离散数学概论

第五章 图的表示与应用

课程QQ号： **819392514**
金耀　软件工程系
fool1025@163.com
13857104418

知识回顾

- 无向图与有向图
- 握手定理
- 图的同构
- 通路与回路（3种）
- 连通性（3种）
- 割集（2类）

1.6 矩阵表示

1.7 路径

1.8 图的着色

1.9 匹配

§6 矩阵表示

一、　邻接矩阵

二、　可达矩阵

三、　关联矩阵

四、　连通性与矩阵关系

❖邻接矩阵

【定义】 $D=<V,E>$为**有向图**，顶点集$V=\{v_1,v_2,\ldots\ldots,v_n\}$，$V$中的结点按下标由小到大编序，构造n阶矩阵$A=(a_{ij})_{n\times n}$，其中：

$$a_{ij}=\begin{cases} \text{m, 若存在m条}v_i\text{到}v_j\text{直接相连的有向边} \\ \text{0, 若不存在}v_i\text{到}v_j\text{直接相连的有向边} \end{cases} \quad (i,j=1,2,\ldots\ldots,n)$$

则称A为有向图D的**邻接矩阵**，记为$A(D)$.

❖邻接矩阵

【定义】 $G=<V,E>$为无向图, 顶点集$V=\{v_1,v_2,\ldots\ldots,v_n\}$,V中的结点按下标由小到大编序, 构造n阶矩阵$A=(a_{ij})_{n\times n}$,其中：

$$a_{ij}=\begin{cases}1, & v_i\text{与}v_j\text{直接相连}\\ 0, & v_i\text{与}v_j\text{不直相连的有向边}\end{cases}\quad (i,j=1,2,\ldots\ldots,n)$$

则称A为有向图G的邻接矩阵, 记为$A(G)$.

邻接矩阵与结点编序有关:

同一个图形结点编序不同得到的邻接矩阵不同，但是表示的都是同一张图. 也就是说这些结点不同编序得到的图都是同构的, 同时它们的邻接矩阵也是相似的.

❖邻接矩阵的性质

(1) 零图的邻接矩阵的元素全为零, 并称它为**零矩阵**.

(2) 图的每一结点都有自回路而再无其他边时，则该图的邻接矩阵是**单位矩阵**.

(3) 简单图的邻接矩阵主对角元素全为零.

(4) 若设简单图D的邻接矩阵$A=(a_{ij})_{n\times n}$,则它的补图$\overline{G}$的邻接矩阵

$\overline{A}=(\overline{a}_{ij})_{n\times n}$为：

$$\overline{a}_{ij}=\begin{cases}1-a_{ij}, & i\neq j\\ 0, & i=j\end{cases} \qquad i,j=1,2,\dots,n$$

有向图的邻接矩阵

定义 设有向图D=<V,E>, $V=\{v_1, v_2, \ldots, v_n\}$, $E=\{e_1, e_2, \ldots, e_m\}$, 令$a_{ij}^{(1)}$为顶点$v_i$邻接到顶点$v_j$边的条数，称$(a_{ij}^{(1)})_{m\times n}$为**D的邻接矩阵**, 记作$A(D)$, 简记为$A$.

性质

(1) $\sum_{j=1}^{n} a_{ij}^{(1)} = d^{+}(v_i), \quad i = 1,2,\ldots,n$

(2) $\sum_{i=1}^{n} a_{ij}^{(1)} = d^{-}(v_j), \quad j = 1,2,\ldots,n$

(3) $\sum_{i,j} a_{ij}^{(1)} = m$ ---- D中长度为1的通路数

(4) $\sum_{i=1}^{n} a_{ii}^{(1)}$ ---- D中长度为1的回路数

邻接矩阵

➢ **定理：** 设无向图$G=<V, E>$， $V=\{v_1,v_2,\ldots,v_n\}$的邻接矩阵$A=(a_{ij})_{n\times n}$， 则

$$\deg(v_i)=\sum_{k=1}^{n}a_{ik}+a_{ii}=\sum_{k=1}^{n}a_{ki}+a_{ii}$$

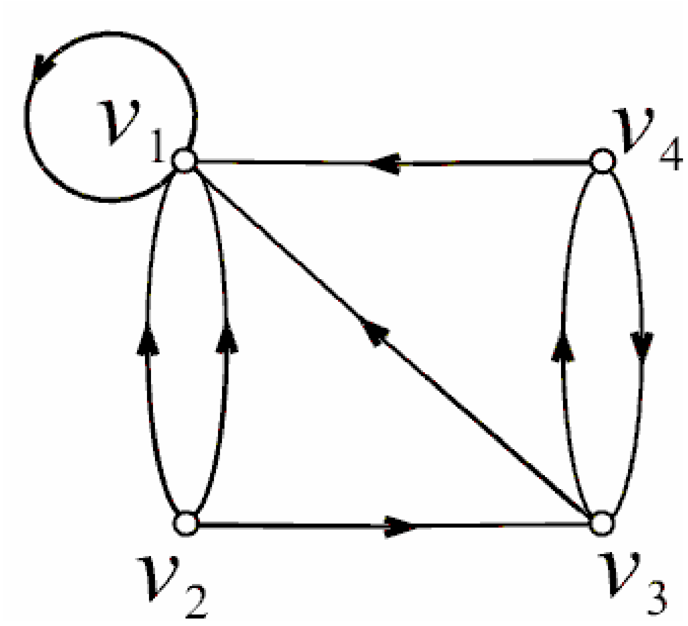
$$\sum_{i=1}^{n}\deg(v_i)=\sum_{i=1}^{n}\left(\sum_{k=1}^{n}a_{ik}+a_{ii}\right)=\sum_{i=1}^{n}\sum_{k=1}^{n}a_{ik}+\sum_{i=1}^{n}a_{ii}$$

设有向图$G=<V, E>$, $V=\{v_1,v_2,\ldots,v_n\}$的邻接矩阵$A=(a_{ij})_{n\times n}$, 则

$$\deg^{+}(v_i)=\sum_{k=1}^{n}a_{ik}\,,\quad \deg^{-}(v_i)=\sum_{k=1}^{n}a_{ki}$$

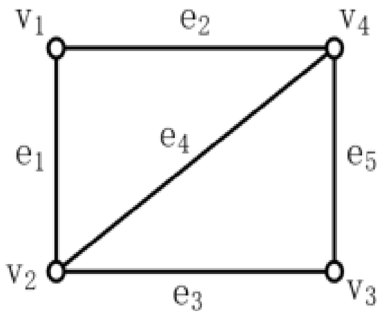
有向图的邻接矩阵实例

$$A=\begin{bmatrix}1&0&0&0\\2&0&1&0\\1&0&0&1\\1&0&1&0\end{bmatrix}$$



一. 邻接矩阵

例：求下图G的邻接矩阵A。



解：邻接矩阵A求解如下：

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

D中的通路及回路数

定理 设A为n阶有向图D的邻接矩阵, 则$A^l(l\geq 1)$中元素

$a_{ij}^{(l)}$为D中v_i到v_j长度为l的通路数,

$a_{ii}^{(l)}$为v_i到自身长度为l的回路数,

$\sum_{i=1}^{n}\sum_{j=1}^{n}a_{ij}^{(l)}$ 为D中长度为l的通路总数,

$\sum_{i=1}^{n}a_{ii}^{(l)}$ 为D中长度为l的回路总数。

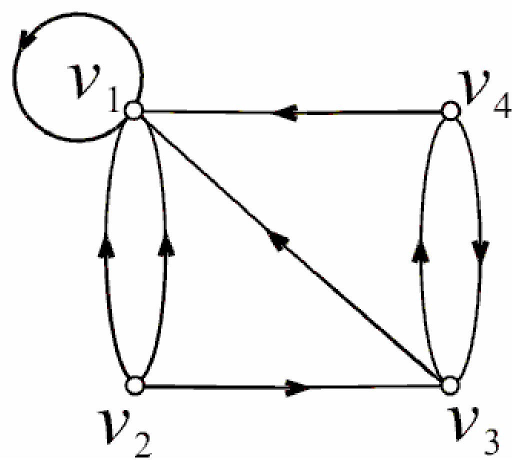
D中的通路及回路数(续)

推论 设$B_l=A+A^2+\ldots+A^l(l\geq 1)$, 则$B_l$中元素

$\sum_{i=1}^{n}\sum_{j=1}^{n} b_{ij}^{(l)}$为$D$中长度小于或等于$l$的通路数,

$\sum_{i=1}^{n} b_{ii}^{(l)}$为$D$中长度小于或等于$l$的回路数。

例 问在有向图D中



(1) 长度为1, 2, 3, 4的通路各有多少条？其中回路分别为多少条？

(2) 长度小于或等于4的通路为多少条？其中有多少条回路？

例(续)

$$A=\begin{bmatrix}1&0&0&0\\2&0&1&0\\1&0&0&1\\1&0&1&0\end{bmatrix}\quad A^2=\begin{bmatrix}1&0&0&0\\3&0&0&1\\2&0&1&0\\2&0&0&1\end{bmatrix}$$

$$A^3=\begin{bmatrix}1&0&0&0\\4&0&1&0\\3&0&0&1\\3&0&1&0\end{bmatrix}\quad A^4=\begin{bmatrix}1&0&0&0\\5&0&0&1\\4&0&1&0\\4&0&0&1\end{bmatrix}$$

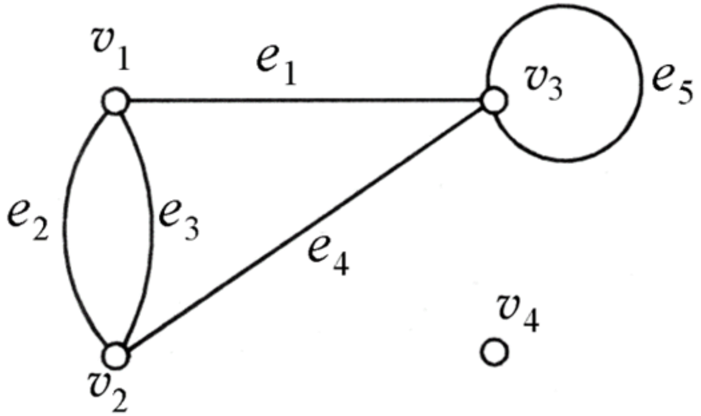
长度	通路	回路
1	8	1
2	11	3
3	14	1
4	17	3
合计	50	8

无向图的关联矩阵

定义 设无向图$G=<V,E>$, $V=\{v_1, v_2, \ldots, v_n\}$, $E=\{e_1, e_2, \ldots, e_m\}$, 令$m_{ij}$为$v_i$与$e_j$的关联次数， 称$(m_{ij})_{n\times m}$为**G的关联矩阵**， 记为$M(G)$.

例

$$M(G)=\begin{bmatrix}1&1&1&0&0\\0&1&1&1&0\\1&0&0&1&2\\0&0&0&0&0\end{bmatrix}$$



无向图的关联矩阵

定义 设无向图$G=<V,E>$, $V=\{v_1, v_2, \ldots, v_n\}$, $E=\{e_1, e_2, \ldots, e_m\}$, 令$m_{ij}$为$v_i$与$e_j$的关联次数, 称$(m_{ij})_{n\times m}$为$G$的关联矩阵, 记为$M(G)$.

性质 (1) 每一列恰好有两个1或一个2

(2) $\sum_{j=1}^{m} m_{ij} = d(v_i) \quad (i = 1,2,\ldots,n)$

(3) $\sum_{i,j} m_{ij} = 2m$

(4) v_i为孤立点当且仅当第i行全为0

(5) 平行边的列相同

有向图的关联矩阵

定义 设无环有向图$D=<V,E>$, $V=\{v_1, v_2, \ldots, v_n\}$, $E=\{e_1, e_2, \ldots, e_m\}$, 令

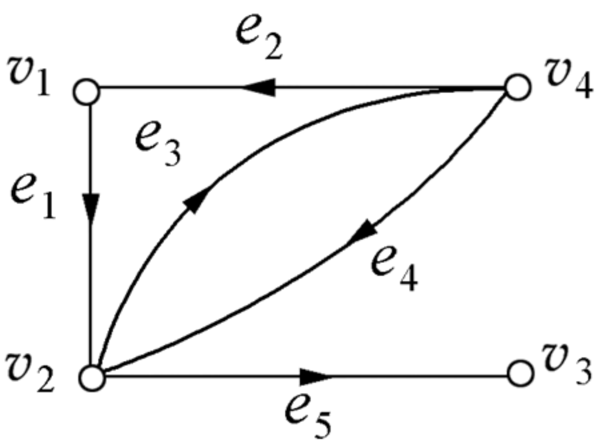
$$m_{ij}=\begin{cases}1, & v_i\text{为}e_j\text{的始点}\\ 0, & v_i\text{与}e_j\text{不关联}\\ -1, & v_i\text{为}e_j\text{的终点}\end{cases}$$

则称$(m_{ij})_{n\times m}$为**D的关联矩阵**，记为$M(D)$。

有向图的关联矩阵(续)

例

$$M(D)=\begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & -1 & 1 & 0 \end{bmatrix}$$



性质

(1) 每一列恰好有一个1和一个-1

(2) 第i行1 的个数等于$d^+(v_i)$, -1 的个数等于$d^-(v_i)$

(3) 1的总个数等于-1的总个数, 且都等于m

(4) 平行边对应的列相同

有向图的可达矩阵

定义 设$D=<V,E>$为有向图, $V=\{v_1, v_2, \ldots, v_n\}$, 令:

$$p_{ij}=\begin{cases}1, & v_i\text{可达}v_j\\ 0, & \text{否则}\end{cases}$$

称$(p_{ij})_{n\times n}$为**D的可达矩阵**, 记作$P(D)$, 简记为P.

性质:

$P(D)$主对角线上的元素全为1.

D强连通当且仅当$P(D)$的元素全为1.

二. 可达矩阵

从图G的邻接矩阵A可以得到可达矩阵P,即令$B_n = A^0+A^1+A^2+A^3+\ldots\ldots+A^{n-1}$,再把$B_n$中非零元素改为1，零元素不变，这种变换后的矩阵就是可达矩阵P。

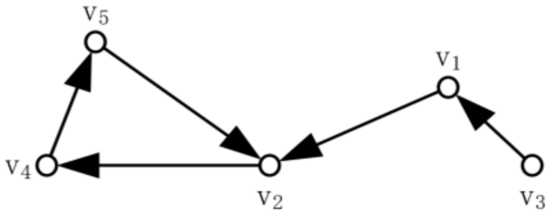
设$G=<V,E>$为线图，A、P分别是G的邻接矩阵和可达性矩阵，则有:

$$P = A^{(0)} \vee A^{(1)} \vee A^{(2)} \vee \cdots \vee A^{(n-1)} = \bigvee_{i=0}^{n-1} A^{(i)}$$

这里，$A^{(i)}$表示i个A进行布尔乘法。

二. 可达矩阵

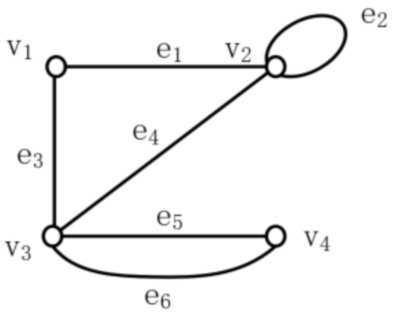
例： 求下图的邻接矩阵和可达性矩阵。



解： 邻接矩阵 $A=\begin{bmatrix}0&1&0&0&0\\0&0&0&1&0\\1&0&0&0&0\\0&0&0&0&1\\0&1&0&0&0\end{bmatrix}$， **可达性矩阵为** $A^2=\begin{bmatrix}0&0&0&1&0\\0&0&0&0&1\\0&1&0&0&0\\0&1&0&0&0\\0&0&0&1&0\end{bmatrix}$， $A^3=\begin{bmatrix}0&0&0&0&1\\0&1&0&0&0\\0&0&0&1&0\\0&0&0&1&0\\0&0&0&0&1\end{bmatrix}$

$A^4=\begin{bmatrix}0&1&0&0&0\\0&0&0&1&0\\0&0&0&0&1\\0&0&0&0&1\\0&1&0&0&0\end{bmatrix}$， $P=E\vee A\vee A^2\vee A^3\vee A^4=\begin{bmatrix}1&1&0&1&1\\0&1&0&1&1\\1&1&1&1&1\\0&1&0&1&1\\0&1&0&1&1\end{bmatrix}$。

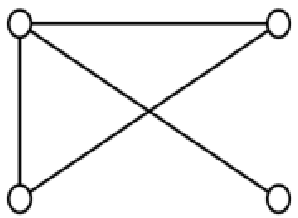
例：求下面多重图的邻接矩阵A和关联矩阵M。



解：$A=\begin{bmatrix}0&1&1&0\\1&1&1&0\\1&1&0&2\\0&0&2&0\end{bmatrix}$； $M=\begin{bmatrix}1&0&1&0&0&0\\1&2&0&1&0&0\\0&0&1&1&1&1\\0&0&0&0&1&1\end{bmatrix}$

三. 可达矩阵

例：求下图G的邻接矩阵A和关联矩阵M。



解：这两个矩阵与顶点和边的排列次序有关。

一种排列次序得到下面的矩阵：

$$A=\begin{bmatrix}0&1&1&1\\1&0&0&1\\1&0&0&0\\1&1&1&0\end{bmatrix};\qquad M=\begin{bmatrix}1&1&1&0\\1&0&0&1\\0&1&0&0\\0&0&1&1\end{bmatrix}$$

四. 连通性与矩阵关系

❖ 连通性与矩阵关系：

- 无向线图G是连通图当且仅当它的可达性矩阵P的所有元素都均为1;
- 有向线图D是强连通图当且仅当它的可达性矩阵P的所有元素都均为1;
- 有向线图D是单向连通图当且仅当它的可达性矩阵P及其转置矩阵P^T经布尔运算加后所得矩阵$P' = P \vee P^T$中除对主角元外的其余元素均为1;
- 有向线图D是弱连通图当且仅当它的邻接矩阵A及其转置矩阵A^T经布尔加运算后所得矩阵$B = A \vee A^T$作为邻接矩阵而求出的可达性矩阵P'中所有元素均为1.

1.6 矩阵表示

1.7 路径

1.8 图的着色

1.9 匹配

§7 路径

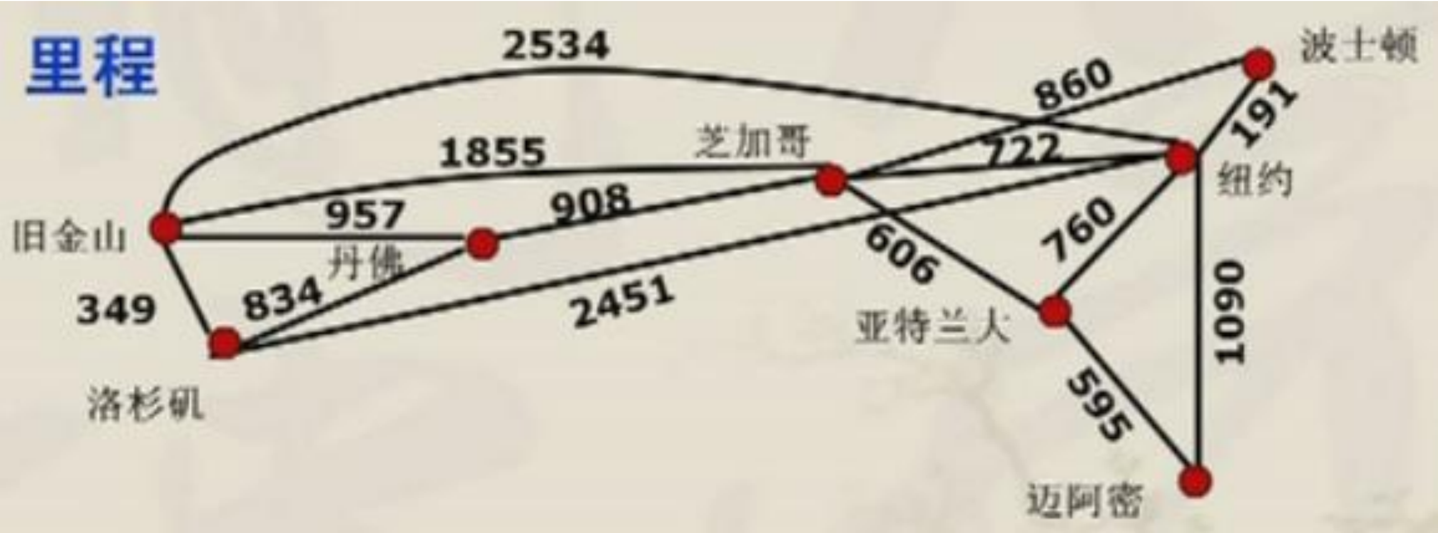
一、 最短路径

二、 Dijkstra算法

三、 拓扑排序和关键路径

❖给边赋权值的图来建模

- **航线系统建模**



➢ 计算从波士顿到洛杉矶之间空中距离**最短**的通路?

一. 最短路径

❖基本概念

➢ 带权图：给每条边赋值权值为一个数的图.

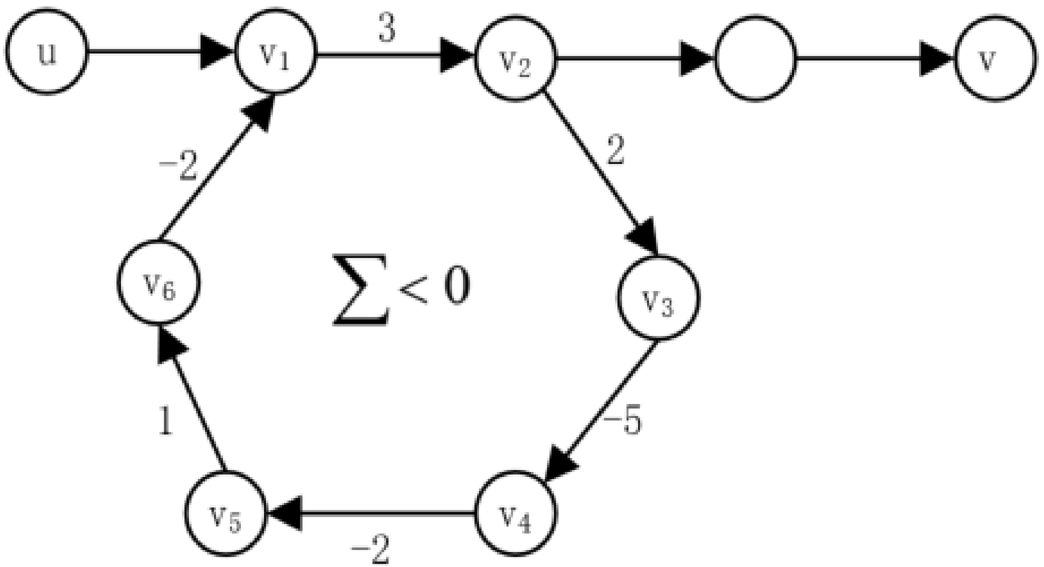
➢ 一条路径的长度：若p:$v_1 \to v_2 \to \ldots v_k$表示带权图$v_1$到$v_k$的一条路径, p的权值为该路径经过的所有边的权值总和, 记为w(p).

➢ 最短路径：若p为从u到v的一条路径,使w(p)最小,此时的p就是最短路径. 最短路径的权值为$\delta(u, v)=\min\{w(p)\}$

最短路径可能不存在：

(1) 存在负权回路（如下图），

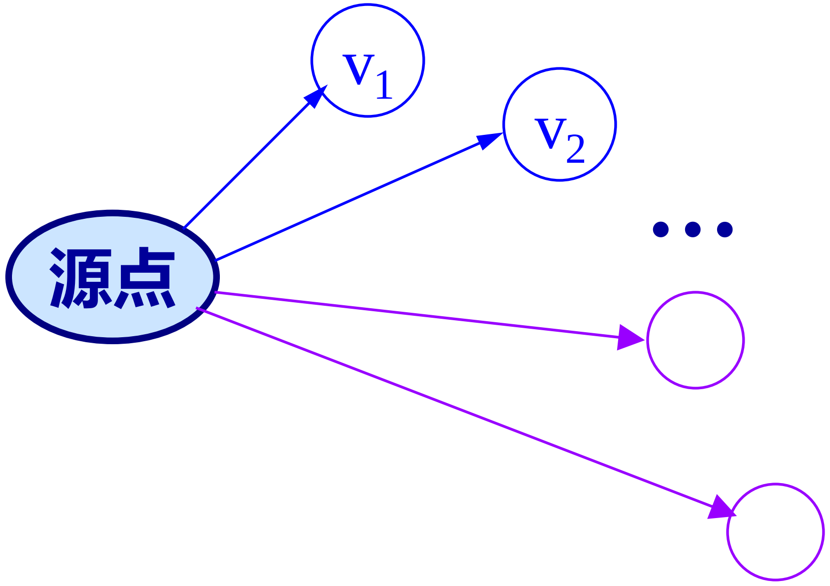
(2) 不存在从u到v的路径, 肯定也不存在最短路径.



一. 最短路径

❖求从源点到其余各点的最短路径的算法的基本思想：

➢ 依最短路径的长度递增的次序求得各条路径



其中，从源点到顶点v_1的最短路径是所有最短路径中长度最短者.

一. 最短路径

❖求从源点到其余各点的最短路径的算法的基本思想：

- 路径长度最短的最短路径的特点：

  在这条路径上，必定只含一条弧，并且这条弧的权值最小。

- 下一条路径长度次短的最短路径的特点：

  它只可能有两种情况：或者是直接从源点到该点(只含一条弧)；或者是从源点经过顶点v_1，再到达该顶点(由两条弧组成)。

一. 最短路径

❖求从源点到其余各点的最短路径的算法的基本思想：

- 再下一条路径长度次短的最短路径的特点：

它可能有三种情况：或者是直接从源点到该点(只含一条弧)；或者是从源点经过顶点v_1，再到达该顶点(由两条弧组成)；或者是从源点经过顶点v_2，再到达该顶点。

- 其余最短路径的特点：

它或者是直接从源点到该点(只含一条弧)；或者是从源点经过已求得最短路径的顶点，再到达该顶点。

❖Dijkstra算法

Dijkstra算法（1959）

设G有n个顶点；边的长度$\ell_{ij}>0$；

- 结点v_i和v_j没有边相连(不是邻接点)，则令$\ell_{ij}=\infty$,
- 对每个结点v_i，令$\ell_{ij}=0$。

❖Dijkstra算法

基本思想：

（1）需要指定起点v_1(即从顶点v_1开始计算);

（2）引进两个集合P和T.

P的作用是记录已求出最短路径的顶点(以及相应的最短路径长度),

T则是记录还未求出最短路径的顶点(以及该顶点到起点s的距离);

（3）初始时,P中只有起点s; T中是除s之外的顶点,并且T中顶点的路径是"起点s到该顶点的路径".然后,从T中找出路径最短的顶点,并将其加入到P中;接着,更新T中的顶点和顶点对应的路径. 然后,再从T中找出路径最短的顶点,并将其加入到P中;接着,更新T中的顶点和顶点对应的路径.重复该操作,直到遍历完所有顶点.

❖Dijkstra算法

算法步骤：

■**Step1:**

初始化：将v_1置为P标号，$d(v_1)=0$，$P=\{v_1\}$，$\forall v_i(i\neq 1)$置v_i为T标号，即$T=V-P$，且

$$d(v_i)=W(v_1, v_i) \quad 若 v_i \ adj \ v_i$$

$$d(v_i)=\infty \quad else$$

❖Dijkstra算法

算法步骤：

■Step2:找最小

寻找具有最小值的T标号的结点。若为v_1，则将v_1的T标号改为P标号，且$P=P\cup\{v_l\}$，$T=T-\{v_l\}$。

❖Dijkstra算法

算法步骤：

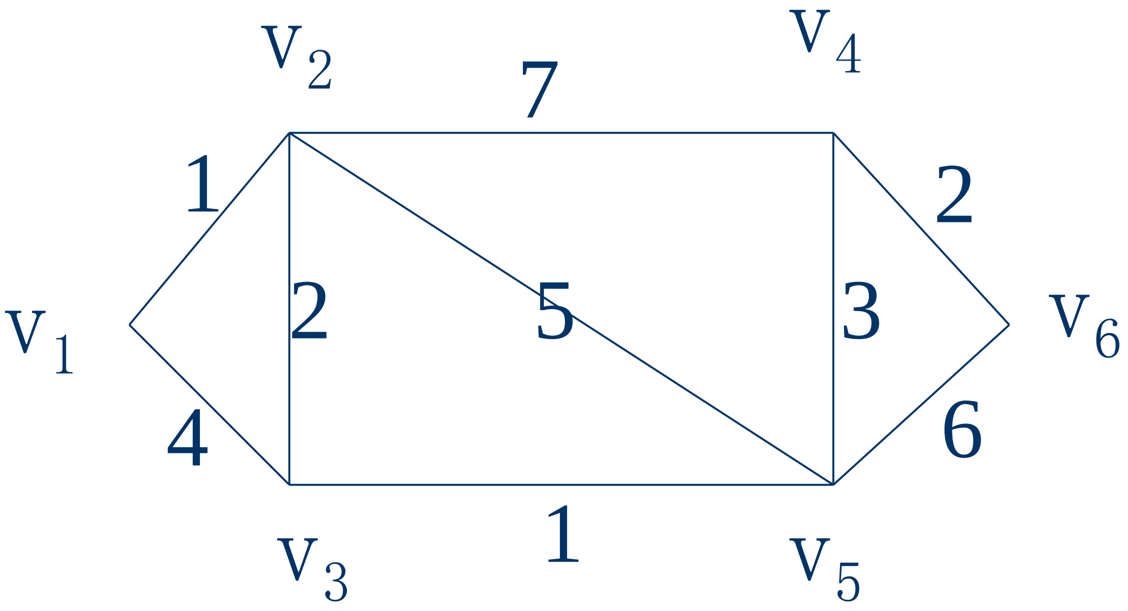
■Step3：修改

修改与v_1相邻的结点的T标号的值. $\forall\ v_i \in T$：

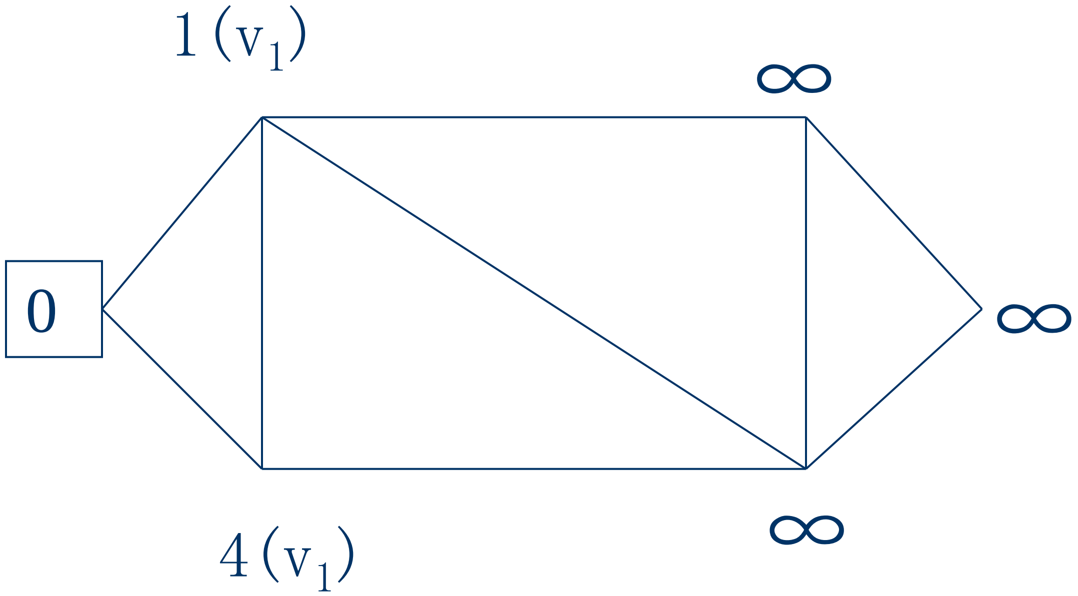
$$d(vi)=\begin{cases} d(v_l)+W(v_l,v_i) & 若 d(v_l)+W(v_l,v_i)<d(v_i) \\ d(v_i) & 否则 \end{cases}$$

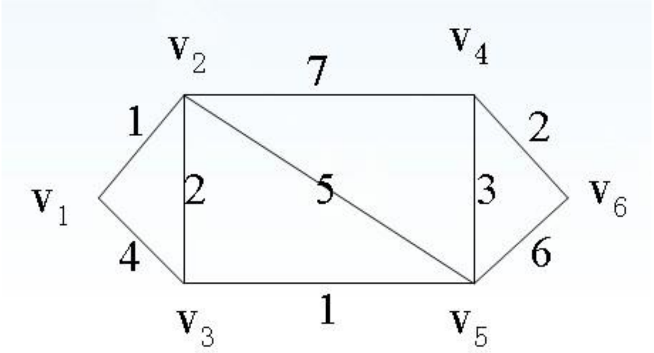
■Step4：重复（2）和（3），直到v_n改为P标号为止。

例：试求无向赋权图中v_1到v_6的最短路径



二. Dijkstra算法(*)

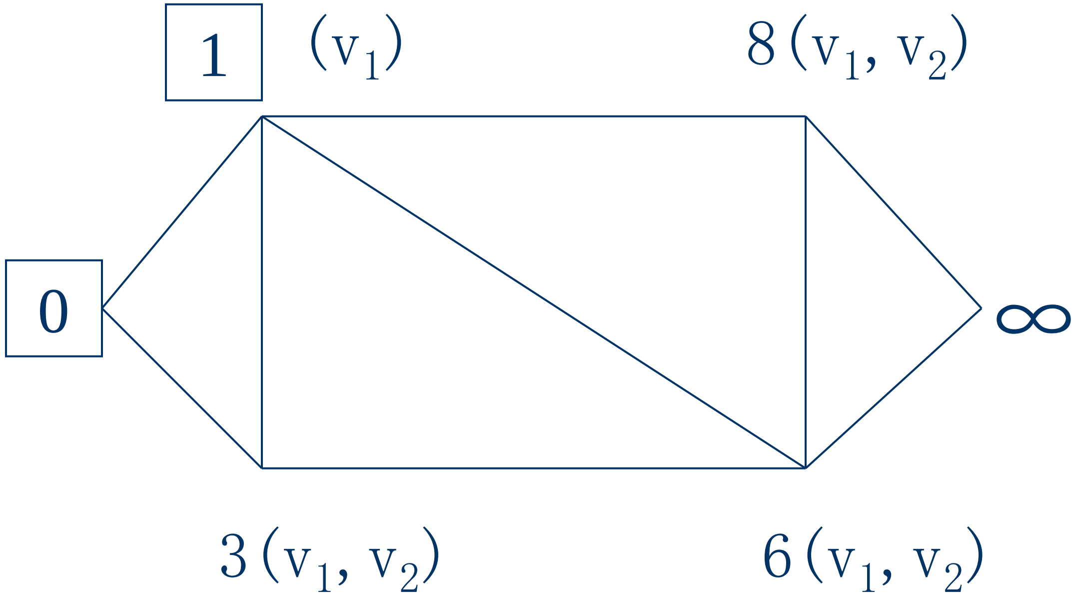


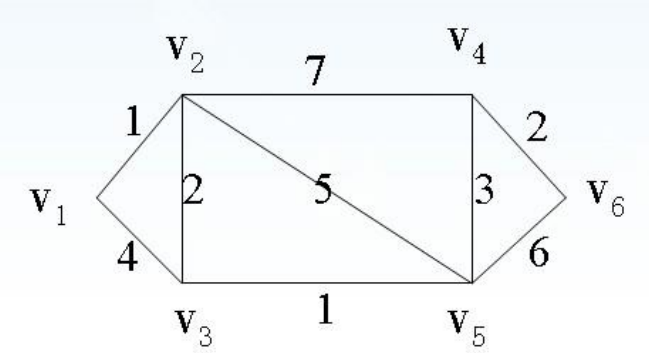


$P=\{v_1\}$

$T=\{v_2, v_3, v_4, v_5, v_6\}$

二. Dijkstra算法(*)

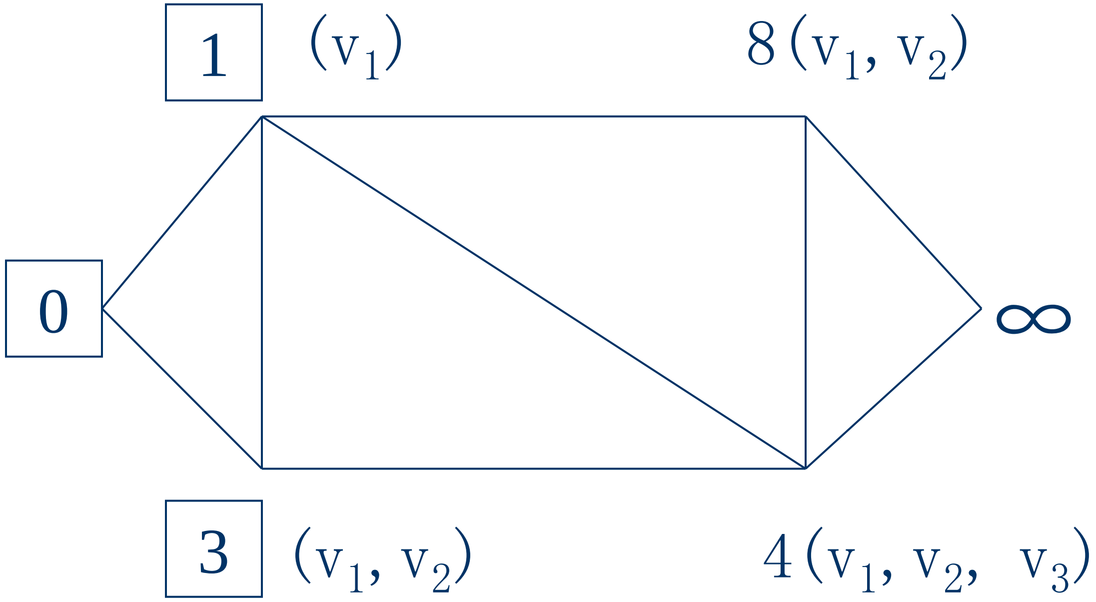




$P=\{v_1,v_2\}$

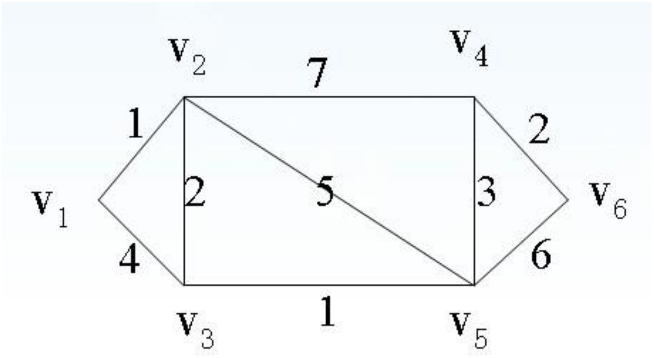
$T=\{v_3,v_4,v_5,v_6\}$

二. Dijkstra算法(*)

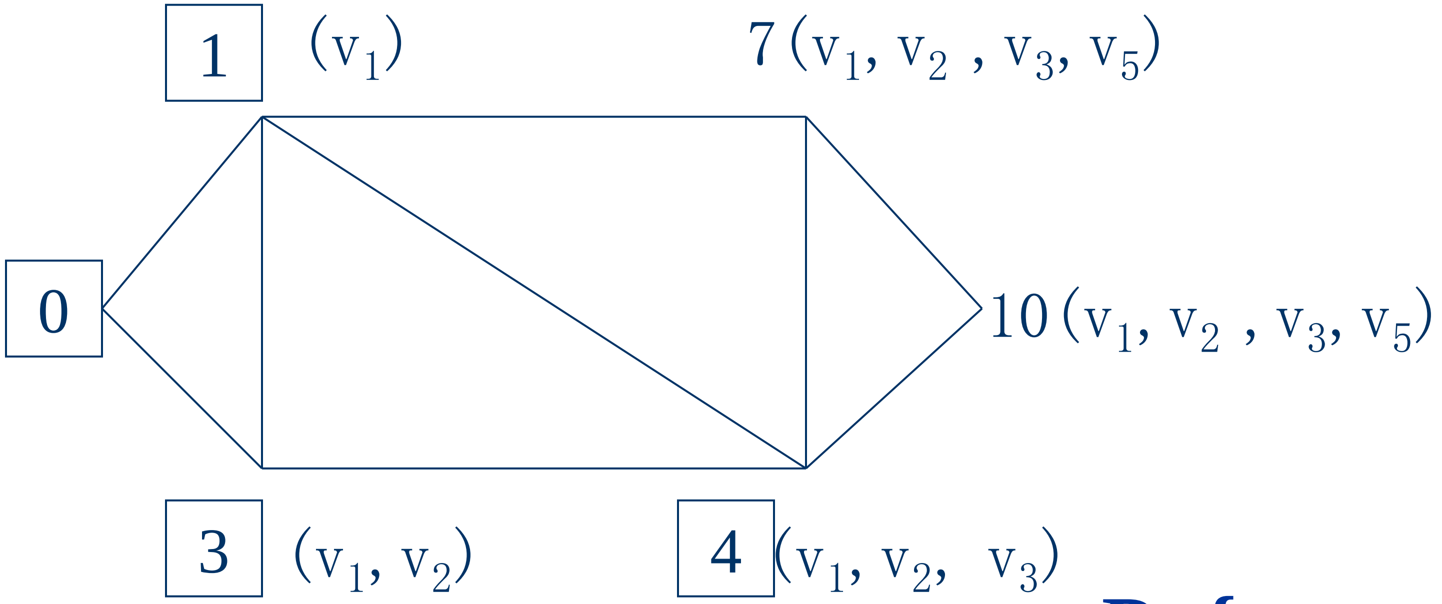


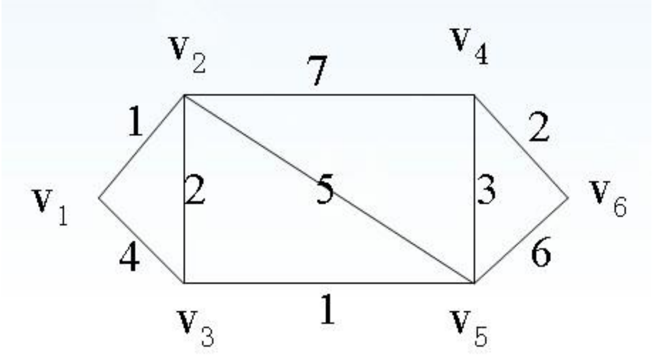
$P=\{v_1,v_2, v_3\}$

$T=\{v_4,v_5,v_6\}$



二. Dijkstra算法(*)

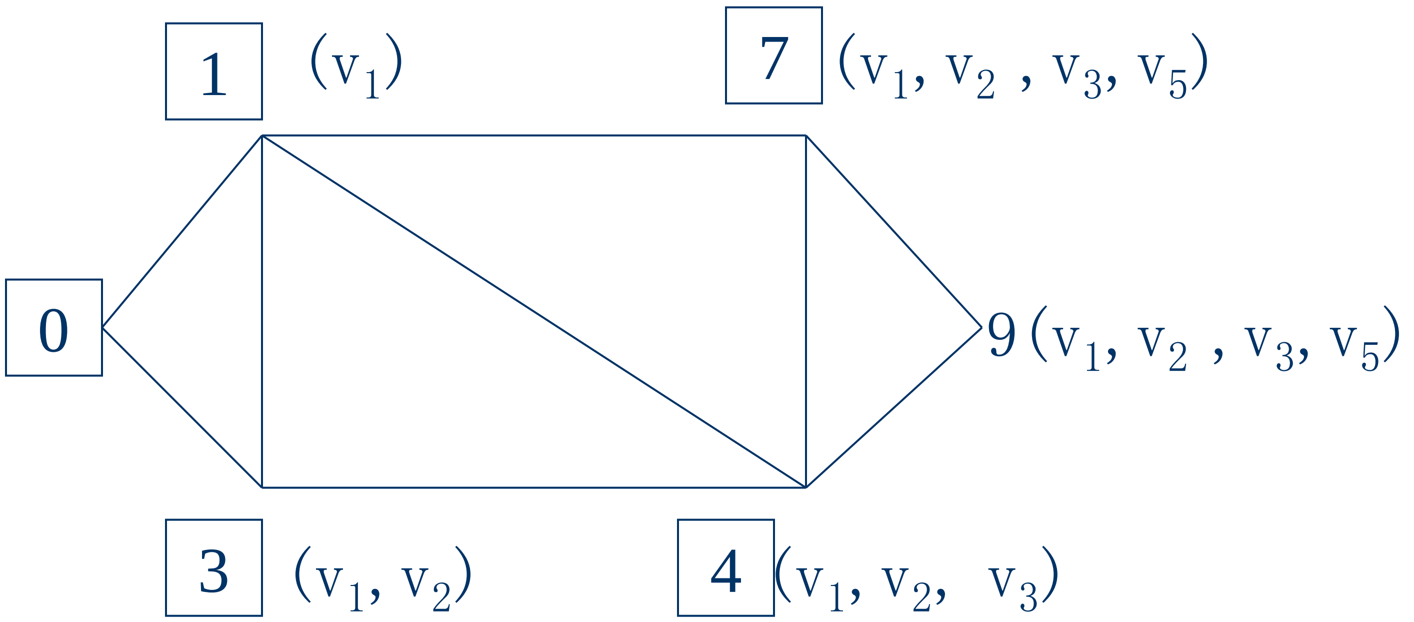


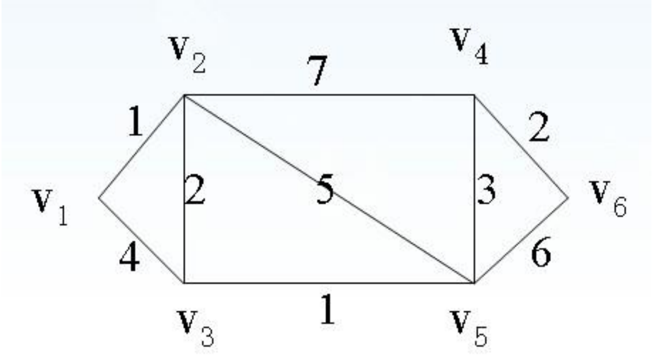


$P=\{v_1,v_2 , v_3, v_5\}$

$T=\{v_4,v_6\}$

二. Dijkstra算法(*)

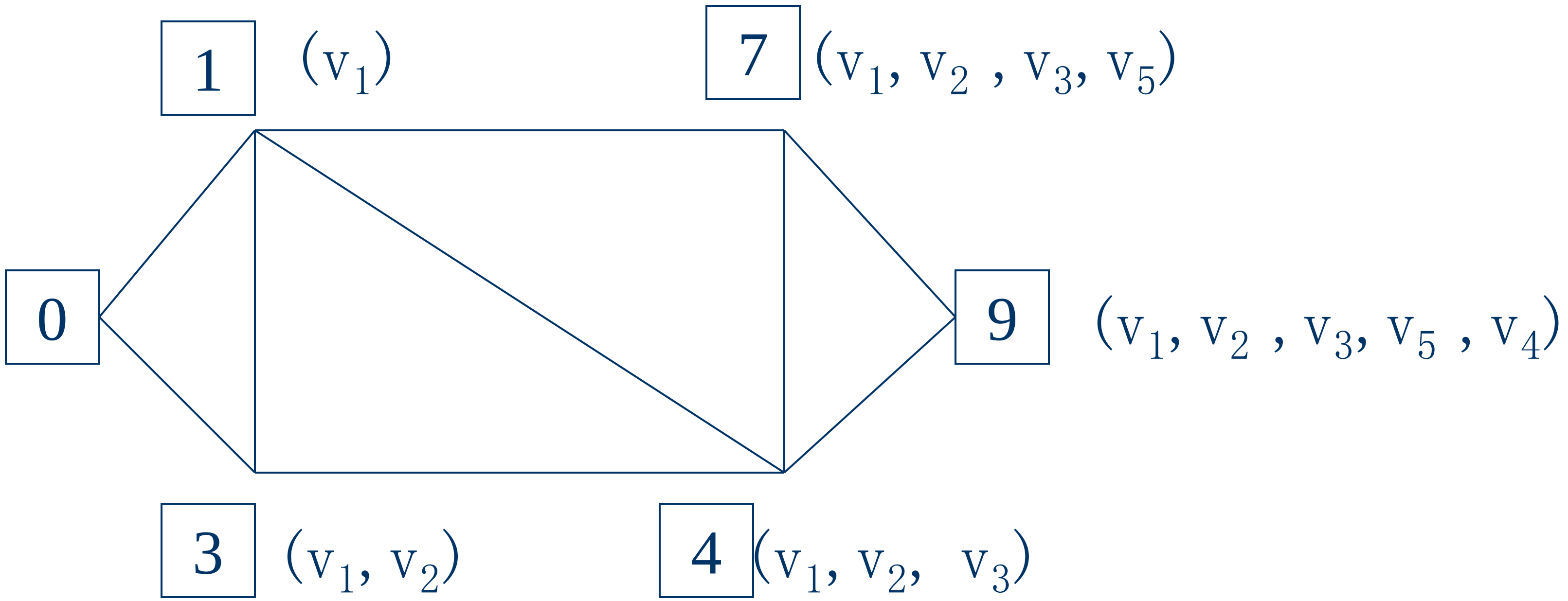


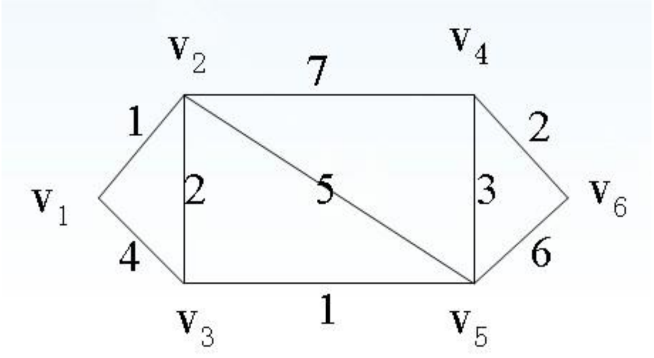


$P=\{v_1,v_2 , v_3, v_5 , v_4\}$

$T=\{v_6\}$

二. Dijkstra算法(*)



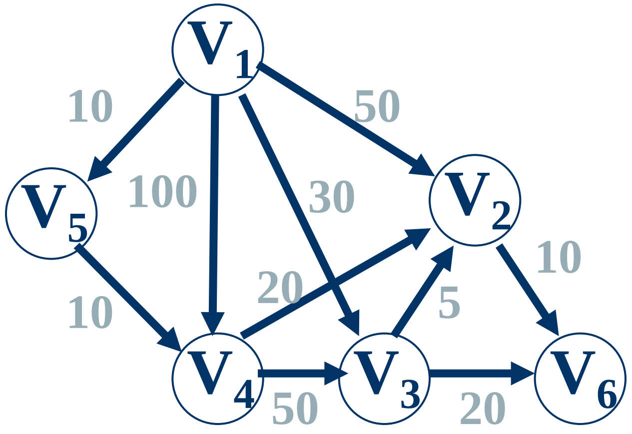


$P=\{v_1,v_2 , v_3, v_5\ v_4 , v_6\}$

$T=\{\}$

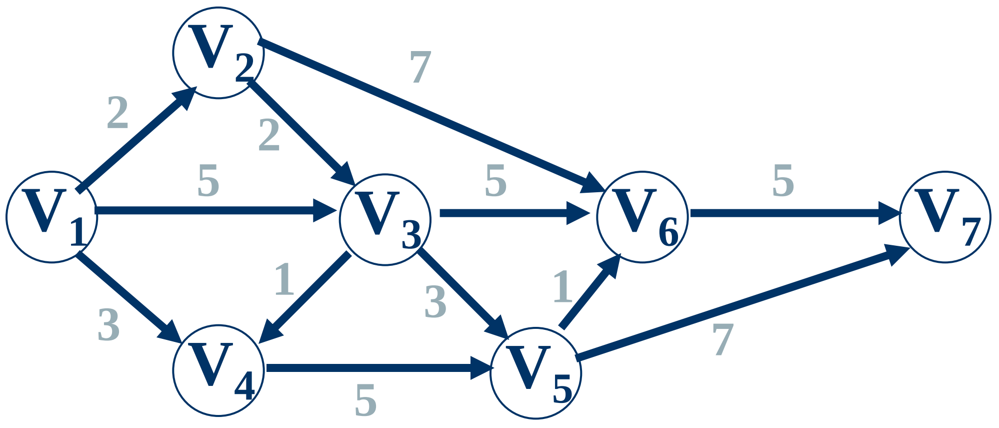
二. Dijkstra算法

	v_2	v_3	v_4	v_5	v_6	
step1	50	30	100	10/V_1	∞	10(v_5)第1短
step2	50	30	20/V_5		∞	20(v_4)第2短
step3	40	30/V_1			∞	30(v_3)第3短
step4	35/V_3				50	35(v_2)第4短
step5					45/V_2	45(v_6)第5短



二. Dijkstra算法(*)

	v_2	v_3	v_4	v_5	v_6	v_7	
step1	2	5	3	∞	∞	∞	2(v_2)第1短
step2		4	3	∞	9	∞	3(v_4)第2短
step3		4		8	9	∞	4(v_3)第3短
step4				7	9	∞	7(v_5)第4短
step5					8	14	8(v_6)第5短
step6						13	13(v_7)第6短



最短路径(*)

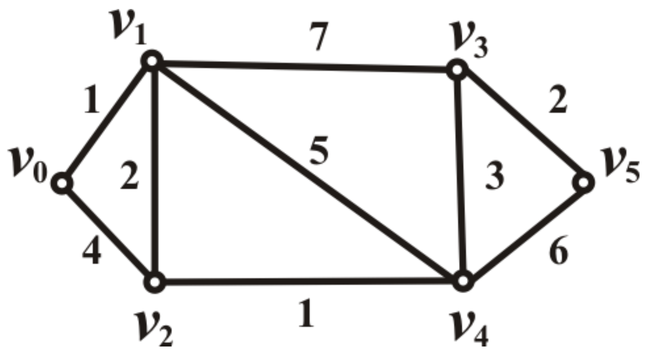
带权图$G=<V,E,w>$, 其中$w:E\to \mathrm{R}$. $\forall e\in E$, $w(e)$称作e的权. $e=(v_i,v_j)$, 记$w(e)=w_{ij}$. 若v_i,v_j不相邻, 记$w_{ij}=\infty$.

通路L的权: L的所有边的权之和, 记作$w(L)$. u和v之间的最短路径: u和v之间权最小的通路.

例 $L_1=v_0v_1v_3v_5$, $w(L_1)=10$,

$L_2=v_0v_1v_4v_5$, $w(L_2)=12$,

$L_3=v_0v_2v_4v_5$, $w(L_3)=11$.



标号法 (E.W. Dijkstra, 1959)（*）

设带权图$G=\langle V,E,w\rangle$, 其中$\forall e\in E$, $w(e)\geq 0$.

设$V=\{v_1,v_2,\ldots,v_n\}$, 求v_1到其余各顶点的最短路径

1. 令 $l_1\leftarrow 0$, $p_1\leftarrow\lambda$, $l_j\leftarrow+\infty$, $p_j\leftarrow\lambda$, $j=2,3,\ldots,n$,

   $P=\{v_1\}$, $T=V-\{v_1\}$, $k\leftarrow 1$, $t\leftarrow 1$. 　　　/ λ表示空

2. 对所有的$v_j\in T$且$(v_k,v_j)\in E$

   令$l\leftarrow\min\{l_j, l_k+w_{kj}\}$, 若$l=l_k+w_{kj}$, 则令$l_j\leftarrow l$, $p_j\leftarrow v_k$.

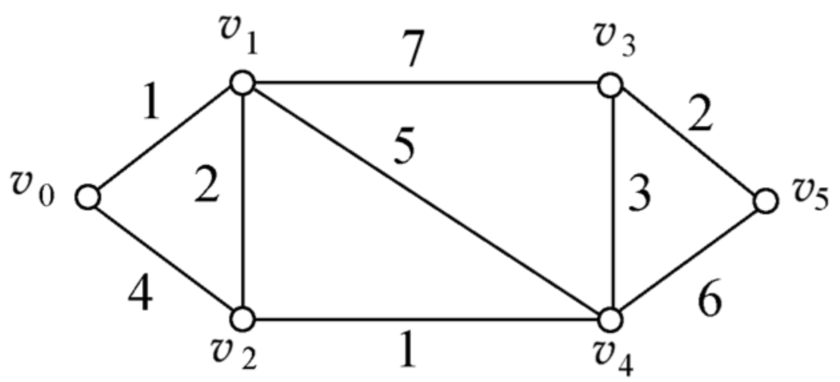
3. 求$l_i=\min\{l_j|\ v_j\in T_t\}$.

   令$P\leftarrow P\cup\{v_i\}$, $T\leftarrow T-\{v_i\}$, $k\leftarrow i$.

4. 令$t\leftarrow t+1$, 若$t<n$, 则转2.

Dijkstra标号法(*)

例 求v_0到v_5的最短路径

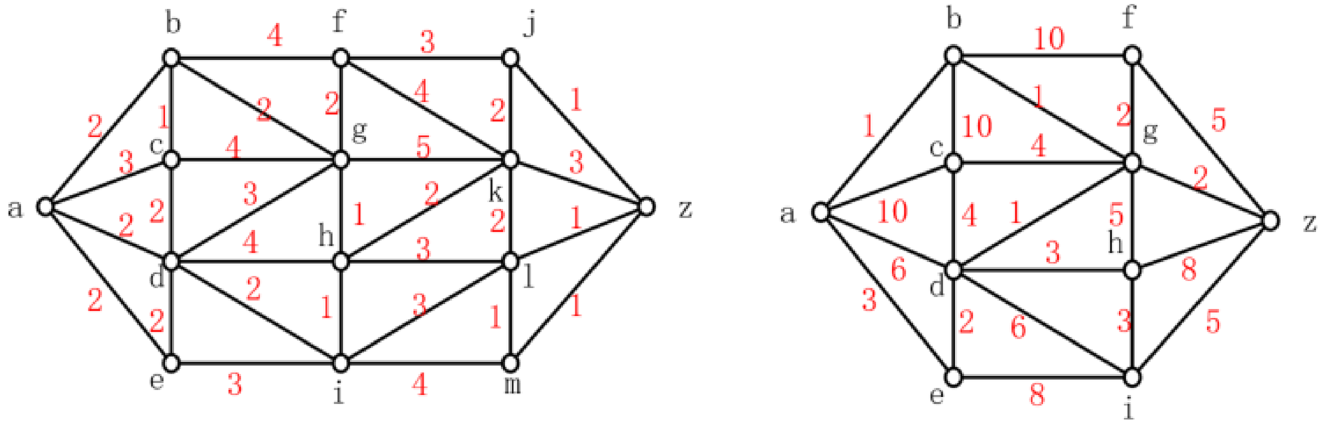


t	v_0	v_1	v_2	v_3	v_4	v_5
1	$(0,\lambda)^*$	$(+\infty,\lambda)$	$(+\infty,\lambda)$	$(+\infty,\lambda)$	$(+\infty,\lambda)$	$(+\infty,\lambda)$
2		$(1,v_0)^*$	$(4,v_0)$	$(+\infty,\lambda)$	$(+\infty,\lambda)$	$(+\infty,\lambda)$
3			$(3,v_1)^*$	$(8,v_1)$	$(6,v_1)$	$(+\infty,\lambda)$
4				$(8,v_1)$	$(4,v_2)^*$	$(+\infty,\lambda)$
5				$(7,v_4)^*$		$(10,v_4)$
6						$(9,v_3)^*$

v_0到v_5的最短路径: $v_0v_1v_2v_4v_3v_5$, $d(v_0,v_5)=9$

二. Dijkstra算法(*)

例： 用Dijkstra算法求下图(a)(b)从a到z的最短路径及其长度.



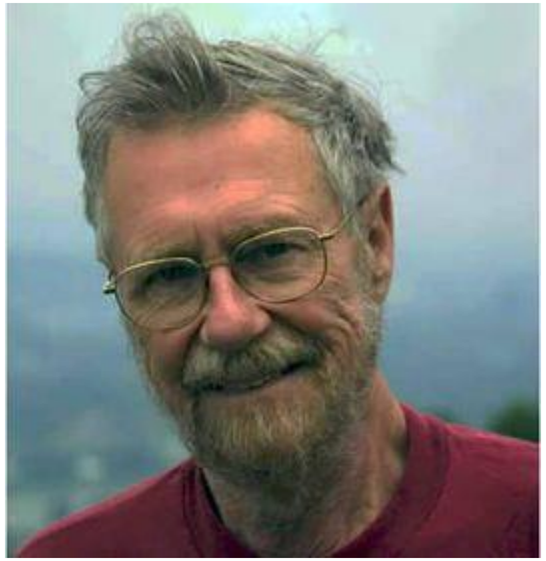
(a) (b)

解： (a)图中a到z的最短路径长为8,路径为（a, d, i, l, z）.

(b)图中a到z的最短路径长为4,路径为（a, b, g, z）.

E.W.Dijkstra (1930~2002)

1 提出“goto有害论”;



2 提出信号量和PV原语;

3 解决了“哲学家聚餐”问题;

4 Dijkstra最短路径算法和银行家算法的创造者;

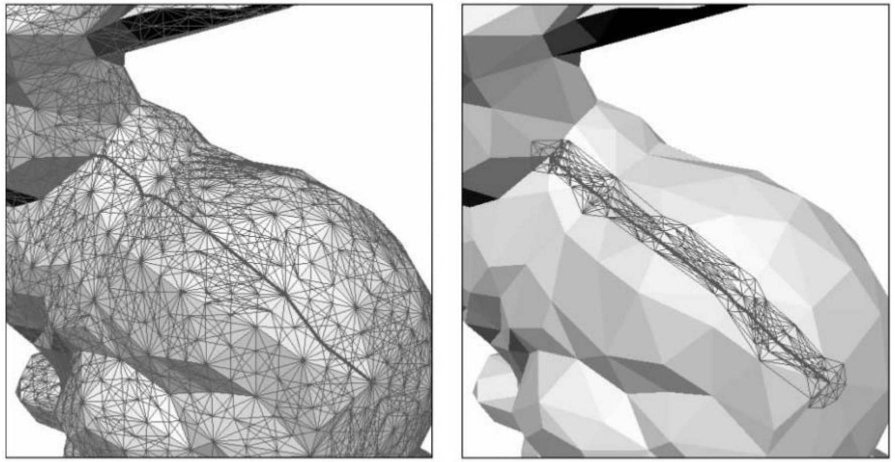
5 第一个Algol 60编译器的设计者和实现者;

6 THE操作系统的设计者和开发者;

与D. E. Knuth并称为我们这个时代最伟大的计算机科学家的人。

与癌症抗争多年，于2002年8月6日在荷兰Nuenen自己的家中去世，享年72岁。

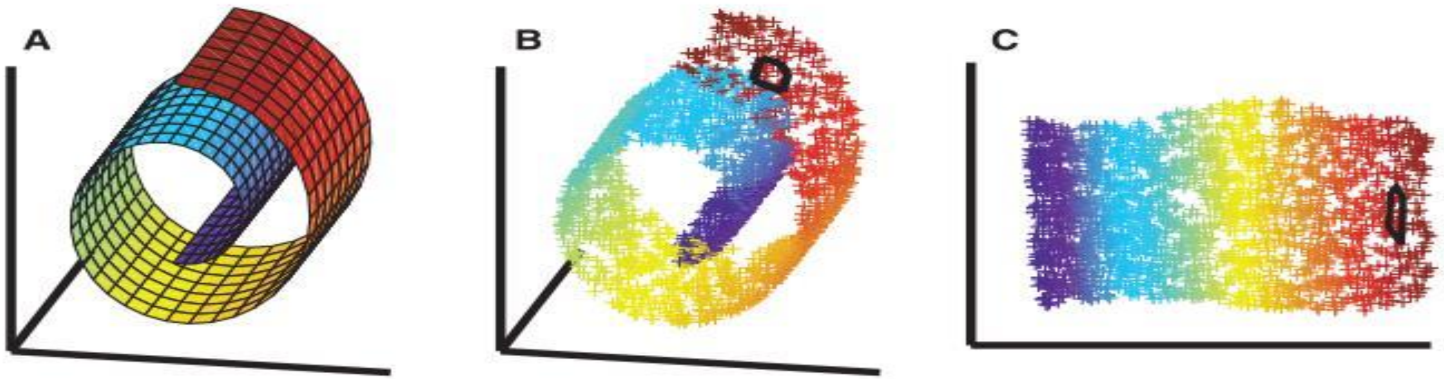
网格模型上的最短路径

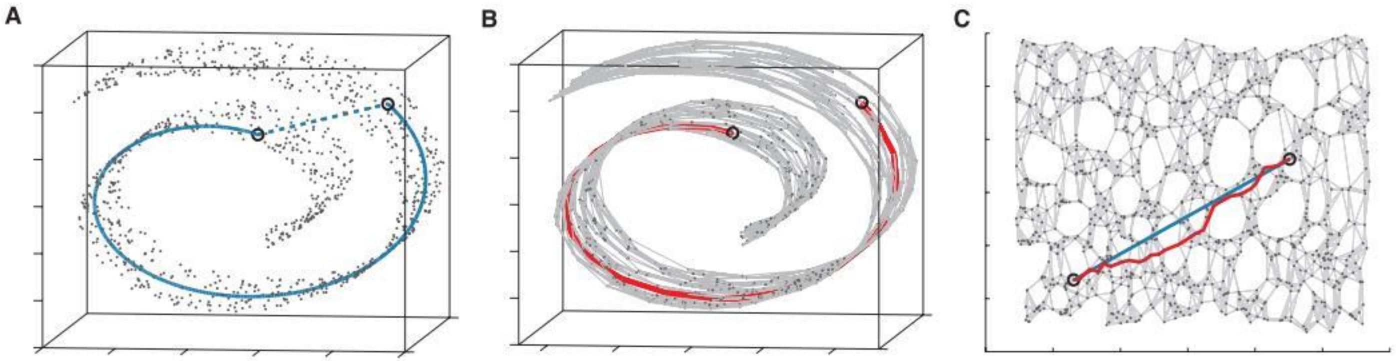


近似最短路径

精确最短路径

流形学习（ISOMAP）





三.拓扑排序和关键路径（*）

对一个工程或者系统,人们最关心的往往是两个方面的问题：

（1）工程能否顺利进行

对AOV网进行拓扑排序

（2）估算整个工程完成所必须的最短时间

对AOE网求关键路径

三.拓扑排序和关键路径（*）

❖AOV网 （activity on vertex network）

【定义】 在一个表示工程的有向图中，用顶点表示活动，用弧表示活动之间的优先关系，称这样的有向图为顶点表示活动的网，简称**AOV网**

思考： AOV网中能不能出现回路？

↓

如何判断AOV网是否有回路？

对AOV网拓扑排序

三.拓扑排序和关键路径（*）

❖拓扑排序

【定义】设$G=(V,E)$是一个有向图，V的顶点序列$v_0, v_1,\ldots,v_{n-1}$当且仅当满足以下条件：

若从顶点v_i到v_j有一条路径，在顶点序列中v_i必须存在于v_j之前，则称此顶点序列为一个**拓扑序列**。

➢ 对一个有向图构造拓扑序列的过程称为**拓扑排序**。

注：拓扑排序的排序结果很可能是不唯一的。

三.拓扑排序和关键路径（*）

❖拓扑排序的过程

过程如下：

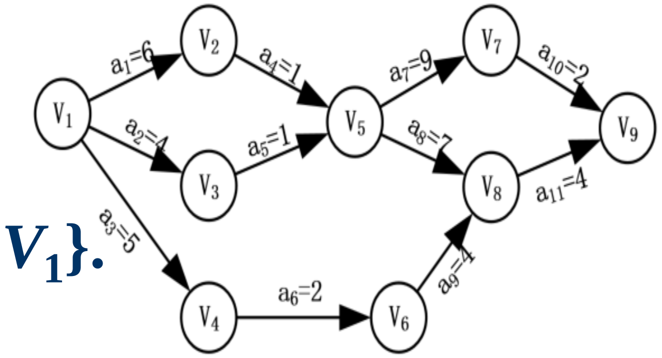
① 每次输出一个入度为0（即没有前驱）的结点,并删除该点与该点指出的有向边.

② 重复此过程直至全部入度为0的结点被输出,得到的结点输出序列就是拓扑序列.

③ 如果所有入度为0的结点都被输出,但图还不为空,说明该有向图中必存在环.

三.拓扑排序和关键路径（*）

❖拓扑排序的过程



例：由右图可得拓扑排序过程如下：

1）入度为0的结点只有V_1, 所以输出V_1, 并删除边a_1, a_2, a_3, $\{V_1\}$.

2）入度为0的结点有V_2, V_3, V_4, 可以任选一个结点输出，
比如先输出V_2, 并删除边a_4, $\{V_1,V_2\}$.

3）入度为0的结点有V_3, V_4, 可以任选一个结点输出，
比如先输出V_3, 并删除边a_5, $\{V_1,V_2,V_3\}$.

…

9）入度为0的结点只有V9,输出V9，全部结点被输出，图为空，拓扑排序完成，最后排序结果为{V1,V2,V3,V4,V5,V6,V7,V8,V9}.

本例题答案不唯一，满足拓扑排序的条件即可.

❖AOE网（Activity On Edge Network）

【定义】AOE网是指用边表示活动的网,是一个带权的有向无环图.

- 顶点：事件（Event），
- 弧：活动（Activity），
- 权值：活动持续的时间.

三.拓扑排序和关键路径（*）

❖关键路径

由于整个工程只有一个开始点和一个完成点，在正常的情况（无环）下，

- 网中只有一个入度为零的点（称作**源点**）
- 一个出度为零的点（称作**汇点**）

【**定义**】完成工程的最短时间指的是从源点到汇点的最长路径的长度,而这个长度最长的路径就叫做**关键路径**.

三.拓扑排序和关键路径（*）

❖关键活动

【定义】假设开始点是v_1, 从v_1到v_i的最长路径长度叫做事件v_i的最早发生时间.

- 这个时间决定了所有以v_i为尾的弧所表示的活动的最早开始时间.
- 活动a_i的最早开始时间通常用$e(i)$表示.

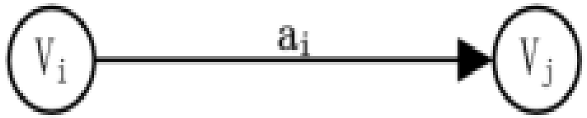
【定义】活动的最迟开始时间$l(i)$是指在不推迟整个工程完成的前提下, 活动a_i最迟必须开始进行的时间.

【定义】我们把$l(i) = e(i)$的活动叫做关键活动.

三.拓扑排序和关键路径（*）

❖关键活动

若活动a_i由弧$<i, j>$表示,持续时间记为$dut(<i, j>)$,则关系如下图所示：



- 活动i的最早开始时间等于事件j的最早发生时间

$$e(i) = v_e(i)$$

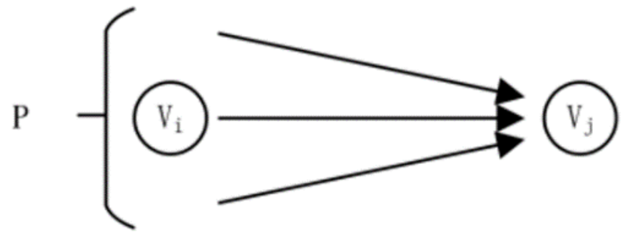
- 活动i的最迟开始时间等于事件k的最迟时间减去活动i的持续时间

$$l(i) = v_l(j) - dut(<i, j>)$$

❖关键活动

$v_e[j]$和$v_l[j]$可以采用下面的递推公式计算,需分两步进行：

（1）向汇点递推

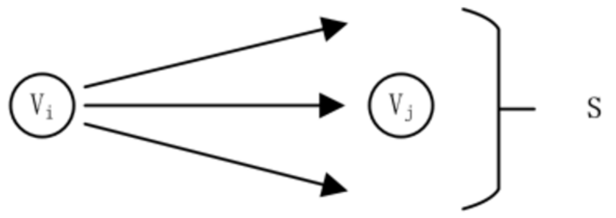


a) 向汇点递推

- v_e(源点) = 0 ;
- $v_e(j) = \mathrm{Max}\{v_e(i) + dut(<i, j>)\}$

➢ **公式意义**：从指向顶点V_j的弧的活动中取最晚完成的一个活动的完成时 间作为V_j的最早发生时间$v_e[j]$, 如右图所示.

❖关键活动



b) 向源点递推

(2) 向源点递推

由上一步的递推,最后总可求出汇点的最早发生时间$v_e[n]$.因汇点就是结束点,最迟发生时间与最早发生时间相同,即$v_l[n]= v_e[n]$.从汇点最迟发生现时间$v_l[n]$开始,利用下面公式:

- v_l(汇点) = v_e(汇点);
- $v_l(i) = Min\{v_l(j) – dut(<i, j>) \}$

➢ **公式意义**: 由从V_i顶点指出的弧所代表的活动中取需最早开始的一个开始时间作为V_i的最迟发生时间,如下图所示.

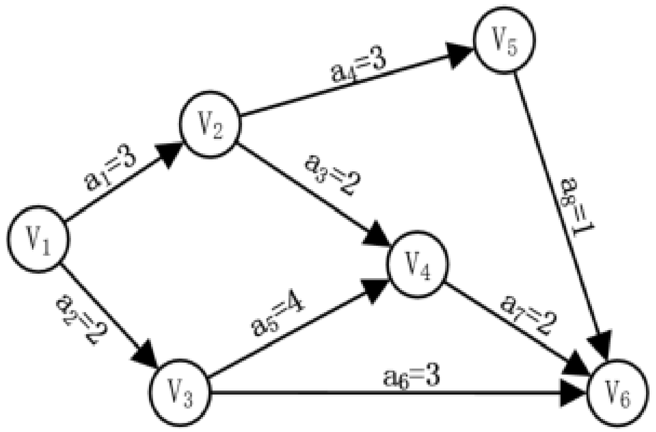
三.拓扑排序和关键路径（*）

例： 求右图中AOE网的拓扑排序和关键路径.

解： 由图可得,拓扑排序为V_1-V_2-V_3-V_4-V_5-V_6.

关键路径求解如下：

顶点	ve	vl	活动	e	l	l-e
v_1	0	0	a_1	0	1	1
v_2	3	4	a_2	0	0	0
v_3	2	2	a_3	3	4	1
v_4	6	6	a_4	3	4	1
v_5	6	7	a_5	2	2	0
v_6	8	8	a_6	2	5	3
			a_7	6	6	0
			a_8	6	7	1



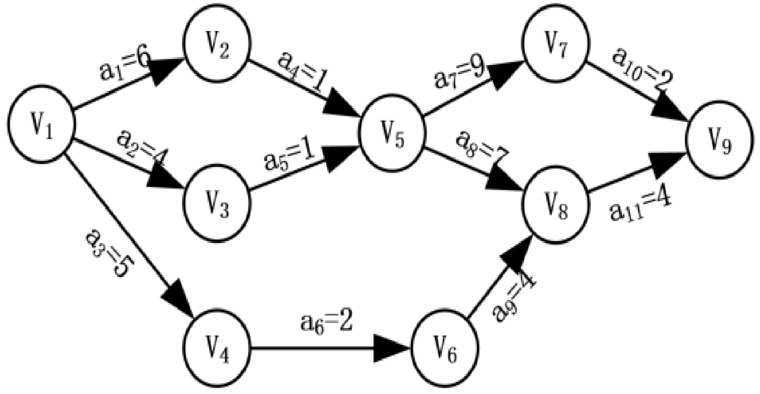
由上表可知,活动a_2、a_5、a_7的最早开始时间和最迟开始时间相等（$e=l$），所以a_2、a_5、a_7为关键活动.故得出关键路径为：V_1-V_3-V_4-V_6.

三.拓扑排序和关键路径（*）

例：求右图中AOE网的关键路径.

解：由图可得,关键路径求解如下：

事件 j	$e_v[j]$	$L_v[j]$	活动 i	e[i]	L[i]	L[i]-e[i]
1	0	0	1	0	0	0
2	6	6	2	0	2	2
3	4	6	3	0	3	3
4	5	8	4	6	6	0
5	7	7	5	4	6	2
6	7	10	6	5	8	3
7	16	16	7	7	7	0
8	14	14	8	7	7	0
9	18	18	9	7	10	3
			10	16	16	0
			11	14	14	0



由上表可知,活动a_1、a_4、a_7、a_8、a_{10}、a_{11}为关键活动,所以关键路径为V_1-V_2-V_5-V_7-V_9或者V_1-V_2-V_5-V_8-V_9.

1.6 矩阵表示

1.7 路径

1.8 图的着色

1.9 匹配

§8 图的着色

一、 对偶图

二、 四色猜想

三、 平面图面着色

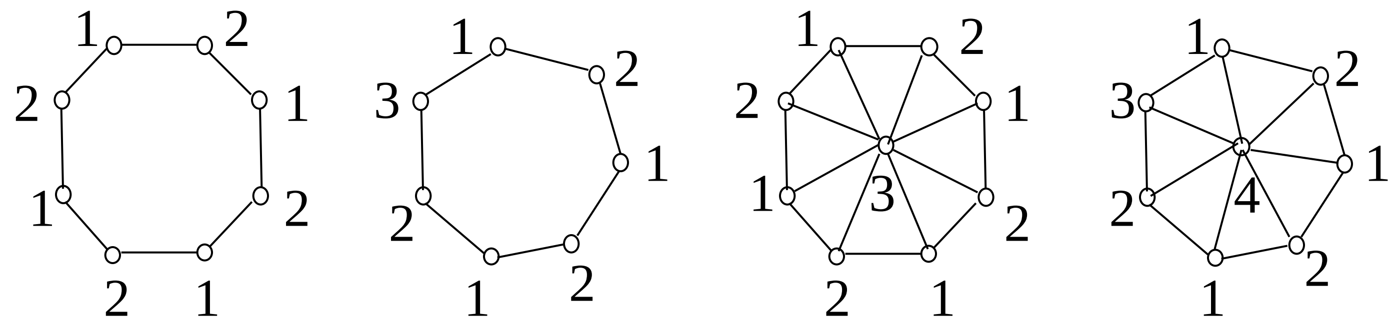
四、 平面图点着色

（点）着色

定义 设无向图G无环，对G的每个顶点涂一种颜色，使相邻的顶点涂不同的颜色，称为图G的一种**点着色**，简称**着色**。若能用k种颜色给G的顶点着色，则称G是**k-可着色**的，记作：$\chi(G) = k$

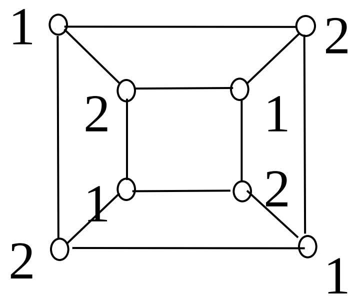
图的着色问题：用尽可能少的颜色给图着色。

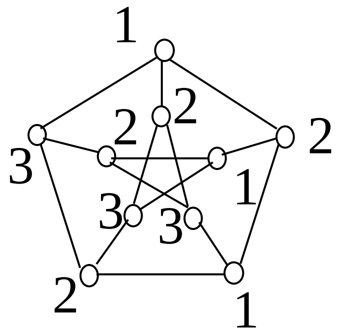
例1

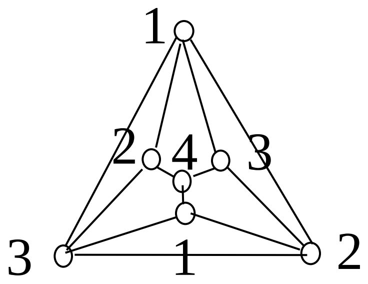


例

例2

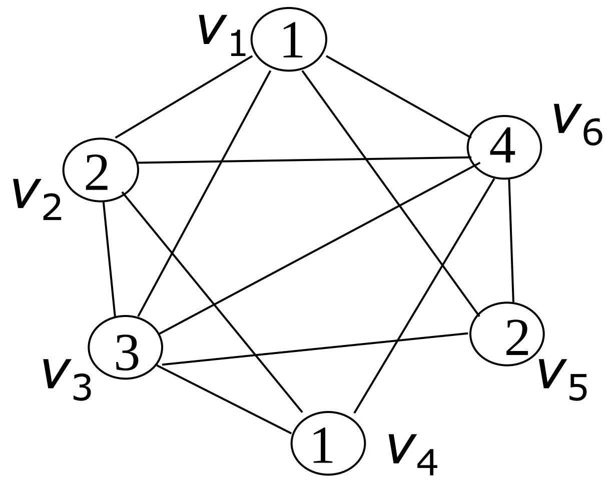






例

例：学生会下设6个委员会，第一委员会={张，李，王}，第二委员会={李，赵，刘}，第三委员会={张，刘，王}，第四委员会={赵，刘，孙}，第五委员会={张，王}，第六委员会={李，刘，王}。每个月每个委员会都要开一次会，为了确保每个人都能参加他所在的委员会会议，这6个会议至少要安排在几个不同时间段？



至少要4个时段

第1时段：一，四

第2时段：二，五

第3时段：三

第4时段：六

应用

- 有n项工作，每项工作需要一天的时间完成。有些工作由于需要相同的人员或设备不能同时进行，问至少需要几天才能完成所有的工作？
- 计算机有k个寄存器，现正在编译一个程序，要给每一个变量分配一个寄存器。如果两个变量要在同一时刻使用，则不能把它们分配给同一个寄存器。如何给变量分配寄存器？
- 无线交换设备的波长分配。有n台设备和k个发射波长，要给每一台设备分配一个波长。如果两台设备靠得太近，则不能给它们分配相同的波长，以防止干扰。如何分配波长？

1.6 矩阵表示

1.7 路径

1.8 图的着色

1.9 匹配

§9 匹配

一、二部图

二、匹配与最大匹配

三、 霍尔定理

引例

- 每学年评奖学金，把一等奖（1项），二等奖（2项），三等奖（3项）颁给某班同学，如何描述奖学金与同学之间的关系？

- 运动会颁奖，如何描述名次与运动员之间的关系？

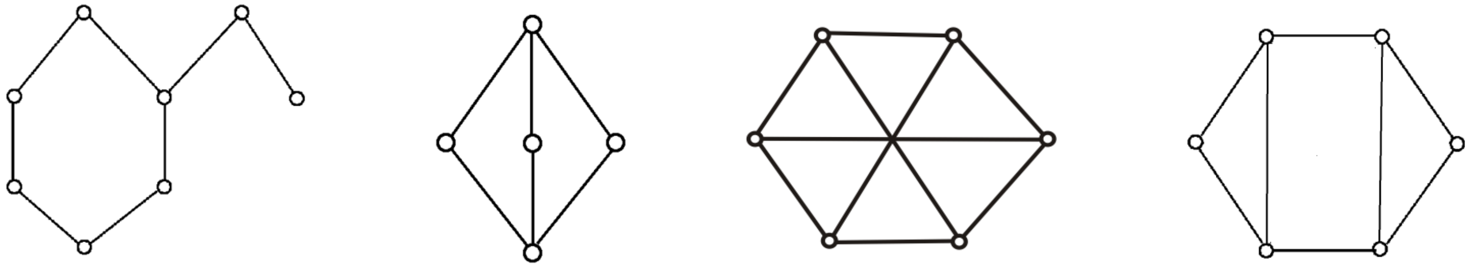
二部图

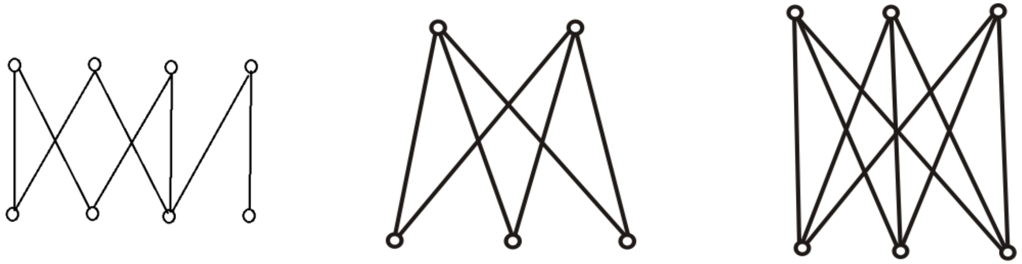
定义：设无向图 $G=<V,E>$，若能将V 划分成V_1 和 V_2 ($V_1 \cup V_2=V$, $V_1 \cap V_2=\varnothing$)，使得G中的每条边的两个端点都一个属于V_1，另一个属于V_2，则称G为二部图，记为$<V_1,V_2,E>$，称V_1和V_2为互补顶点子集. 又若G是简单图, 且V_1中每个顶点都与V_2中每个顶点相邻, 则称G为完全二部图, 记为$K_{r,s}$, 其中$r=|V_1|$, $s=|V_2|$.

注意: n 阶零图为二部图.

二部图(续)

例 下述各图是否是二部图?

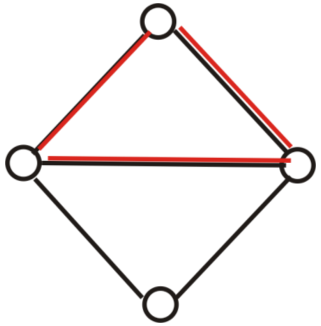




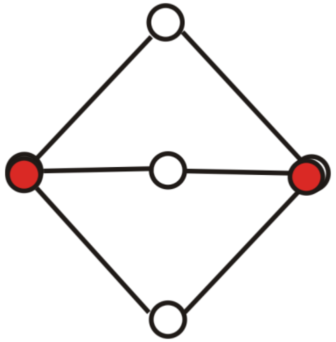
不是

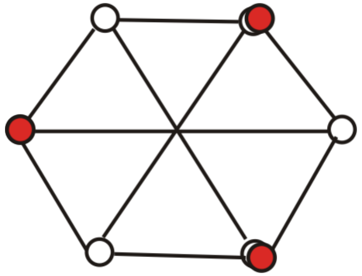
定理 无向图$G=<V,E>$是二部图当且仅当G中无奇圈

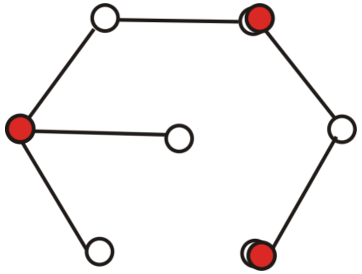
二部图(续)

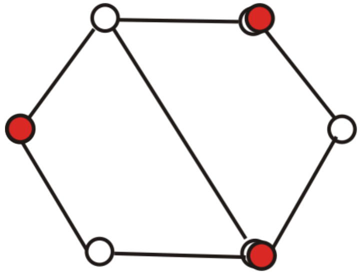


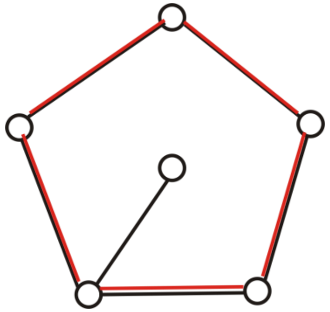
非二部图











非二部图

匹配（单射）

设 $G=<V,E>$,

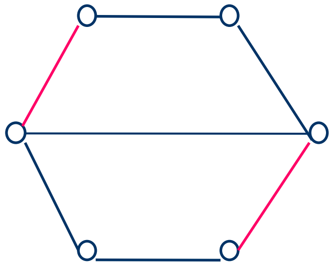
匹配(边独立集): 任2条边均不相邻的边子集

极大匹配: 添加任一条边后都不再是匹配的匹配

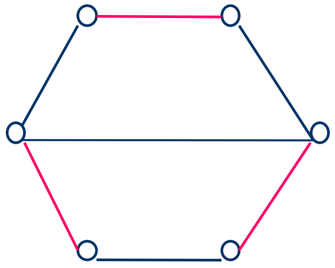
最大匹配: 边数最多的匹配

匹配数: 最大匹配中的边数, 记为 β_1

例



极大匹配



最大匹配 $\beta_1=3$

匹配 (续)

设M为G中一个匹配

v_i与v_j被M匹配: $(v_i,v_j)\in M$

v为M饱和点: M中有边与v关联

v为M非饱和点: M中没有边与v关联

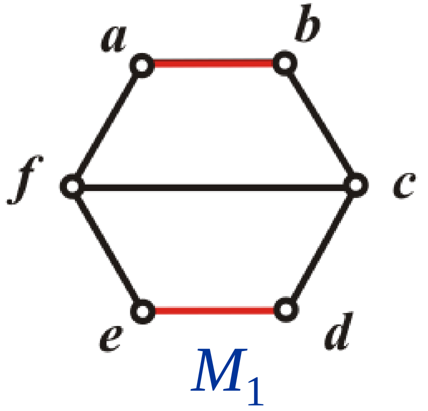
M为完美匹配: G的每个顶点都是M饱和点

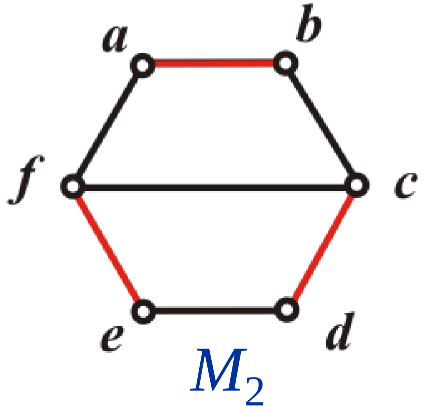
例 关于M_1, a,b,e,d是饱和点

f,c是非饱和点

M_1不是完美匹配

M_2是完美匹配

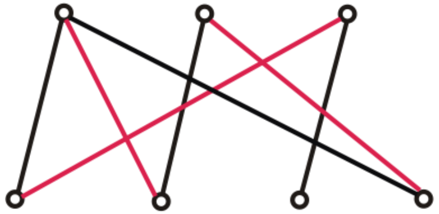




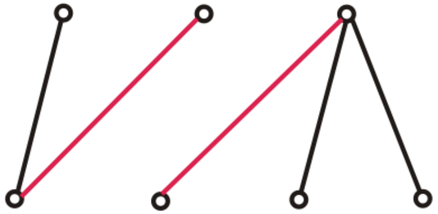
二部图中的匹配

定义 设$G=<V_1,V_2,E>$为二部图, $|V_1|\le|V_2|$, M是G中最大匹配, 若V_1中顶点全是M饱和点，则称M为G中V_1到V_2的完备匹配. 当$|V_1|=|V_2|$时, 完备匹配变成完美匹配。

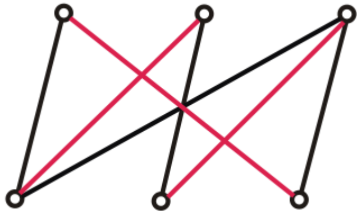
例



完备,不完美



不完备



完美

Hall（霍尔）定理

定理(Hall定理) 设二部图$G=<V_1,V_2,E>$中，$|V_1|\le|V_2|$. G中存在从V_1到V_2的完备匹配当且仅当V_1中任意k 个顶点至少与V_2中的k个顶点相邻($k=1,2,\ldots,|V_1|$).

—相异性条件

由Hall定理, 上一页第2个图没有完备匹配.

定理 设二部图$G=<V_1,V_2,E>$中, 如果存在$t\ge 1$, 使得V_1中每个顶点至少关联 t 条边, 而V_2中每个顶点至多关联t条边，则G中存在V_1到V_2的完备匹配.

—t 条件

证 V_1中任意k个顶点至少关联kt条边, 这kt条边至少关联V_2中的k个顶点, 即V_1中任意k个顶点至少邻接V_2中的k个顶点. 由Hall定理, G中存在V_1到V_2的完备匹配.

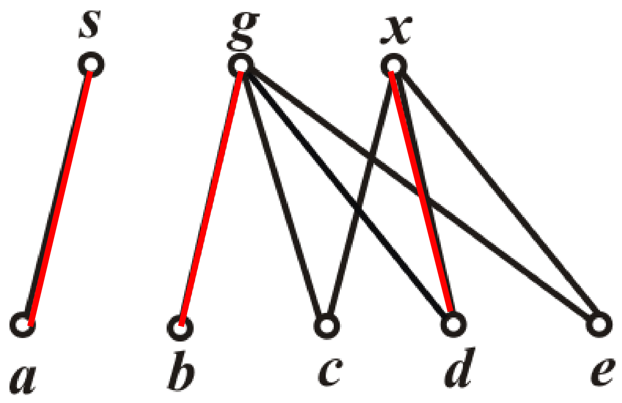
应用实例（一）

例 某课题组要从a, b, c, d, e 5人中派3人分别到上海、广州、香港去开会. 已知a只想去上海，b只想去广州，c, d, e都表示想去广州或香港. 问该课题组在满足个人要求的条件下，共有几种派遣方案?

解 令$G=<V_1,V_2,E>$, 其中$V_1=\{s, g, x\}$, $V_2=\{a, b, c, d, e\}$,

$E=\{(u,v) \mid u\in V_1, v\in V_2, v想去u\}$,

其中s, g, x分别表示上海、广州和香港.



G 满足相异性条件，红边是一个完备匹配, 对应的派遣方案:

a–上海, b–广州, d–香港

应用实例（一）

例 某中学有3个课外活动小组：数学组，计算机组和生物组．有赵，钱，孙，李，周5名学生，问分别在下述3种情况下，能否选出3人各任一个组的组长？

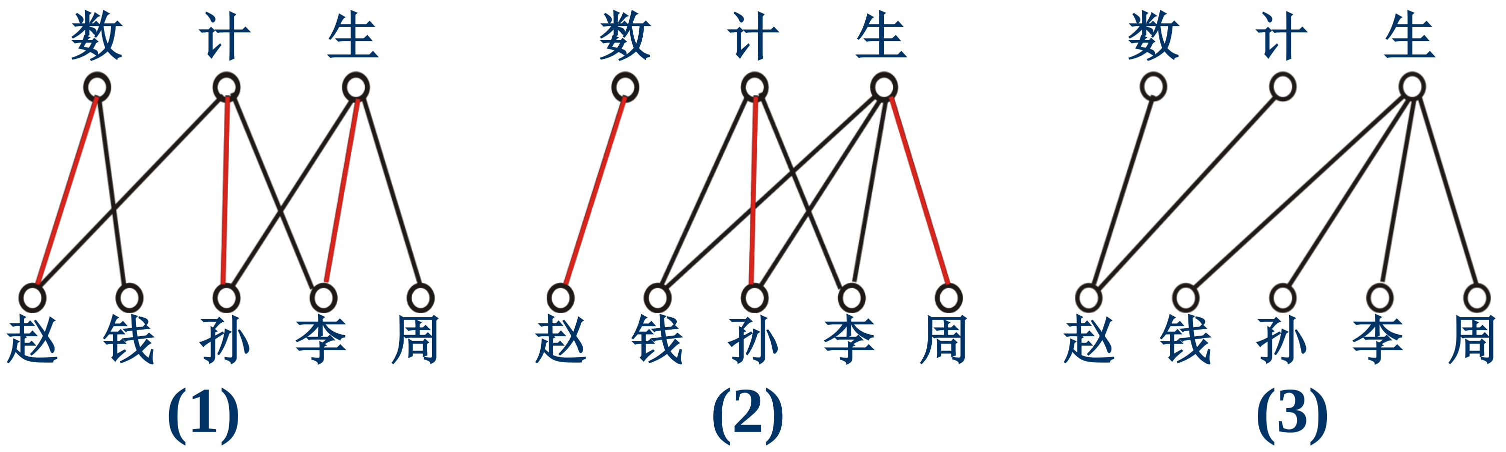
(1) 赵，钱为数学组成员，赵，孙，李为计算机组成员，孙，李，周为生物组成员．

(2) 赵为数学组成员，钱，孙，李为计算机组成员，钱，孙，李，周为生物组成员．

(3) 赵为数学组和计算机组成员，钱，孙，李，周为生物组成员．

应用实例（二）续

解

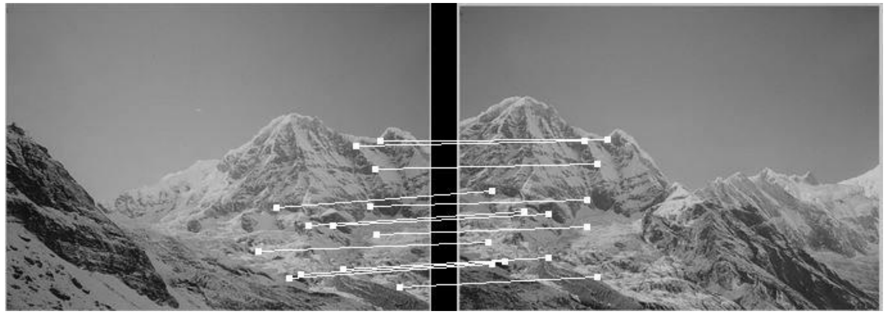


一个完备匹配对应一个方案

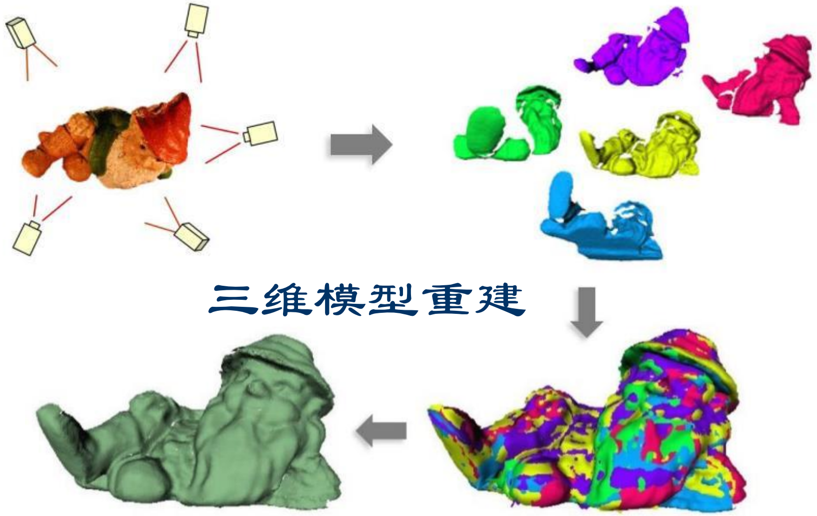
(1), (2)存在完备匹配, 且有多种方案,

(3)不满足相异性条件, 不存在完备匹配。

图像/网格匹配

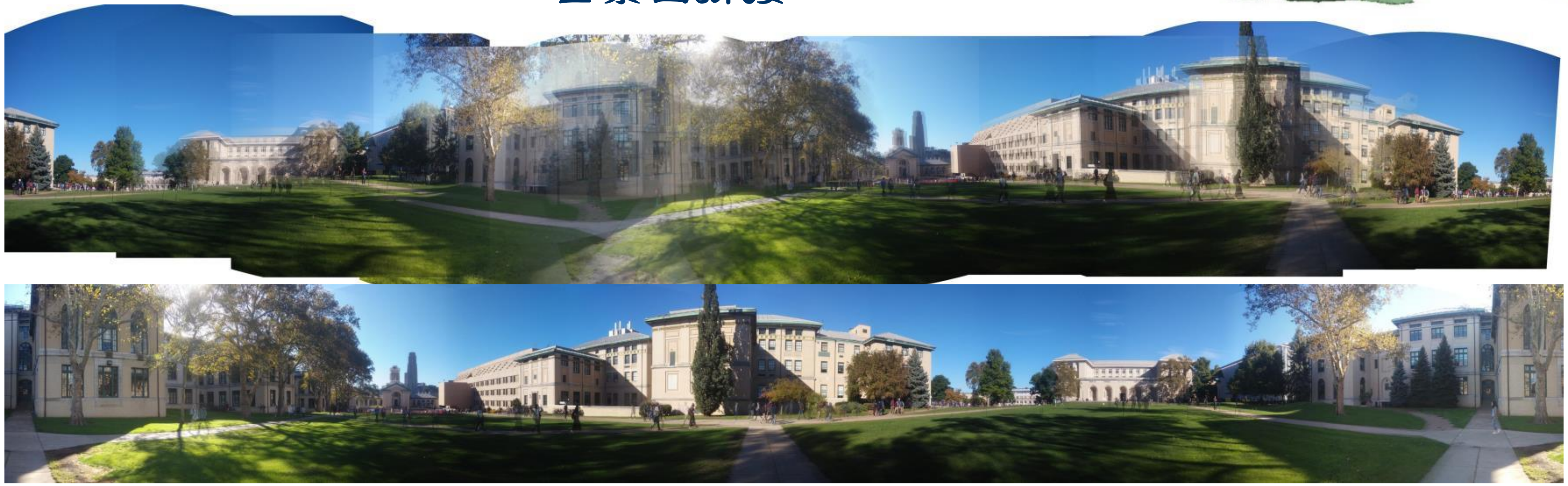


图像配准

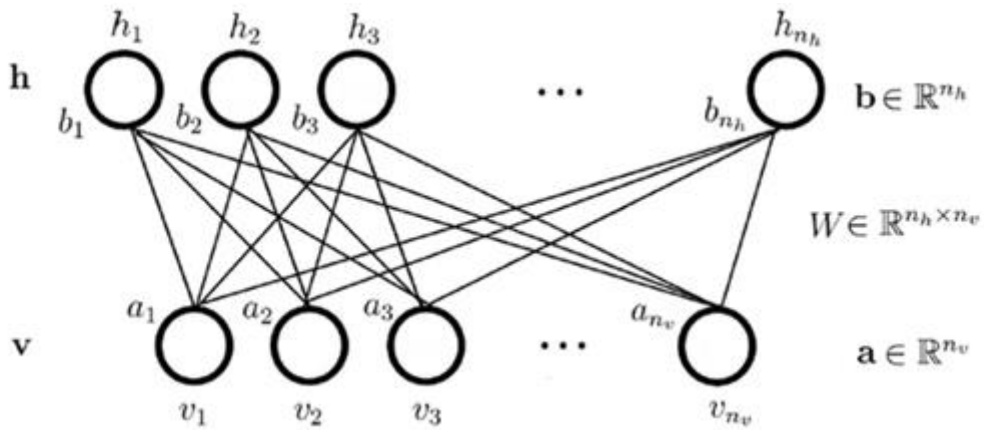


三维模型重建

全景图拼接



受限玻尔兹曼机RBM



v层： 可见层，输入特征。（好比黑白图片，v层就是某处是否为白色）

h层： 隐含层

常量： n_v, n_h --> 可见层和隐含层神经元数目　　num of visiable /hidden

变量： $w_{i,j}$ –> 权值矩阵　a –> 可见层偏置向量　b –> 隐含层偏置向量θ=（w,a,b）–>把所有变量放到一起

状态： v=(v1,v2,...)T　h=(h1,h2,...)T 可见层和隐含层的状态向量

习题

1. 5.18 有向图D如图5-1所示，求D中在定义意义下长度为4的通路总数，并指出其中有多少条是回路？又有几条是v_3到v_4的通路？ (40)

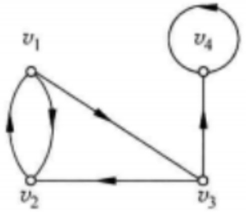


图 5-1

2. 5.21 计算机系期末要安排7门公共课的考试，课程编号为1到7。下列每一对课程有学生同时选修：1和2，1和3, 1和4，1和7, 2和3, 2 和4，2 和5，2和7，3和4, 3和 6，3和7，4和5，4和6，5和6，5和 7,6和7.这7门课的考试至少要安排在几个不同的时间段？给出一个安排方案。 (30)

3. 6.5 今有工人甲、乙、丙要完成3项任务a, b, c,已知甲能胜任a, b, c这3项任务，乙能胜任a, b两项任务，丙能胜任b, c两项任务。你能给出一种安排方案，使每个工人各完成一项他们能胜任的任务吗？ (30)