放置在位置2。

综上所述，最终选定的放置方案为情况2。

Step2：考虑初始情况。RGV从位置1出发，由于位置1、2、4各有个加工第一道工序的CNC，故初始仅需要对3台CNC进行上料，上料的顺序共有以下2种情况：



**图5.2.5：上料顺序图**

故模型将基于2种初始状态给出2种最佳调度方案，取2种方案中的最优作为结果。

Step3：对任意一种初始状态，RGV下一步开始对目前CNC上的3块物料进行操作，直到3块物料都完成先前给出的加工流程，RGV会开始对其它物料进行作业。因此，我们将每3块物料绑定为一组；对第一组，从初始放置结束为起始，到第3块物料完成清洗为结束；对接下来的第*x*组，以*x*−1组结束为起始，第3*x*个物料清洗完成为结束；循环这个流程，直到8小时工作结束。

Step4：对于每一组而言，第1块物料完成第一道工序后有5个可选择的第二道工序CNC进行放置；第2块物料完成第一道工序后，有4个可选择的第二道工序CNC进行放置。以此类推，得到总的选择数为5×4×3=60。

同样，对于每一组而言，在3块进行第一道工序的物料中选择第一块加入第二道工序的物料，共有3个可能的选择；选择第二块加入第二道工序的物料，共有2个可能的选择。依次类推，得到总的可选择数为6。

因此，得到每一组3块物料加工过程的可选调度方案数为60×6=360。

设第*x*组的开始时间点为𝑓1𝑥，结束时间为点𝑓2𝑥，则对每一组的24×24种可选方案，取最小时间的方案为该组最终调度方案。公式表示为：



其中，𝑓0表示完成一组调度需要的总时间。

Step5：考虑最后一组的情况。当RGV完成对最后一组第*m*块物料的清洗后返回初始位置时，整个流程时间恰好已超出28800秒，则放弃对第*m*块物料的加工，直接返回初始位置。设最后一组加工完成的物料数为d，则：



Step6：确定目标函数。设每种初始状态对应的最大加工物料数分别为𝐷1、𝐷2。则对任意一种初始状态，最大加工物料数表示为：



其中，*n*为28800秒内完整完成的组数。

故目标函数表示为：



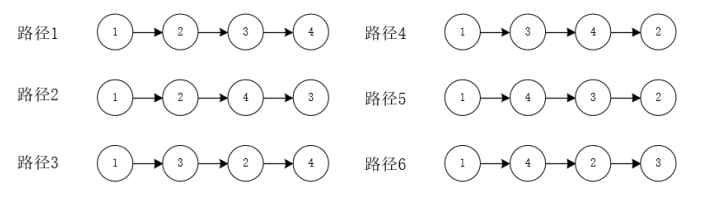
Step7：根据RGV抓取生料的顺序，将第*k*个被抓取的生料记为物料*k*。记录每个*k*，并记录每个*k*对应的第一道工序上料开始时间点𝑢3𝑘，下料开始时间点𝑢4𝑘；第二道工序上料开始时间点𝑢5𝑘，下料开始时间点𝑢6𝑘。

* 5:3的调度方案

5:3型的调度方案与3:5型有许多相似之处，因此接下来只对改变的步骤进行解释。

Step1的改变：根据同一位置有效的法则，5:3型固然先将4个第一道工序的CNC放置在同侧；在3:5型的放置方案中，第一道工序的CNC位于位置1、2、4，为了不影响更优的位置，将第5个第一道工序的CNC放置在位置3的另一侧。故CNC放置方案确定。

Step2的改变：考虑初始情况。RGV从位置1出发，由于位置1、2、3、4都存在加工第一道工序的CNC，故初始需要对5台CNC进行上料，结合同一位置优先的法则，上料的顺序共有以下6种情况：



**图5.2.6：初始上料顺序图**

故模型将基于6种初始状态给出6种最佳调度方案，取6种方案中的最优作为结果。

Step4的改变：对于每一组而言，第1块物料完成第一道工序后有3个可选择的第二道工序CNC进行放置；第2块物料完成第一道工序后，有2个可选择的第二道工序CNC进行放置。以此类推，得到总的选择数为6。

同样，对于每一组而言，在3块进行第一道工序的物料中选择第一块加入第二道工序的物料，共有5个可能的选择；选择第二块加入第二道工序的物料，共有4个可能的选择。依次类推，得到总的可选择数为60。

因此，得到每一组3块物料加工过程的可选调度方案数为60×6=360。

Step6的改变：由于初始状态有6种，故目标函数表示变为：



* 更换刀具

考虑更换刀具的情况，首先明确换刀的目的。当可以换刀时，不再有CNC的的配比放置方案，一切以提高效率为准。

Step1：显然，初始的最佳状态为8台CNC都进行第一道工序。

Step2：当RGV为物料更换工序时，利用一道工序的调度模型选取下一台CNC执行第二道工序。此时对该CNC进行刀具更换工作，并拾起该CNC先前加工好第一道工序的物料，放置前一个需要加工第二道工序的物料。

Step3：重复Step2的步骤，直到完成全部的调度。这里我们不难发现，为了提高效率而制定的上述调度方案中换刀是极其频繁的，每执行一次操作时都大概率进行换刀，具体表现为满足两个法则：

1．若前一个完工的CNC执行了第一道工序，则RGV下一个联动的CNC需更换为第二道工序的刀具。

2．若前一个完工的CNC执行了第二道工序，则RGV下一个联动的CNC需更换为第一道工序的刀具。

在实际中换刀的时间大约为2−5*s*，因此不会对加工效率产生明显影响。但是另有3点我们必须考虑到：

1．加工系统是否容易发生故障取决于频繁的操作，如果为了提高效率盲目进行上述频繁换刀的操作，会使得系统发生故障的概率大大提升，得不偿失。

2．且在实际情况中，不会出现为提高效率而频繁换刀的操作。

3．初步对频繁换刀的理想状态进行估计，相比不换刀的最佳方案，28800秒内增加的物料加工数量均在5个左右，效率提高并不明显。

上述个问题直接从多方面否定了在调度中频繁更换刀具的不合理性。因此，我们在确定后续的调度方案中，不考虑CNC更换刀具的情况，直接将其排除。

* 调度模型的求解算法

对于情况二，所有的求解步骤都已经在模型中一一给出；因此只需要依据模型，将遍历搜索算法与直接计算方法结合，利用MATLAB依次得到模型需要的结果，即可完成对情况二的求解。

### 5.2.2 任务二

* 两道工序加工模型的结果

首先考虑不换刀的情况，将题目给出的三组参数分别带入模型，利用MATLAB编程求解。经过配比检验，得到第一组参数下确定使用4:4的放置方案；第二、三组参数需要分别带入4:4、3:5、5:3的模型中求解，第二组4:4得到成品209个成品、3:5得到200个成品、5:3得到159个成品；第三组4:4得到成品232个成品、3:5得到174个成品、5:3得到236个成品（三种配比情况的结果表详见附录二十六），取其中最优的放置方案。从得到的结果发现：第一、二组适用4:4，第三组适用5:3。（5个一号刀，3个二号刀）其中，两道工序加工的第一组结果如下(代码文件Ques201.m详见附录十二)：

**表5.2.1：两道工序调度模型第一组数据结果表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 加工物料序号 | 工序1的CNC编号 | 上料开始时间 | 下料开始时间 | 工序2的CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 428 | 2 | 456 | 884 |
| 2 | 3 | 48 | 507 | 4 | 535 | 988 |
| 3 | 5 | 96 | 586 | 6 | 614 | 1092 |
| …… | | | | | | |
| 251 | 5 | 27584 | 28026 | 6 | 28054 | 28496 |
| 252 | 7 | 27688 | 28130 | 8 | 28158 | 28600 |
| 253 | 1 | 27818 | 28260 | 2 | 28288 | 28730 |

两道工序加工第二组结果如下(代码文件 Ques201.m 详见附录十二)：

**表5.2.2：两道工序调度模型第二组数据结果表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 加工物料序号 | 工序1的CNC编号 | 上料开始时间 | 下料开始时间 | 工序2的CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 310 | 2 | 340 | 875 |
| 2 | 3 | 53 | 398 | 4 | 428 | 993 |
| 3 | 5 | 106 | 486 | 6 | 516 | 1111 |
| …… | | | | | | |
| 207 | 3 | 27273 | 27808 | 6 | 27861 | 28396 |
| 208 | 7 | 27414 | 27949 | 8 | 27979 | 28514 |
| 209 | 5 | 27532 | 28067 | 2 | 28160 | 28695 |

两道工序加工第三组结果如下(代码文件 Ques203.m 详见附录十四)：

**表5.2.3：两道工序调度模型第三组数据结果表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 加工物料序号 | 工序1的CNC编号 | 上料开始时间 | 下料开始时间 | 工序2的CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 4 | 18 | 505 | 3 | 537 | 957 |
| 2 | 8 | 82 | 596 | 7 | 628 | 837 |
| 3 | 5 | 132 | 673 | 1 | 732 | 1059 |
| …… | | | | | | |
| 234 | 5 | 27582 | 28193 | 3 | 28238 | 28474 |
| 235 | 4 | 27679 | 28442 | 3 | 28474 | 28683 |
| 236 | 6 | 27813 | 28308 | 7 | 28358 | 28567 |

结果表明，第1、2组参数均在4:4放置CNC的情况下得到更好的解，第3组参数在5:3放置CNC的情况下得到更好的解，这也验证了之前判断配比的法则是有必要且正确的。

* 两道工序模型的实用性

从结果来看，模型针对3组参数的求解情况均符合实际，既没有出现异常的上下料时间、也没有异常的两个相邻编号的物料上下料开始时间间隔。而这3组数据各有特点，反应了不同工序的不同耗时配比，故3组结果均合理足以证明模型具有较强的实用性，可针对任何符合实际的一组参数进行求解，得到合理的最佳效率方案。分析最优的结果序列，虽然不全为1-8的自然序列，但都会陷入一个循环路径。这与实际的加工过程中，RGV的运动通常存在某种规律、不会做无规则的复杂运动相符合，进一步证明了模型的实用性。

* 两道工序模型系统的作业效率

理想情况下，不考虑RGV移动所消耗的时间，则对于之前已经验证出一、二两组的4:4类型的CNC分布。作业效率如下：



*tti*表示加工一个第*i*道工序的时间，*e*1*i*表示加工第一道工序的时间与上下料时间之和，*e*2*i*表示加工第二道工序的时间与上下料时间之和。



*N*2*i*表示理想情况下，8小时能加工两道工序的第*i*组的成品数。*t*0*i*表示第一次上下料所消耗的时间。得到*N*21=268，*N*22=216



*η*2*i*表示两道工序情况下，利用该模型系统的作业效率。

得到：*η*21=94.40%，*η*22=96.76%。

第三组是适用于5:3刀具类型的CNC分布，如果想向之前那样具体算出理想情况下8小时能生产的成品数，比较繁琐。那么，仍以4:4刀具分布的CNC的理想情况作为比较基准，先得到*N*23=236再求得*η*23=100%。

## 5.3 随机故障模型的建立与求解

### 5.3.1 任务一

对于情况三而言，首先需要给出加工系统的故障概率模型，再将模型分别嵌入情况一、二的模型中，分别考虑其影响。

* 概率故障模型

已知CNC在加工过程中发生故障的概率为1%,即每台CNC在每一次加工时，都有1%的概率的出现故障。故给出第*i*台CNC的故障状态如下：



其中，具体每次加工中，𝐶𝑖状态随机选取，有1%的概率为0，99%的概率为1。当CNC发生故障时，其故障时长可能为10-20分钟。我们不妨假设故障时长服从均匀分布，再针对情况一、二的模型进行嵌入，分别固定带入这3个故障市场对结果进行求解[5]。

* 一道工序的故障模型

Step1：将概率故障模型嵌入一道工序的“八步看一步”的调度模型，从初始放置步骤开始进行调度操作。

Step2：假设在随机的过程中，第*k*个物料在第*i*台CNC上加工时发生了故障；此时需要明确，故障的修复是人工操作，RGV无法预知修复完成的时间。因此，从CNC发生故障的时刻开始，在RGV的视野中第*i*台CNC消失，系统中仅仅存在7台CNC。

Step3：在上述情况下进行一道工序的调度方案，以未来8步决策1步作业，直到CNC修复完成，重新回归到RGV的视野中[6]。设第*w*步的时间区间为[*𝑤*1,*𝑤*2]，此处需要进行考虑，当CNC发生故障时，RGV正处于第*w*步的时间区间中，则RGV需将第*w*步完成，在考虑下一步时忽略故障CNC。

Step4：故障在一次加工过程中发生时，以秒为单位服从均匀分布，即随机在加工时间里的任意一秒开始出现故障；故障发生的时长也服从均匀分布，故随机选取10-20分钟内的任意一个加工时长。因此，对于发生故障的CNC正在加工的物料*k*，记录*k*及*k*的故障开始时间、结束时间。

Step5：故障排除后恢复原先的调度过程，直到下一个随机故障出现，重复上述步骤，一直到28800秒结束，记录一道工序调度模型需要得到的结果。

* 两道工序的故障模型

两道工序的故障模型首先因基于情况二的完整过程。对于情况二，我们已经通过三组数据验证了当不更换刀具时，4:4型为解决一般问题的最佳CNC配比方案，因此我们只对4:4型的调度方案嵌入概率故障模型，并给出其结果。

由于两道工序的故障模型与一道工序的故障模型区别仅仅存在于Step3，故直接延用一道工序的故障模型，并对Step作出改变。

Step1：将概率故障模型嵌入两道工序的调度模型，从初始放置步骤开始进行调度操作。

Step2：假设在随机的过程中，第*k*个物料在第*i*台CNC上加工时发生了故障；此时需要明确，故障的修复是人工操作，RGV无法预知修复完成的时间。因此，从CNC发生故障的时刻开始，在RGV的视野中第*i*台CNC消失，系统中仅仅存在7台CNC。

Step3：两道工序的调度方案是以4块物料完成加工为一组进行分步调度。设对第*k*块物料进行某道工序的加工过程中出现故障。设此时系统处于第*x*组物料的加工过程中，故先将该组的加工流程继续完成，跳过故障CNC对应的步骤；可以得出，由于故障原因，此过程可能出现多余的往返。在进行下一组物料的加工时，由于某一类型的CNC有1个缺失，故可能的选择情况由24×24转变为144。因此，从故障开始时，每一组由144种方案中选取最优的调度方案进行；但需要注意，这一组加工的物料数可能为4，也可能为3，这取决于出现故障的CNC类型。当故障被修复时，同理令系统完成当前一组的加工，从下一组加工开始恢复原先的调度方案。

Step4：故障在一次加工过程中发生时，以秒为单位服从均匀分布，即随机在加工时间里的任意一秒开始出现故障；故障发生的时长也服从均匀分布，故随机选取10-20分钟内的任意一个加工时长。因此，对于发生故障的CNC正在加工的物料*k*，记录*k*及*k*的故障开始时间、结束时间。

Step5：故障排除后恢复原先的调度过程，直到下一个随机故障出现，重复上述步骤，一直到28800秒结束，记录一道工序调度模型需要得到的结果。

Step6：对不同可选CNC配比放置下的初始情况分别重复上述步骤，比较不同配比下故障对系统的影响。

* 模型的求解算法

基于原先的求解算法，更改相关的函数、参数，依照概率合理利用随机生成，对模型步骤依次进行编程求解，最终可以得到模型需要的相关结果。

### 5.3.2任务二

* 故障模型的结果

利用MATLAB求解一道工序随机故障模型(代码文件Ques301.m详见附录十六)。一道工序的情况下，第一组随机故障模型结果如下：

**表5.3.1：一道工序模型第一组随机故障模型结果表**

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 加工CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 588 |
| 2 | 2 | 28 | 641 |
| …… | | | |
| 365 | 1 | 28049 | 28693 |

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 加工CNC编号 | 上料开始时间 | 下料开始时间 |
| 66 | 2 | 5046 | 5647 |
| 213 | 6 | 16048 | 16968 |
| 244 | 6 | 18797 | 19869 |
| 351 | 3 | 27059 | 28190 |

一道工序的情况下，第二组随机故障模型：

**表5.3.2：一道工序模型第一组随机故障模型结果表**

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 加工CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 610 |
| 2 | 2 | 30 | 670 |
| …… | | | |
| 339 | 7 | 28067 | 28677 |

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 故障CNC编号 | 故障开始时间 | 故障结束时间 |
| 78 | 6 | 6443 | 7413 |
| 201 | 2 | 16240 | 16886 |
| 244 | 5 | 20715 | 21522 |
| - | - | - | - |

一道工序的情况下，第三组随机故障模型：

**表5.3.3：一道工序模型第一组随机故障模型结果表**

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 加工CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 588 |
| 2 | 2 | 28 | 641 |
| …… | | | |
| 365 | 1 | 28049 | 28693 |

|  |  |  |  |
| --- | --- | --- | --- |
| 加工物料序号 | 故障CNC编号 | 故障开始时间 | 故障结束时间 |
| 66 | 2 | 5046 | 5647 |
| 213 | 6 | 16048 | 16968 |
| 244 | 6 | 18797 | 19869 |
| 351 | 3 | 27059 | 28190 |

利用MATLAB求解两道工序随机故障模型(代码文件Ques302.m详见附录十七)。两道工序的情况下，第一组随机故障模型结果如下：

**表5.3.4：两道工序模型第一组随机故障模型结果表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 加工物料序号 | 工序1的CNC编号 | 上料开始时间 | 下料开始时间 | 工序2的CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 428 | 2 | 456 | 865 |
| 2 | 3 | 48 | 507 | 4 | 535 | 944 |
| 3 | 5 | 96 | 586 | 6 | 614 | 1023 |
| …… | | | | | | |
| 238 | 5 | 27026 | 27468 | 2 | 27671 | 28077 |
| 239 | 3 | 27130 | 27747 | 4 | 28149 | 28558 |
| 240 | 7 | 27364 | 28238 | 8 | 28266 | 28675 |

**表5.3.5：两道工序模型第一组随机故障模型故障信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| 故障时的物料序号 | 故障CNC  编号 | 故障开  始时间 | 故障结  束时间 |
| 96 | 7 | 10489 | 11243 |
| 199 | 4 | 22782 | 23772 |
| 235 | 6 | 27513 | 28149 |

两道工序的情况下，第二组随机故障模型结果：

**表5.3.6：两道工序模型第二组随机故障模型结果表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 加工物料序号 | 工序1的CNC编号 | 上料开始时间 | 下料开始时间 | 工序2的CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 428 | 2 | 456 | 865 |
| 2 | 3 | 48 | 507 | 4 | 535 | 944 |
| 3 | 5 | 96 | 586 | 6 | 614 | 1023 |
| …… | | | | | | |
| 238 | 5 | 27026 | 27468 | 2 | 27671 | 28077 |
| 239 | 3 | 27130 | 27747 | 4 | 28149 | 28558 |
| 240 | 7 | 27364 | 28238 | 8 | 28266 | 28675 |

**表5.3.7：两道工序模型第二组随机故障模型故障信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| 故障时的物料序号 | 故障CNC  编号 | 故障开  始时间 | 故障结  束时间 |
| 41 | 3 | 5360 | 6192 |
| 119 | 2 | 16831 | 17984 |
| 124 | 6 | 18107 | 18768 |

两道工序的情况下，第三组随机故障模型结果：

**表5.3.8：两道工序模型第二组随机故障模型结果表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 加工物料序号 | 工序1的CNC编号 | 上料开始时间 | 下料开始时间 | 工序2的CNC编号 | 上料开始时间 | 下料开始时间 |
| 1 | 1 | 0 | 310 | 2 | 456 | 865 |
| 2 | 3 | 53 | 398 | 4 | 535 | 944 |
| 3 | 5 | 106 | 486 | 6 | 614 | 1023 |
| …… | | | | | | |
| 198 | 1 | 27235 | 27770 | 2 | 27822 | 28357 |
| 199 | 7 | 27411 | 27946 | 8 | 27981 | 28516 |
| 200 | 5 | 27534 | 28069 | 6 | 28099 | 28634 |

**表5.3.9：两道工序模型第二组随机故障模型故障信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| 故障时的物料序号 | 故障CNC  编号 | 故障开  始时间 | 故障结  束时间 |
| 140 | 8 | 17259 | 18006 |
| 198 | 7 | 24004 | 25088 |

* 故障模型的实用性

关于故障模型，我们需要通过模型的稳定性判断模型是否实用，即判断故障对模型产生的影响程度是否在某一限度内。首先依次选择加工物料总数、故障物料总数为样本。分别对随机生成修复故障时间的样本、设定10分钟修复故障时间的样本、设定15分钟的样本及设定20分钟的样本做平均值与标准差，标准差计算公式表示为：



利用MATLAB分别求解4种情况下20个样本的标准差。

对一道工序，得到的结果如下(代码文件Ques303.m详见附录二十)：

**表5.3.10：随机故障模型对一道工序20次样本分析结果表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | 成品  平均数 | 成品  标准差 | 故障发生  平均次数 | 故障数  标准差 |
| 随机 | 第一组 | 364.6 | 12.20 | 4.20 | 1.881 |
| 第二组 | 330.6 | 15.57 | 3.50 | 1.638 |
| 第三组 | 377.2 | 10.03 | 3.50 | 1.573 |
| 10min | 第一组 | 365.0 | 9.55 | 4.25 | 1.860 |
| 第二组 | 326.0 | 18.68 | 4.00 | 1.806 |
| 第三组 | 376.1 | 13.41 | 4.20 | 2.726 |
| 15min | 第一组 | 366.0 | 11.95 | 3.85 | 1.927 |
| 第二组 | 327.7 | 19.55 | 3.60 | 1.759 |
| 第三组 | 378.0 | 8.92 | 3.60 | 1.536 |
| 20min | 第一组 | 362.1 | 9.82 | 4.15 | 1.755 |
| 第二组 | 326.3 | 14.68 | 3.75 | 1.517 |
| 第三组 | 366.6 | 13.73 | 4.45 | 1.572 |

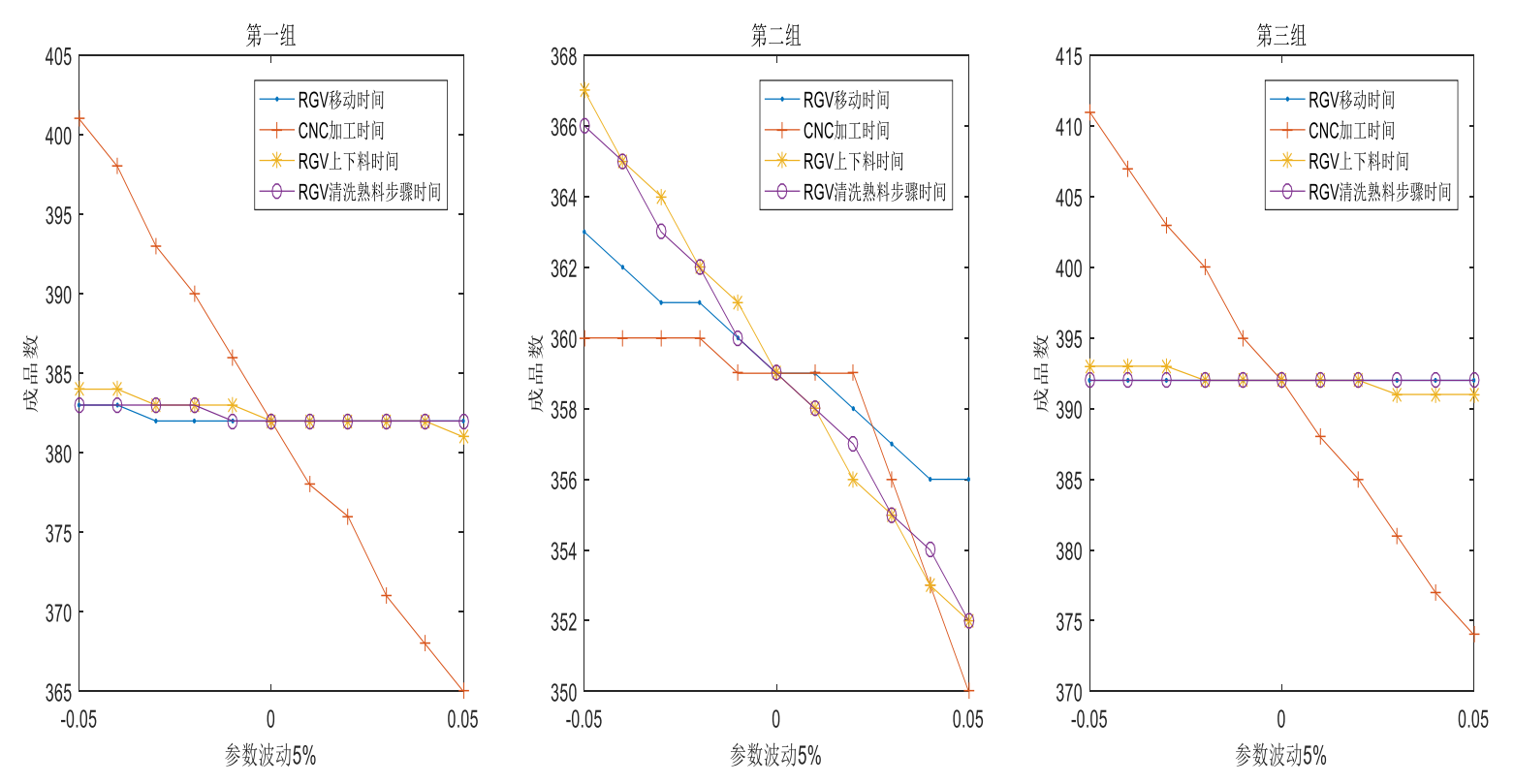
对两道工序情况（针对4:4情况），结果如下(代码文件Ques304.m详见附录二十一)：

**表5.3.11：随机故障模型对两道工序20次样本分析结果表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | 成品  平均数 | 成品  标准差 | 故障发生  平均次数 | 故障数  标准差 |
| 随机 | 第一组 | 240.8 | 12.20 | 2.30 | 1.881 |
| 第二组 | 196.7 | 7.85 | 3.20 | 1.824 |
| 第三组 | 121.9 | 7.67 | 3.15 | 1.755 |
| 10min | 第一组 | 241.3 | 9.55 | 3.05 | 1.860 |
| 第二组 | 202.6 | 3.39 | 3.65 | 1.309 |
| 第三组 | 226.5 | 4.58 | 2.95 | 1.669 |
| 15min | 第一组 | 242.6 | 11.95 | 2.35 | 1.927 |
| 第二组 | 201.2 | 5.87 | 2.25 | 1.446 |
| 第三组 | 222.1 | 7.72 | 2.80 | 1.765 |
| 20min | 第一组 | 233.6 | 9.82 | 2.85 | 1.755 |
| 第二组 | 196.4 | 9.69 | 2.70 | 1.689 |
| 第三组 | 218.0 | 10.13 | 3.00 | 1.589 |

## 5.4 模型的灵敏度分析

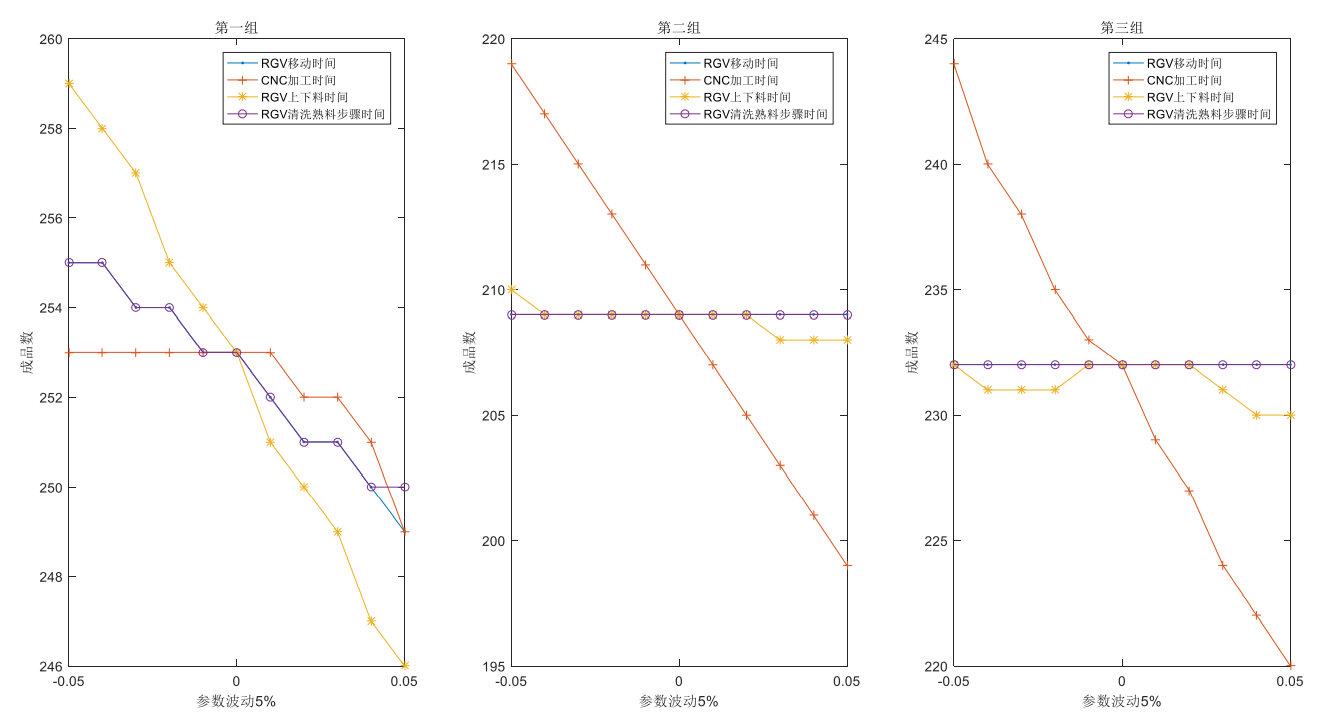
对题目给定的3组数据，令RGV移动时间、CNC加工时间、上下料时间、清洗时间均上下连续波动5%，利用MATLAB绘制4个参数分别与物料成品数量之间的函数关系图。一道工序情况下的模型灵敏度分析图如下(代码文件LingMin1.m详见附录二十四)：



**图5.4.1：一道工序模型的灵敏度分析**

如图所示，对于第一、三组数据而言，两道工序加工的调度模型对CNC加工时间较为敏感，而对第二组数据而言，RGV移动时间、CNC加工时间、上下料时间、清洗时间的变化都会对最终能完成的成品数产生较大影响。因此，可以推断得出，一般情况下，模型对CNC加工时间都较为敏感，在实际加工过程中，需要谨慎考虑加工时间的设置，以保证加工效率。

绘制两道工序情况下的模型灵敏度分析图如下(代码文件LingMin2.m详见附录二十五)：



**图5.4.2：两道工序模型的灵敏度分析**

观察上图结果，对于第二、三组数据而言，一道工序加工的调度模型对CNC加工时间较为敏感，而对第一组数据而言，RGV移动时间、CNC加工时间、上下料时间、清洗时间的变化都会对最终能完成的成品数产生较大影响。因此，可以推断得出，一般情况下，模型对CNC加工时间都较为敏感，在实际加工过程中，需要谨慎考虑加工时间的设置，以保证加工效率。

上述两种情况表明，无论采用几道加工工序，CNC单次加工时间的设定最为关键，而其它三个参数会因加工工序的数量不同而呈现不同的灵敏度，有些情况下敏感，有些情况下稳定。

# 六、模型的评价与推广

## 6.1优点分析

本文在考虑未来步骤对当前步骤的影响时，从多方面进行分析，确定了由未来几步确定当前一步可以时结果无限接近全局最优；既没有进行复杂的全局搜索，也没有进行不确定性较强的随机生成，但却得到了理想的最优结果。

本文在考虑两道工序的加工过程时，针对不同的CNC布局进行了全面的分析与验证，最终给出了不同情形下最佳的布局方式，增强了模型的实用性。

本文突出的特点在于考虑了更换刀具的问题。在实际问题中，更换刀具是常见的操作；针对题目，频繁的更换刀具确实可以使问题得到更优的结果；最终要的一点是，合理的换刀并没有违背题目的任何要求。基于上述三点，更换刀具对于两道工序的加工过程是有意义的，值得推广的。

本文在考虑随机生成故障时，对RGV的指令进行了分析，确定了故障出现时RGV是否应继续当前指令以及故障期间RGV指令发生了怎样的改变。这使得本文对故障的考虑更加实际、更加精细。

本文在检验模型的步骤中，使用方差来判断模型的稳定性，使用理想状态与实际状态的对比情况来判断模型的效率，进一步增强了模型的可靠性。

## 6.2缺点分析

本文在考虑不同配比的CNC时，忽略了2:6、1:7的情况；但是在实际的操作中，当两种CNC的加工时间差超过一定限度时，2:6、1:7的配比将成为最优的放置方案。此处模型直接忽略了少数情况，考虑欠妥当。

本文在模型假设时假设了传送带的总能及时给到生料与熟料空位，但在实际情况中，传送带的运行是有一定的速度的，在加工系统运行时传送带会影响整个调度方案；模型的假设与实际略不相符，故存在缺陷。

## 6.3模型的推广

本文主要利用动态规划对智能加工系统制定了高效的调度方案，其中运用最多的“未来多步确定一步”的思想可以推广到更多的调度模型中；对于问题本身，此模型在RGV位置增多，CNC数量增多的情况下依然使用，可能仅仅需要对“未来多步”进行略微的调整。但是RGV的位置与CNC的数量不能无限的增加下去，当其达到一定值时，RGV会应运动时间过长而效率降低，此时就需要在原先模型的基础上增加RGV数量，得到更高效的调度方案。

# 七、参考文献

[1]查振元,李计星,绳润涛,等.智能平移轨道导引车的应用[J].机器人技术与应用,2017(5):42-43.

[2]顾红,邹平,徐伟华.环行穿梭车优化调度问题的自学习算法[J].系统工程理论与实践,2013,33(12):3223-3230.

[3]陈华.基于分区法的2-RGV调度问题的模型和算法[J].工业工程与管理,2014,19(6):70-77.

[4]金芳,方凯,王京林,基于排队论的AGV调度研究[J].仪器仪表学报,2004,25(s1):844-846.

[5]周广文,贾亚洲.专用计算机数控机床故障率实验研究[J].组合机床与自动化加工技术,2005(1):45-47.

[6]乔非,吴启迪.SJ—FMS中RGV的实时调度与故障调度[J].组合机床与自动化加工技术,1995(03):39-43.