

存在设备有关延迟约束的综合调度算法^①

谢志强^{②*} 邵 侠^{**} 杨 静^{*}

(* 哈尔滨工程大学计算机科学与技术学院 哈尔滨 150001)

(** 哈尔滨理工大学计算机科学与技术学院 哈尔滨 150080)

摘 要 针对目前综合调度中没有考虑设备有关延迟约束(DDC)影响调度效果的问题,提出了存在设备有关延迟约束的综合调度的算法。该算法在标准工序、设备有关延迟工序(DDP)和包含设备有关延迟约束的加工工艺树等概念的基础上,将等待延迟时间转化为设备有关延迟工序,使存在 DDC 的调度问题转变为较易解决的一般综合调度问题,再用拟关键路径法(ACPM)确定工序的调度次序,最后用前沿贪心规则确定工序的开始时间。为了使设备有关延迟工序影响的工序尽早开始,进一步提出了设备有关延迟工序配合调整的策略。实验表明,该调度算法能够有效解决存在设备有关延迟约束的综合调度问题,可在不提高算法复杂度的前提下,提高调度结果的精度并减少产品总的加工时间。

关键词 延迟约束,设备有关延迟约束(DDC),综合调度,拟关键路径法(ACPM),算法

0 引 言

车间调度问题是 NP-C 问题^[1],对这个问题的研究虽然已取得一定成果^[2,4],但所用算法解决的是无约束工件的纯加工或有约束工件的纯装配的调度问题,而对多品种小批量产品,特别是具有树状结构的复杂单产品的生产调度问题不适用,因为分别进行加工和装配必然割裂产品内在加工与装配间的并行关系,其调度结果不如加工和装配一同处理的综合调度^[5]。于是有关加工和装配一同处理的综合调度的算法被提出,如文献[6]提出的拟关键路径法(ACPM)、文献[7]提出的动态关键路径法和文献[8]提出的紧密衔接工序组联动的调度算法。对于工序间存在延迟约束的问题,有关学者提出了针对车间调度问题的带有传输时间^[9]和带有非负时间滞差^[10]的调度算法,以及针对综合调度问题的设备无关延迟约束的调度算法^[11,12],但目前还没有解决存在设备有关延迟约束(device-dependent deferred constraint, DDC)的调度问题的方法,而实际综合调度中确实存在与设备有关的延迟约束的情况,例如在能调整刀具和夹具的数控机床或加工中心上

进行产品加工时,两个工序因加工类型不同需要等待调整刀具或调整夹具(设备调整)的一段时间为设备有关的延迟约束。以往的调度算法是将延迟约束形成的等待时间分配到前后工序的加工时间内^[13],限制了等待时间的灵活移动,导致调度结果不精确。为此,本文对存在设备有关的延迟约束的综合调度问题进行了研究,并提出了调度算法。

1 问题分析与解决方案设计

通常情况下,综合调度问题一般用文献[6]的数学模型来描述。考虑存在设备有关延迟约束的综合调度问题时,应注意到工序间不只存在顺序约束,还存在设备有关延迟约束,即需要等待设备有关延迟约束形成的时间段。

为了解决存在设备有关延迟约束的综合调度问题,本文将设备有关延迟约束时间虚拟为可在一定范围内前后移动的虚拟工序,即将该问题简化为只存在顺序约束的一般综合调度问题。

1.1 工序分类

为了便于问题的描述和分析,将加工工序分为标准工序和设备有关延迟工序(device-dependent

① 国家自然科学基金(60873019,61073043),黑龙江省自然科学基金(F200901,F201101),中国博士后科学基金(20090460880)和哈尔滨市优秀学科带头人(2010RFXXG054,2011RFXXG015)资助项目。

② 男,1962 年生,博士,教授;研究方向:机械制造及其自动化,数据库与知识工程;联系人,E-mail: xzq0111@tom.com
(收稿日期:2010-10-26)

deferred procedure, DDP), 相关定义如下。

定义 1 标准工序 产品的实际加工工序称为标准工序。

定义 2 设备有关延迟约束 设 O_i, O_j 是同一设备 M_k 上 2 个相邻加工的标准工序, 当加工完 O_i , 需调整 M_k 的加工工具才能加工 O_j 所形成的延迟约束为设备有关延迟约束。

定义 3 设备有关延迟工序 将设备有关延迟约束形成的等待时间虚拟成的一道工序, 称为设备有关延迟工序。

1.2 存在设备有关延迟约束的加工工艺树模型的设计

通过对一般加工工艺树的研究与分析, 提出了一般加工工艺树中加入设备有关延迟工序: 存在设备有关延迟约束的加工工艺树。

定义 4 存在设备有关延迟约束的加工工艺树 在加工和装配综合调度中, 根据标准工序和设备有关延迟工序间的工艺约束关系形成的树状结构称为存在设备有关延迟约束的加工工艺树。

存在设备有关延迟约束的加工工艺树中, 每个标准工序结点由工序名、所需设备名、工序加工时间和加工类型组成; 每个设备有关延迟工序结点由工序名、设备名和加工时间组成。

对于有 L 个工件, 每个工件的工序数为 J_i , ($i = 1, 2, \dots, L$), 总工序数 $n = \sum_{i=1}^L J_i$ 在 m 个设备上加工的产品, 存在设备有关延迟约束的综合调度问题的数学模型为 $T_{\text{total}} = \min(\max(S_{iz} + t_{iz}))$, 其中 S_{iz} 表示工序 x 在设备 M_z 上的开始加工时间, E_{jz} 表示工序 j 在设备 M_z 上的结束时间, D_{zy} 表示标准工序 x 和 j 间需要等待延迟时间, t_{iz} 表示工序 i 在设备 M_z 上的加工时间。

当标准工序 i 和 j 间存在关于设备 M_z 的延迟时间 D_{zy} 时, $S_{iz} \geq E_{jz} + D_{zy}$, $i, j = 1, 2, \dots, n$; $z = 1, 2, \dots, m$; 即标准工序 i 和 j 存在设备有关延迟工序, 其在标准工序 i 和 j 间加工。

2 等待延迟时间转化为设备有关延迟工序的策略

针对加工类型不同的标准工序在加工树中位置的 4 种情况, 提出延迟时间转换为设备有关延迟工序的策略。

(1) 情况 1。若标准工序 O_i, O_k 在同一设备 M_i 上加工, 加工时间分别为 $(t_i + t_i^1), (t_k + t_k^1)$, 其中

O_k 是 O_i 的前续工序, 二者之间的关系如图 1(左)所示。因加工类型不同, O_k 加工结束后需调整设备 M_i 上的加工工具, 才能加工 O_i , 即加工完 O_k 之后在加工 O_i 之前存在等待延迟时间 $D_{ik} = t_i^1 + t_k^1$, 将等待延迟时间 D_{ik} 从 O_k, O_i 的加工时间中分离出来, 虚拟为一道设备有关延迟工序 O_{ik} , 加工设备是设备 M_i , 转换后三者之间的关系如图 1(右)所示, 则 O_k, O_{ik}, O_i 的加工次序依次是 O_k, O_{ik}, O_i 。

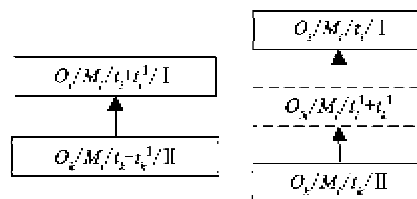


图 1 情况 1 示意图

(2) 情况 2。若标准工序 O_k, O_j 和 O_i , 其中 O_k 和 O_i 在同一设备 M_i 上加工, 加工时间分别为 $(t_i + t_i^1), (t_k + t_k^1)$, O_j 在另一设备 M_j 上加工, 三者之间的关系如图 2(左)所示。因 O_k 和 O_i 的加工类型不同, O_k 加工完后, 需要调整设备 M_i 的加工工具, 再加工 O_i , 即 O_k 和 O_i 存在等待延迟关系, 将等待延迟时间 $D_{ik} = t_i^1 + t_k^1$ 从 O_k, O_i 的加工时间中分离出来, 虚拟为一道设备有关延迟工序 O_{ik} , 转换之后工序间的约束关系如图 2(右)所示, 则 O_k, O_{ik}, O_j 和 O_i 的加工次序为 O_k, O_j, O_{ik}, O_i 。当 O_j 的加工时间为零时, 此种情况转变为情况 1。

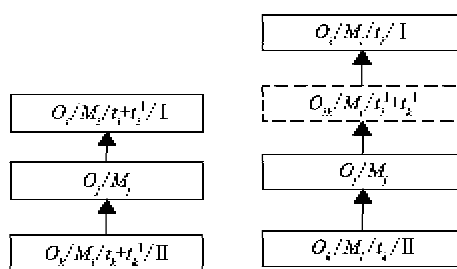


图 2 情况 2 示意图

(3) 情况 3。若标准工序 O_k, O_i 在同一设备 M_i 上加工, 加工时间分别为 $(t_i + t_i^1), (t_k + t_k^1)$, 工序间的约束关系如图 3(左)所示。因 O_k, O_i 的加工类型不同, O_k 加工结束后在 O_i 加工前需要调整设备 M_i 的加工工具, 即 O_k 和 O_i 存在等待延迟关系, 将等待延迟时间 $D_{ik} = t_i^1 + t_k^1$ 从 O_k, O_i 的加工时间中分离出来, 将等待延迟时间段虚拟为一道设备有关延迟工序 O_{ik} 。由于 O_k 在关键路径上, 因为关键路径上工

序的加工时间直接影响产品总的加工时间,将 O_{ik} 转换为 O_i 的紧前工序,转换后三者的关系如图 3 (右)所示。三者在设备 M_i 上的加工次序为 O_k, O_{ik} 及 O_i 。

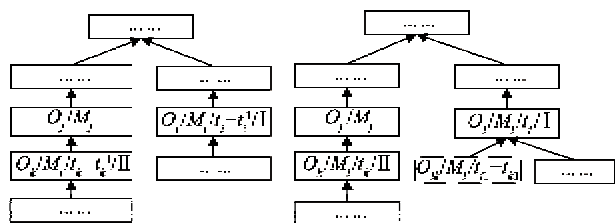


图3 情况3示意图

(4) 情况4。若标准工序 O_k, O_j 和 O_i , 在同一设备 M_i 上加工, 加工时间分别为 $(t_k + t_k^1), (t_j + t_j^1), (t_i + t_i^1)$, 其中 O_k, O_j 是 O_i 的前续工序, 工序间的约束关系如图 4 (左) 所示。因加工类型不同, 在加工完 O_k, O_j 后因装配关系需要调整设备 M_i 的加工工具才能加工 O_i 即 O_k, O_j 和 O_i 之间存在等待延迟关系, 将等待延迟时间 $D_{ij} = t_i^1 + t_k^1 + t_j^1$ 从 O_k, O_j 和 O_i 的加工时间中分离出来, 将等待延迟时间段虚拟为一道设备有关延迟工序 O_{ijk} , 则 O_k, O_j, O_i, O_{ik} 转换后的关系如图 4 (右) 所示, 转换后在设备 M_i 上的加工次序为 O_j, O_k, O_{ik}, O_i 。

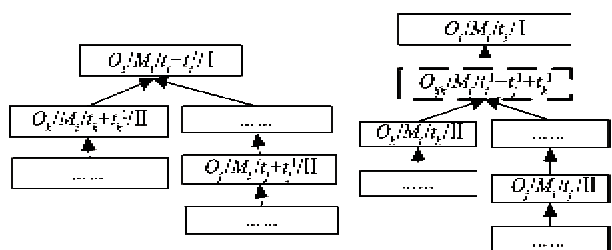


图4 情况4示意图

3 算法设计

对于设备有关延迟约束问题, 为了使设备有关延迟时间灵活移动, 缩短产品加工时间, 本文先采取将等待延迟时间转化为设备有关延迟工序的策略, 将一般加工工艺树转换为存在设备有关延迟约束的加工工艺树, 然后再采用拟关键路径法确定工序的调度次序。

3.1 拟关键路径法

拟关键路径法的基本思想利用循环递归算法优先调度对加工时间有重要影响的关键路径上的工序, 是具有 2 次复杂度且调度效果较好的综合调度

算法^[6]。

3.2 算法策略

根据存在设备有关延迟约束的加工工艺树的特点, 设计算法策略: 首先, 对存在设备有关延迟约束的加工工艺树上的工序采用拟关键路径法确定工序的调度次序; 其次在加工过程中, 标准工序采用采用文献[6]的调度方法加工; 为使受设备有关延迟工序影响的标准工序尽早开始加工, 采用设备有关延迟工序配合调整的策略。

3.3 设备有关延迟工序配合调整的策略

设存在设备有关延迟约束的加工工艺树中, 标准工序 O_p, O_i 和 O_j 在同一设备 M_i 上加工, 其中 O_p 是 O_i, O_j 的前续工序, O_p 和 O_j 加工类型相同, O_p 和 O_i 的加工类型不同, 即如果 O_p 和 O_i 连续加工, 存在设备有关延迟工序 O_{pi} 。由于 O_{pi} 可以在其所有前续工序结束后和紧后工序开始加工前之间任意时刻加工, O_{pi} 的加工情况分为如下两种:

(1) 若 O_j 可在 O_p 和 O_i 间加工, 应先加工 O_j , 这样可以去掉按 O_p, O_i, O_j 顺序加工形成 O_{pi} 和 O_{ij} 之中的 O_{ij} , 可减少整个产品加工过程中设备有关延迟时间。

(2) 若 E_{pk} 和 S_{ik} 间的空闲时间大于 O_{pi} 的加工时间 D_{pi} 时, O_{pi} 尽可能的延后加工, 这样剩余空闲时间, 可以加工与 O_p 加工类型相同的可加工工序。

4 算法实现

步骤1 设置工序属性。

根据加工和装配综合调度的要求和树状结构产品的特点, 为每个工序设置 5 个属性: (1) 工序名, 用 O_i 表示; (2) 工序 O_i 的加工时间, 用 t_i 表示; (3) 工序 O_i 的加工设备, 用 M_i 表示; (4) 工序 O_i 的紧后工序, 用 N_i 表示工序 O_i 的唯一紧后工序; (5) 工序 O_i 的加工类型, 用 f_i 表示。于是, 工序 O_i 可表示为 $O_i/t_i/M_i/N_i/f_i$ 。

步骤2 (1) 为工序的属性赋初值, 设共有 n 道工序, 工序名 $O_i, i \in \{0, 1, \dots, n-1\}$; (2) 初始 t_i 中的时间值为 0; (3) 初始 M_i 中的设备为 Null; (4) 唯一紧后工序; (5) 初始工序的加工类型 f_i 为 Null。

步骤3 确定工序的加工顺序。

按文献[6]中的拟关键路径法对工序进行排序。

步骤4 等待延迟时间转换为设备有关延迟工序, 具体实现方法: 按步骤3中的排序, 若标准工序

O_j 在标准工序 O_i 后加工, 首先比较 O_i 、 O_j 的加工设备 M_i 、 M_j , 若 $M_i = M_j$, 再比较工序的加工类型是否相同, 若 $f_i \neq f_j$, 则把调整设备的时间从其前后工序的加工时间内分离出来, 转换为顺序约束的设备有关延迟工序, 并更改工序间的前后约束关系。

步骤 5 工序加工。

(1) 对于标准工序采用文献[6]中的前沿贪心规则加工。(2) 对于设备有关延迟工序采用设备有关延迟工序配合调整的策略确定其开始加工时间。

步骤 6 若调度结点为根结点, 调度结束, 输出甘特图。

5 复杂度分析

假设所有产品的总工序数为 n , 设备数为 m , 设备有关延迟工序数为 F , 算法的复杂度分析如下:

(1) 等待延迟时间转换为设备有关延迟工序后, 产品总的工序数为 $n + F$ 。若每个设备上平均加工工序数为 n/m , 最坏情况下, 每个设备上设备有关延迟工序数最多为 $n/m - 1$, m 个设备上设备有关延迟工序数最多为 $F = m \cdot (n/m - 1) = n - m$ 。转换后产品的总工序数为 $2n - m$ 。

(2) 采用拟关键路径法确定工序的调度次序, 根据文献[6], 其排序需要的比较次数为 $(n + F)(n + F - m) / (2m) = (2n - m) \cdot (2n - 2m) / (2m) = 2n^2m - 3n + m$, 因为 $1 \leq m \ll n$, 即排序复杂度为 $O(n^2)$ 。

(3) 设备有关延迟约束转换为设备有关延迟工序。首先按工序加工顺序比较工序设备属性, 最坏情况下, 需和其他 $n - 1$ 道工序比较, 比较 $n - 1$ 次, 所以比较所有工序设备属性最多为 $n \cdot (n - 1)$; 其次对同一设备上的工序比较相邻工序的加工类型属性, 若平均每台设备上有 n/m 道工序, 最坏情况下, 每台设备比较加工类型 $2n/m$, 所有设备比较加工类型次数最多为 $2n$ 次。即设备有关延迟约束转换为设备有关延迟工序最多需要比较 $n \cdot (n - 1) + 2n$ 次, 所以转换设备有关延迟工序的复杂度为 $O(n^2)$ 。

(4) 确定设备有关延迟工序的开始加工时间。若标准工序 O_i 和 O_j 间存在设备有关延迟工序 O_{ij} , 先按工序加工顺序查找 O_i 和 O_j 间是否存在和 O_i 同设备且同加工类型的标准工序, 最坏情况下, 需和其他 $n - 2$ 道标准工序进行比较, 比较次数最多为 $2(n - 2)$ 次。由于设备有关延迟工序数最多为 $F = n -$

m , 所以确定设备有关延迟工序的开始加工时间最多比较 $2(n - 2) \cdot (n - m)$ 次, 即确定设备有关延迟工序开始加工时间的复杂度为 $O(n^2)$ 。

以上分析表明, 采用步骤分解的方式调度, 本文算法的时间复杂度为 $O(n^2)$ 。虽然具有 2 次复杂度算法的计算次数为工序数的平方, 计算时间数量级比工序数的数量级多 1, 用现代计算能力每秒千万亿次的计算机处理, 即使调度工序数 $n = 10000$ 的产品时, 也只需要约 10^{-3} 秒, 所以, 该算法受具体例子规模影响不大。

6 示例分析

考虑到具体例子规模对该算法影响不大, 本文采用如图 5 所示的产品 A 说明, 图中长方框内符号分别为: 工序名/加工设备号/加工时间/加工类型。采用拟关键路径法对图 5 所示的加工工艺树的调度次序为: A4、A5、A6、A1、A2、A3、A7、A8、A10、A13、A9、A11、A12、A14。在同一设备上相邻加工工序, 因工序的加工类型不同, 需要设备有关延迟约束时间信息表如表 1 所示。

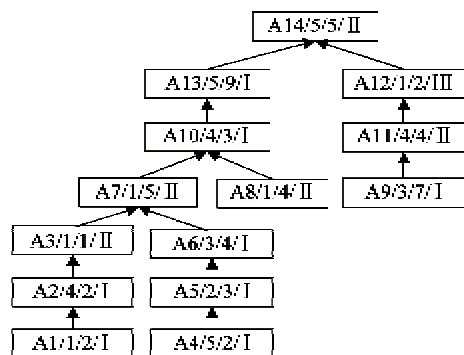


图 5 产品 A 的加工工艺树

表 1 设备有关延迟约束时间信息表

工序	调整的设备	设备有关延迟 约束时间(工时)
A1→A3	M1	1
A10→A11	M4	4
A8→A12	M1	3
A13→A14	M5	5

在以往的综合调度算法中, 处理加工工艺树中工序间的延迟约束通常是等待延迟时间加入到前后工序的加工时间内或忽略等待延迟时间, 从而导致调度结果不精确。对于产品 A 的一般加工工艺

树中存在等待延迟约束的问题,先通过本文提出的等待延迟时间到设备有关延迟工序的转换策略将存在等待延迟时间的一般加工工艺树转换为设备有关延迟约束的加工工艺树,如图6所示。其中V1,V2,V3,V4为设备有关延迟工序,加工设备分别为M1,M4,M1,M5。

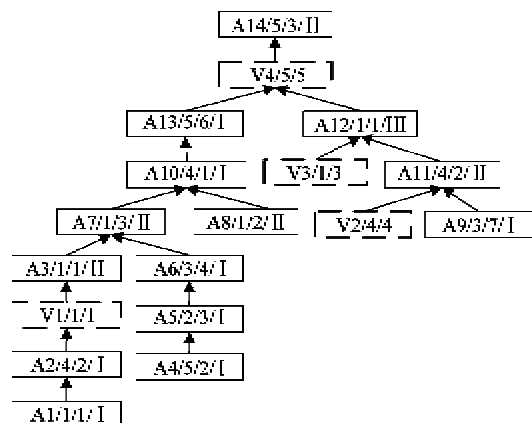


图6 产品A的设备有关延迟约束的加工工艺树

(1)采用文献[6]的调度算法即拟关键路径法(ACPM)和BFSM,对图5所示的一般加工工艺树调度,需先比较各叶子结点的路径长度(A1工时为27,A4工时为31,A8工时为21,A9工时为18),路径最长的工序序列(A14、A13、A10、A7、A6、A5、A4)为关键路径。按照文献[6]中ACPM调度各工序的次序为:A4、A5、A6、A1、A2、A3、A7、A8、A10、A13、A9、A11、A12、A14。调度甘特图如图7所示,图中纵坐标表示设备,横坐标表示时间,长方框表示工序,框中数字表示工序名,框的长度按工序时间比例确定(下同)。

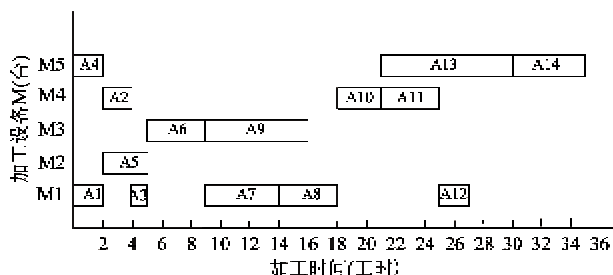


图7 采用文献[6]的算法调度图5的甘特图(35个工时)

(2)采用本文的算法对图6所示加工工艺树的调度时,也需先比较各叶子结点的路径长度(A1工时为23,A4工时为27,A8工时为17,A9工时为18,V2工时为15,V3工时为12),路径最长的工序序列(A14、V4、A13、A10、A7、A6、A5、A4)为关键路

径。按照文献[6]中的拟关键路径法(ACPM)调度的各工序的次序为:(A4、A5、A6、A1、A2、V1、A3、A7、A8、A10、A13、A9、V2、A11、V3、A12、V4、A14),按照本文算法的调度甘特图如图8所示。

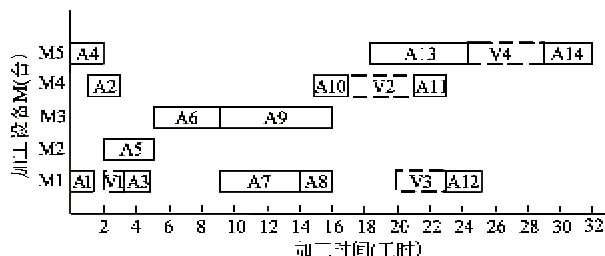


图8 采用本文算法调度图6的甘特图(32个工时)

在采用本文算法调度图6所示的加工树的过程中,对于加工树中设备有关延迟工序的调度采用设备有关延迟工序配合调整的策略确定设备有关延迟工序的开始加工时间,如图8中的V3,在A7完成后,A12加工之前都可以加工,因在A7和A12间有加工类型和A7加工类型相同的A8,应先加工A8再加工V3,这样可以使得A8及其紧后工序提前加工,可使产品总的加工时间缩短,否则会延迟产品总的加工时间;如图8中的设备有关延迟工序V1,在A1完成后、A3加工前可以直接加工,根据本文调整策略尽量让设备有关延迟工序V1延后加工,这样可以让其他产品的和A1的加工类型相同的工序提前加工。

由图7和图8可以看出,采用文献[6]的调度算法,对图5所示加工工艺树加工,总加工时间为35个工时;采用本文的调度算法,对图6所示的设备有关延迟约束的加工工艺树加工,总加工时间为32个工时,总加工时间缩短3个工时。之所以缩短是因为,本文的算法分离了工序实际加工时间和由调整刀具或夹具造成的等待延迟时间,再通过本文提出的等待延迟时间到设备有关延迟工序的转换策略,使得等待延迟时间不再是固定在实际工序的加工时间内,而是在满足约束条件下,转换为可以在一定范围内活动的虚拟工序:设备有关延迟工序,便于其他标准工序提前加工,这样将会使得产品总的加工时间缩短。

该例子虽然简单,但树状结构产品可能出现的设备有关延迟工序的各种情况均已考虑,如:A1和A3之间的延迟是转换策略的情况2;当A1和A3之间A2的加工时间不为零时,此种情况转变为情况1;A8和A12之间的延迟约束是转换策略的情况3;

A13 和 A14 之间的延迟约束是转换策略的情况 4。因此该例子所反映的存在设备有关延迟约束的综合调度问题具有普遍意义。

7 结 论

本文针对综合调度问题中存在设备有关延迟约束的情况,首次提出了解决该问题的算法,为解决此类问题提供了参考,具有一定的理论和实际应用价值。结论如下:

(1)考虑设备有关延迟约束的方法可把设备有关延迟时间转换为在一定范围内处理的虚拟工序,便于其它标准工序提前加工,缩短综合调度总加工时间。

(2)本文提出的算法,可以在二次多项式内较好的解决存在设备有关延迟约束的综合调度问题,因此本文提出的算法简便、可行。

(3)本文只研究了存在设备有关延迟约束的综合调度问题,如果将本文算法与柔性调度算法相结合,可扩展解决存在设备有关延迟约束的柔性综合调度问题。

参考文献

[1] Brucker P, Schl R. Job shop scheduling with multi-purpose machines. *Computing*, 1990, 45 (4): 369-375

[2] 张长胜,孙吉贵,欧阳丹彤等. 求解车间调度问题的自适应混合粒子群算法. *计算机学报*, 2009, 32 (11): 2017-2146

[3] Goncalves J F, De Magalhaes Mendes J J, Resende M G C. A hybrid genetic algorithm for the job shop scheduling problems. *European Journal of Operation Research*, 2005, 167(1): 77-95

[4] 崔健双,李铁克. 求解作业车间调度问题的全局邻域搜索方法. *计算机集成制造系统*, 2009, 15 (7): 1383-1388

[5] 谢志强. 工件间有约束的复杂产品工序调度研究[博士学位论文]. 哈尔滨:哈尔滨理工大学计算机科学与技术学院, 2009. 5-11

[6] 谢志强,刘胜辉,乔佩利. 基于 ACPM 和 BFSM 的动态 Job Shop 调度算法. *计算机研究与发展*, 2003, 40 (7): 977-983

[7] 谢志强,杨静,周勇等. 基于工序集动态关键路径多产品制造调度算法. *计算机学报*, 2011, 34 (2): 406-412

[8] 谢志强,滕宇峰,杨静. 紧密衔接工序组联动的综合调度算法. *自动化学报*, 2011, 37 (3): 371-379

[9] Brucker P, Kunst S. Scheduling chains with identical jobs and constant delays on a single machine. *Mathematical Methods of Operations Research*, 2006, 63 (1): 63-75

[10] Alix M, Francis S. Scheduling chains on a single machine with nonnegative time lags. *Mathematical Methods Operations Research*, 2003, 53 (1)

[11] 谢志强,莫涛,谭光宇. 非紧密衔接工序动态车间调度算法[J]. *机械工程学报*, 2008, 44 (1): 155-166

[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. *机械工程学报*, 2011, 47 (4): 177-185

[13] 谢志强,刘胜辉,乔佩利. 电机产品加工的动态调度算法. *计算机集成制造系统*, 2003, 9 (6): 444-448

An integrated scheduling algorithm with device dependence deferred constraint

Xie Zhiqiang^{* **}, Shao Xia^{**}, Yang Jing^{*}

(* College of Computer Science and Technology, Harbin Engineering University, Harbin 150001)

(** College of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080)

Abstract

Aiming at the problem that the present integrated scheduling ignores the device-dependent deferred constraint (DDC) thus the scheduling result is affected, an integrated scheduling algorithm with the device-dependent deferred constraint is proposed. On the basis of the concepts of standard procedure, device-dependent deferred procedure (DDP) and the processing tree with the device-dependent deferred constraint, the algorithm converts the waiting delay time into the device-dependent deferred procedure, thus the scheduling problems with the device-dependent deferred constraint are converted into the general integrated scheduling problem being easy to solve. Then, the order of the procedures is determined by the allied critical path method (ACPM), and the start time of procedure is determined by the forward greedy rule. The strategy for device-dependent delay procedure cooperating adjustment is proposed to make affected standard procedures start early. The experimental results show that the proposed algorithm can solve the integrated scheduling problems with device-dependent deferred constraint. The scheduling result is more accurate, and the total processing time is reduced without increasing the algorithm complexity.

Key words: deferred constraint, device-dependent deferred constraint (DDC), integrated scheduling, allied critical path method (ACPM), algorithm