

Mathematical Experiments

图论算法

——最短路径应用案例



重庆大学数学与统计学院



A

案例1：人狼羊菜渡河问题

B

案例2：最短运输路线问题

C

Dijkstra算法的MATLAB实现



案例1 人狼羊菜渡河问题

Mathematical Laboratory

摆渡人 **Ferryman**

狼 **Wolf**

羊 **Sheep**

菜 **Vegatable**





南岸状态 : $C_4^4 + C_4^3 + C_4^2 + C_4^1 + C_4^0 = 2^4 = 16$ **种** ,

其中WS, SV, WSV, 从而FV, FW, F为**不允许状态**

10种允许状态 :

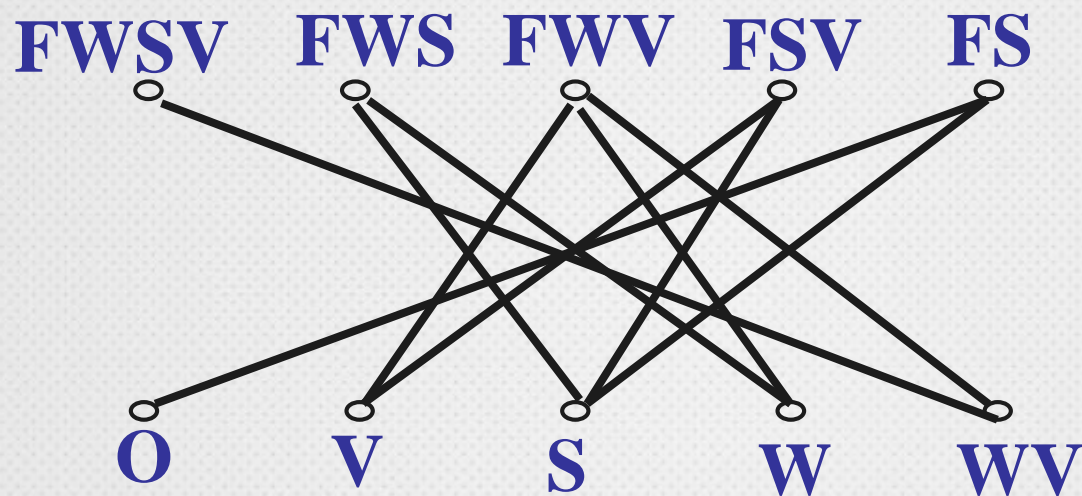
FWSV	FWS	FWCV	FSV	FS
O	V	S	W	WV



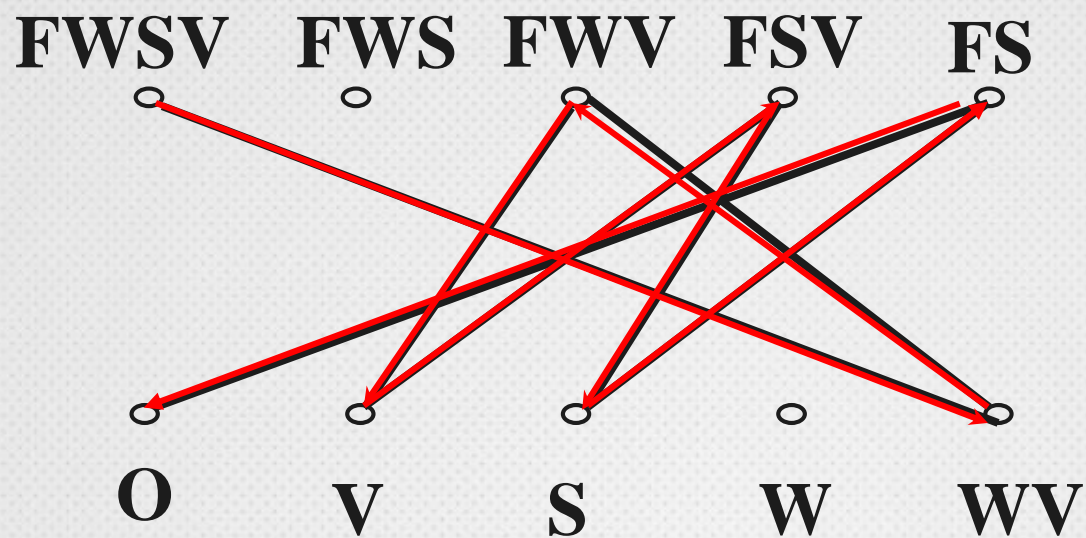
建立图G的模型

顶点：南岸的允许状态，共10个顶点，

边：当且仅当两个状态可以通过一次摆渡互相转化时，就连一边。



原问题  求初始状态顶点“ FWSV” 到末状态顶点“ O” 之间的最短路径问题.



渡河方案：对应于图中从顶点“FWSV”到顶点“O”的**路径**。

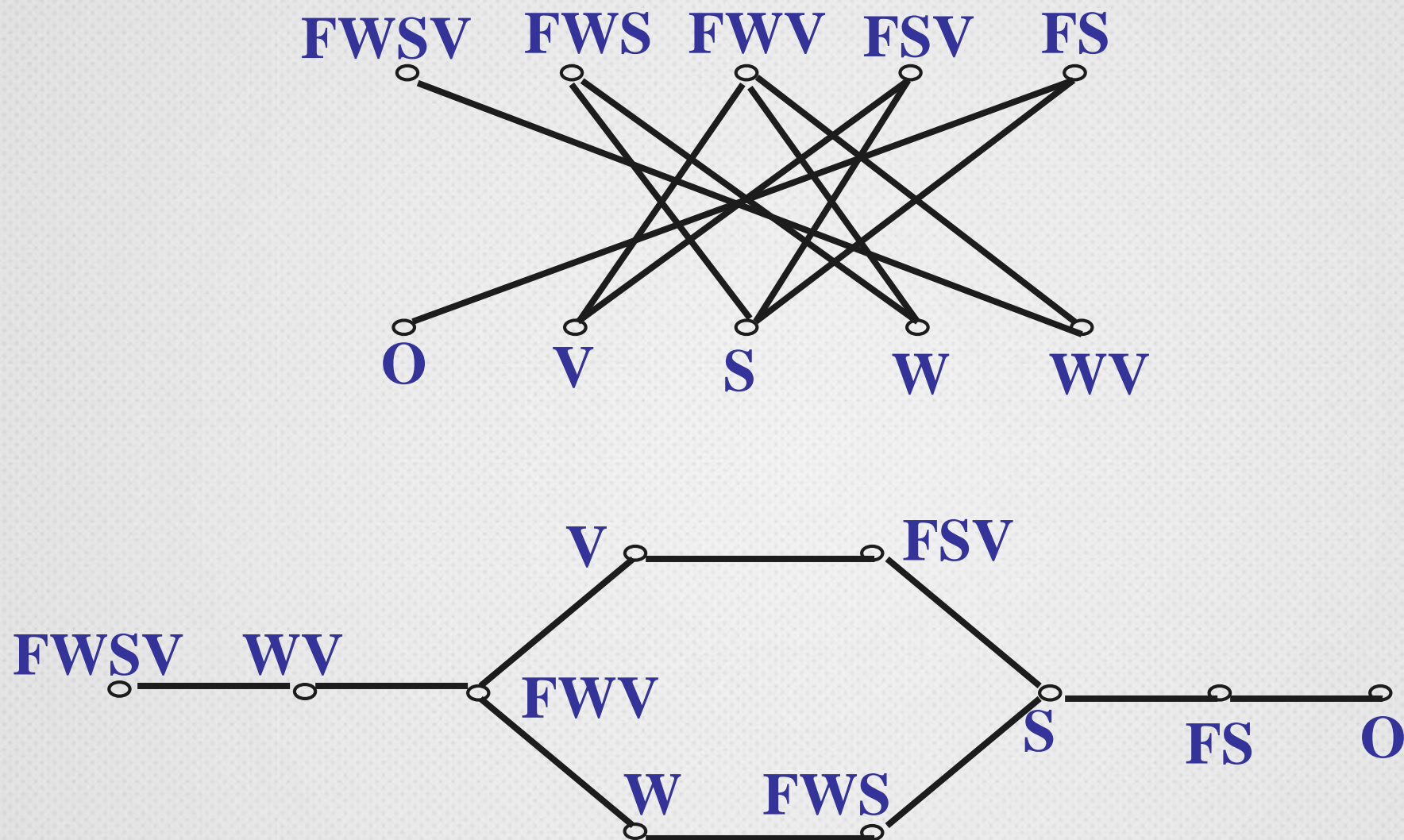


该方案是否是最优的渡河方案？



案例1 人狼羊菜渡河问题

Mathematical Laboratory

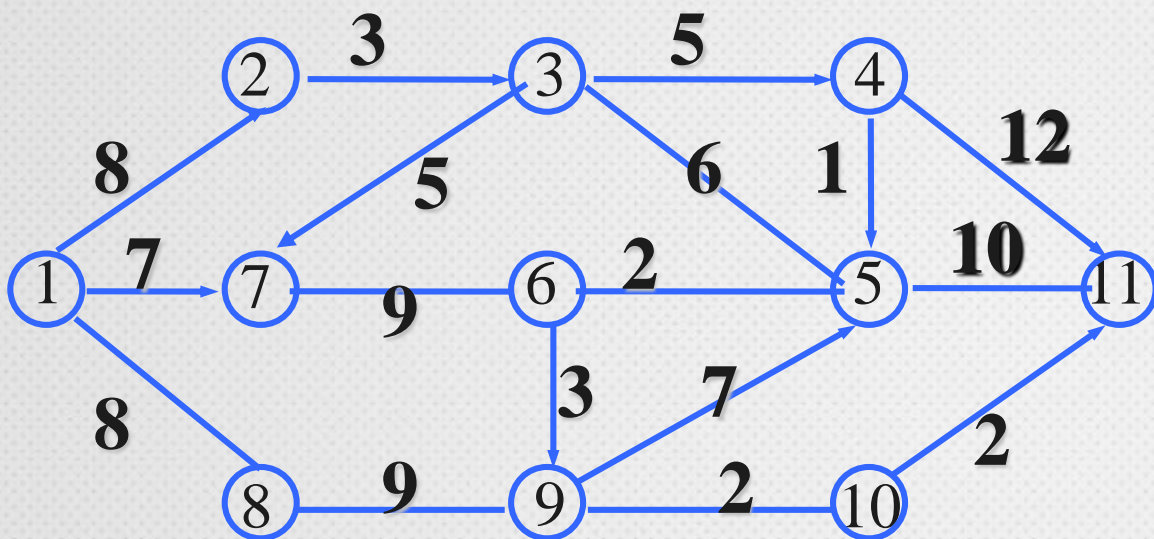




边：道路； **顶点**：道路交汇处

边权：行驶时间

有向边：单行道； **无向边**：可双向行驶。



若有一批货物要从1号顶点运往11号顶点，问运货车应沿哪条线路行驶，才能最快地到达目的地？



```
function [min,path]=dijkstra(w,start,terminal)
n=size(w,1); label(start)=0; f(start)=start;
for i=1:n
    if i~=start
        label(i)=inf;
    end, end
s(1)=start; u=start;
while length(s)<n
    for i=1:n
        ① ins=0;
        for j=1:length(s)
            if i==s(j)
                ins=1;
            end, end
            if ins==0
                v=i;
                if label(v)>(label(u)+w(u,v))
                    label(v)=(label(u)+w(u,v)); f(v)=u;
                end, end, end
            end, end, end
        end
    end
end, end, end
```

```
min=label(terminal);
path(1)=terminal;
i=1;
while path(i)~=start
    path(i+1)=f(path(i));
    i=i+1 ;
end
path(i)=start;
L=length(path);
path=path(L:-1:1);
③
```

```
v1=0;
k=inf;
for i=1:n
    ② ins=0;
    for j=1:length(s)
        if i==s(j)
            ins=1;
        end, end
        if ins==0
            v=i;
            if k>label(v)
                k=label(v); v1=v;
            end, end, end
        end, end, end
        s(length(s)+1)=v1;
        u=v1;
    end
end
```



Dijkstra算法程序的使用说明：

调用格式为

$[min, path] = \text{dijkstra}(w, start, terminal),$

w：图的**带权邻接矩阵**，

start：最短路径的**起点号码**

terminal：最短路径的**终点号码**。

path：最短路径; **min**：最短路径的**长度**。

注意：顶点的编号从1开始连续编号。



案例2：最短运输路线问题

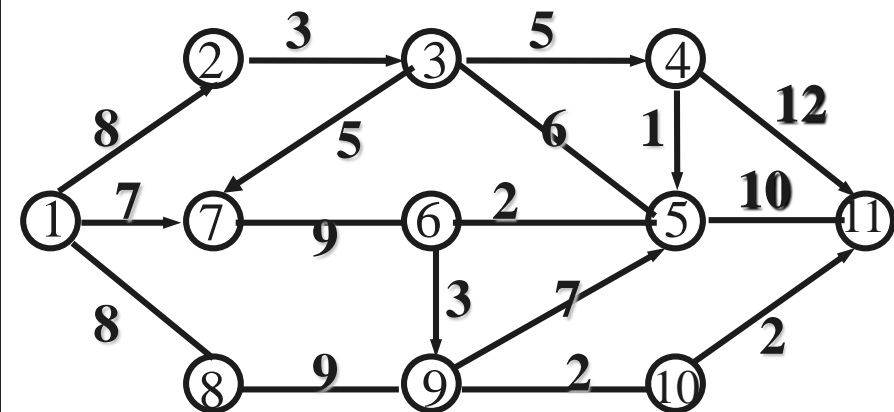
Mathematical Laboratory

```
edge= [ 2,3,1,3,3,5,4, 4,1,7,6,6,5, 5,11, 1,8,6,9,10,8,9, 9,10;...  
        3,4,2,7,5,3,5,11,7,6,7,5,6,11, 5, 8,1,9,5,11,9,8,10,9;...  
        3,5,8,5,6,6,1,12,7,9,9,2,2,10,10,8,8,3,7, 2, 9,9, 2, 2];
```

边
权
矩
阵

求带权邻接矩阵

```
n=11; weight=inf*ones(n, n);  
for i=1:n  
    weight(i, i)=0;  
end  
for i=1:size(edge,2)  
    weight(edge(1, i), edge(2, i))=edge(3, i);  
end  
[dis, path]=dijkstra(weight, 1, 11)
```





运行上页程序输出：

dis =

21

path =

1 8 9 10 11

因此顶点1到顶点11的**最短路径为1→8→9→10→11, 其长度为21。**



另一种求法

MATLAB的生物信息工具箱中也有求最短路径的函数

graphshortestpath

调用格式为

$[min, path, pred] = \text{graphshortestpath}(G, start, terminal),$

G : 图的 $n \times n$ 稀疏矩阵，非零元素为边权

start : 最短路径的起点号码; **terminal**: 最短路径的终点号码。

min:最短路径的长度；**path**: 最短路径。

pred: 各顶点的父亲点。



构建
稀疏
矩阵

```
G=sparse( [ 2,3,1,3,3,5,4, 4,1,7,6,6,5, 5,11, 1,8,6,9,10,8,9, 9,10],...  
          [3,4,2,7,5,3,5,11,7,6,7,5,6,11, 5, 8,1,9,5,11,9,8,10,9],...  
          [3,5,8,5,6,6,1,12,7,9,9,2,2,10,10,8,8,3,7, 2, 9,9, 2, 2]);
```

```
[dist, path, pred]=graphshortestpath(G,1,11)
```

```
h = view(biograph(G,[],'ShowWeights','on'))
```

```
set(h.Nodes(path),'Color',[1 0.4 0.4])
```

```
edges = getedgesbynodeid(h,get(h.Nodes(path),'ID'));
```

```
set(edges,'LineColor',[1 0 0])
```

```
set(edges,'LineWidth',1.5)
```

作
网
络
图



运行上页的程序得到输出

dist=

21,

path =

1 8 9 10 11

pred =

0 1 2 3 3 7 1 1 8 9 10

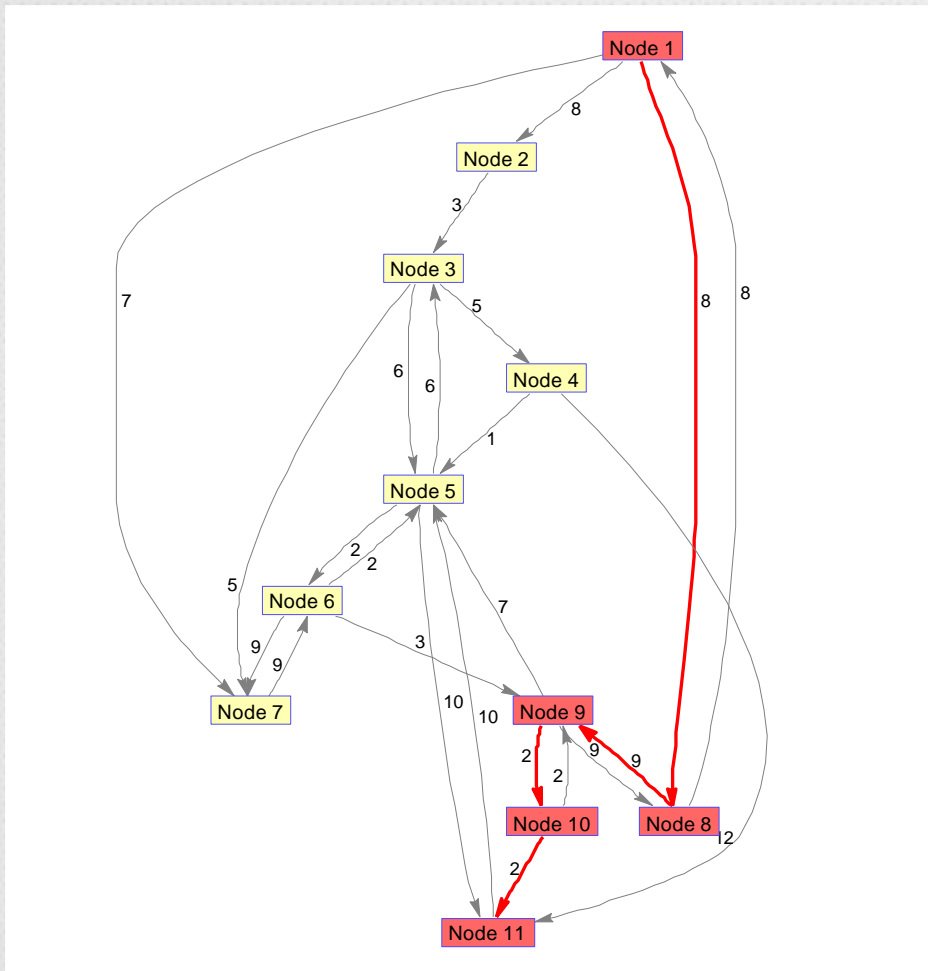
Biograph object with 11 nodes and 24 edges.

前两个输出与我们自编的程序dijkstra一样，多了一个第3输出，父亲点。

另外，还给出信息“生物图形对象具有11个顶点，24条边”。



另外上述MATLAB程序作出的网络图为





假设一个城市里有地铁，公交车，如果我们要求一个城市里两个站点之间转乘次数最少（即最方便）的乘坐地铁或公交车的乘坐方式，**如何构建图的模型 G ，使问题转化为求 G 的最短路径问题？**

如果公共交通的票价有分段计价和单一票价两种方式，又**该如何构建图的模型来求最小费用的乘车方式呢？**

Thanks



重庆大学数学与统计学院