



网络优化算法：Floyd算法



《美国数学建模竞赛》

完整课程请长按下方二维码





4.2.2 Floyd算法

- Floyd算法可以一次性求出所有点之间最短距离的算法。
- Floyd算法的基本思想是：对于有 n 个顶点的图，每次插入一个顶点 v ，然后将始点到终点的当前最短路径与插入顶点 v 作为中间点的最短路径作比较，取较小值以得到新的距离矩阵，如此循环迭代下去，递推产生一个矩阵序列 $D_0, D_1, \dots, D_k, \dots, D_n$ ，其中 $D_k(i, j)$ 表示从顶点 v_i 到顶点 v_j 的路径上所经过的顶点序号不大于 k 的最短路径长度，最后 D_n 就是任意两点之间的最短距离矩阵。



- Floyd算法步骤如下：
- ① 赋初值. 对所有 i, j , $d_{ij}=a_{ij}$, $r_{ij}=j$, $k=1$ 。 转向②
- ② 更新 d_{ij}, r_{ij} , 对所有 i, j , 若 $d_{ik}+d_{kj}<d_{ij}$, 则令 $d_{ij}=d_{ik}+d_{kj}$, $r_{ij}=r_{kj}$ 转向③, .
- ③ 终止判断. 若 $d_{ii}<0$, 则存在一条含有顶点 v_i 的负回路, 终止; 或者 $k=n$ 终止; 否则令 $k=k+1$, 转向②.
- 其中最短路线可由 r_{ij} 得到.



- 该算法的适用条件和范围：
- *a)* 所有成对点的最短路径；
- *b)* 稠密图效果最佳；

Floyd最短路算法的MATLAB函数文件floyd.m:

```
function [D,R]= floyd (A)
```

```
%采用floyd算法计算图中任意两点之间最短路程，可以有负权。  
%参数D为连通图的权矩阵, R是路径矩阵
```



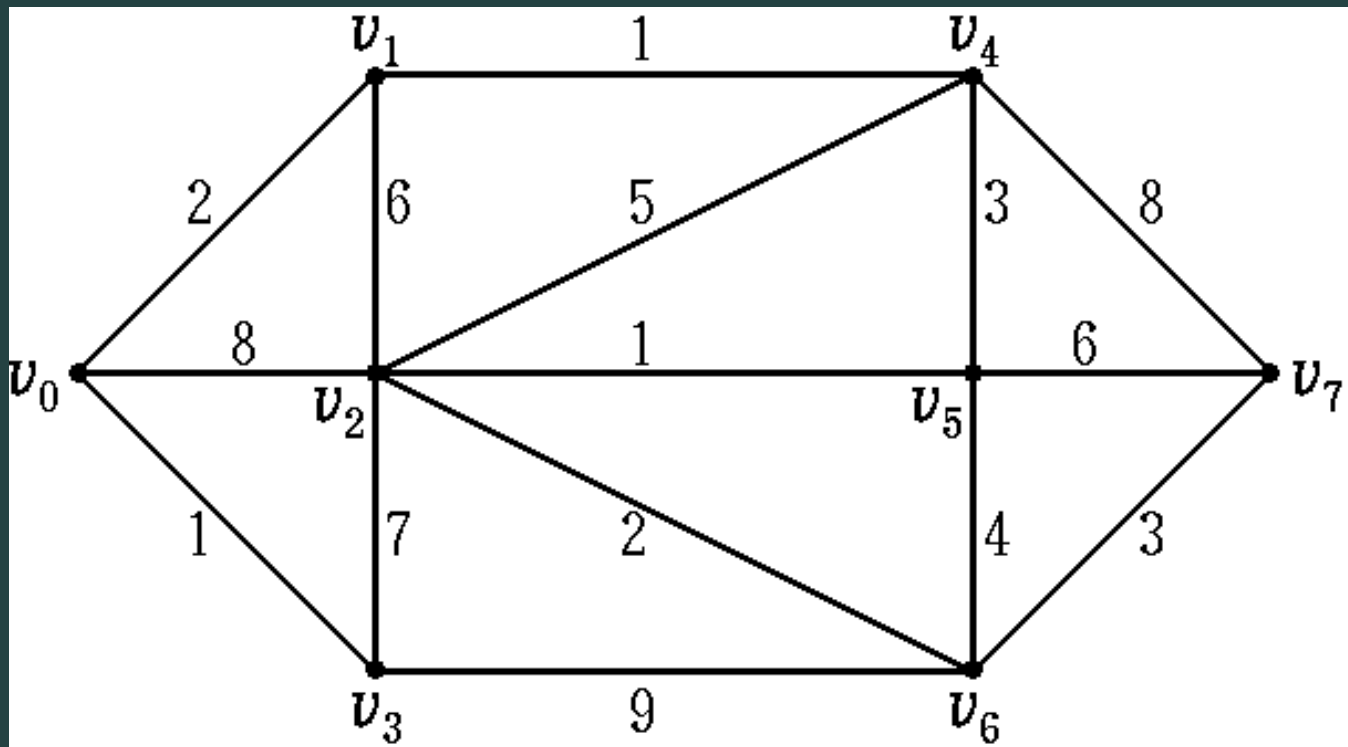
Floyd算法程序

- function [D,R]= floyd (A)
- %采用floyd算法计算图中任意两点之间最短路程，可以有负权。
- %参数D为连通图的权矩阵，R是路径矩阵
- D=A; n=length(D); %赋初值
- for(i=1:n)
- for(j=1:n)
- R(i,j)=j;
- end;
- end %赋路径初值
- for (k=1:n)
- for(i=1:n)
- for (j=1:n)
- if (D(i,k)+D(k,j)<D(i,j))
- D(i,j)=D(i,k)+D(k,j); %更新dij，说明通过k的路程更短
- R(i,j)=R(k,j);
- end;
- end;
- end %更新rij，需要通过k

- pd=0;
- for i=1:n %含有负权时
- if (D(i,i)<0)
- pd=1;
- break;
- end;
- end %跳出内层的for循环 存在一条含有顶点vi的负回路
- if (pd==1)
- fprintf('有负回路');
- break;
- end %存在一条负回路, 跳出最外层循环 终止程序
- end %程序结束



例 求下图中任意两点间的最短路.





- 解:在主程序窗口输入

- >> A=[0 2 8 1 Inf Inf Inf Inf
- 2 0 6 Inf 1 Inf Inf Inf
- 8 6 0 7 5 1 2 Inf
- 1 Inf 7 0 Inf Inf 9 Inf
- Inf 1 5 Inf 0 3 Inf 8
- Inf Inf 1 Inf 3 0 4 6
- Inf Inf 2 9 Inf 4 0 3
- Inf Inf Inf Inf 8 6 3 0];
- >> [D,R]=floyd(A)



MATLAB R2016a

主 页 绘图 应用程序 编辑器 发布 视图

新建 打开 保存 比较 转至 注释 断点 运行 运行并前进 运行并计时

文件 导航 编辑 断点 运行

当前文件夹: D:\matlab\bin

名称

- zhuij.m
- worker.bat
- process3_1.m
- path.m
- mw_mpiexec.bat
- mexutils.pm
- mexsetup.pm
- mexext.bat
- mex.pl
- mex.bat
- mcc.bat
- mbuild.bat
- matlab.exe
- lcldata_utf8.xml
- lcldata.xsd
- lcldata.xml
- he.m
- floyd.m
- dijkstra.m
- deploytool.bat
- win64
- util
- registry
- ms2imgconverter

详细信息

选择文件以查看详细信息

命令行窗口

```

2      6

>> A=[ 0  2  8  1  Inf Inf Inf Inf
       2  0  6  Inf 1  Inf Inf Inf
       8  6  0  7  5  1  2  Inf
       1  Inf 7  0  Inf Inf 9  Inf
       Inf 1  5  Inf 0  3  Inf 8
       Inf Inf 1  Inf 3  0  4  6
       Inf Inf 2  9  Inf 4  0  3
       Inf Inf Inf Inf 8  6  3  0];

[D,R]=floyd(A)
    
```

fx 编辑器 - D:\matlab\bin\floyd.m

dijkstra.m path.m floyd.m +

```

1 function [D,R]= floyd (A)
2 %采用floyd算法计算图中任意两点之间最短路程，可以有负权。
3 %参数D为连通图的权矩阵，R是路径矩阵
4 D=A; n=length(D); %赋初值
5 for(i=1:n)
6 for(j=1:n)
7 R(i,j)=j;
8 end;
9 end %赋路径初值
10 for(k=1:n)
11 for(i=1:n)
12 for(j=1:n)
13 if (D(i,k)+D(k,j)<D(i,j))
14 D(i,j)=D(i,k)+D(k,j); %更新dij, 说明通过k的路程更短
15 R(i,j)=R(k,j);
16 end;
17 end;
    
```

工作区

名称

- A
- ans
- D
- i
- R
- t

命令历史记录

```

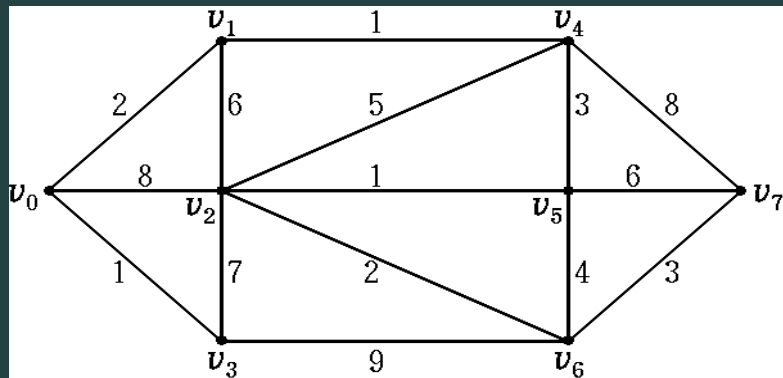
path(t,1,6)
path(t,1,5)
path(t,1,2)
path(t,1,3)
path(t,2,6)
A=[ 0281...
206Inf1Inf...
8607512Inf
1Inf70Inf1...
Inf15Inf03...
InfInf1Inf...
InfInf29In...
InfInfInfI...
[D,R]=floy...
    
```

行 32 列 1

16:39
2020/1/31



- 其中矩阵D中的 $D(i, j)$ 为 $i \rightarrow j$ 的最短距离，矩阵R中的 $R(i, j)$ 为 $i \rightarrow j$ 的最短路中j的负顶点，所以需要逆向追踪找到完整的最短路径。



D =

0	2	7	1	3	6	9	11
2	0	5	3	1	4	7	9
7	5	0	7	4	1	2	5
1	3	7	0	4	7	9	12
3	1	4	4	0	3	6	8
6	4	1	7	3	0	3	6
9	7	2	9	6	3	0	3
11	9	5	12	8	6	3	0

R =

1	1	6	1	2	5	3	5
2	2	6	1	2	5	3	5
2	5	3	3	6	3	3	7
4	1	4	4	2	5	4	5
2	5	6	1	5	5	3	5
2	5	6	1	6	6	3	6
2	5	7	7	6	3	7	7
2	5	7	1	8	8	8	8



- **例** 某城市要建立一个消防站，位于该城市所属的七个区服务之一（如下图）。问消防站应设在那个区，才能使它至最远区的路径最短。





- 分析与建模：上图是一个赋权无向图，顶点表示区，两顶点之间的边表示连接两区之间的道路，边上的权表示两区之间的距离。问题转化为从图中确定顶点 v_i ，使到距其最远顶点的路径最短。根据图的最短路算法，可设计算法如下

- 算法：

- (1) 用Floyd算法求出距离矩阵 $D = (d_{ij})_{v \times v}$ 。
- (2) 计算在各点 v_i 设立服务设施的最大服务距离 $S(v_i)$ ：
$$S(v_i) = \max_{1 \leq j \leq v} \{d_{ij}\}, i = 1, 2, \dots, v$$

- (3) 求出顶点 v_k ，使 $S(v_k) = \min_{1 \leq i \leq v} \{S(v_{ij})\}$ ，则 v_k 就是要求的建立消防站的地
点，此点称为图的中心点。



- 对此问题，用MATLAB编程求解如下：
- $a = [0 \ 3 \ \text{inf} \ \text{inf} \ \text{inf} \ \text{inf} \ \text{inf}]$
- $\begin{matrix} 3 & 0 & 2 & \text{inf} & 18 & 2.5 & \text{inf} \end{matrix}$
- $\begin{matrix} \text{inf} & 2 & 0 & 6 & 2 & \text{inf} & \text{inf} \end{matrix}$
- $\begin{matrix} \text{inf} & \text{inf} & 6 & 0 & 3 & \text{inf} & \text{inf} \end{matrix}$
- $\begin{matrix} \text{inf} & 18 & 2 & 3 & 0 & 4 & \text{inf} \end{matrix}$
- $\begin{matrix} \text{inf} & 2.5 & \text{inf} & \text{inf} & 4 & 0 & 1.5 \end{matrix}$
- $\begin{matrix} \text{inf} & \text{inf} & \text{inf} & \text{inf} & \text{inf} & 1.5 & 0 \end{matrix}$
- $\gg D = \text{floyd}(a)$



• $D =$

• 0	3.0000	5.0000	10.0000	7.0000	5.5000	7.0000
• 3.0000	0	2.0000	7.0000	4.0000	2.5000	4.0000
• 5.0000	2.0000	0	5.0000	2.0000	4.5000	6.0000
• 10.0000	7.0000	5.0000	0	3.0000	7.0000	8.5000
• 7.0000	4.0000	2.0000	3.0000	0	4.0000	5.5000
• 5.5000	2.5000	4.5000	7.0000	4.0000	0	1.500
• 7.0000	4.0000	6.0000	8.5000	5.5000	1.5000	0

$$S(v_1) = 10, S(v_2) = 7, S(v_3) = 6, S(v_4) = 8.5, S(v_5) = 7, S(v_6) = 7, S(v_7) = 8.5$$

因为 $S(v_3) = 6$ 最小，故应将消防站设在 v_3 处。