## 若有疑问的小伙伴留言微信公众号: 数模自愿分享交流

# 第五节 指派问题 (Assignment Problem)

### 1. 标准指派问题的提法及模型

指派问题的标准形式是: 有n个人和n件事,已知第i个人做第j件事的费用为 $c_{ij}$ (i,j=1,2,…,n),要求确定人和事之间的一一对应的指派方案,使完成这n件事的总费用最小。

设
$$n^2 \wedge 0 - 1$$
变量  $x_{ij} = \begin{cases} 1 & \text{若指派第i} \wedge \text{人做第j} \wedge \text{件事} \\ 0 & \text{若不指派第i} \wedge \text{人做第j} \wedge \text{件事} \end{cases}$  (i,j=1,2,..., n)

数学模型为:

$$\min Z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

$$\begin{cases} \sum_{i=1}^{n} x_{ij} = 1 \\ \sum_{j=1}^{n} x_{ij} = 1 \\ x_{ij} = 0 \quad or \quad 1, \quad i, j = 1, 2, \dots, n \end{cases}$$

其中矩阵C称为是效率矩阵或系数矩阵。

其解的形式可用0-1矩阵的形式来描述,即(xii)nxn。

标准的指派问题是一类特殊的整数规划问题,又是特殊的0-1规划问题和特殊的运输问题。1955年W.W. Kuhn利用匈牙利数学家 D. Konig关于矩阵中独立零元素的定理,提出了解指派问题的一种算法,习惯上称之为匈牙利解法。

#### 2. 匈牙利解法

匈牙利解法的关键是指派问题最优解的以下性质: 若从指派问题的系数矩阵 $C=(c_{ij})$  的某行(或某列)各元素分别减去一个常数k,得到一个新的矩阵 $C'=(c'_{ij})$ ,则以C和C'为系数矩阵的两个指派问题有相同的最优解。(这种变化不影响约束方程组,而只是使目标函数值减少了常数k,所以,<u>最优解并不改变。</u>)

作变换, 其不变性是最优解

对于指派问题,由于系数矩阵均非负,故若能在在系数矩阵中找到n个位于不同行和不同列的零元素(<u>独立的0元素</u>),则对应的指派方案总费用为零,从而一定是最优的。

### 匈牙利法的步骤如下:

步1:变换系数矩阵。对系数矩阵中的每行元素分别减去该行的最小元素;再对系数矩阵中的每列元素分别减去该列中的最小元素。若某行或某列已有0元素,就不必再减了(不能出现负元素)。

步2:在变换后的系数矩阵中确定独立0元素(试指派)。若独立0元素已有n个,则已得出最优解:若独立0元素的个数少于n个,转步3。

确定独立0元素的方法: 当n较小时,可用观察法、或试探法; 当n较大时,可按下列顺序进行

- 从只有一个0元素的行(列)开始,给这个0元素加圈,记作◎,然后划去◎所在的列(行)的其它0元素,记作♦。
- •给只有一个0元素的列(行)的0加圈,记作◎,然后划去◎所在行的0元素,记作♠。
- •反复进行,直到系数矩阵中的所有0元素都被圈去或划去为止。
- •如遇到行或列中0元素都不只一个(存在0元素的闭回路),可任选其中一个0元素加圈,同时划去同行和同列中的其它0元素。被划圈的0元素即是独立的0元素。

- •步3: 作最少数目的直线,覆盖所有0元素(目的是确定系数矩阵的下一个变换),可按下述方法进行
- 1) 对没有⑩的行打"✓"号;
- 2) 在已打"✓"号的行中,对φ所在列打"✓"
- 3) 在已打"✓"号的列中,对◎所在的行打"✓"号;
- 4) 重复2) 3), 直到再也找不到可以打"✓"号的行或列为止;
- 5) 对没有打"✓"的行划一横线,对打"✓"的列划一纵线,这样就得到覆盖所有0元素的最少直线数。

步4:继续变换系数矩阵,目的是增加独立0元素的个数。方法是在未被直线覆盖的元素中找出一个最小元素,然后在打"✓"行各元素中都减去这一元素,而在打"✓"列的各元素都加上这一最小元素,以保持原来0元素不变(为了消除负元素)。得到新的系数矩阵,返回步2。以例说明匈牙利法的应用。

例2 某大型工程有五个工程项目,决定向社会公开招标,有五家建筑能力相当的建筑公司分别获得中标承建。已知建筑公司 $A_i$  (I=1, 2, 3, 4, 5) 的报价 $c_{ij}$  (百万元)见表,问该部门应该怎样分配建造任务,才能使总的建造费用最小?

工程 公司	B1	B2	В3	B4	B5
<b>A</b> 1	4	8	7	15	12
A2	7	9	17	14	10
A3	6	9	12	8	7
A4	6	7	14	6	10
A5	6	9	12	10	6

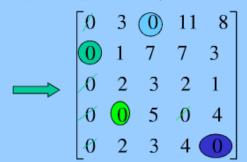
min 
$$Z = 4x_{11} + 8x_{12} + \dots + 10x_{54} + 6x_{55}$$

$$\begin{cases} \sum_{i=1}^{5} x_{ij} = 1 & j = 1, 2, \dots, 5 \\ \sum_{j=1}^{5} x_{ij} = 1 & i = 1, 2, \dots, 5 \\ x_{ij} = 0 & or \quad 1 & i, j = 1, 2, \dots, 5 \end{cases}$$

解:第一步:系数矩阵的变换(目的是得到某行或列均,,,)0元素)

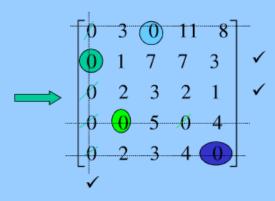
$$\begin{bmatrix} 4 & 8 & 7 & 15 & 12 \\ 7 & 9 & 17 & 14 & 10 \\ 6 & 9 & 12 & 8 & 7 \\ 6 & 7 & 14 & 6 & 10 \\ 6 & 9 & 12 & 10 & 6 \end{bmatrix} \longrightarrow \begin{bmatrix} 0 & 3 & 0 & 11 & 8 \\ 0 & 1 & 7 & 7 & 3 \\ 0 & 2 & 3 & 2 & 1 \\ 0 & 0 & 5 & 0 & 4 \\ 0 & 2 & 3 & 4 & 0 \end{bmatrix}$$

第二步:确定独立0元素,即加圈

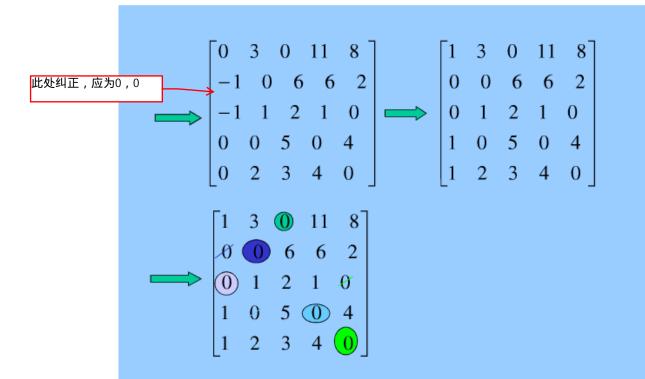


◎元素的个数m=4,而n=5,进 行第三步。

第三步:作最少的直线覆盖所有的0元素,目的是确定系数矩阵的下一个变换。



第四步:对上述矩阵进行变换,目的是增加独立0元素个数。方法是在未被直线覆盖的元素中找出一个最小元素,然后在打"✓"行各元素中都减去这一元素,而在打"✓"列的各元素都加上这一最小元素,以保持原来0元素不变(消除负元素)。得到新的系数矩阵。(它的最优解和原问题相同,为什么?因为仅在目标函数系数中进行操作)



此矩阵中已有5个独立的0元素,故可得指派问题的最优指派方案为:

也就是说,最优指派方案为:让A1承建B3,A2承建B2,A3承建B1,A4承建B4,A5承建B5。这样安排建造费用为最小,即

7+9+6+6+6=34(百万元)

求解下列指派问题,已知指派矩阵为

```
    3
    8
    2
    10
    3

    8
    7
    2
    9
    7

    6
    4
    2
    7
    5

    8
    4
    2
    3
    5

    9
    10
    6
    9
    10
```

```
程序: 指派优化
                                        运行环境: Matlab2011a
c=[3 8 2 10 3;8 7 2 9 7;6 4 2 7 5 ;8 4 2 3 5;9 10 6 9 10];
c=c(:);
a=zeros(10,25);
for i=1:5
a(i,(i-1)*5+1:5*i)=1;
a(5+i,i:5:25)=1;
end
b=ones(10,1);
[x,y]=bintprog(c,[],[],a,b);% bintprog 函数功能: 提供的新的 0-1 线性规划求解函数
x=reshape(x,[5,5])
y%最小值
                                   运行结果
x =
     0
          0
                 0
                      0
                            1
     0
          0
                 1
                      0
                             0
     0
           1
                 0
                      0
                            0
     0
          0
                0
                       1
                            0
     1
          0
                0
                      0
                             0
    21
```

程序: 指派优化	运行环境: Lingo11
model:	
sets:	
var/15/;	
link(var,var):c,x;	
endsets	
data: c=3 8 2 10 3	
87297	
64275	
84235	
9 10 6 9 10;	
enddata	
min=@sum(link:c*x);	

```
@for(var(i):@sum(var(j):x(i,j))=1);
@for(var(j):@sum(var(i):x(i,j))=1);
@for(link:@bin(x));
End
```