**云南大学数学与统计学院《数学建模实验》上机实验报告**

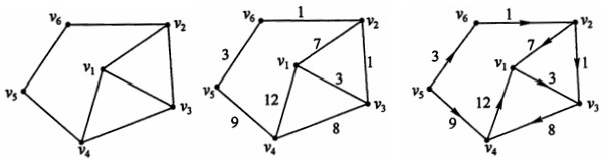
|  |  |  |
| --- | --- | --- |
| **课程名称**：数学建模实验 | **学期：**2023-2024学年秋季学期 | **成绩**： |
| **指导教师**：杨莹 | **姓名**：枫叶 | **学号**： |
| **实验名称**：实验五 图与网络模型及方法 | | |
| **实验编号**：No. 5 | **实验日期**：2023.11.13 | **实验学时**：3 |
| **学院：数学与统计学院** | **专业：统计学** | **年级**：2021级 |

**一、实验目的**

掌握图模型的建模与相关问题的MATLAB求解。

1. **实验内容**

1、用MATLAB分别画出下列图形（图1）:



(a) 非赋权图 (b) 赋权图 (c) 有向图

图1 三种图

2、在图2中，用点表示城市，现有*A*、*B*1、*B*2、*C*1、*C*2、*C*3、*D*共7个城市。点与点之间的连线表示城市间有道路相连。连线旁的数字表示道路的长度。现计划从城市*A*到城市*D*铺设一条沿道路的天然气管道，请用Dijkstra算法求解出最小长度管道铺设方案，并绘制出图形。

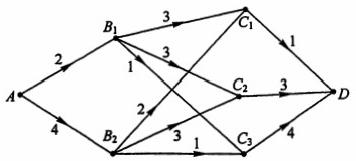


图2 7个城市间的连线图

3、用Prim算法求图3所示赋权图的最小生成树，绘制图形，高亮最小生成树。

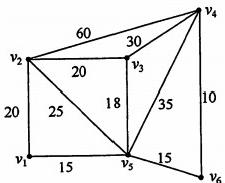


图3 赋权无向图

4、在图3中求从*v*1到*v*4的最短路径和最短距离，并绘制出图形，高亮最短路径。

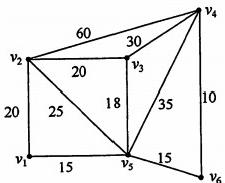


图3 赋权无向图

5、建立整数规划模型，求图4所示网络的最小费用最大流，并显示最小费用最大流对应的矩阵。弧上的第1个数字为单位流的费用，第2个数字为弧的容量。

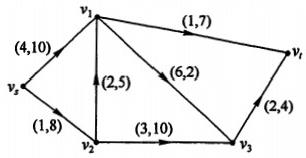


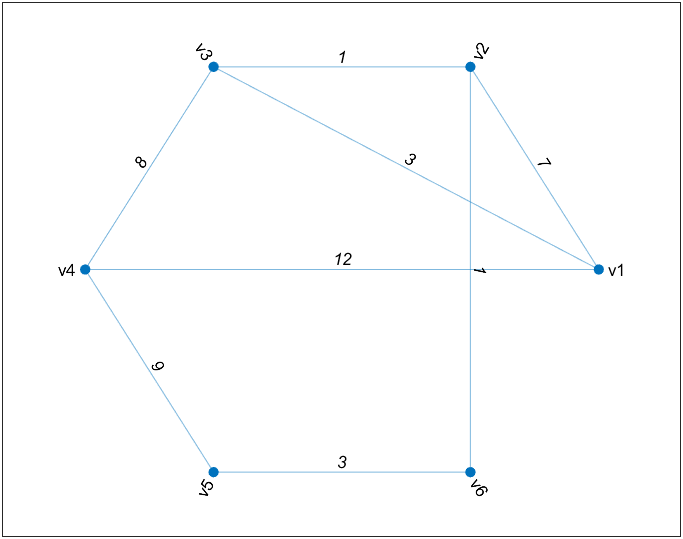
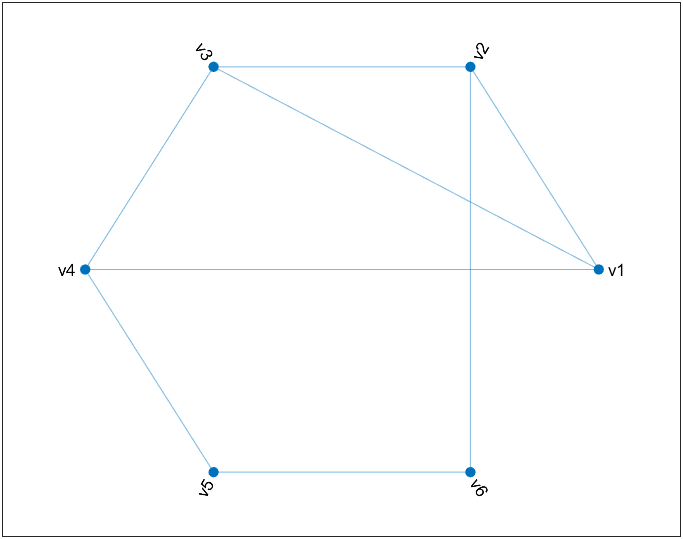
图4 最小费用最大流的网络图

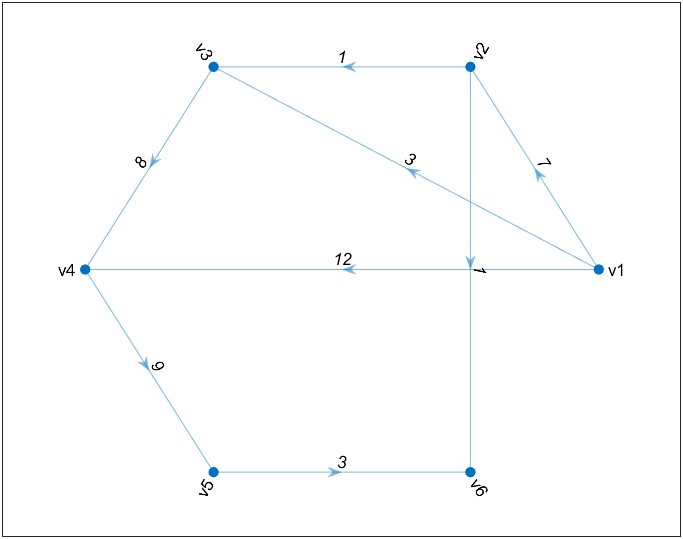
**三、使用环境**

**MATLAB R2021b**

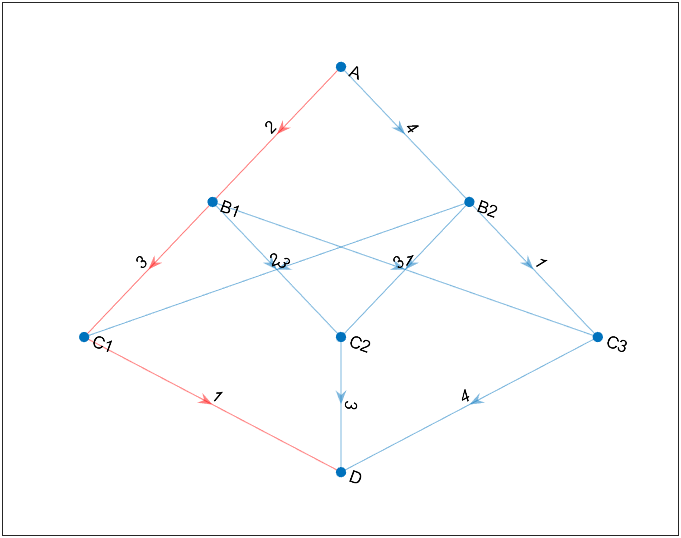
**四、实验过程**

**题目1**

****

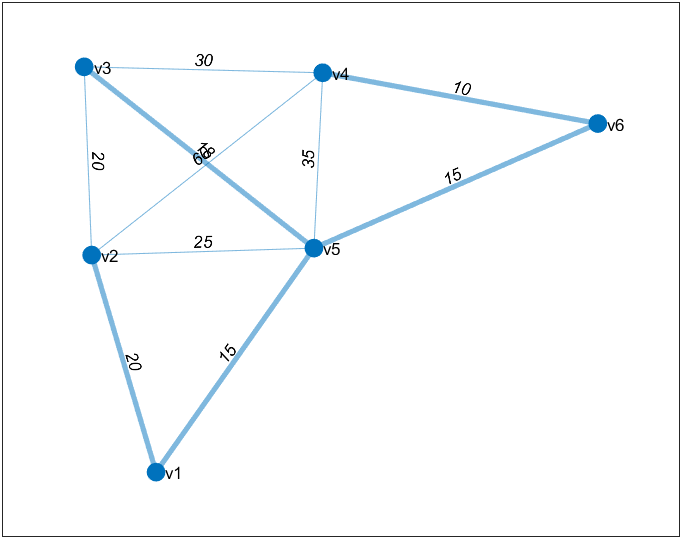
****

**题目2**



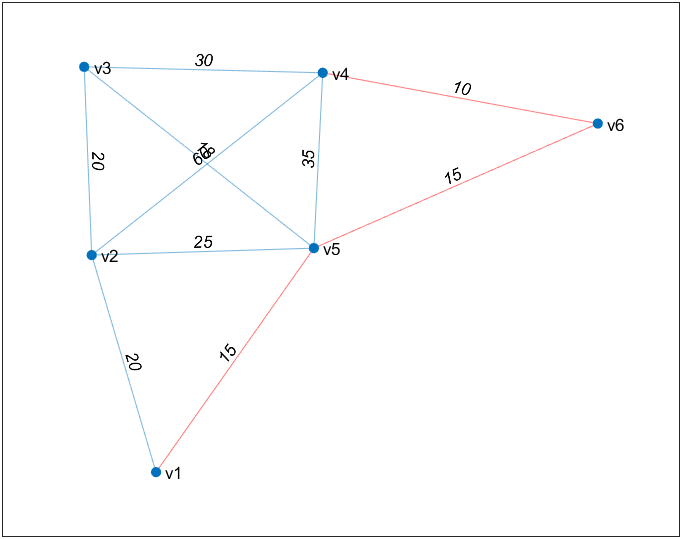
最小长度管道铺设方案为A-B1-C1-D，长度为6，如上图所示

**题目3**



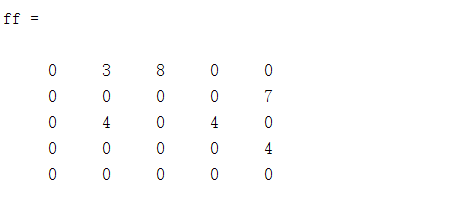
最小生成树长度为78

**题目4**

****

最短路径为40

**题目5**

****

求得最大流为11，最小费用为55

**五、实验代码**

**题目1**

|  |
| --- |
| %1  %非赋权图  E = [1,2;1,3;1,4;2,3;2,6;3,4;4,5;5,6];  s = E(:,1);  t = E(:,2);  nodes = cellstr(strcat("v",int2str([1:6]')));  G = graph(s,t,[],nodes);  plot (G,'Layout','circle')  %赋权图  weights = [7,3,12,1,1,8,9,3];  G2 = graph(s,t,weights,nodes);  plot(G2, 'EdgeLabel', G2.Edges.Weight,"Layout","circle")  %有向图  E2 = [2,1;1,3;4,1;2,3;6,2;3,4;5,4;5,6];  s2 = E(:,1);  t2 = E(:,2);  G2 = digraph(s,t,weights,nodes);  plot(G2, 'EdgeLabel', G2.Edges.Weight,"Layout","circle") |

**题目2**

|  |
| --- |
| %2  clear  L = {'A','B1',2;'A','B2',4;'B1','C1',3;'B1','C2',3;'B1','C3',1;'B2','C1',2;  'B2','C2',3;'B2','C3',1;'C1','D',1;'C2','D',3;'C3','D',4};  G = digraph(L(:,1),L(:,2),cell2mat(L(:,3)));  p = plot(G,'EdgeLabel',G.Edges.Weight);  [path,d] = shortestpath (G,'A','D','Method','positive')  highlight(p,path,'EdgeColor','r') |

**题目3**

|  |
| --- |
| %3  clear  E = [1,2;1,5;2,3;2,4;2,5;3,4;3,5;4,5;4,6;5,6];  s = E(:,1);  t = E(:,2);  weights = [20,15,20,60,25,30,18,35,10,15];  nodes = cellstr(strcat("v",int2str([1:6]')));  G = graph(s,t,weights,nodes);  p = plot(G, 'EdgeLabel', G.Edges.Weight);  T = minspantree(G);L=sum(T.Edges.Weight)  p1 = highlight(p,T) |

**题目4**

|  |
| --- |
| %4  clear  E = [1,2;1,5;2,3;2,4;2,5;3,4;3,5;4,5;4,6;5,6];  s = E(:,1);  t = E(:,2);  weights = [20,15,20,60,25,30,18,35,10,15];  nodes = cellstr(strcat("v",int2str([1:6]')));  G = graph(s,t,weights,nodes);  p = plot(G, 'EdgeLabel', G.Edges.Weight);  [path,d] = shortestpath (G,'v1','v4','Method','positive')  highlight(p,path,'EdgeColor','r') |

**题目5**

|  |
| --- |
| %5  clear  L = {'vs','v1',4,10;'vs','v2',1,8;'v1','v3',6,2;'v1','vt',1,7;'v2','v1',2,5;  'v2','v3',3,10;'v3','vt',2,4};  G1 =digraph(L(:,1),L(:,2),cell2mat(L(:,3)));  G2 =digraph(L(:,1),L(:,2),cell2mat(L(:,4)));  [M,F] = maxflow (G2,'vs' ,'vt')  b = full (adjacency (G1 ,'weighted'));%导出费用邻接矩阵  c= full (adjacency (G2 ,'weighted'));%导出容量邻接矩阵  f = optimvar('f',5 ,5,'LowerBound',0);  prob = optimproblem;prob.Objective = sum(sum(b.\*f));  con1 = [sum(f(1,:))==M  sum(f(:,[2:end-1]))'==sum(f([2:end-1],:),2)  sum(f(:,end))==M];  prob.Constraints.con1 = con1 ;  prob.Constraints.con2 = f<=c;  [sol,fval] = solve (prob)  ff = sol.f %显示最小费用最大流对应的矩阵 |

**六、实验总结**

**七、参考文献**

**八、教师评语**