Mathematical Experiments

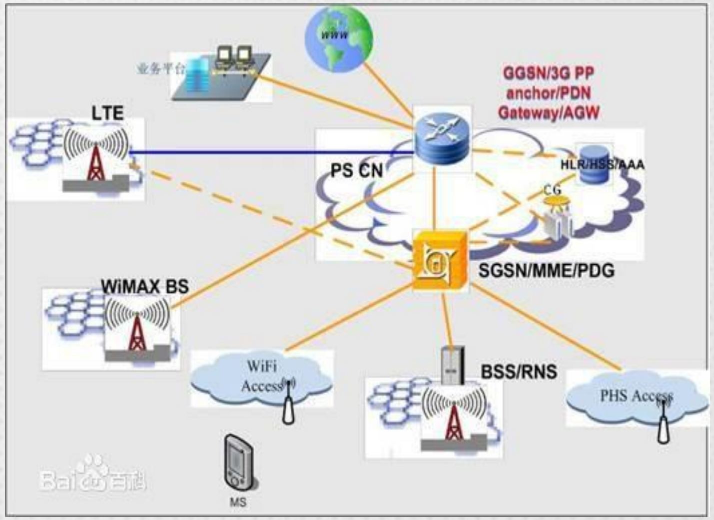
图论算法

——最短路径

重庆大学数学与统计学院

引言

哪条路最可靠?最快捷？
哪个生产方案成本最低？
哪个投资计划利润最大？



主要内容

A 基本概念

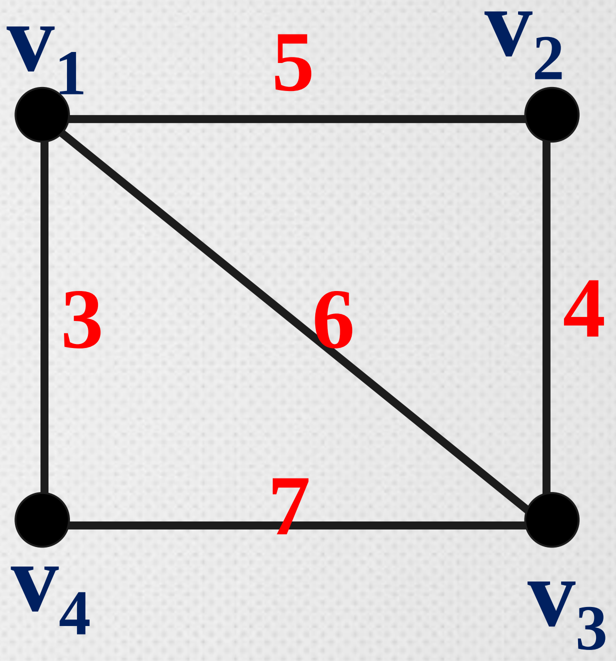
B Dijkstra算法

C 一个算例

定义：

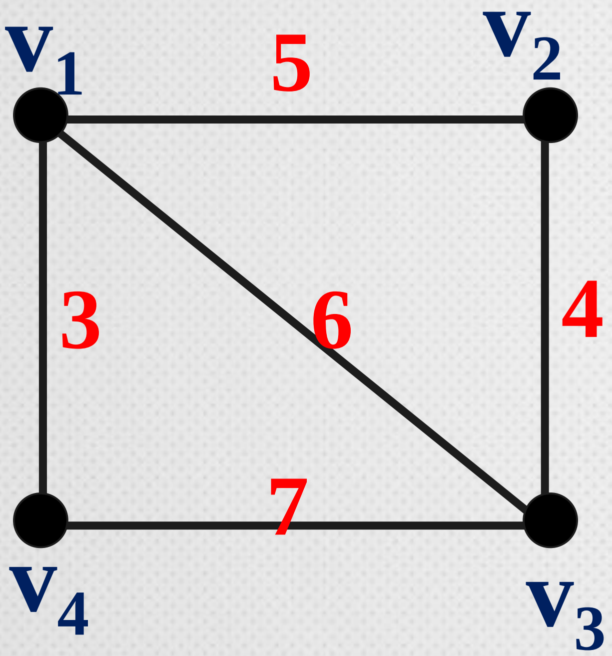
1) 设P(u,v)是加权图G中从u到v的路径, 则该路径上的边权之和称为该**路径的权或长度**, 记为**w(P)**。

2) 从u到v可能有多条路径，其中权最小的那条路径 P*(u,v)称为**u到v的最短路径**, 该最短路径的权称为**u到v的距离**，记为**d(u,v)**。



计算机如何表示图?

加权图的带权邻接矩阵：$W=(w_{ij})\upsilon\times\upsilon$， 其中$w_{ij}$（$i\neq j$）是边$(v_i,v_j)$（若有这条边）上的权，若不存在边（$v_i,v_j$），则$w_{ij}$为无穷大， W的对角线上的元素为0。



$$
\begin{array}{c}
 \\ V_1 \\ V_2 \\ V_3 \\ V_4
\end{array}
\begin{array}{c}
\begin{array}{cccc} V_1 & V_2 & V_3 & V_4 \end{array} \\
\begin{bmatrix}
0 & 5 & 6 & 3 \\
5 & 0 & 4 & \infty \\
6 & 4 & 0 & 7 \\
3 & \infty & 7 & 0
\end{bmatrix}
\end{array}
$$

Dijkstra算法（E.W.Dijkstra，1959）

使用范围:

1) 寻求从一固定顶点到其余各点的最短路径;

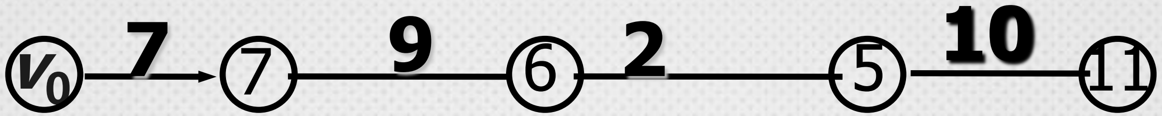
2) 有向图、无向图和混合图;

3) 权非负.

时间复杂度$O(n^2)$，是一个多项式时间算法，有效算法，

Dijkstra算法

迭代算法



依据： 最短路径上的任一子段也是最短路径。

算法思想：按与v_0的距离由近及远地逐个求出各顶点的最短路径和长度。

算法思路：设置一个集合S，存放已求出其最短路径长度的顶点。

1）$S \leftarrow \{v_0\}$；

2）求出$S' = V-S$ 中与 v_0 距离最近的顶点u，将u加入到S中；

3）重复2）直到S′为空集 。

Dijkstra算法

$l(v)$: v的标记，记录从v_0到v的当前最短路径长度;

$f(v)$: 该路径上v的前一个点（称为v的父亲点），用以确定最短路径。

算法步骤

输入加权图的带权邻接矩阵$w=[w(v_i,v_j)]_{n\times n}$.

1) **初始化**： 令$l(v_0)=0$, $\forall\ v\neq v_0$,$l(v)=\infty$; $S=\{v_0\}$，$u= v_0$;

2) **更新$l(v)$, $f(v)$**: 对S′ 中所有顶点v,

   a) 若 $l(v)>l(u)+w(u,v)$, 则 $l(v)\leftarrow l(u)+w(u,v)$, $f(v)\leftarrow u$;

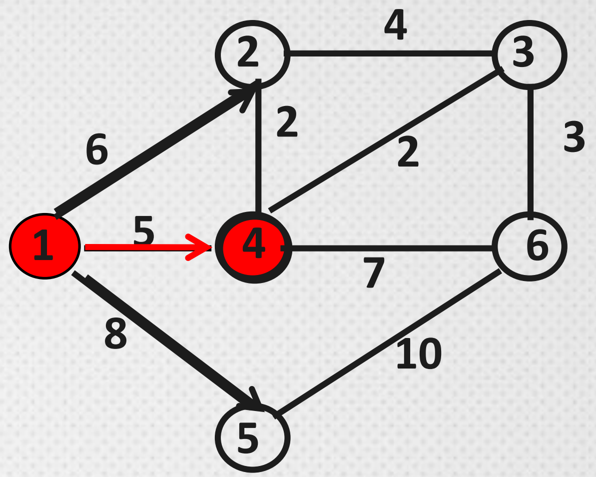
   b) 否则, $l(v)$, $f(v)$ 不变

3) **更新S, u**：在S′ 中找到使$l(u)$最小的顶点u，把u加入到S中；

4) **重复步骤2), 3),** 直到所有顶点都在S中为止.

一个算例

例：用Dijkstra**算法**求下图从1号顶点到6号顶点的最短路径。



解：初始化u=1，S={1}

v	1	2	3	4	5	6
$l(v)$	0	∞	∞	∞	∞	∞

第一次迭代：S′ ={2,3,4,5,6}中顶点的新标号由公式

$l(v)=\min\{l(v),l(u)+w(u,v)\}$确定，修正$l(v)$,并为$f(v)$赋值

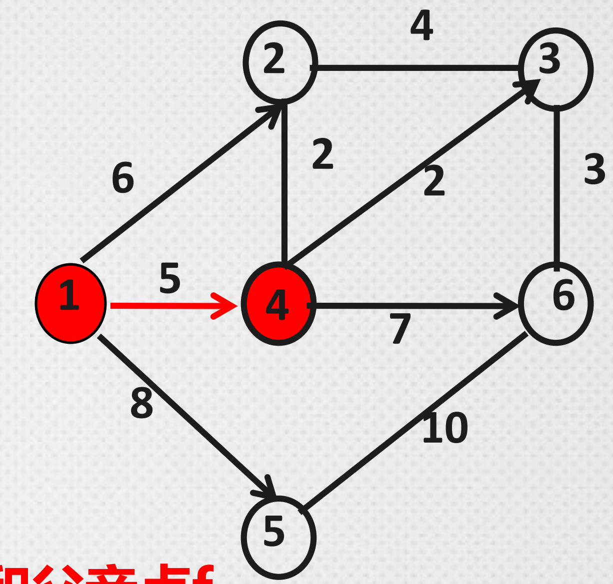
v	1	2	3	4	5	6
$l(v)$	0	6	∞	5	8	∞
$f(v)$		1		1	1	

*表中红色数字表示本次迭代更新了的标记l和父亲点f

u=4 $S \leftarrow S\cup\{u\}$, S={1,4}

第二次迭代：S′ ={2,3,5,6} 中顶点的新标号由公式 $l(v)=\min\{l(v),l(u)+w(u,v)\}$确定，修正$l(v)$,并为$f(v)$赋值

v	1	2	3	4	5	6
$l(v)$	0	6	7	5	8	12
$f(v)$		1	4	1	1	4



*表中红色数字表示本次迭代更新了的标记l和父亲点f

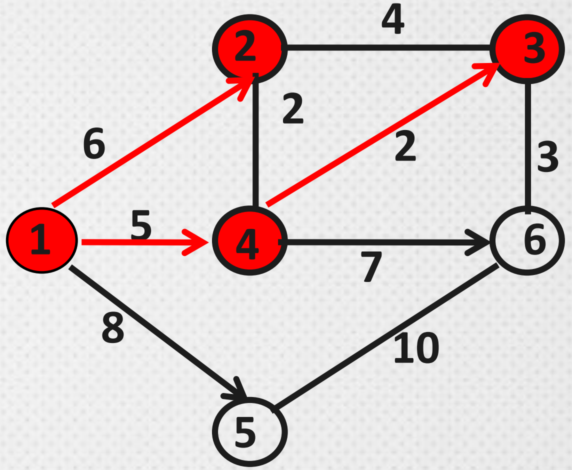
2号顶点是S′ 中标记最小的顶点

u=2, $S \leftarrow S \cup \{u\}$, S={1，4，2}

一个算例

第三次迭代：S′ ={3,5,6}中顶点的新标号由公式 $l(v)=\min\{l(v),l(u)+w(u,v)\}$确定，修正$l(v)$,并为$f(v)$赋值

v	1	2	3	4	5	6
$l(v)$	0	6	7	5	8	12
$f(v)$		1	4	1	1	4



3号顶点是S′ 中标记最小的顶点

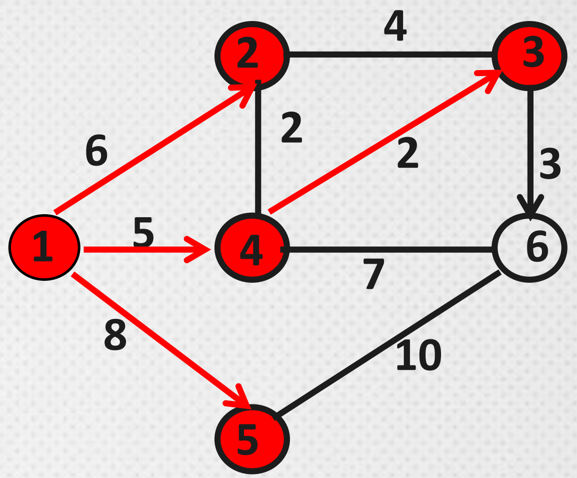
u=3,　　$S\leftarrow S\cup\{u\}$, S={1,4,2,3}

一个算例

第四次迭代：S′ ={5,6} 中顶点的新标号由公式

$l(v)=\min\{l(v),l(u)+w(u,v)\}$确定，修正$l(v)$,并为$f(v)$赋值

v	1	2	3	4	5	6
$l(v)$	0	6	7	5	8	10
$f(v)$		1	4	1	1	3



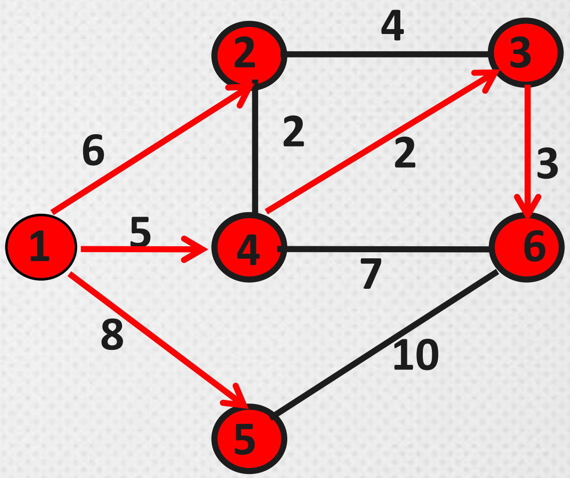
***表中红色数字表示本次迭代更新了的标记l和父亲点f**

$u=5$, $S\leftarrow S\cup\{u\}$, $S=\{1,4，2，3，5\}$

第五次迭代：S′ ={6}中顶点的新标号由公式

$l(v)=\min\{l(v), l(u)+w(u,v)\}$确定，修正$l(v)$,并为$f(v)$赋值

v	1	2	3	4	5	6
$l(v)$	0	6	7	5	8	10
$f(v)$		1	4	1	1	3



u=6, $S \leftarrow S \cup \{u\}$,

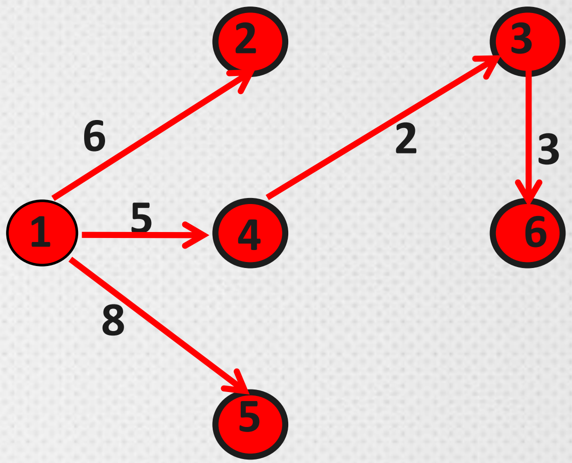
$S=\{1,4, 2, 3, 5, 6\}$

一个算例

例：用Dijkstra算法求下图从1号顶点到6号顶点的最短路径。

最终的标记和父亲点记录：

v	1	2	3	4	5	6
$l(v)$	0	6	7	5	8	10
$f(v)$		1	4	1	1	3

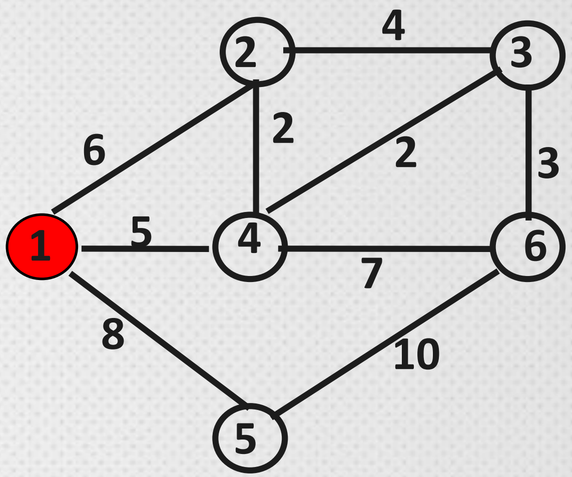


f(6)=3,
f(3)=4,
f(4)=1

从1号顶点到6号顶点的**最短路径**　$6 \leftarrow 3 \leftarrow 4 \leftarrow 1$

其**长度**为$l(6)=10$

例：用Dijkstra算法求下图从1号顶点到6号顶点的最短路径。



总结

每次迭代：S′ 中顶点的新标号由公式 $l(v)=\min\{l(v),l(u)+w(u,v)\}$ 确定，并为$f(v)$赋值, 然后找出S′ 中标记最小的顶点加入到S中。

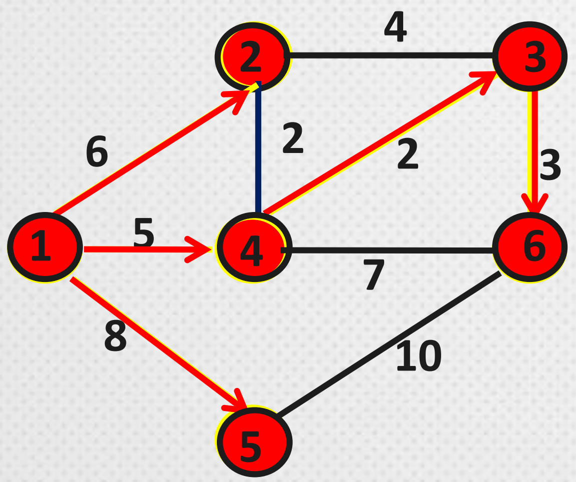
	S	S′	l(2)	l(3)	l(4)	l(5)	l(6)
1	1	2,3,4,5,6	6	∞	5	8	∞
2	1,4	2,3,5,6	6	7		8	12
3	1,4,2	3,5,6		7		8	12
4	1,4,2,3	5,6				8	10
5	1,4,2,3,5	6					10
6	1,4,2,3,5,6						

每次迭代： 求出一个离1号顶点最近的顶点u

l(u)称为**永久标记**

将u加入到S中.

永久父子关系（红色）的形成过程



结束语

- 如何去求**每对顶点的最短路径**及距离
- 如何去求两点之间**第2、第3、…、第k短**的路径
- 如何去求**带约束的最短路径**

    比如，包含一些指定顶点的最短路径

    包含一些指定边的最短路径

- 如何去求**路径权为其上边权的其他函数形式**时的最小（或最大权）路径

    如，路径权为其上边权之积。

Thanks

重庆大学数学与统计学院