图论中最优树问题的Lingo求解

树 连通且不含圈的无向图称为树．常用*T*表示。

树中的边称为树枝，树中度为1的顶点称为树叶

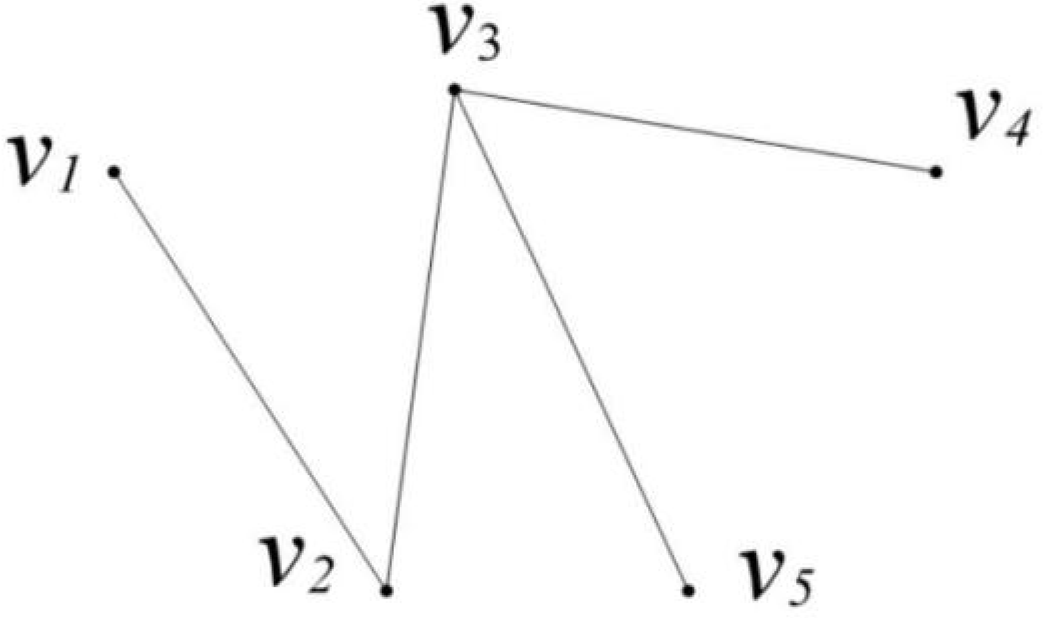


图1 树的示例

生成树　若 T 是包含图 G 的全部顶点的子图,它又是树,则称 T 是 G 的生成树.

最小生成树　设 $T=(V,E_1)$ 是赋权图 $G=(V,E)$ 的一棵生成树，称 T 中全部边上的权数之和为**生成树的权**，记为 $w(T)$，即 $w(T)=\sum_{e\in T} w(e)$.

如果生成树 T^* 的权 $w(T^*)$ 是 G 的所有生成树的权中最小者，则称 T^* 是 G 的最优树，即 $w(T^*)=\sum_{T}\min\{w(T)\}$

在许多实际问题中，如在许多城市间建立公路网、输电网或通信网络，都可以归结为赋权图的最优树问题。

如在一个城市中，对若干个居民点要供应自来水，已经预算出连接各点间管道的造价，要求给出一个总造价最小的铺设方案。

图论中最优树的的求解通常有两种算法：

*Kruskal*算法（或避圈法）和Prim算法（破圈法）.

这里我们给出利用LINGO求解最优树的方法。

设无向图共有 n 个节点，其赋权图的邻接矩阵为 $d_{n\times n}$ 。

d_{ij} 表示节点 i 到 j 的距离。d 为对称矩阵。令 $d_{ii}=0$ 。

现求根节点 1 到各节点生成的最优树，要求各线路上的权值和最小。

其线性规划模型为：

决策变量：设 $x_{ij}=\begin{cases}1 & 节点i与节点j连通\\ 0 & 节点i与节点j不连通\end{cases}$

目标函数为寻找一条从起始点 1 到各节点生成的最优树，

要求各线路上的权值和最小，故目标函数为：

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

1) 对起始点 1 至少有一条路出去，故有：$\sum^{n} x_{1j} \geq 1$

2)对其余各节点，恰有一条路进入，有：$\sum_{\substack{k=1 \\ k\neq i}}^{n} x_{ki} = 1 \quad i = 2,\cdots,n$

3) 所有节点不出现圈，约束为：

$$u_i - u_j + n.x_{ij} \leq n-1 \quad i,j = 1,2,\cdots,n$$

总线性规划模型为：

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

$$s.t.\begin{cases}\sum\limits_{j=2}^{n} x_{1j}\geq 1 \\ \sum\limits_{\substack{k=1\\k\neq i}}^{n} x_{ki}=1 \qquad i=2,3,\cdots,n \\ u_i-u_j+nx_{ij}\leq n-1 \qquad i,j=1,2,\cdots,n \\ x_{ij}=0\text{或}1\end{cases}$$

问题1 某有10个城镇见下图，它们之间的距离见表1。城镇1处有一条河流，现需要从各城镇之间铺设管道，使城镇1处的水可以输送到各城镇，求铺设管道最少的设计方式。

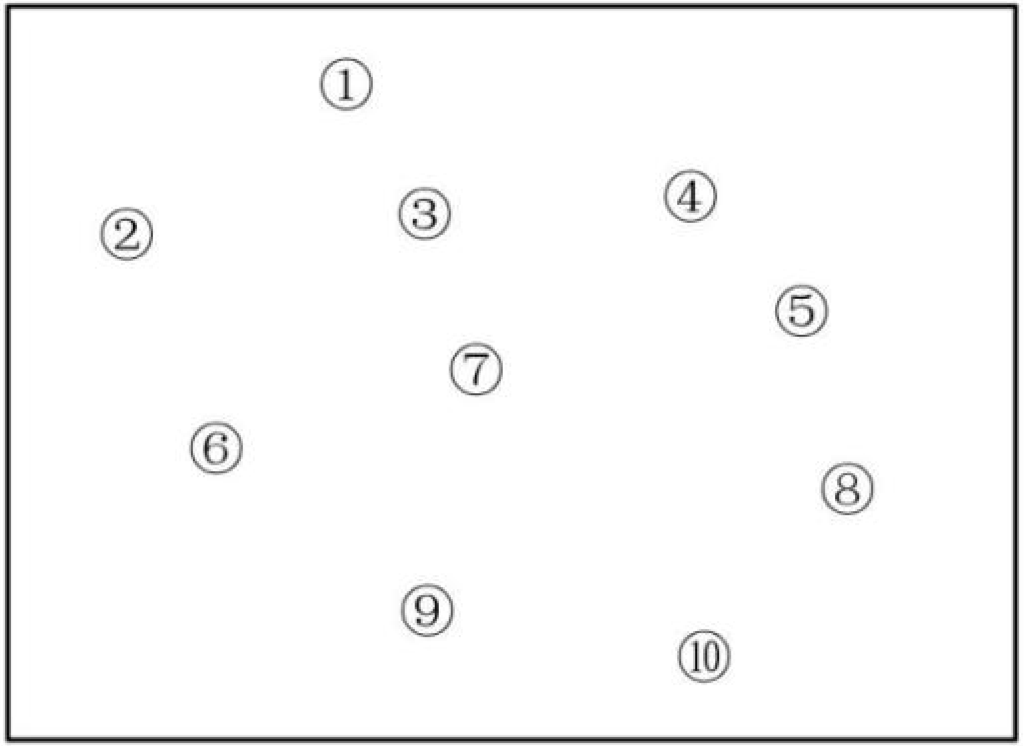


表1 10个地区之间的距离（单位：公里）

地区	1	2	3	4	5	6	7	8	9	10
1	0	8	5	9	12	14	12	16	17	22
2	8	0	9	15	16	8	11	18	14	22
3	5	9	0	7	9	11	7	12	12	17
4	9	15	7	0	3	17	10	7	15	15
5	12	16	9	3	0	8	10	6	15	15
6	14	8	11	17	8	0	9	14	8	16
7	12	11	7	10	10	9	0	8	6	11
8	16	18	12	7	6	14	8	0	11	11
9	17	14	12	25	15	8	6	11	0	10
10	22	22	17	15	15	16	11	11	10	0

该问题实际上是求从点1出发的最优树问题。

Lingo实现程序为：

```
! 最优树的Lingo程序;
model:
sets:
point/1..10/:u;
link(point,point):d,x;
endsets
data:
d=0,8,5,9,12,14,12,16,17,22,
  8,0,9,15,16,8,11,18,14,22,
  5,9,0,7,9,11,7,12,12,17,
9,15,7,0,3,17,10,7,15,15,
 12,16,9,3,0,8,10,6,15,15,
 14,8,11,17,8,0,9,14,8,16,
 12,11,7,10,10,9,0,8,6,11,
 16,18,12,7,6,14,8,0,11,11,
 17,14,12,25,15,8,6,11,0,10,
 22,22,17,15,15,16,11,11,10,0;
@text()=@writefor(link(i,j)|x(i,j)
 #GT#0:'x(',i,',',j,')=',x(i,j),' ');
 enddata
```

```
min=@sum(link(i,j)|i#ne#j:d(i,j)*x(i,j));
n=@size(point);
@sum(point(j)|j#gt#1:x(1,j))>=1;
@for(point(i)|i#ne#1:@sum(point(j)|j#ne#i:x(j,i))=1);
@for(link(i,j):@bin(x(i,j)));
@for(link(i,j)|i#ne#j:u(i)-u(j)+n*x(i,j)
    <=n-1); !不构成圈;
end
```

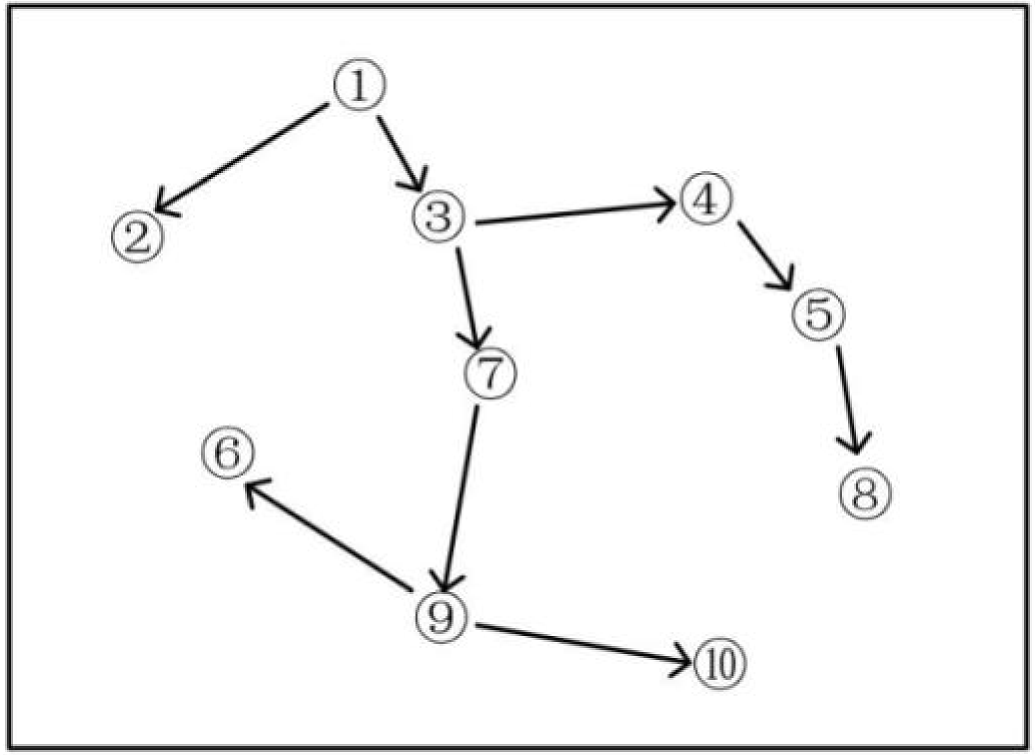
结果为minZ=60

x(1,2)=1 x(1,3)=1 x(3,4)=1 x(4,5)=1

x(9,6)=1 x(3,7)=1 x(7,9)=1 x(5,8)=1

x(9,10)=1

故最优树(最佳铺设管道方式)见图.



谢 谢！