工作排序问题及其数学模型

洪 文 胡雁玲

(安徽大学,安徽 合肥 230039; 合肥联合大学,安徽 合肥 230022)

[摘要] 利用 LINGO4.0给出了工作排序问题的数学模型。模型的数据与公式完全分离,调试模型只要在数据存放的文件中进行,非常容易。模型的类型属线性规划模型并用 LINGO 语言编写,求解迅速。该模型具有较强的实用性和通用性。

[关键词] 密集; 属性; 约束

[分类号] O220 [文献标识码] A [文章编号] 1008- 6056(2002)01- 0095- 05

Work- piece Ordering and Its Math Model

HONG Wen (Anhui Uniuersity, Hefei 230039)

HU Yan- ling(Hefei Union Univ., Hefei Anhui 230022)

Abstract: This paper introduces a math model for work- piece ordering with LINGO 4.0. The debugging model can be easily carried out within the data files due to the complete seperation of data from the formula of the model. The model, written with LINGO language, belonging to linear programming, can solve problems more rapidly, and therefore can be widely used.

Key words: denseness; attribute; restriction

0 背景

在工作排序问题中, 每个工件将被指派到不同的机床上进行加工作业, 目标是求出所有工件的加工方案, 使在最短的时间内完成全部加工作业。对于每一个工件来说, 它要在不同的机床上进行加工作业, 而且还要按一定的顺序进行; 对每一个机床来说, 它要加工不同的工件, 而且必须按照一定的顺序进行。虽然我们无论如何都可以求出工件的加工方案, 但是, 如果安排不当就必然会出现瓶颈现象——要床等待加工的时间较长。这样就会使得完成全部工件加工的总时间较长。由于工件加工作业之间具有优先关系, 有些作业必须等到其它作业完成后才可进行, 这使得工件排序问题变得更加复杂。指派到机床上的加工作业必须服从优先关系。

1 问题

假设用 4 台机床加工 3 个工件。各个工件的机床加工顺序, 以及工作 i 在机床 j 上的加工时间(j= 1, 2, 3; j= 2, 3, 4) 见下表: (单位: 上时)

表 1 工件加工及时间表

	机术 1→	机床 2→	机床 3→	机床 4
工件 1	3	4	9	2
工件 2	2	6	3	4
工件 3	1	2	4	8

[收稿日期] 2002- 01- 10

[作者简介] 洪 文(1958-), 男, 安徽合肥人, 学士, 副教授.

我们要找出一种工件加工作业方案使得完成全部工件加工的总时间达到最小。

2 模型

```
! 该模型是处理工件加工问题;
! 建立工件集合、机床集合、时间集合及它们的乘积;
sets:
        GONGJ/ 1..3/ ;
JICHUANG/ 1..4/ ;
SHIJIAN/ 1..8/ ;
! 属性 x 的奇数列表示开始时间; 偶数列表示结束时间。;
    GONGSHI( GONGJ, SHIJIAN) : x;
! 属性 A 存储工件在每个机床上所用时间;
! 属性 Y 是控制变量;
    LINKS( GONGJ, JICHUANG) : A, y;
endsets
data:
! 从文件‘ 工件排序. xls’ 中输入加工时间, 加工时间数据用域“ 加工时间” 定义;
A= @ OLE(‘ \ lingo \ 工作排序. xls’ ,‘ 加工时间’ ) ;
! 输出解答到文件‘ 工作排序. xls’ 中的‘ 起止时间”域止;
      @ OLE(‘ \ lingo \ 工作排序. xls’ ,‘ 起止时间’ ) = x;
enddata
! 求出两个基本集合 GONGJ 和 JICHUANG 的大小;
s= @ size( GONGJ) ;
m= @ size( JICHUANG) ;
! 目标函数; min= w ; @ for( GONGJ( i) : w > = (i, m+ 3) + a( i, m) ) ;
! 每个工件加工次序的约束;
@ for( LINKS( i, j) | j# lt# m: x( i, 2* j- 1) + a( i, j) < = x( i, 2* j+ 1) ) ;
! 每个机床加工次数的约束;
@ for( GONGJ( k) | k# lt# s: / @ for( LINKS( i, j) | i# lt# s- k+ 1: / x( i, 2* j- 1) + a( i, j) <
= x( i+ k, 2* j- 1) + 1000* - y( i+ k- 1, j) ) ) ; /
! 限制 y( i, j) 为 0- 1 变量;
@ for( GONGJ( i) : @ for( JICHUANG( j) : @ bin( y( i, j) ) ) ) ; / ! 为结束时间赋值;
@ for( GONGJ( l) : @ for( SHIJIAN( k) | k# le# m : x( 1, 2* k) = x( 1, 2* k- 1) + a( l, k) ) ) ;
```

3 模型注释

上面的数学模型是用 LINGO4. 0for Windows 编写的线性规划模型。运行后立刻就能得到结果。下面就模型的主要部分给出解释。

3.1 集合

我们定义三个基本集合——GONGJ(工件)、JICHUANG(机床) 和 SHIJAN(时间)。我们

用这三个基本集合产生两个派生集合。一个是 LINKS, 它是一个密集, 是由集合 GONGJ 与 JICHUANG 相乘而得。利用 LINKS 我们产生两个属性。属性 A 存储工作在每个机床上加工所用时间; 属性 Y 是控制变量。另一个派生集合是 GONGSHI, 它也是一个密集, 是由集合 GONGJ 和 SHIJIAN 相乘而得。利用 GONGSHI 我们产生一个属性 X, 它表示工件在各个机床上加工的开始时间和结束时间。

3.2　变量

模型里的决策变量是属性 X。X(i, 1)、X(i, 3)、X(i, 5)、X(i, 7) 表示工件 i 在 4 个机床上的加工开始时间, X(i, 2)、X(i, 4)、X(i, 6)、X(i, 8) 表示工件 i 在 4 个机床上的加工结束时间(i= 1, 2, 3); 当 j 是奇数时, X(1, j)、X(2, j)、X(3, j) 表示机床(j+ 1)/2 加工 3 个工件的开始时间, 当 j= 1, 2, ... 8). 所有的 X(i, j) 构成了工件的加工时间表。属性 Y 是控制变量, 它只取 0 和 1, 用来控制机床加工工件的顺序。

3.3　数据

在此模型中, 输入的数据是存放在文件工件排序. xls 里的' 加工时间' 域上。具体内容见表 1 中的数据部分。

将数据放在模型之外可以使得模型的调试变得很容易。如果加工时间有变化, 只要改动表 1 中的数据, 模型不需作任何改动就可获得相应的解答。

输入数据是用下面的公式完成:

A= @OLE(' \lingo\ 工件排序. xls', ' 加工时间');

3.4　目标函数

这个模型的目标很简单, 它就是求出加工完全部工件的总时间的最小值。可用下面的表达式给出:

min= w;

@for(GONGJ(i): W>= x(i, m+ 3)+ a(i, m)); 这里, M 表示最后一个机床。

3.5　约束

在这个模型里, 我们有下面两类约束:

3.5.1　每个工件加工次序的约束

对于每个工件来说, 加工次序必须得到满足。即工件 i(i= 1, 2, 3) 在某个机床上的开始加工时间加上加工所有时间必须小于等于在下一个机床上的开始加工时间。可以用下面的表达式表达:

@for(LINKS(i, j)| j# lt# m: x(i, 2* j- 1)+ a(i, J)< = x(i, 2* j+ 1));

3.5.2　每个机床加工次序的约束

对于每个机床来说某一时刻只能加工一个工件。例如, 我们只考虑工件 1 和工件 2 在机床 1 上的加工情况, 则必须满足: X(1, 1)+ a(1, 1) ≤X(2, 1) 或 X(2, 1)+ a(2, 1) ≤X(1, 1)。我们可以用下面的表达式表达机床加工次序的约束:

@for(GONGJ(k)| k# lt# s: @for(LINKS(i, j)| i# lt# s- k+ 1:

x(i, 2* j- 1)+ a(i, j)< = x(i+ k, 2* j- 1)+ 1000* y(i+ k- 1, j)));

@for(GONGJ(k)| k# lt# s: @for(LINKS(i, j)| i# lt# s- k+ 1:

x(i+ k, 2* j- 1)+ a(i+ k, j)< = x(i, 2* j- 1)+ 1000* (1- y(i+ k- 1, j))));

! 限制 y(i, j) 为 0- 1 变量; @for(GONGJ(i): @for(JICHUANG(j): @bin(y(i< j))));

4 解答

模型求解后，我们将解答送回到文件工件排序.xls 里的‘起止时间’域上。可用下式完成：

@OLE(‘\lingo\工件排序.xls’,‘起止时间’)= x

输出的具体结果见下表中的数据部分：

表 2 工件加工时间表

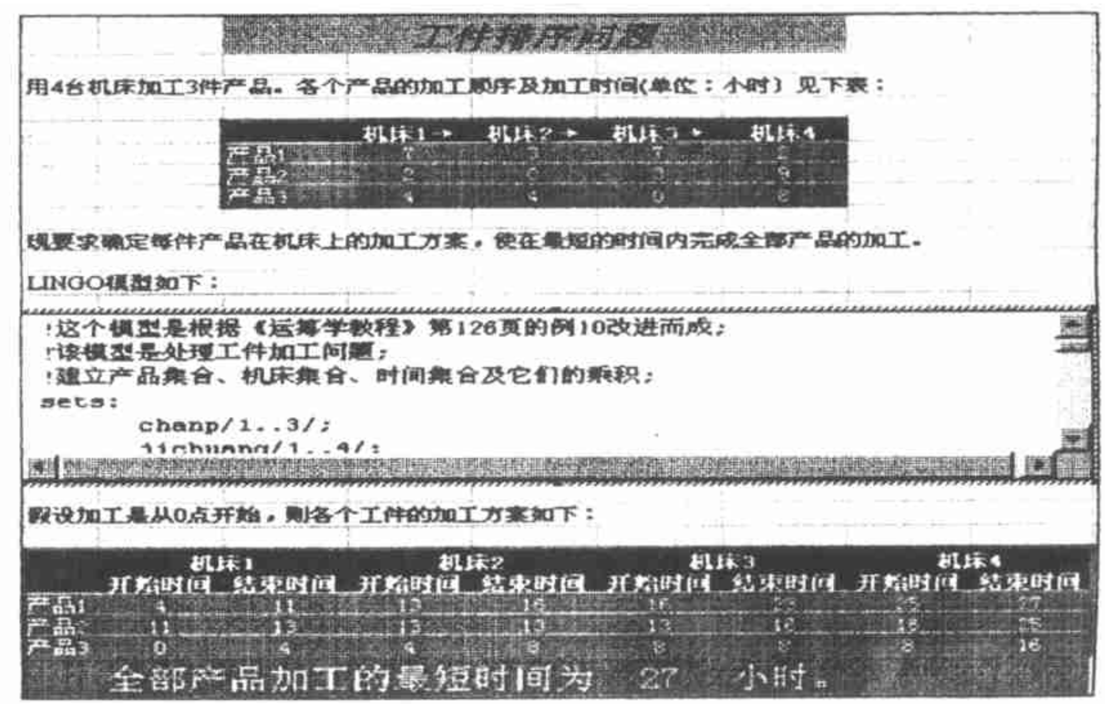
	机床 1		机床 2		机床 3		机床 4	
	开始时间	结束时间	开始时间	结束时间	开始时间	结束时间	开始时间	结束时间
工件 1	3	6	9	13	13	22	22	24
工件 2	1	3	3	9	9	12	15	19
工件 3	0	1	1	3	3	7	7	15

5 说明

5.1 模型中的工件数和机床数都是固定的，分别是 3 和 4。如果发生变化，例如工件数为 s，机床数为 m，则只需将“GONGJ\1..3\;”、“JICHUANG\1..4\;”和“SHIJIAN\1..8\;”中的 3 改为 s，4 改为 m，8 改为 2m 即可。同时文件工件排序.xls 中的“起止时间”和“加工时间”域也要作相应的变化。

5.2 如果只是工件的加工时间发生变化，则只需在文件工件排序.xls 中改变相应的数字即可，而对模型不需作任何改动。

5.3 本模型也可直接嵌入到文件工件排序.xls 中，这样使用起来更加方便。下图就是嵌入模型并变动加工时间后的运算结果：



5　结束语

本文通过对工件排序问题的研究给出了这一问题的数学模型。虽然实际问题中的工件排序问题还有很多其他类型, 这里不能一一给出相应的模型, 但是本文介绍的方法是值得借鉴的, 相应的模型公式与数据完全分离, 具有通用性强、易于模拟等优点。另外, LINGO forWindows 对变量, 尤其是对整数变量有限制。当问题较大时就可能导致模型无法运行, 这时候不要使用更高版本 LINGO forWindows。

[参考文献]

[1]　胡运权. 运筹学教程, [M]. 北京: 清华大学出版社, 1998.

[2]　卢开澄. 单目标、多目标与整数规划[M]. 北京: 清华大学出版社, 1999.

[3]　郭耀煌. 运筹学原理与方法, [M]. 成都: 西南交通大学出版社, 1994.

[4]　郭耀煌. 运筹学与工程系统分析, [M]. 北京: 中国建筑工业出版社, 1986.

[5]　李宗元. 运筹学 ABC, [M]. 北京: 经济管理出版社, 2000.

[6]　张建中. 线性规划, [M]. 北京: 科学出版社, 1999.