

基于拟关键路径的二车间综合调度算法

谢志强 周含笑 桂忠艳 郑付萍

(哈尔滨理工大学计算机科学与技术学院 哈尔滨 150080)

摘要 针对如何将复杂产品工序有效地分配到具有相同设备资源的二车间加工的问题,提出了基于拟关键路径法的二车间综合调度算法。为了让二车间负载均衡并进行充分的并行处理,尽早结束产品加工,该算法按拟关键路径法(ACPM)对工序排序,再采取二车间加工结束时间接近的预调度策略进行调度。为了减少二车间工序的迁移次数,该算法将入度不小于2的工序放入其紧前工序分配较多的车间;将入度小于2且其紧后工序的入度不小于2的工序分配到能让其尽早结束的车间;对于其他唯一紧前紧后工序与其叶子节点所形成的工序串按预调度策略进行整串调度。实例表明,该算法可以在二次复杂度内较优地实现具有相同设备资源的二车间分布式综合调度。

关键词 拟关键路径法,二车间,分布式,综合调度

中图分类号 TP278 **文献标识码** A

Integrated Scheduling Algorithm of Two Workshops Based on ACPM

XIE Zhi-qiang ZHOU Han-xiao GUI Zhong-yan ZHENG Fu-ping

(School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China)

Abstract Aiming at the problem that how processes of complex products are effectively allocated to two workshops with the same equipment resources, this paper proposed integrated scheduling algorithm of two workshops based on ACPM. In order to let the two workshops load balancing and fully parallel processed, and complete products processing as soon as possible, the algorithm according to the ACPM, takes the pre-scheduled strategy for scheduling for the end of the two workshops in which processed time is close. In order to reduce the number of relocation of two workshops, the algorithm assigns fork point process for in-degree no less than two into the workshop in which predecessor assigned is more; assigns the processes of their in-degree less than 2, at the same time having the successor with the in-degree no less than 2 into the workshop which makes them early finished, to take pre-scheduled strategy for the other processes which have the only predecessor and successor formed by their leaf nodes to finishing a whole string. Examples show that the algorithm has better achieved the two workshop distributed scheduling with the same equipment resources in which in the two time complexity.

Keywords ACPM, Two workshops, Distributed, Integrated scheduling

1 引言

产品制造调度问题是最困难的组合优化问题和典型的NP-Hard问题之一^[1],多年来,一直倍受国内外许多学者的广泛关注。以往研究的产品制造调度方法主要针对纯加工调度^[2-4]或纯装配调度^[5,6],相关的调度方法主要解决大批相同产品面向流水线生产过程的控制^[7]。

随着社会对产品需求的多样化,多品种小批量产品将成为企业的主要生产对象,特别是具有树状结构的复杂单产品,如果采用先加工后装配的生产方式,必然割裂产品内在的加工和装配并行处理关系,影响制造效率。为此,谢志强提出了产品加工和装配一同处理的综合调度^[8],随着综合调度的研究,目前已经取得了一定的成果,已解决了一般综合调度问题^[9,10]、工序间存在紧密衔接约束的特殊综合调度问题^[11]、

存在设备无关延迟约束的综合柔性调度问题^[12]等。

虽然有学者对分布式调度进行了一定的研究^[13-16],但其并非针对综合调度。对于实际生产中经常存在的单件复杂产品在两个具有相同资源车间进行的分布式制造的综合调度问题,目前尚无相关研究成果,因此,有必要进行二车间综合调度算法的研究。由于二车间综合调度比单车间综合调度更加复杂,为了有效地利用二车间资源,提高产品制造效率,需要考虑二车间充分并行处理和车间间工序迁移次数,于是提出了基于拟关键路径的二车间分布式调度算法。

由于关键路径是影响产品加工周期的主要因素,采用拟关键路径法调度不仅复杂度低,而且调度效果好^[17],于是考虑采用拟关键路径法进行工序调度;为了使调度的工序在二车间充分并行处理,需考虑在二车间每调度完成一个工序或一组工序时,能够让二车间的相互等待时间尽量小,即完工时

到稿日期:2012-07-01 返修日期:2012-10-17 本文受国家自然科学基金(60873019),黑龙江省自然科学基金(F201101),哈尔滨市优秀学科带头人(2010RFXXG054),黑龙江省2011年研究生创新科研基金项目(YJSCX2011-035HLJ)资助。

谢志强(1962-),女,博士,教授,CCF高级会员,主要研究方向为智能计算、数据挖掘和网络优化,E-mail: xzq011@tom.com.

间的差值尽量小,可将选取的某一工序或某一组工序进行二车间的预调度,于是设计了二车间充分并行处理的基于拟关键路径的二车间预调度策略;又由于算法主体是按照拟关键路径选取工序,必然会出现成串工序的调度,为了减少工序迁移次数,设计了以串为单位进行二车间分配预调度的成串处理策略和入度不小于2的叉点工序放入其紧前工序分配较多的车间方案;考虑产品加工尽早结束,分配入度小于2且其紧后工序的入度不小于2的工序到能让其尽早结束的车间。综合以上因素设计了可兼顾二车间工序充分并行处理和减少工序迁移次数的工序调整策略。最后通过实验验证。

2 问题描述

产品的加工工艺图呈树状结构,一般称为产品加工树。树中的节点代表工序,边代表工艺约束的偏序关系,叶节点为初始可加工工序,根节点为最后加工的工序,当根节点加工完毕,表明该产品加工完毕。

由于一般单一车间综合调度是将加工和装配工序统一为加工工序,加工设备和装配设备统一为加工设备并一同调度,因此一般单一车间综合调度的约束条件为:

1. 一台设备在某一时刻只能加工一道工序;
2. 一道工序在某一时刻只能被一台设备加工;
3. 一台设备一旦加工某道工序,则直到该工序加工完毕后,这台设备才能加工其它工序;
4. 每道工序都必须在其紧前工序加工完后,方可开始加工;
5. 每道工序的加工时间已知,且与加工顺序无关;
6. 允许工序之间等待,允许设备在工序达到之前闲置。

由于产品最后工序的开始加工时间必需等其前面的工序加工完毕,产品全部加工完毕的时间为各设备完工的最大时间值,因此希望产品尽早完工则需要工序尽早开始加工,于是所求问题的数学描述为:

$$T = \min(\max(x_{ij} + y_{ij})) \quad (1)$$

$$\text{s. t. } \min(x_{ij}) \quad (2)$$

$$x_{xy} \geq x_{ij} + y_{ij} \quad (3)$$

式中, x_{ij} 是 x_{xy} 的紧前工序, m 为设备数, $i=1,2,\dots,m,j=1,2,\dots,n,x_{ij}$ 为设备 i 的第 j 个工序的开始加工时间, y_{ij} 为设备 i 的第 j 个工序的连续加工时间。式(1)表示产品加工的时间目标;式(2)表示每个工序尽早开始加工;式(3)表示工艺紧后工序需要在其紧前工序加工完毕后才可开始加工。

当将产品工序分配到二车间调度时(设这两个车间为车间 p 与车间 q),从二车间整体上看,综合调度的数学模型是不变的,主要不同是紧前工序可能与紧后工序不在同一车间加工,需要设计相应的调度策略,以缩短生产周期,提高产品制造效率,减少产品工序间的迁移次数。

3 问题分析与策略设计

3.1 问题分析与目标

二车间调度是将产品加工工艺树上的工序分配到两个拥有相同资源的车间上进行处理,与一般的单一车间的综合调度问题有很大区别,需要特别考虑的有:①产品的拆分;②工序迁移。

需要特别考虑①产品的拆分,是因为产品在两个车间加工的工序不一致,车间之间必然会产生加工结束的时间差,如果这个值较大,就说明二车间之间工序的并行性比较差且加工负载不平衡,导致设备利用率低,加工效果差。

需要特别考虑②工序迁移,是因为产品工艺的紧后工序可能与其紧前工序不在同一车间加工,于是不可避免地产生工序在二车间迁移。这个次数越多,二车间之间工序迁移的时间开销会越大,影响产品完工时间。

于是有必要设计出一种综合调度算法,针对①产品的拆分,达到减少车间之间相互等待时间的目标;针对②工序迁移,达到减少工序在二车间迁移次数的目标。

由于按文献[10]中的层优先、短用时和长路径策略的调度算法可以较好地考虑纵横调度优化,因此首先设计了基于层优先策略的二车间综合调度算法;按照层优先、短用时、长路径策略将所有工序排序;再将工序依次放入车间 p 与车间 q 进行预调度,比较二车间的用时差,判断车间负载平衡。

由于基于层优先策略的二车间综合调度算法只采用预调度平衡负载的策略,不易控制工序迁移次数,因此本文又设计了改进算法。

由于关键路径是影响产品加工周期的主要因素,采用拟关键路径法调度不仅复杂度低,而且调度效果好,因此本文考虑采用拟关键路径法进行工序调度。

为了使调度的工序在二车间充分并行处理,即达到目标一,需考虑二车间每调度完成一个工序或一组工序时,二车间相互等待的时间尽量小,即完工时间的差值尽量小。由于将选取的某一工序或某一组工序先进行二车间的预调度再分配车间可减少二车间完工时间的差值,因此设计了二车间充分并行处理的基于拟关键路径的二车间预调度策略,而且进行过预调度的工序,其完成时间可为后文提到的工序的调整策略起到辅助作用;又由于算法主体是按照拟关键路径选取工序,必然会出现成串工序的调度,为了减少工序迁移次数,即达到目标二,成串工序以整体为单位进行二车间分配预调度处理,于是设计了减少工序迁移次数的成串处理策略和入度不小于2的叉点工序的分配。因此,所设计的方法要兼顾两个目标。下面进行本文策略的具体设计与分析。

3.2 拟关键路径调度策略

复杂产品的加工工艺树都有自己的一条关键路径,无论是二车间分布式的综合调度还是传统单一车间的综合,关键路径上的工序对产品完工时间都有较大的影响。由于分段关键路径处理的拟关键路径法调度不仅复杂度低,而且调度效果好,本文采用文献[17]的拟关键路径法进行工序调度排序。具体方式为:优先调度最长路径上的工序,并确定所选取的工序分配到哪个车间。

3.3 预调度策略

按拟关键路径法调度的工序可能在不同的车间上加工,为了让二车间之间相互等待的时间尽量小,可将调度的工序分别在二车间进行预调度,选取二车间时差小即负载尽量均衡的工序分配方案,即预调度策略。

具体方式为:按拟关键路径法选取出某一工序或某一组工序,将这些工序依次放入车间 p 与车间 q ,得到二个车间目前的最后完成时间,这时比较两个车间目前的最后完成时间

的差值,为了比较该差值,需要记录二车间动态完成时间 T_p 与 T_q ,初始时 $T_p=0, T_q=0$;还有预调度后二车间的完工时间 T_p' 与 T_q' ,选择时差小的车间分配方案。

该策略的好处是,让两车间负载近于平衡,可以达到减少车间之间相互等待时间的目标。

3.4 拟关键路径上成串工序的处理策略

对于入度小于2且其紧后工序入度不小于2工序以外的唯一紧前紧后工序与其叶子节点所形成的工序串按预调度策略进行整串处理。

具体方式为,在按照拟关键路径调度时,先将这些串放入 p 车间与 q 车间进行预调度,比较二车间的当前完成时间,最终确定工序串的加工车间方案是:二车间的用时差尽量小;若用时差相同,则考虑让该串尽早结束;若该串在二车间的结束时间相同,则考虑让该串分配时产生的迁移更少;若迁移数目相同,则该串可在任意车间调度。

例如如图1所示的产品加工树,图中的方框表示工序,方框中的信息为:工序号|设备号|加工时间。根据工序信息可知,不能进行成串调度的工序有入度小于2且其紧后工序入度不小于2的A4、A5、A6和入度不小于2的A3,其余的工序可按照其路径形成整体加工的工序串: $\{A11, A10, A8\}$ 、 $\{A7, A9\}$ 和 $\{A1, A2\}$ 。

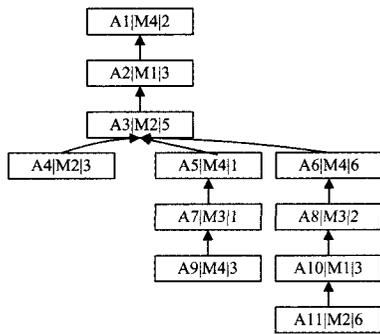


图1 产品加工树

该策略的好处是,按照拟关键路径算法形成的工序串在同一车间加工,减少了迁移次数。

3.5 特殊工序的调整策略

复杂产品工艺树上的工序种类繁多,如果不进行分类处理,可能会导致车间资源分配不均衡或工序迁移次数过大,为此提出了对特殊工序的调度进行适当调整的策略。

1. 对于入度小于2且其紧后工序为入度不小于2的工序,如图1所示的A4、A5、A6,只有当此类工序尽早结束,才能让其后续工序尽早开始加工。因此对于这类工序的车间分配,应首先考虑其能尽早完工,再次考虑二车间负载的平衡与较少的迁移次数。

具体调整策略为:预先将该类工序分别放入二车间调度一次,然后比较该工序在二车间的完工时间,将该工序分配到结束时间较早的车间;若该工序在二车间的完工时间相同,则应继续进行负载平衡的判断,选取二车间用时差较小的分配方式;若时间差相同,应继续判断哪个车间分配了该工序的紧前工序,将其分配到紧前工序所在的车间;若该工序不存在紧前工序,放入任意车间调度皆可。

2. 对于入度不小于2的工序,如图1中的A3,此类工序如果分配不好,会引起较多工序迁移。于是分配该类工序时,

应首先考虑其迁移次数尽量小,再次考虑二车间负载的平衡与该工序较早的完工时间。

具体调整策略为:首先判断该工序的紧前工序在哪个车间分配较多,将该工序分配到较多的车间;若紧前工序在二车间分配的数目相同,则将该工序进行预调度,判断负载的平衡,将其分配到时差较小的车间;若时间差相同,则将其分配到结束时间较早的车间。

4 算法详细设计

根据以上分析,为了实现单件复杂产品二车间分布式综合调度的目标,设计算法的具体步骤如下,相应的流程图如图2所示。

Step1 输入加工产品工序信息。

Step2 深度遍历工艺树,求出所有路径长度。

Step3 按照拟关键算法进行工序排序,形成备选工序集(初始元素个数为 n)。

Step4 记录车间 p 与车间 q 目前的总体完工时间(初始时, $T_p=0, T_q=0$);设工序 i 在 p 车间的完工时间为 E_{pi} ,在 q 车间的完工时间为 E_{qi} ,设每个或每串工序分配车间后产生的迁移次数为 S_i ,总迁移次数 S 的初始值为0。

Step5 从备选工序集中按序依次选取工序。

Step6 判断该工序的入度是否大于等于2。若否,转Step8;若是,计算该工序的紧前工序在二车间的个数,将该工序分配到个数较多的车间,转Step11;

若二车间个数一样多,则判断车间负载平衡:若 $|T_p - T_q| < |T_q - T_p'|$,则分配到车间 q 调度,转Step11;若 $|T_p - T_q| > |T_q - T_p'|$,则分配到车间 p 调度,转Step11;若 $|T_p - T_q| = |T_q - T_p'|$,则比较 E_{pi} 与 E_{qi} ,若 $E_{pi} < E_{qi}$,则分配到车间 p 调度,转Step11;若 $E_{pi} > E_{qi}$,则分配到车间 q 调度,转Step11。

Step7 判断该工序的紧后工序是否为入度大于等于2。若否,则转Step8。若是,则比较 E_{pi} 与 E_{qi} ,若 $E_{pi} < E_{qi}$,分配到车间 p 调度,转Step11,若 $E_{pi} > E_{qi}$,分配到车间 q 调度,转Step11;若 $E_{pi} = E_{qi}$,则判断车间负载平衡:若 $|T_p - T_q| < |T_q - T_p'|$,则分配到车间 q 调度,转Step11,若 $|T_p - T_q| > |T_q - T_p'|$,则分配到车间 p 调度,转Step11;若 $|T_p - T_q| = |T_q - T_p'|$,则将该工序放入紧前工序所在的车间,转Step11;若无紧前工序,可分配到任意车间调度,转Step11。

Step8 形成可成串处理的工序集合(集合中初始元素个数为0)。

Step9 判断工序串在二车间的负载平衡:若 $|T_p - T_q| < |T_q - T_p'|$,则分配到车间 q 调度,转Step10;若 $|T_p - T_q| > |T_q - T_p'|$,则分配到车间 p 调度,转Step10;若 $|T_p - T_q| = |T_q - T_p'|$,则比较集合中最后工序的结束时间 E_{pi} 与 E_{qi} ,若 $E_{pi} < E_{qi}$,则分配到车间 p 调度,转Step10;若 $E_{pi} > E_{qi}$,则分配到车间 q 调度,转Step10;若 $E_{pi} = E_{qi}$,则分配到其首个工序的紧前工序所在的车间,转Step10;若无紧前工序,可分配到任意车间调度,转Step10。

Step10 将分配完的工序串所在的集合清空。

Step11 从备选工序集中删除已经分配调度好的工序,并记录迁移次数 $S=S+S_i$ 。

Step12 判断备选工序集合是否为空,若不为空,转到

Step5.

Step13 二车间分别输出甘特图。

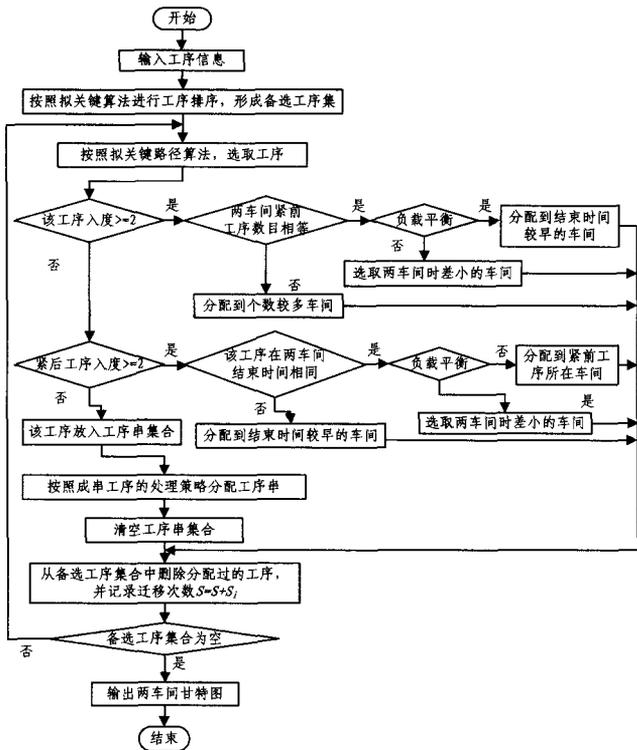


图2 算法流程

5 复杂度分析

1. 拟关键路径调度策略: 根据文献[17], 可得到通过拟关键路径调度算法对工序进行排序的复杂度为 $O(n^2)$ 。

2. 预调度工序: 最坏情况下每个工序在两个车间都要调度1次, 即 n 个工序进行2遍拟关键路径调度, 所以预调度工序的复杂度为 $O(n)$ 。

3. 选取工序串或判断特殊工序: 选取工序串或判断特殊工序是根据工序在工艺树中的属性来判断, 每个工序判断1次, 所以其复杂度为 $O(n)$ 。

4. 记录迁移次数: 在确认每个工序加工的车间后, 该工序涉及的迁移次数最多为其紧前工序个数。由于全部工序的紧前工序个数为 $n-1$, 因此记录迁移次数的复杂度为 $O(n)$ 。

综上所述, 基于拟关键路径的二车间综合调度算法复杂度不超过2次项。

6 实例分析与对比

6.1 调度实例

接下来用本文所设计的算法对图3所示的复杂产品加工树实例进行二车间的分配与调度。

按照成串工序处理策略, 调度的工序串有: $\{A_{20}, A_{16}\}$ 、 $\{A_{15}, A_{10}\}$ 、 $\{A_{21}, A_{18}, A_{13}\}$ 和 $\{A_{14}, A_9, A_5\}$, 采取平衡负载的策略调度; 对于入度小于2且其紧后工序入度不小于2的工序: $A_1, A_2, A_6, A_8, A_{11}, A_{12}, A_{21}$ 和 A_{22} , 采取结束时间较早的特殊调整策略调度; 对于入度不小于2的工序: A_9, A_7, A_4 和 A_0 , 采取分配到紧前工序所在车间个数较多的特殊调整策略。当调度完成后, 输出二车间的甘特图, p 车间甘特图如图4所示, q 车间甘特图如图5所示。

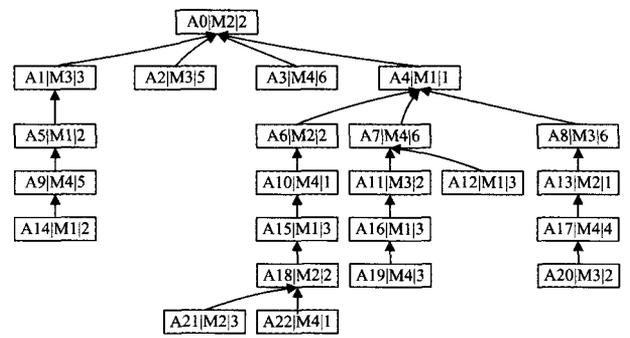


图3 产品加工树

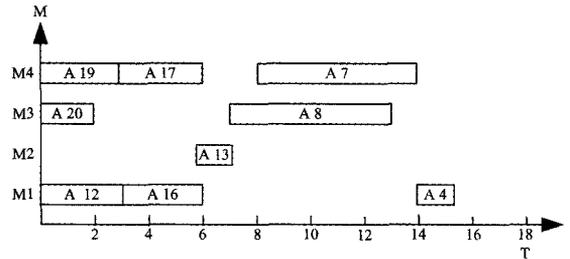


图4 p 车间甘特图

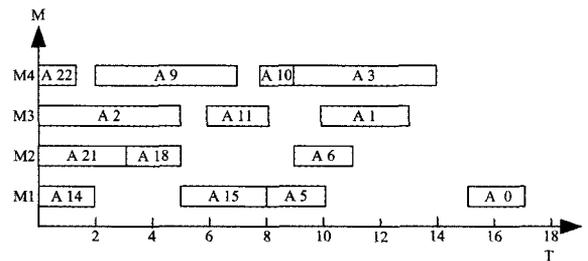


图5 q 车间甘特图

由图4和图5可知: 产品完工时间为17工时, 二车间用时差为2工时。

由算法调度过程可知: 加工 A_{11}, A_7, A_4 和 A_0 时, 各产生1次迁移, 即工序之间的迁移次数 $S=4$ 。

6.2 对比算法的分析

为了充分说明本文算法较优, 下面先介绍基于层优先策略的二车间综合调度算法。

基于层优先策略的二车间综合调度算法的思想是: 首先按照层优先、短用时、长路径策略将所有工序排序; 再将工序依次放入车间 p 与车间 q 进行预调度, 比较二车间的用时差, 进行车间负载均衡的判断, 若用时差不同, 则放入差较小的车间, 若用时差相同, 则放入紧前工序所在较多的车间。若紧前工序在二车间的数目相同, 则放入能让其较早结束的车间。若结束时间相同, 则放入任意车间调度。

算法具体步骤如下, 流程图如图6所示。

Step1 输入加工产品工序信息。

Step2 深度遍历工艺树, 求出所有路径长度。

Step3 按照层优先、短用时、长路径策略将所有工序排序。

Step4 形成备选工序集合(初始元素个数为 n)。

Step5 记录车间 p 与车间 q 目前的总体完工时间(初始时, $T_p=0, T_q=0$); 设工序 i 在 p 车间的完工时间为 E_{pi} , 在 q 车间的完工时间为 E_{qi} , 设每个工序分配车间后产生的迁移次数为 S_i , 总迁移次数 S 的初始值为0。

Step6 从备选工序集中按序依次选取工序。

Step7 将该工序依次放入车间 p 与车间 q 进行预调度,记录此时车间的完工时间 T_p' 与 T_q' ,判断车间负载平衡:若 $|T_p - T_q| < |T_q' - T_p'|$,则分配到车间 q 调度,转 Step8;若 $|T_p - T_q| > |T_q' - T_p'|$,则分配到车间 p 调度,转 Step8;若 $|T_p - T_q| = |T_q' - T_p'|$,则放入紧前工序所在较多的车间,转 Step8;若该工序的紧前工序在二车间的数目相同,则比较 E_{p_i} 与 E_{q_i} ,若 $E_{p_i} < E_{q_i}$,则分配到车间 p 调度,转 Step8;若 $E_{p_i} > E_{q_i}$,则分配到车间 q 调度,转 Step8;若 $E_{p_i} = E_{q_i}$,放入任意车间调度,转 Step8。

Step8 从备选工序集中删除已经分配调度好的工序,并记录迁移次数 $S = S + S_i$ 。

Step9 判断备选工序集合是否为空,若备选工序集合不为空,转到 Step5。

Step10 二车间分别输出甘特图。

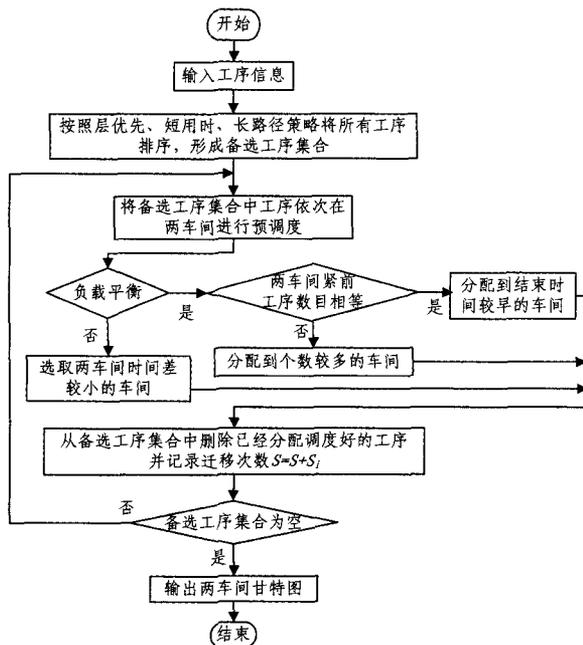


图6 对比算法流程

用上述描述的算法对本文的实例进行调度,分别输出二车间的甘特图, p 车间甘特图如图 7 所示, q 车间甘特图如图 8 所示。

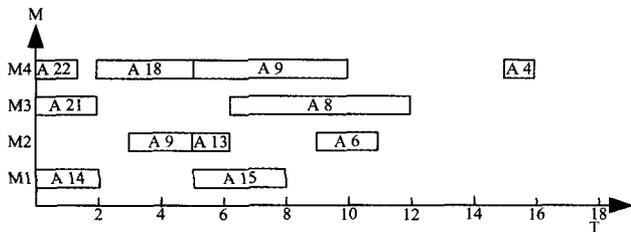


图7 p 车间甘特图

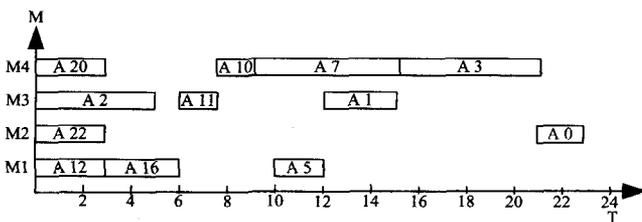


图8 q 车间甘特图

由图 7 和图 8 可知:完工时间为 23 工时,二车间用时差为 7 工时。

由算法调度过程可知:加工 A9、A10、A6、A5、A4 和 A0 时,各产生 1 次迁移,即工序之间的迁移次数 $S=6$ 。

通过两种算法对比可知,基于拟关键路径的算法比基于层优先的算法在用时上缩短了 5 个工时,迁移次数减少了 2 次,而且二车间的加工时间差也缩短了 5 个工时。

之所以迁移次数更少,是因为采用了成串工序的处理策略和针对入度不小于 2 的工序的调整策略有效地控制了工序间的迁移次数。

之所以基于拟关键路径的算法完工时间更少,是因为针对入度小于 2 且其紧后工序入度不小于 2 的工序采取了结束时间较早的特殊调整策略。例如,图 5 中的 A1 比图 8 中的 A1 完成时间早,而 A1 的提前完工使得 A3、A4 和 A0 提前完成。

之所以基于拟关键路径的算法设备使用更加均衡,是因为基于拟关键路径的算法可同时考虑成串的多个工序的车间分配和工序尽早完成,使得产品在二车间加工时间差较小。

所以,本文设计的基于拟关键路径的二车间综合调度算法在复杂产品的二车间综合调度的问题上有较好的效果。

结束语 针对二车间分布式综合调度问题,本文首次实现了基于拟关键路径法的二车间综合调度算法,主要结论如下:

1. 二车间加工制造更加充分并行和均衡;
2. 产品总完成时间更短;
3. 工序的迁移次数更少;
4. 算法复杂度较低,只有 2 次多项式。

由于所提算法在考虑完成时间和工序的迁移次数的同时有较低的复杂度,因此该研究有一定的理论和实际意义。

参考文献

- [1] 曾立平,黄文奇.求解 JobShop 调度问题的一种新的邻域搜索算法[J].计算机研究与发展,2005,32(5):48-53
- [2] 张维存,郑丕涛,吴晓丹.蚁群遗传算法求解能力约束的柔性作业车间调度问题[J].计算机集成制造系统,2007,13(2):127-131,156
- [3] 何彦,刘飞,曹华军,等.面向绿色制造的机械加工系统任务优化调度模型[J].机械工程学报,2007,43(4):27-31
- [4] Yamamotoa H, Sunb J, Matsuc M. A study on limited-cycle scheduling problem with multiple periods[J]. Computers & Industrial Engineering, 2010, 59: 675-681
- [5] 羌磊,肖田元.应用扩展贝叶斯进化算法求解混流装配调度问题[J].计算机集成制造系统,2007,13(2):111-116
- [6] 李原,张开富,王挺,等.基于遗传算法的飞机装配序列规划优化方法[J].计算机集成制造系统,2006,12(2):30-33
- [7] Wu S, David, Richard A W. An Application of Discrete-event Simulation to On-line Control and Scheduling in Flexible Manufacturing [J]. Prod. Res, 1989, 27(9):1603-1623
- [8] 谢志强.工件间有约束的复杂产品工序调度研究[D].哈尔滨:哈尔滨理工大学,2009
- [9] 谢志强,张磊,杨静.基于调度长路径的复杂产品综合调度算法[J].计算机科学,2010,37(2):156-159
- [10] 谢志强,杨静,周勇,等.基于工序集的动态关键路径多产品制造调度算法[J].计算机学报,2011,34(2):216-222
- [11] 谢志强,滕宇峥,杨静.紧密衔接工序组联动的综合调度算法

[12] 谢志强, 邵侠, 杨静. 存在设备无关延迟约束的综合柔性调度算法[J]. 机械工程学报, 2011, 47(4): 181-189
 [13] 刘世平, 张洁, 饶运清, 等. 分布式车间管理控制系统研究[J]. 中国机械工程, 2001, 12(12): 1432-435
 [14] 包振强, 李长仪, 周鑫. 分布式混合优化调度方法研究[J]. 中国机械工程, 2006, 17(18): 1908-1912
 [15] Chung S H, Chan Felix T S, Chan H K. A modified genetic algo-

rithm approach for scheduling of perfect maintenance in distributed production scheduling[J]. Engineering Applications of Artificial Intelligence, 2009, 22(7): 1005-1014

[16] Oike S, Tanaka T. Robust production scheduling using autonomous distributed systems[J]. Key Engineering Materials, 2012, 516: 166-169
 [17] 谢志强, 刘胜辉, 乔佩利. 基于 ACPM 和 BFSM 的动态 Job-Shop 调度算法[J]. 计算机研究与发展, 2003, 40(7): 79-85

(上接第 171 页)

图 4 展示了基于反馈的相似时序查询结果, 用户标识第 1 个、第 2 个、第 4 个、第 5 个、第 8 个结果相似, 可见, 通过融合正、负反馈信息后的查询结果, 提高了用户的满意度。另外, 反馈后的检索结果中即使被用户标识为不相似的时序, 也与初始查询结果中被用户标识为相似的时序匹配程度趋近一致, 同时还与初始阶段被标识为不相似的主题时序在起始点有了明显区别。

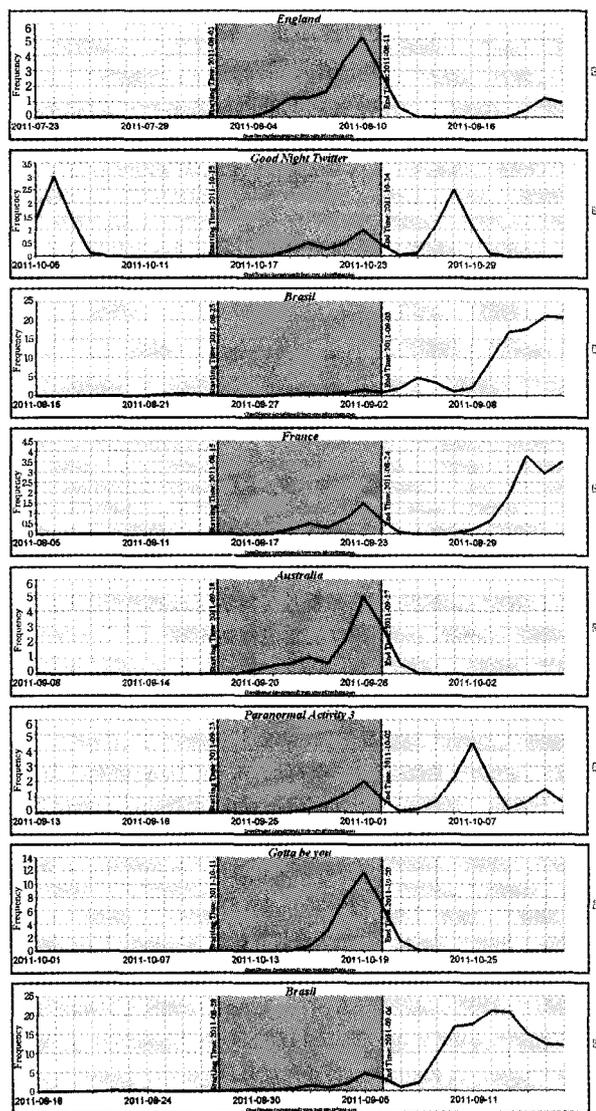


图 4 基于用户反馈的主题“London”相似时序结果展示(用户感兴趣时序以对号标示)

用户综合分析图 4 所示的时序相似主题“England”、“France”等在时序相似时间段后的发展变化情况, 发现其具有“先快速下降继而平缓发展再上升”的发展趋势。基于该分

析结果, 用户进而可对查询主题“London”在“2011-09-27”之后的发展趋势进行预测, 便于用户对其感兴趣的趋势(例如平缓发展)进行监控。

结束语 微博已经成为最受用户欢迎的了解网络热点主题发展状况的社会媒体平台之一, 如何为微博用户提供相似主题时序查询也相应成为目前的研究热点。为了更好地帮助用户获取满足自身喜好的相似主题时序, 本文提出了一种综合考虑正、负反馈信息的微博相似主题时序查询方法。实验结果表明, 此方法能从用户标识为与查询时序是否相似的时序中挖掘出用户的兴趣, 从而实现了体现用户偏好的时序相似性的计算, 更好地为用户提供了满足其偏好的相似主题时序查询结果。未来将进一步考虑主题的内容信息, 结合主题用户关注度时序为用户提供更全面的信息服务。

参考文献

[1] He Y, Su W, Tian Y, et al. Summarizing microblogs on network hot topics[C]//ITAP: the 2011 International Conference on Internet Technology and Applications, 2011: 1-4
 [2] Yang J, Leskovec J. Patterns of temporal variation in online media[C]//WSDM'11: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, 2011: 177-186
 [3] Song S, Li Q, Bao H. Detecting dynamic association among twitter topics[C]//WWW 2012: Proceedings of the 2012 ACM Conference on the World Wide Web, 2012: 605-606
 [4] Keogh E J, Pazzani M J. Relevance feedback retrieval of time series data[C]//SIGIR 1999: the 22nd Annual ACM Conference on Special Interest Group on Information Retrieval, 1999: 183-190
 [5] 郑斌祥, 席裕庚, 杜秀华. 利用反馈的时序模式挖掘算法研究[J]. 控制与决策, 2002, 17(5): 527-531
 [6] 秦吉胜, 王淑静, 宋瀚涛. 基于小波变换和反馈的时间序列相似模式搜索算法[J]. 北京理工大学学报, 2004, 24(12): 1069-1073
 [7] Pawling A, Madey G. Feature Clustering for Data Steering in Dynamic Data Driven Application Systems[C]//ICCS 2009, Part II, Lecture Notes in Computer Science, Volume 5545, 2009: 460-469
 [8] Meij E, Weerkamp W, Rijke M D. Adding Semantics to Microblog Posts[C]//WSDM'12: Proceedings of the fourth ACM international conference on Web search and data mining, 2012: 563-572
 [9] Griery C, Thomas K, Paxson V, et al. @spam: The Underground on 140 Characters or Less[C]//CCS'10: Proceedings of the 17th ACM Conference on Computer and Communications Security, 2010: 27-37