Chapter11 图与网络分析

(Graph Therory and Network Analysis)

Graph Theory and Network Analysis

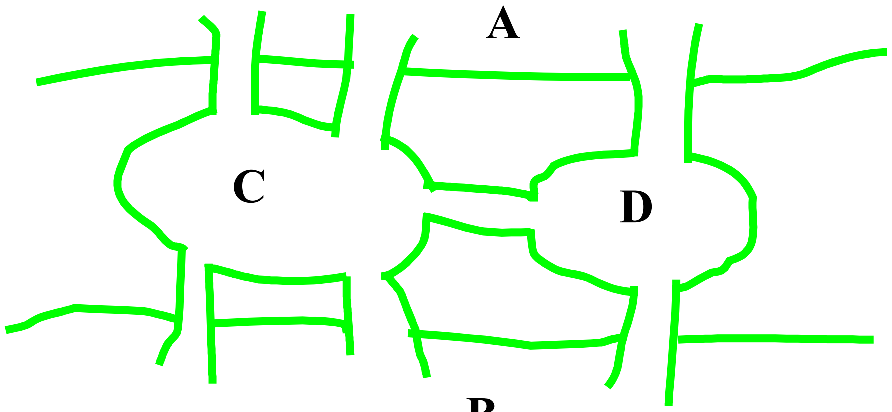
第十一章　图与网络分析

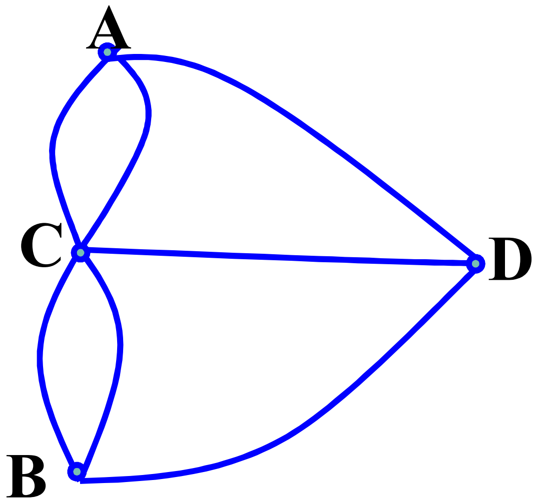
本章主要内容：

- 图与网络的基本知识
- 中国邮路问题
- 最短路问题
- 最大流问题

图与网络的基本知识

图论起源——哥尼斯堡七桥问题

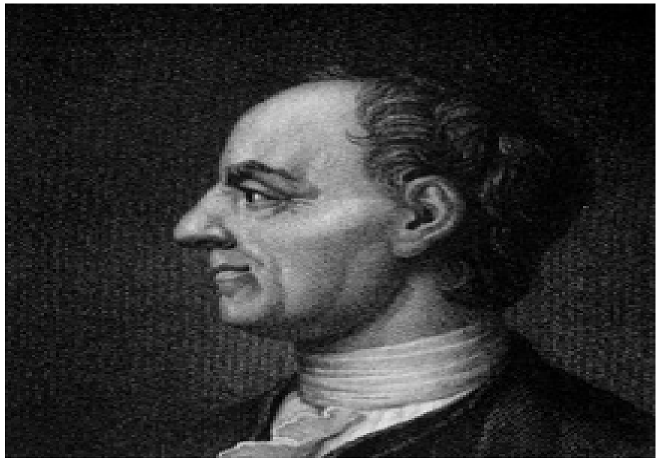




一笔画问题

问题：一个散步者能否从任一块陆地出发，走过七座桥，且每座桥只走过一次，最后回到出发点？ 1736年29岁的欧拉

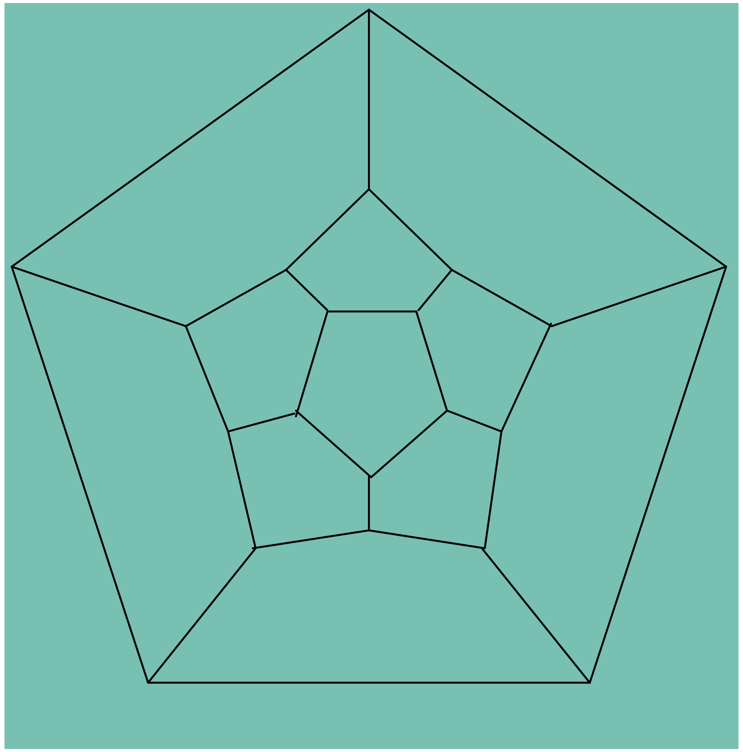
结论：不能。每个结点关联 的边数要均为偶数。



图与网络的基本知识

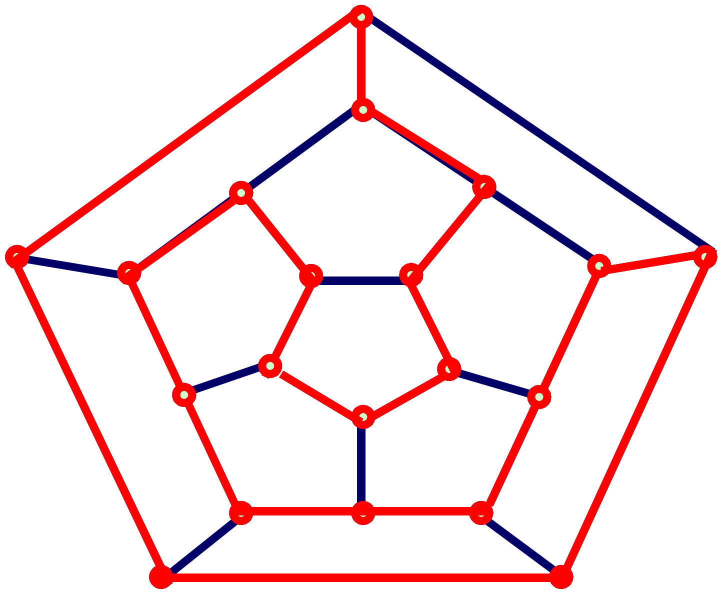
环球旅行问题:

英国数学家**哈密尔顿（Hamilton）**发明了一种游戏
他用一个实心正12面体象征地球，正12面体的20个顶点分别表示世界上20座名城
要求游戏者从任一城市出发，寻找一条可经由每个城市一次且仅一次再回到原出发点的路，这就是“环球旅行”问题。
要在图中找一条经过每个点一次且仅一次的路，称为**哈密尔顿回路**。

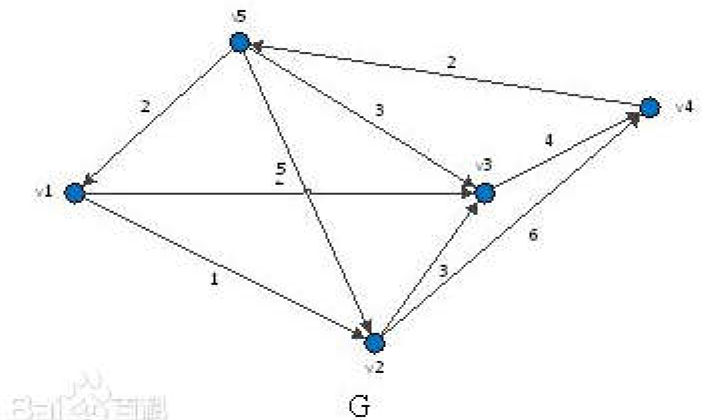


图与网络的基本知识

环球旅行问题的解



另一个著名的问题:
中国邮路问题



给定一个图G，每边e有非负的长度，要求一条回路经过每条边至少一次，且满足总长最短。

图与网络的基本知识

图论中图是由点和边构成，可以反映一些对象之间的关系。一般情况下图中点的相对位置如何、点与点之间联线的长短曲直，对于反映对象之间的关系并不是重要的。

图的定义:

若用点表示研究的对象，用边表示这些对象之间的联系，则图G可以定义为点(顶点)和边的集合，记作:

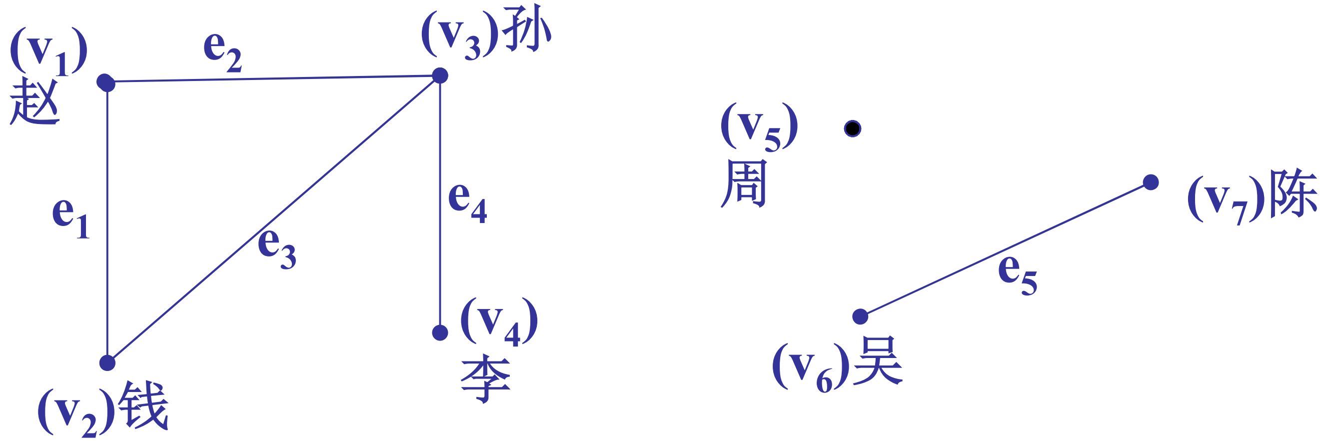
$$G=\{V,E\}$$

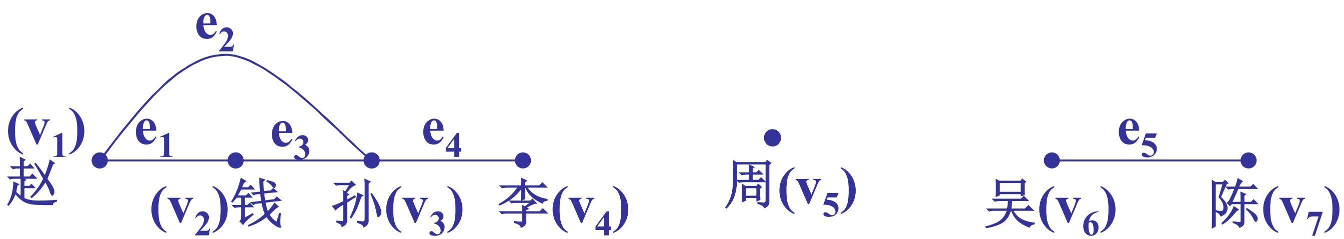
其中: V——点集　E——边集

※ 图G区别于几何学中的图。这里只关心图中有多少个点以及哪些点之间有连线。

图与网络的基本知识

例如：在一个人群中，对相互认识这个关系我们可以用图来表示。





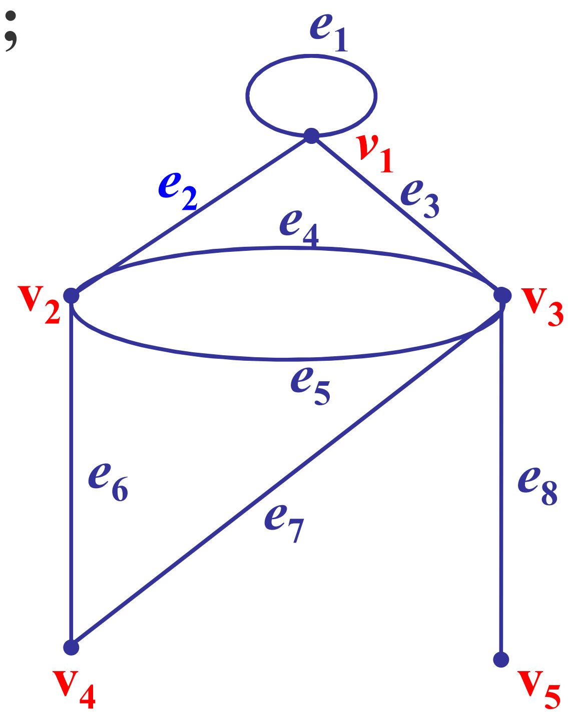
可见图论中的图与几何图、工程图是不一样的。

图与网络的基本知识

定义: 图中的点用v表示，边用e表示。对每条边可用它所连接的点表示，记作：$e_1=[v_1,v_1]$; $e_2=[v_1,v_2]$;

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

$$E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\},$$



端点,关联边,相邻

若有边e可表示为$e=[v_i,v_j]$，称v_i和v_j是边e的端点，反之称边e为点v_i或v_j的关联边。若点v_i、v_j与同一条边关联，称点v_i和v_j相邻；若边e_i和e_j具有公共的端点，称边e_i和e_j相邻。

边数：$m(G)=|E|=m$

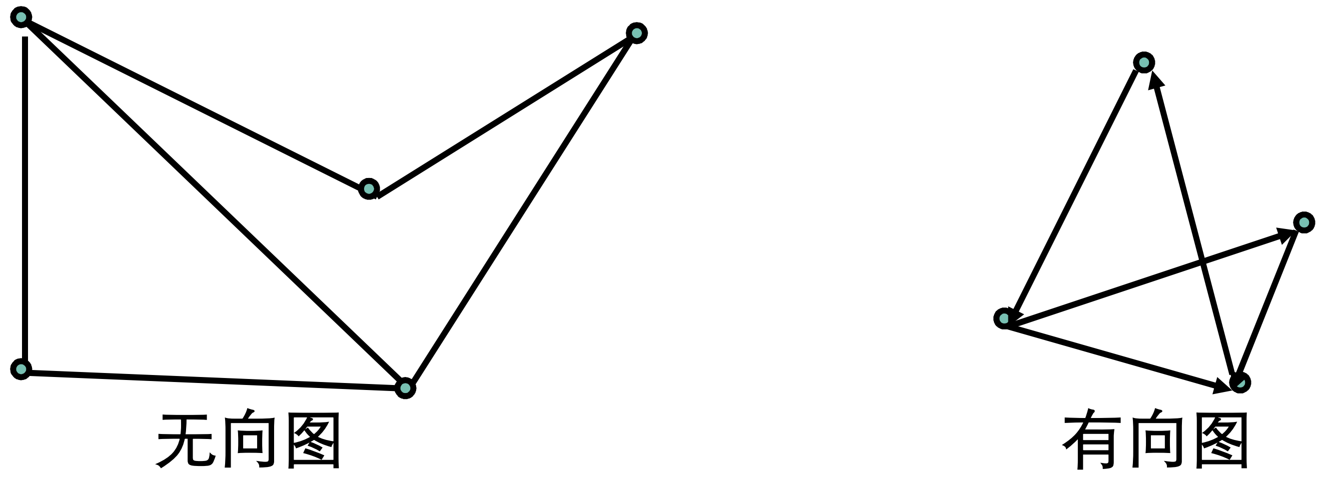
顶点数：$n(G)=|V|=n$

图与网络的基本知识

无向图,有向图

无向边与无向图：若图中任一条边的端点无序，即(v_i, v_j)与(v_j, v_i)是同一条边，则称它为无向边，此时图称为无向图。

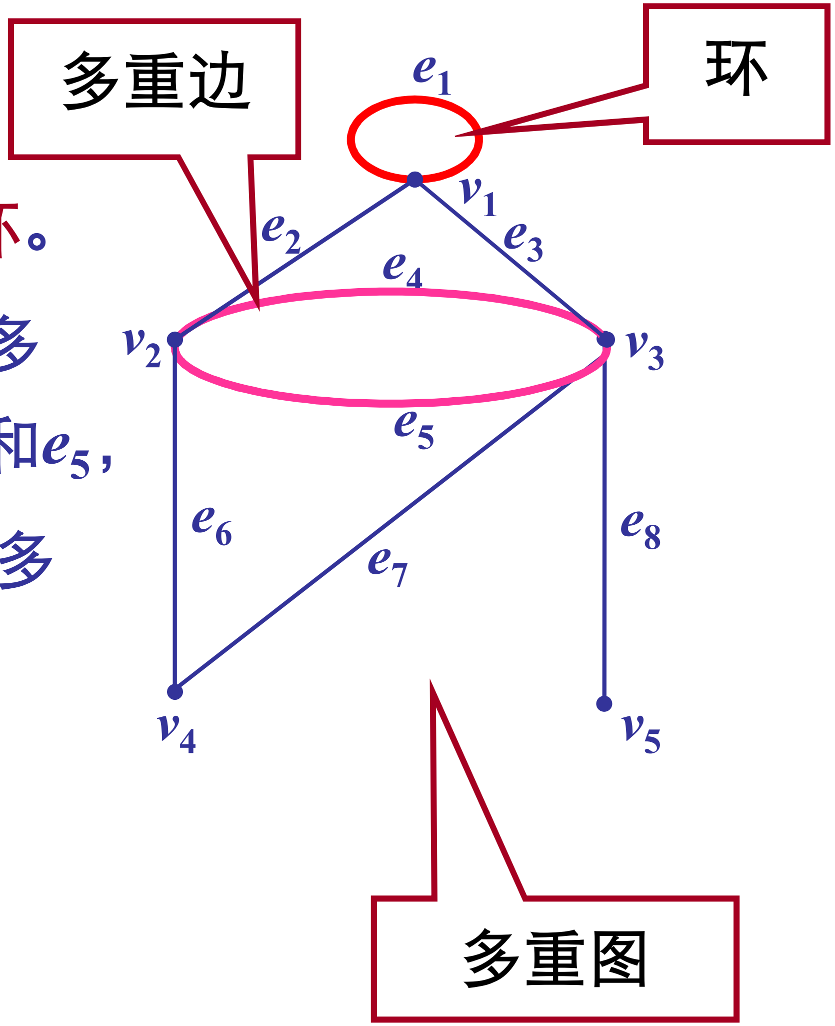
有向图：若图中边(v_i, v_j)的端点是有序的，则称它是有向边（或弧），v_i与v_j分别称为这条有向边的始点和终点，相应的图称为有向图。

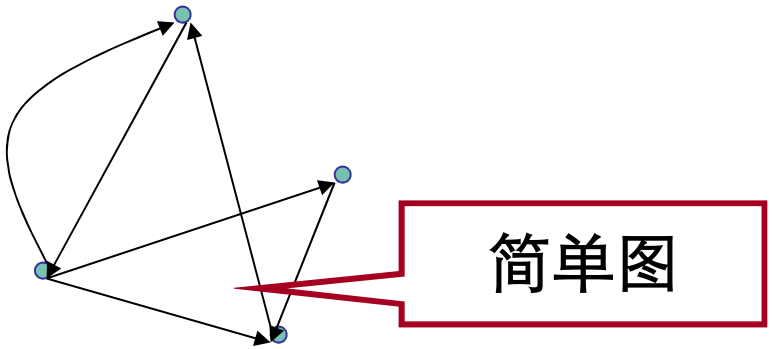


图与网络的基本知识

环, 多重边, 简单图

如果边e的两个端点相重，称该边为环。如右图中边e_1为环。如果两个点之间多于一条，称为多重边，如右图中的e_4和e_5，无环、无多重边的图称作简单图。含多重边的图称为多重图。





图与网络的基本知识

完全图

每一对顶点间都有边相连的无向简单图称为无向完全图；有向完全图是指每一对顶点间有且仅有一条有向边的简单图。

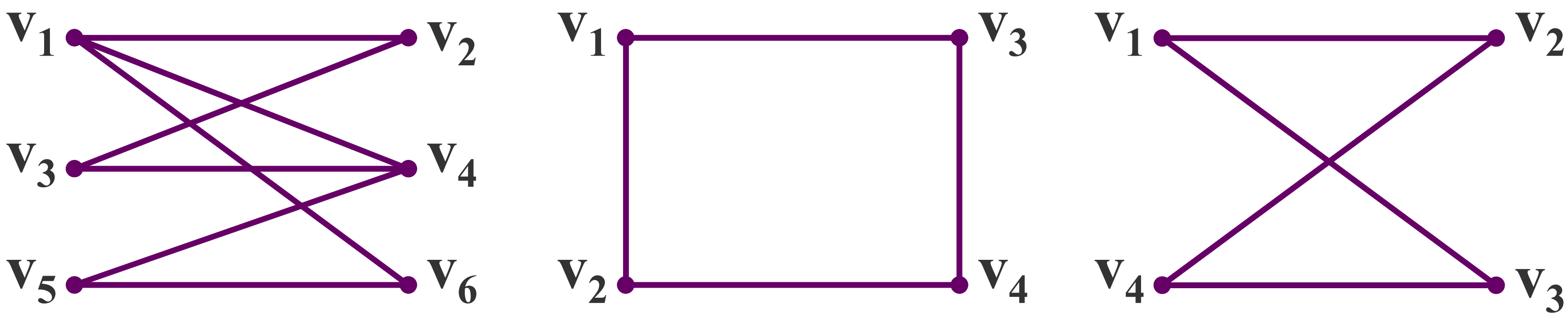
完全图顶点数n与边数m间成立如下关系：

$$m=n(n-1)/2$$

图与网络的基本知识

二部图（偶图）

图G=(V,E)的点集V可以分为两各非空子集X，Y，集X∪Y=V,X∩Y=Ø,使得同一集合中任意两个顶点均不相邻，称这样的图为二部图（偶图）。



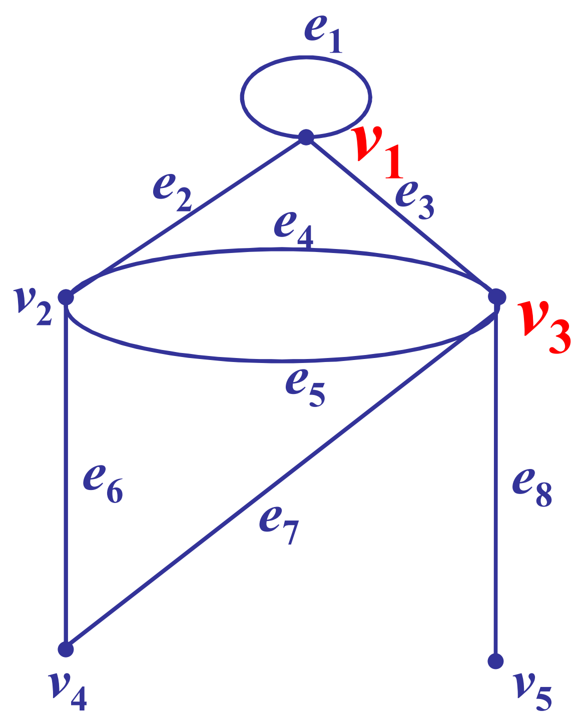
(a) (b) (c)

(a)明显为二部图，(b)也是二部图，但不明显，改画为(c)时可以清楚看出。

图与网络的基本知识

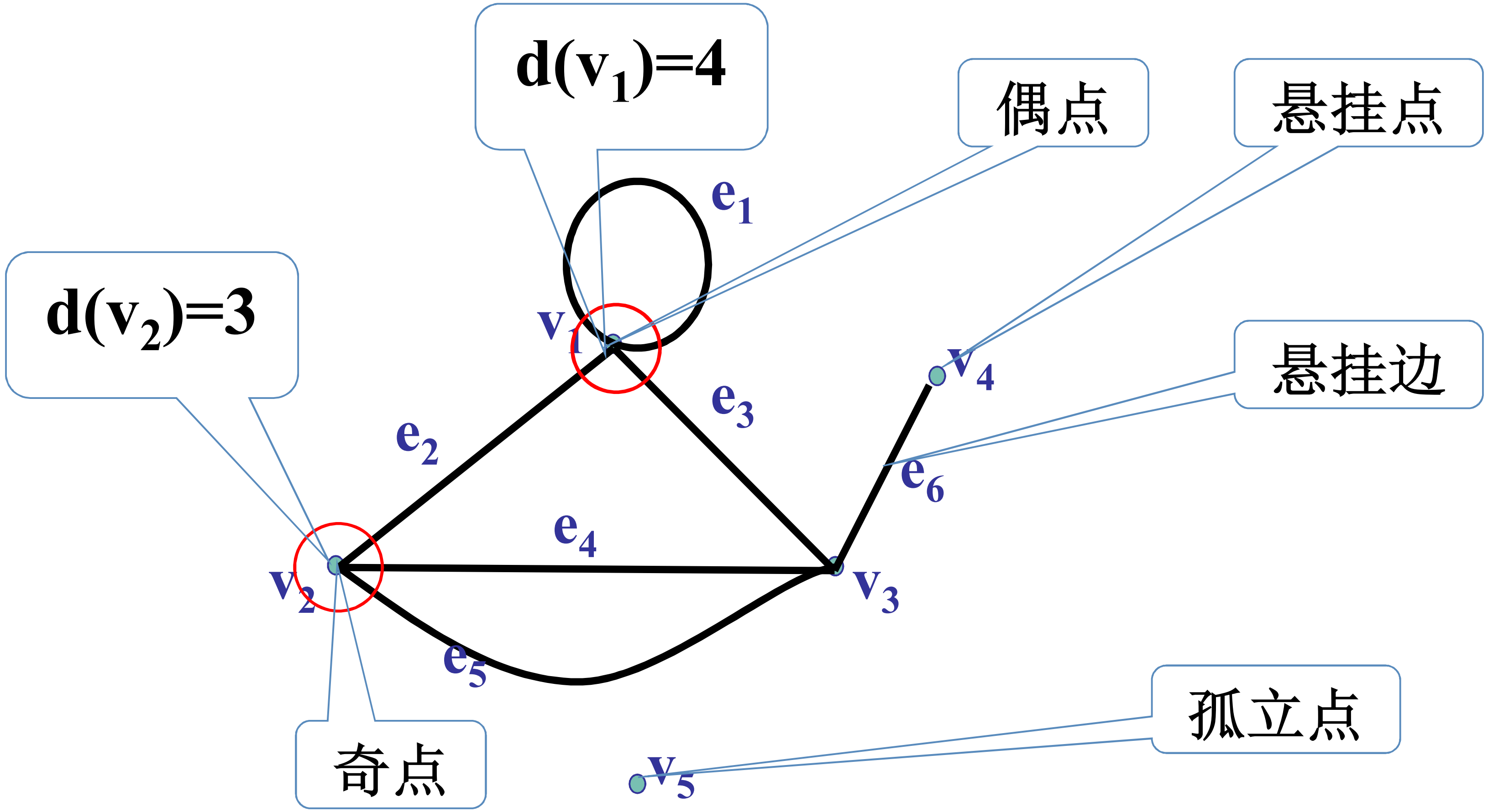
次，奇点，偶点，孤立点

与某一个点v_i相关联的边的数目称为点v_i的次（也叫做度），记作$d(v_i)$。右图中$d(v_1)=4$,$d(v_3)=5$,$d(v_5)=1$。次为奇数的点称作奇点，次为偶数的点称作偶点，次为1的点称为悬挂点，次为0的点称作孤立点。



图的次: 一个图的次等于各点的次之和。

图与网络的基本知识



图的基本概念与模型

图中顶点次的基本性质:

定理1 任何图中，顶点次数之和等于所有边数的2倍。

证明：由于每条边必与两个顶点关联，在计算点的次时，每条边均被计算了两次，所以顶点次数的总和等于边数的2倍。

定理2 任何图中，次为奇数的顶点必为偶数个。

证明：设V1和V2分别为图G中奇点与偶点的集合。由定理1可得:

$$\sum_{v\in V_1} d(v) + \sum_{v\in V_2} d(v) = \sum_{v\in V} d(v) = 2m$$

2m为偶数，且偶点的次之和$\sum_{v\in V_2} d(v)$也为偶数，所以 $\sum_{v\in V_1} d(v)$ 必为偶数，即奇数点的个数必为偶数。

图与网络的基本知识

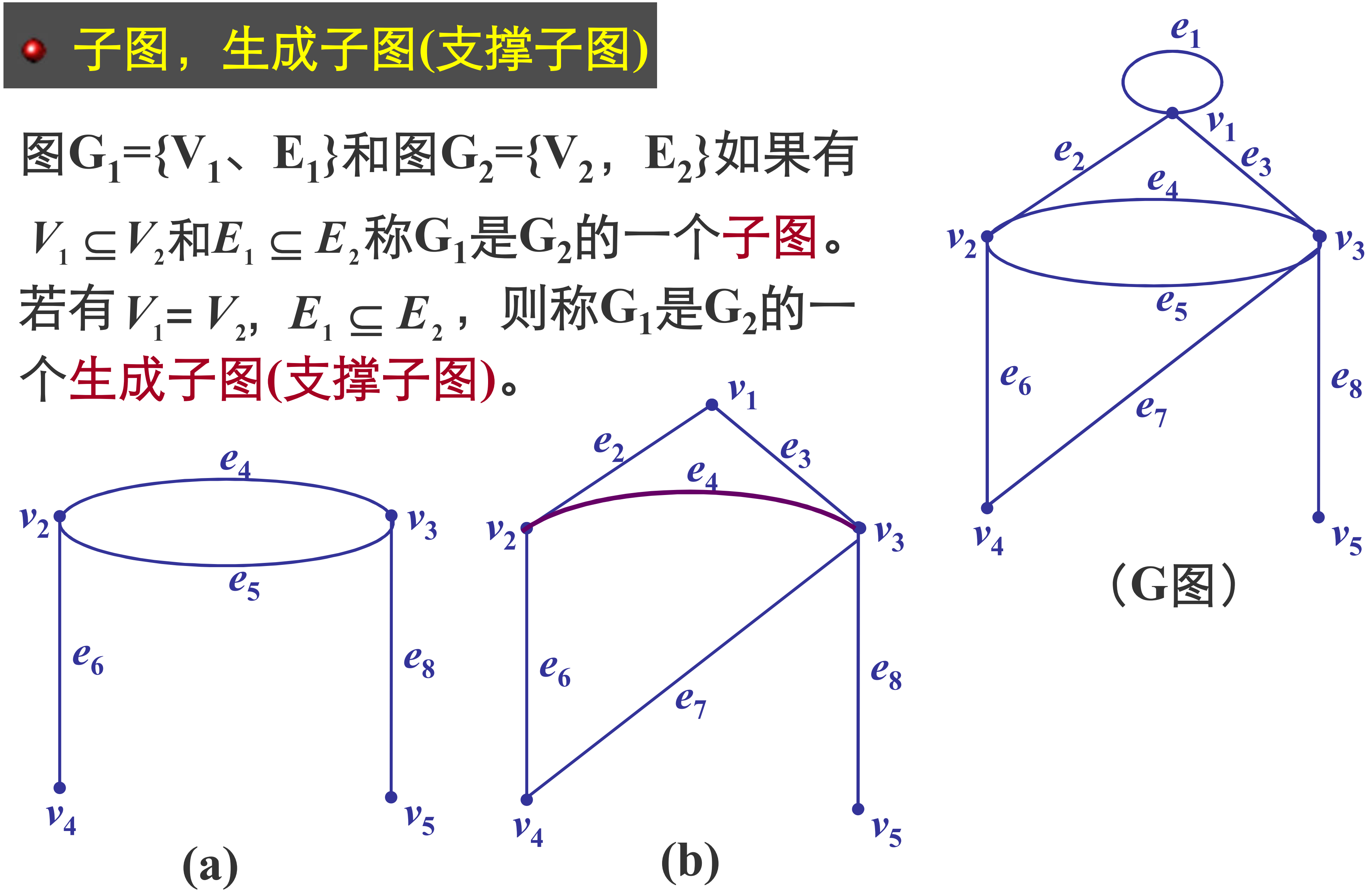
定义6 在有向图中，以顶点v为始点的边数称为顶点v的出次，记为$d^+(v)$；以v为终点的边数称为v的入次，记为$d^-(v)$。顶点v的出次与入次的和称为点v的次。

定义7 图G=(V, E), 若E'是E的子集，若V'是V的子集，且E'中的边仅与V'中的顶点相关联，则称G' = (V', E')为图G的一个子图，特别地，若V' =V, 则称G'为G的一个生成子图（支撑子图）。

图与网络的基本知识

- 子图，生成子图(支撑子图)

图G_1={V_1、E_1}和图G_2={V_2，E_2}如果有 $V_1 \subseteq V_2$和$E_1 \subseteq E_2$称G_1是G_2的一个子图。若有 $V_1 = V_2$，$E_1 \subseteq E_2$，则称G_1是G_2的一个生成子图(支撑子图)。



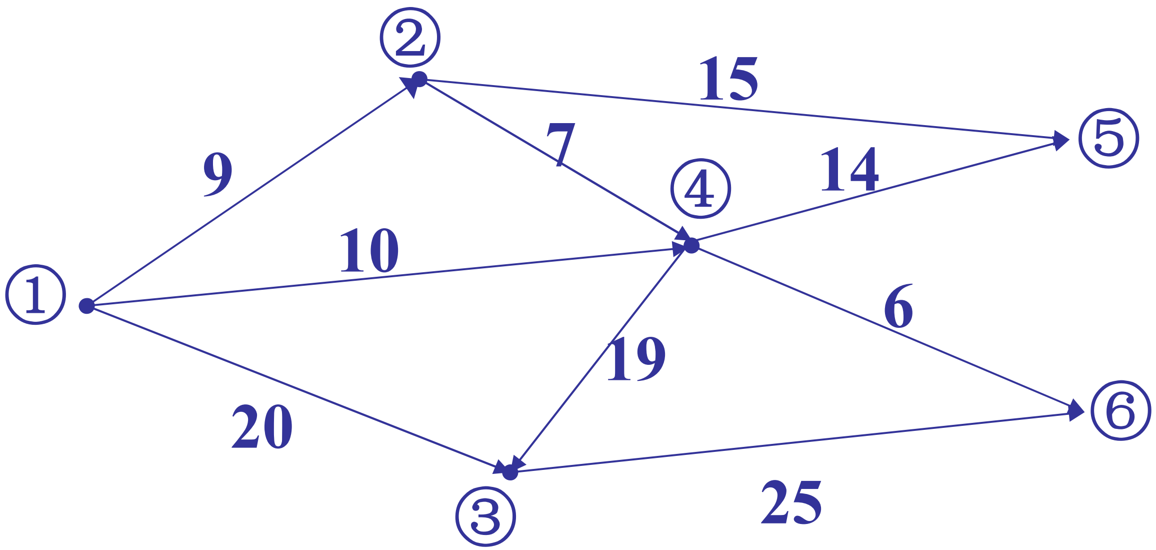
图与网络的基本知识

网络（赋权图）

设图G＝（V，E），对G的每一条边(v_i,v_j)相应赋予数量指标w_{ij}，w_{ij}称为边(v_i,v_j)的权,赋予权的图G称为网络(或赋权图）。

权可以代表距离、费用、通过能力(容量)等等。

端点无序的赋权图称为无向网络，端点有序的赋权图称为有向网络。

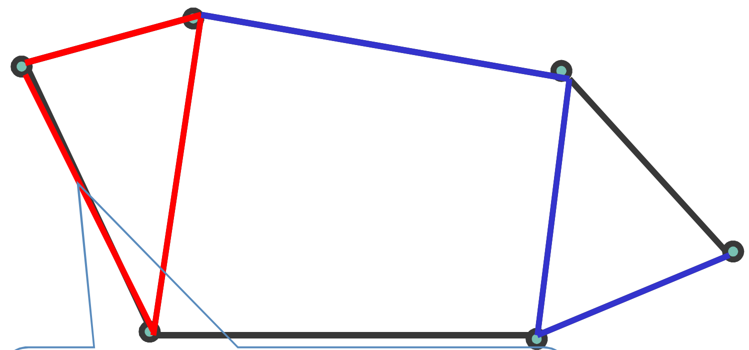


图与网络的基本知识

链，圈，连通图

定义8　无向图中一个点、边交错的序列，序列中的第一个和最后一个元素都是点，若其中每条边以序列中位于它之前和之后的点为端点，则称这个点边序列为图中连接其第一个点与最后一个点的称为**链**。链中所含的边数称为**链长**。

$$\mu = \{v_0, e_1, v_1, \cdots, e_k, v_k\}$$



链，但只是简单链而非初等链

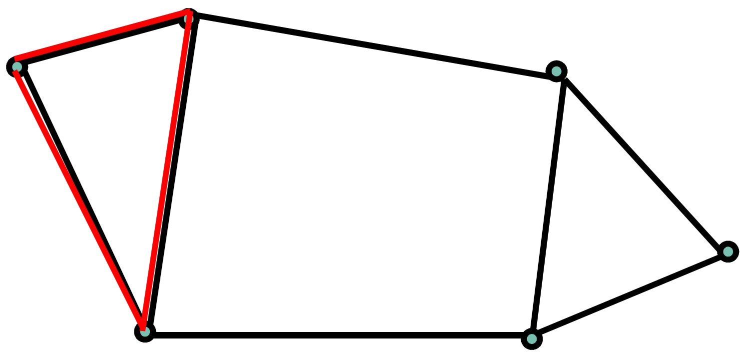
简单链：没有重复边；**初等链**：既无重复边也无重复点。对有向图可类似定义链，如果各边方向一致，则称为**道路**。

图与网络的基本知识

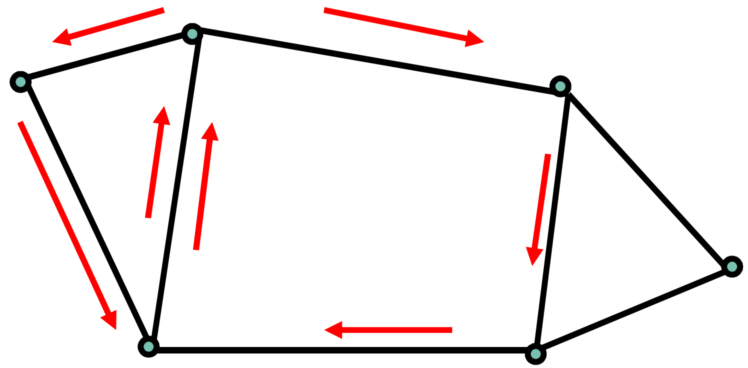
链，圈，连通图

定义9　　若在无向图中，一条链的第一个点与最后一个点重合，则称这条链为圈。只有重复点而无重复边的圈为**简单圈**，既无重复点又无重复边的圈为**初等圈**。

初等圈

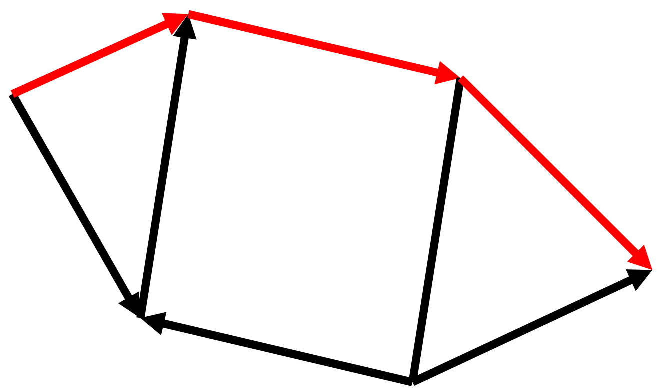


非简单的圈



图与网络的基本知识

有向图	无向图
道路	链（或道路）
回路	圈（或回路）

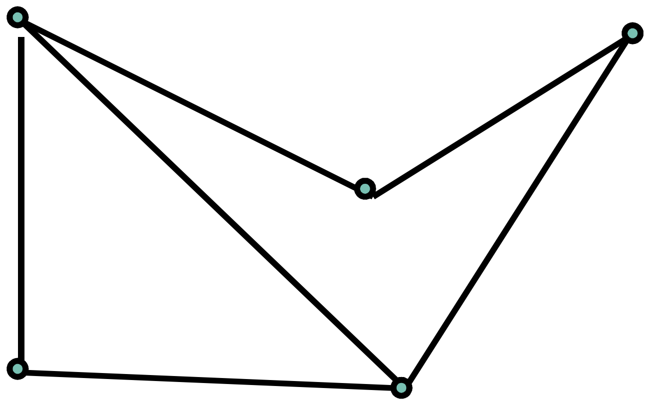


道路(边的方向一致)

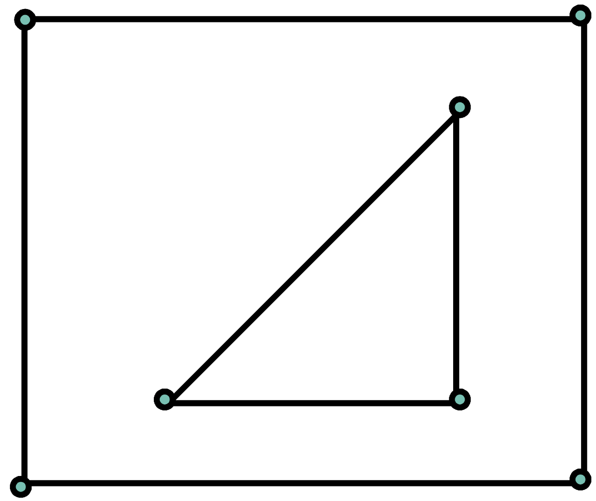
图与网络的基本知识

连通图

定义10 一个图中任意两点间至少有一条链相连，则称此图为连通图。任何一个不连通图总可以分为若干个连通子图，每一个称为原图的一个分图（连通分支）。



连通图

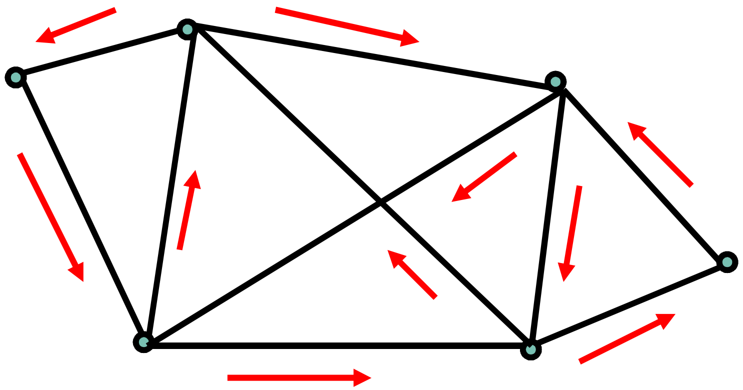


非连通图

欧拉回路

定义13　连通图G中，若存在一条道路，经过每边一次且仅一次，则称这条道路为欧拉道路。若存在一条回路经过每边一次也仅一次，则称这条回路为欧拉回路。

具有欧拉回路的图称为欧拉图（E图）。



定理3 无向连通图G是欧拉图，当且仅当G中无奇点

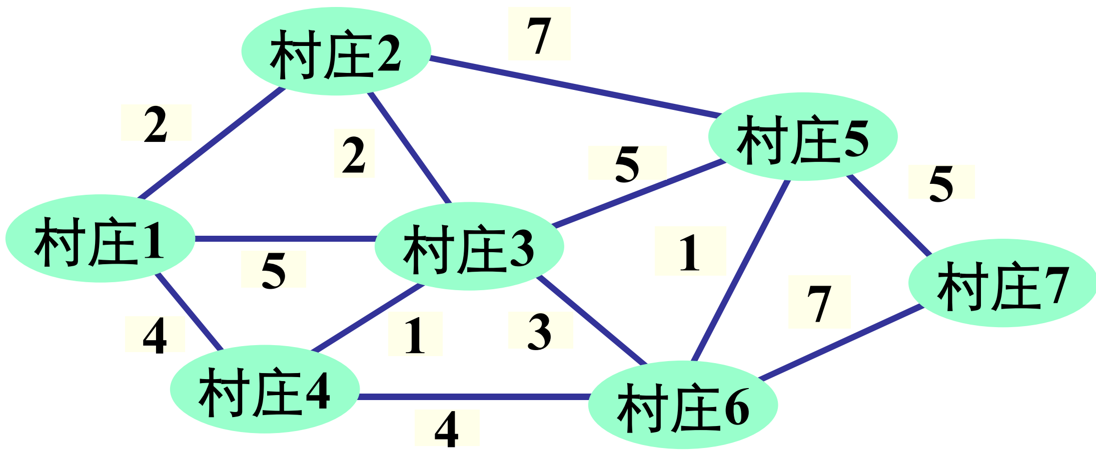
树与最小生成树

7个村庄要在他们之间架设电话线，要求任何两个村庄都可以互相通电话(允许中转)，并且电话线根数最少？

分析：用七个点代表村庄，如果在某两个村庄之间架设电话线，则相应的在两点之间连一条边，这样电话线网就可以用一个图来表示，并且满足如下要求：

为什么？

- 连通图
- 图中有圈的话，从圈中任去掉一条边，余下的图仍连通

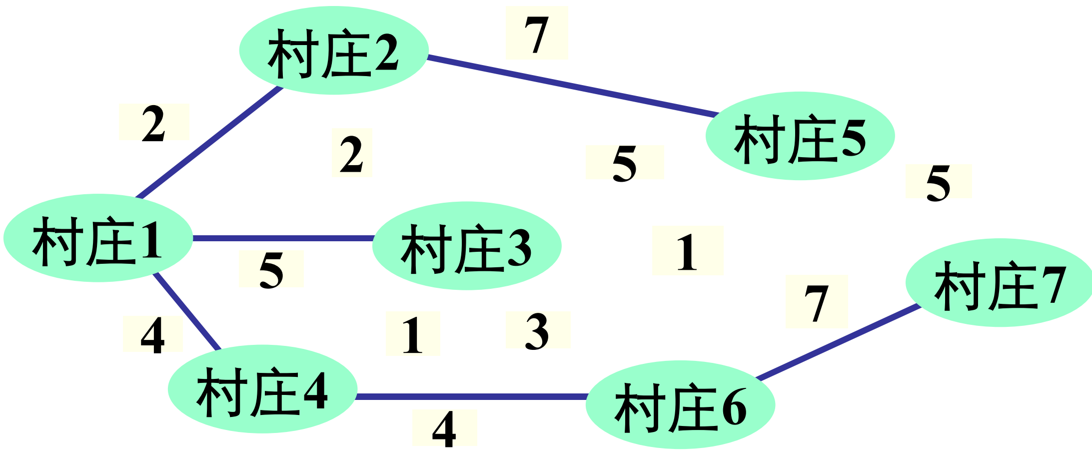


树与最小生成树

如果G=（*V*，*E*）是一个无圈的连通图，则称G为树。

- 树中任两点必有一条链且仅有一条链；
- 在树的两个不相邻的点之间添加一条边，就得到一个圈；反之，去掉树的任一条边，树就成为不连通图；
- *n*个顶点的树有（*n*-*1*）条边。

树是无圈连通图中边数最多的，也是最脆弱的连通图！



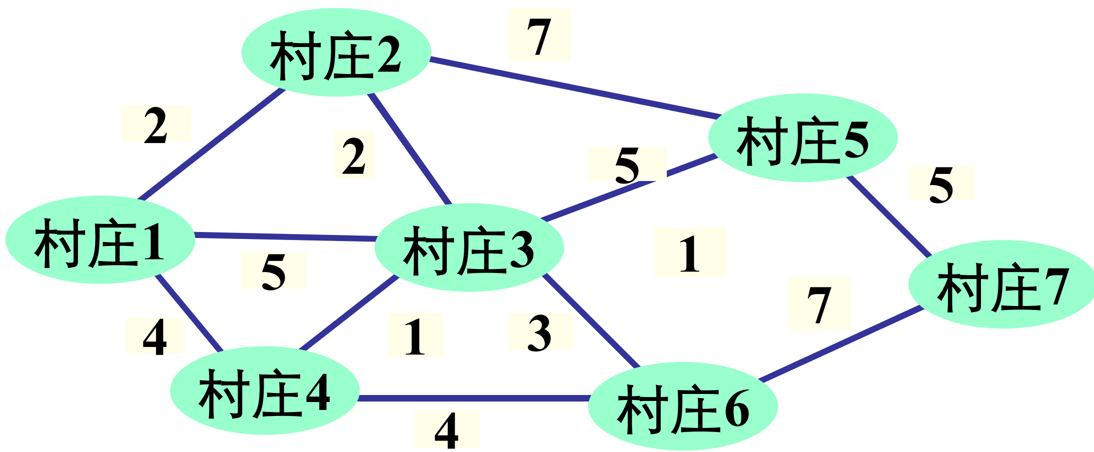
图的生成树

点保留
边可去
仍是树
不唯一

- 如果图G=（*V*，*E*）的部分图G’=（*V*，*E*’）是树，则称G’是G的生成树。

- 生成树上各树枝上权值的和称为它的长度，其中长度最短的生成树，称其为该图的最小生成树。

思考：如何铺设电话线，使得电话线长度最少？

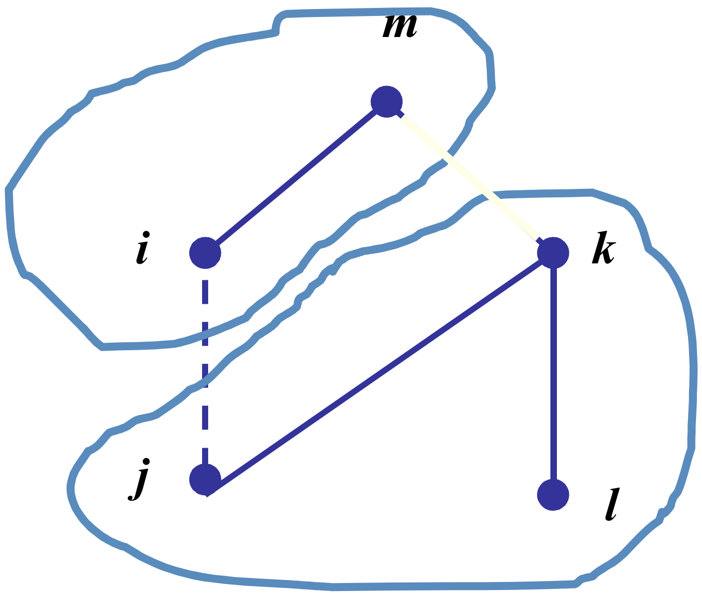


树与最小生成树

定理： 图中任一个点i，若j是与相邻点中距离最近的，则边$[i,j]$一定必含在该图的最小生成树内。

推论： 把图的所有点分成V 和$\overline{V}$ 两个集合，则两集合之间连线的最短边一定包含在最小生成树内。

V



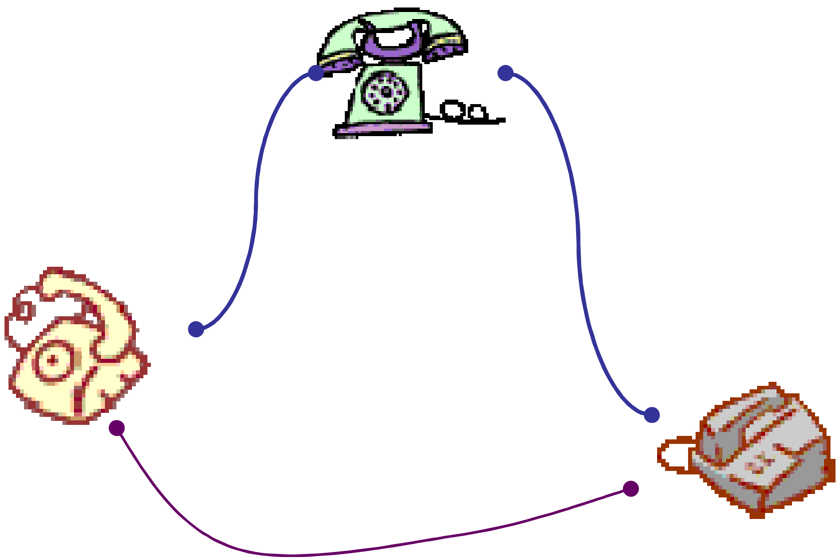
$\overline{V}$

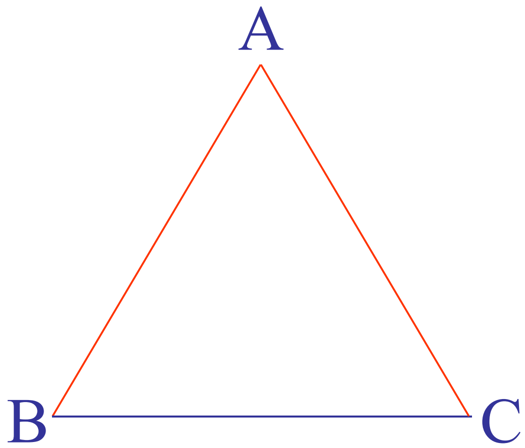
- 避圈法
- 破圈法

最短路问题

如何用最短的线路将三部电话连起来?

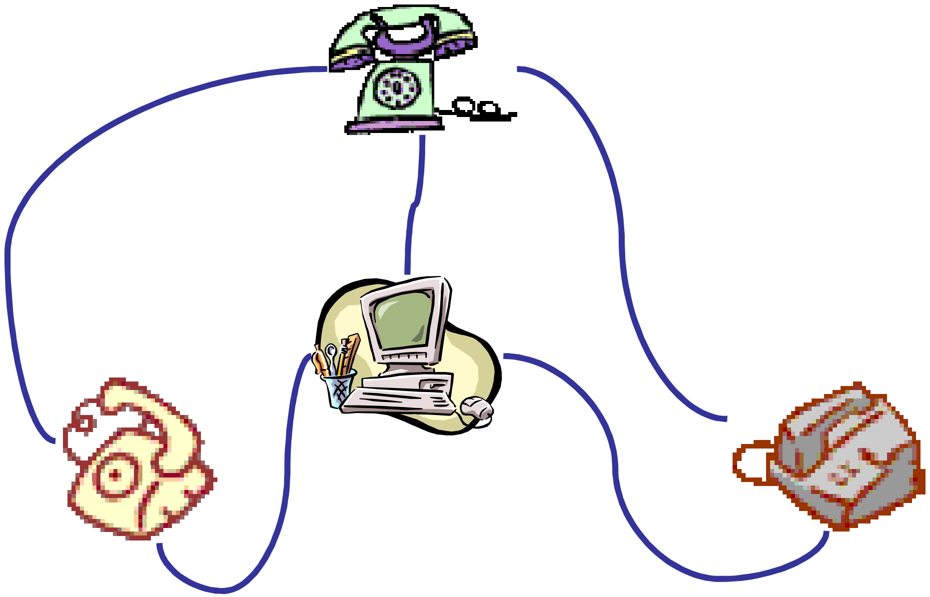
此问题可抽象为设△**ABC**为等边三角形，连接三顶点的路线（称为网络）。这种网络有许多个，其中最短路线者显然是二边之和（如**AB**∪**AC**）。

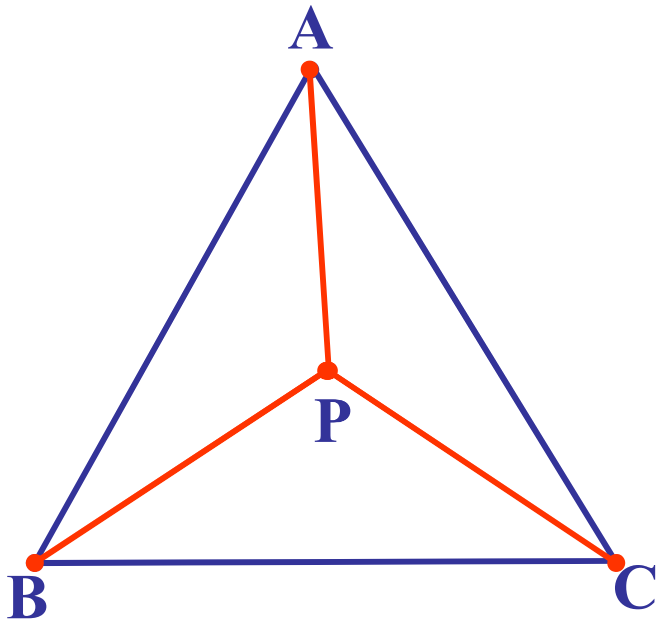




最短路问题

但若增加一个周转站（新点P），连接4点的新网络的最短路线为PA＋PB＋PC。最短新路径之长N比原来只连三点的最短路径O要短。这样得到的网络不仅比原来节省材料，而且稳定性也更好。





最短路问题

问题描述：

就是从给定的网络图中找出一点到各点或任意两点之间距离最短的一条路．

有些问题，如选址、管道铺设时的选线、设备更新、投资、某些整数规划和动态规划的问题，也可以归结为求最短路的问题。因此这类问题在生产实际中得到广泛应用。

最短路问题

例6.4 渡河游戏

一老汉带了一只狼、一只羊、一捆草料要从南岸过河到北岸，河上只有一条独木舟，每次除了人以外，只能带一样东西；另外，如果人不在，狼就要吃羊，羊就要吃草料，问应该怎样安排渡河，才能做到既把所有东西都运过河去，并且在河上来回次数最少？这个问题就可以用求最短路方法解决。

最短路问题

定义:

1）人—M（Man），狼—W（Wolf）， 羊—G（Goat），草—H（Hay）

2） 点—— v_i 表示河岸的状态

3） 边—— e_k 表示由状态 v_i 经一次渡河到状态 v_j

4） 权——边 e_k 上的权定为 1

我们可以得到下面的加权有向图

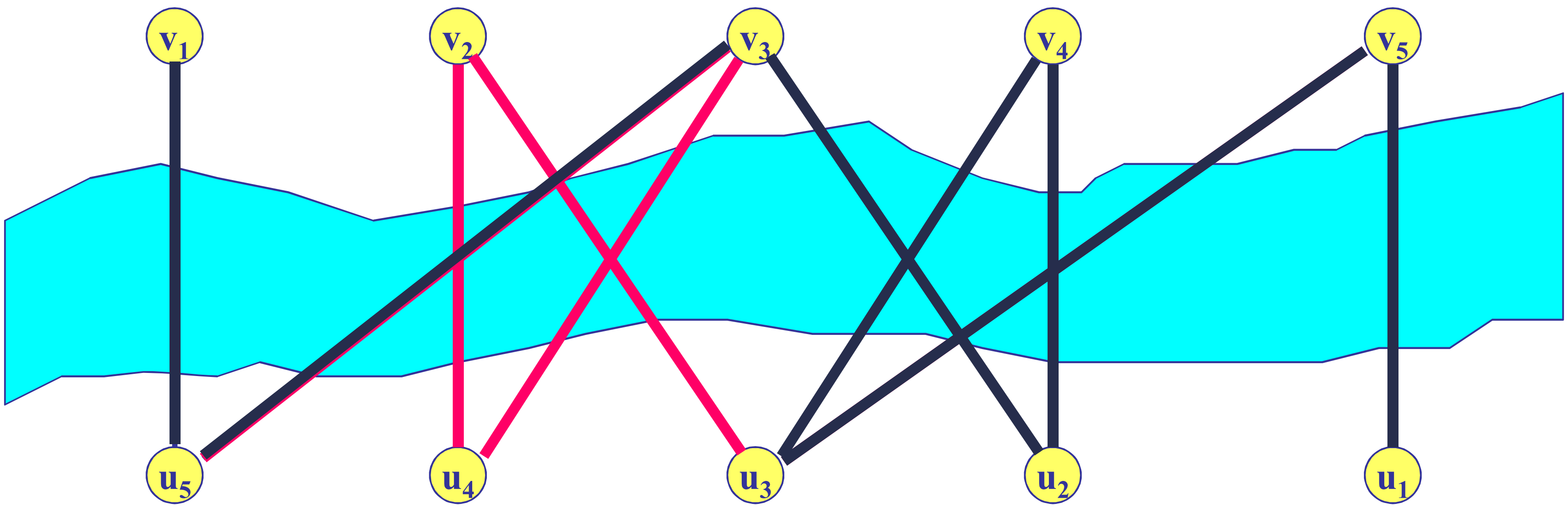
最短路问题

状态说明:

v_1,u_1 =(M,W,G,H); v_2,u_2 =(M,W,G); v_3,u_3 =(M,W,H);

v_4,u_4=(M,G,H); v_5,u_5 =(M,G)

此游戏转化为在下面的二部图中求从 v_1 到 u_1 的最短路问题。



最短路问题

求最短路有两种算法:

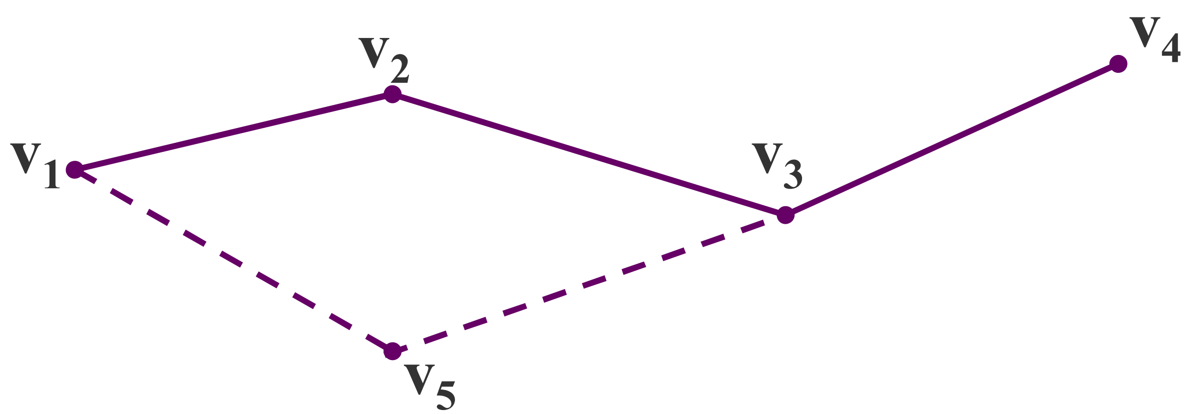
- 狄克斯屈拉(Dijkstra)标号算法
- 逐次逼近算法

最短路问题

狄克斯屈拉(Dijkstra)标号算法的基本思路:

若序列{ $v_s, v_1 \ldots\ldots v_{n-1}, v_n$ }是从v_s到v_t间的最短路，则序列{ $v_s, v_1 \ldots\ldots v_{n-1}$ } 必为从v_s 到v_{n-1}的最短路。

假定$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4$是$v_1 \rightarrow v_4$的最短路，则$v_1 \rightarrow v_2 \rightarrow v_3$一定是$v_1 \rightarrow v_3$的最短路，$v_2 \rightarrow v_3 \rightarrow v_4$也一定是$v_2 \rightarrow v_4$的最短路。



最短路问题

求网络图的最短路，设图的起点是v_s,终点是v_t ,以v_i为起点v_j为终点的弧记为 (i, j) 距离为d_{ij}

P标号(点标号)：b(j) —起点v_s到点v_j的最短路长；

T标号(边标号)： k(i,j)=b(i)+d_{ij},

步骤：

1. 令起点的标号；$b(s) = 0$。

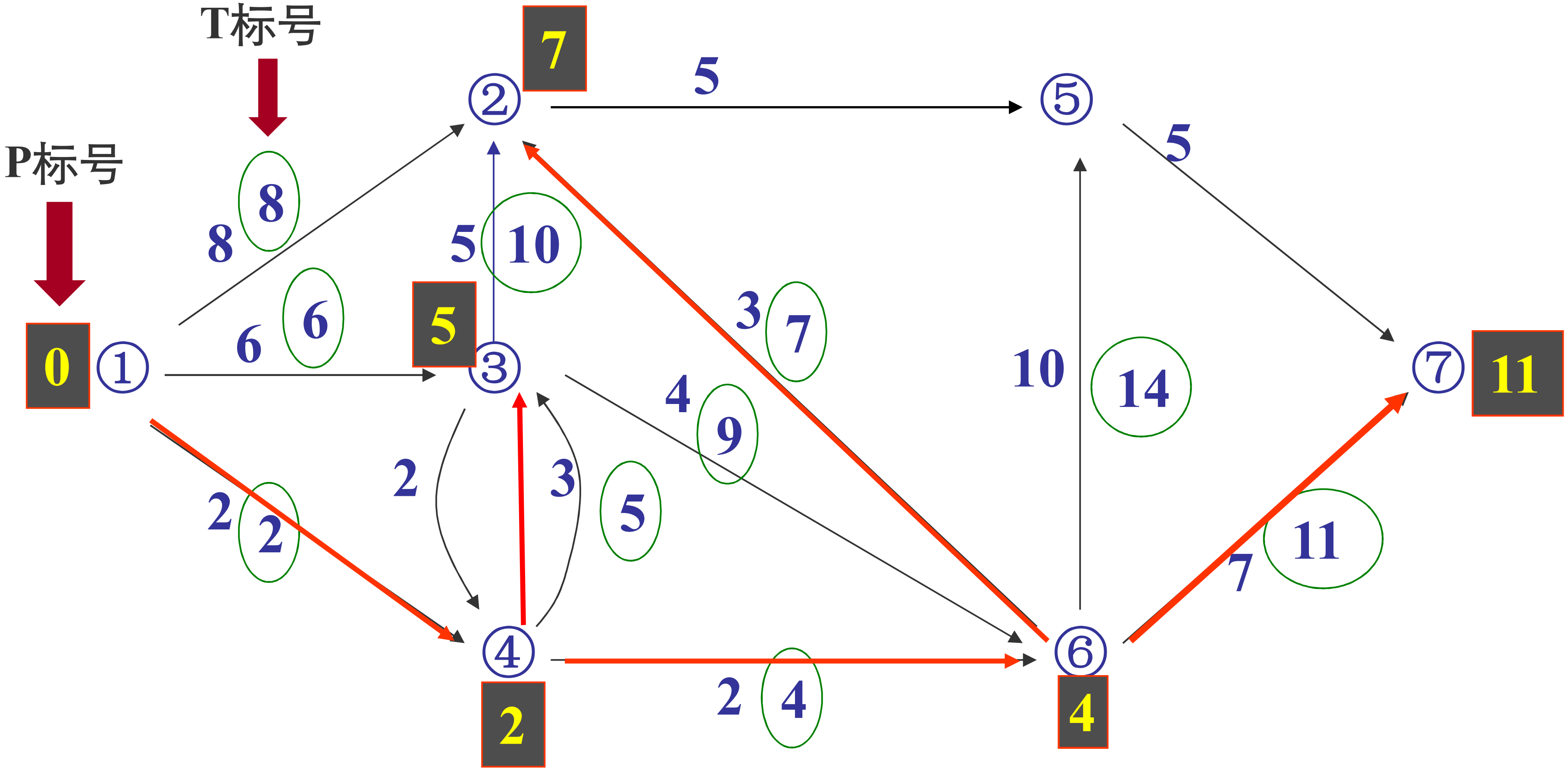
2. 找出所有v_i已标号v_j未标号的弧集合 B={(i, j)} 如果这样的弧不存在或v_t已标号则计算结束；

3. 计算集合B中弧$k(i，j)=b(i)+d_{ij}$的标号

4. 选一个点标号 $b(l) = \min_j \{k(i, j) \mid (i, j) \in B\}$, 在终点$v_l$处标号b(l)，返回到第2步。

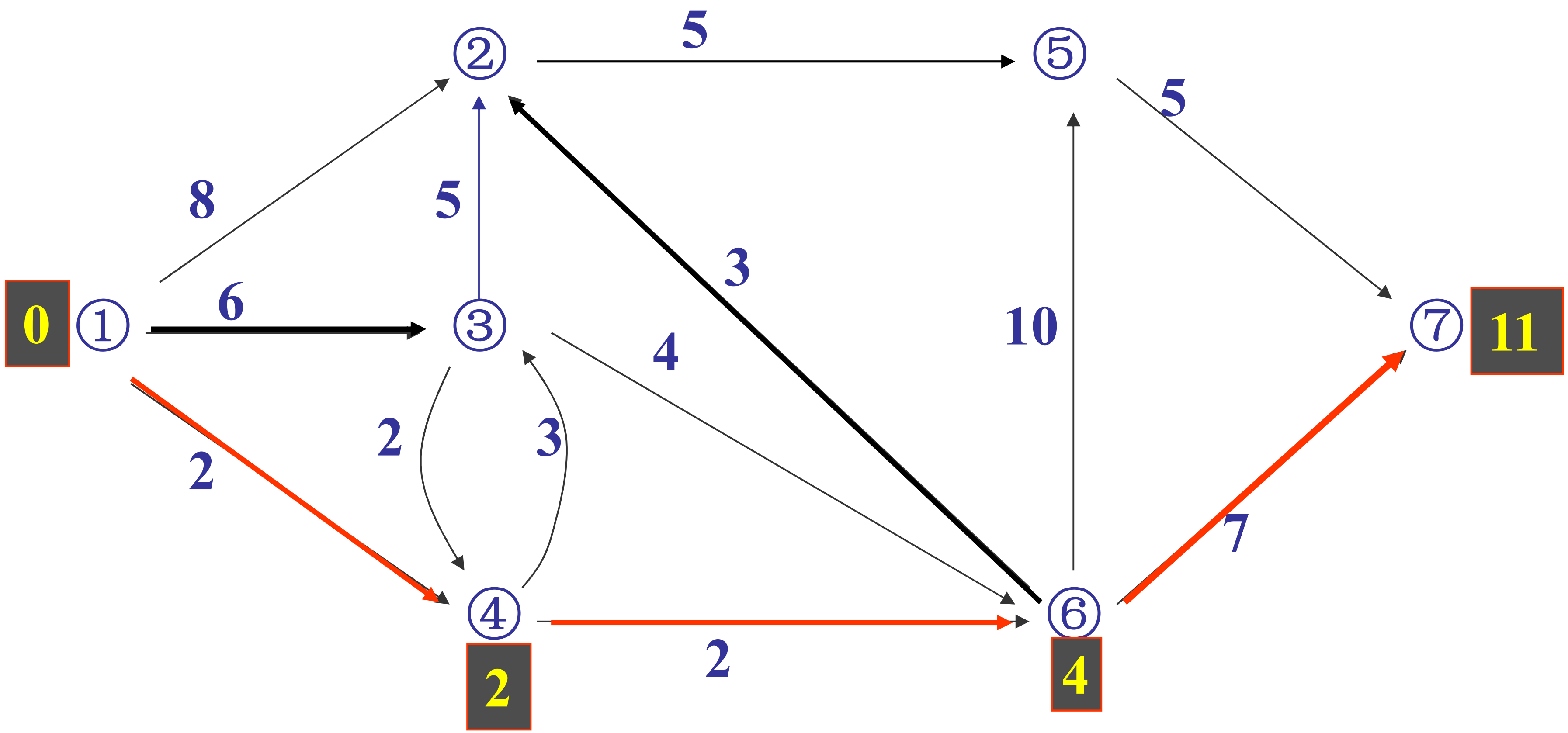
最短路问题

例6.5 求下图v_1到v_7的最短路长及最短路线



最短路问题

v_1到v_7的最短路长及最短路线如图所示：



v_7已标号，计算结束。从v_1到v_7的最短路长是 11，

最短路线：$v_1 \rightarrow v_4 \rightarrow v_6 \rightarrow v_7$

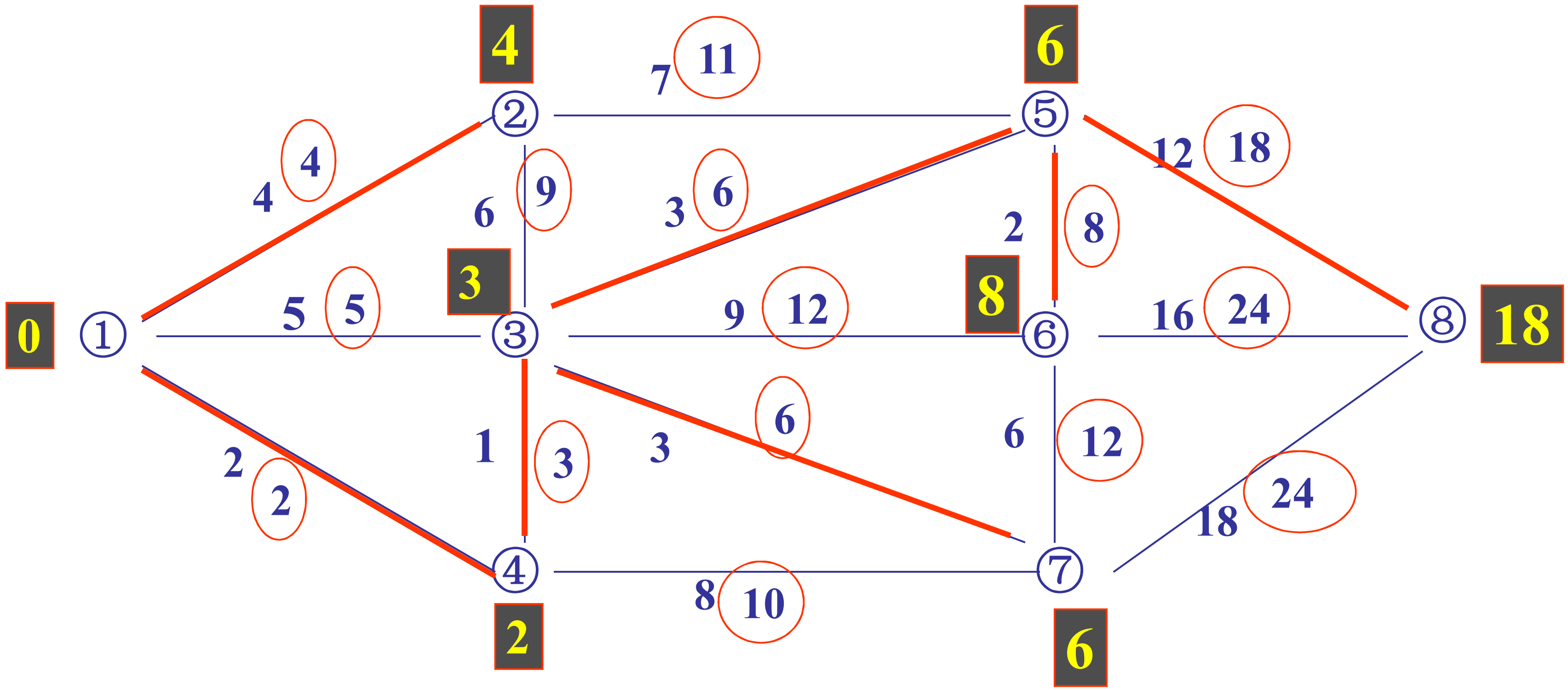
最短路问题

从上例知，只要某点已标号，说明已找到起点v_s到该点的最短路线及最短距离，因此可以将每个点标号，求出v_s到任意点的最短路线，如果某个点v_j不能标号，说明v_s不可达v_j。

注：无向图最短路的求法只将上述步骤2将弧改成边即可。

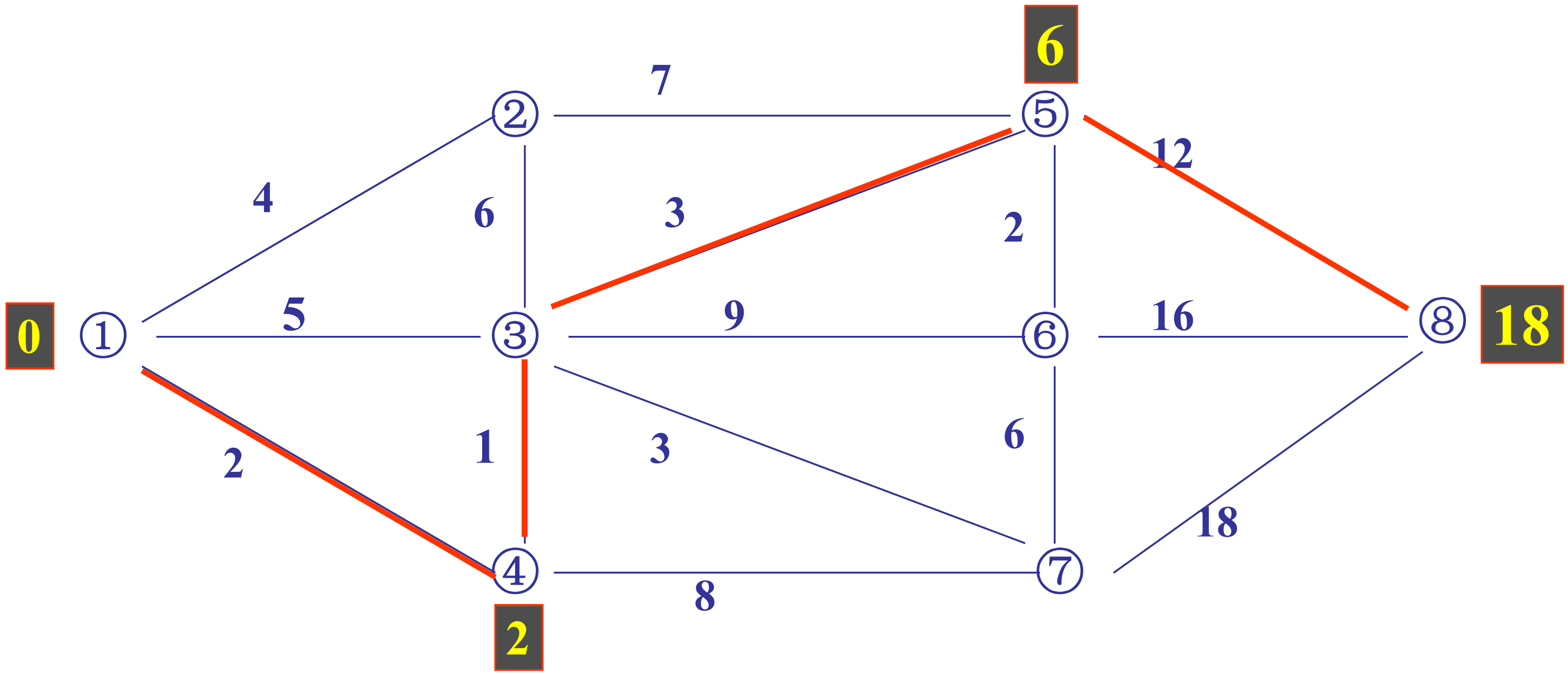
最短路问题

例6.6 求下图v_1到各点的最短距离及最短路线。



最短路问题

v_1到各点的最短距离及最短路线如图所示:

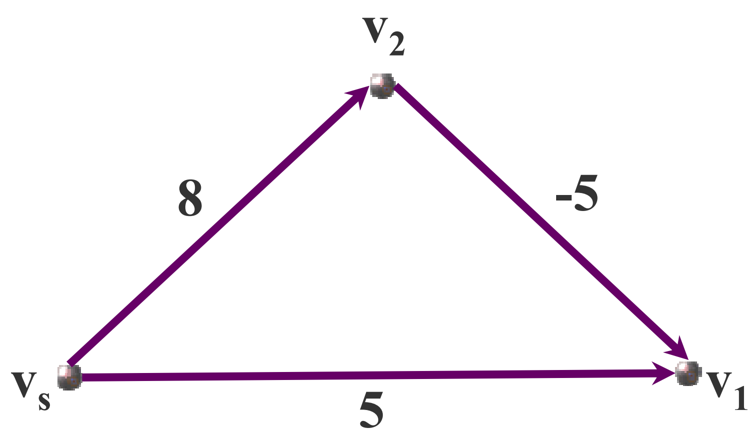


所有点都已标号，点上的标号就是v_1到该点的最短距离，最短路线就是红色的链。

最短路问题

算法适用条件：

Dijkstra算法只适用于全部权为非负情况，如果某边上权为负的，算法失效。



例6.7 如右图所示中按dijkstra算法可得$P(v_1)=5$为从$v_s \to v_1$的最短路长显然是错误的，从$v_s \to v_2 \to v_1$路长只有3。

那么若有某些 $w_{ij} < 0$，如何求解？

最短路问题

设 $d^{(k)}(v_s,v_j)$ 为 v_s 至 v_j 的至多含 $k-1$ 个中间点的最短路长，则

i) $d^{(1)}(v_s,v_j)=w_{sj} \quad (j=1,\cdots,|V|)$

ii) 对 $k=2,3,\cdots$

$$d^{(k)}(v_s,v_j)=\min_i\{d^{(k-1)}(v_s,v_i)+w_{ij}\} \quad (j=1,\cdots,|V|)$$

若 G 不含权和为负数的圈，则 v_s 至每个 v_j 的最短路至多有 $|V|-2$ 个中间点，此时，v_s 至 v_j 的最短路长为

$d^{(|V|-1)}(v_s,v_j)$，

且 $d^{(|V|)}(v_s,v_j)=d^{(|V|-1)}(v_s,v_j)$.

最短路问题

例6.8 设备更新问题。某公司使用一台设备，在每年年初，公司就要决定是购买新的设备还是继续使用旧设备。如果购置新设备，就要支付一定的购置费，当然新设备的维修费用就低。如果继续使用旧设备，可以省去购置费，但维修费用就高了。请设计一个五年之内的更新设备的计划，使得五年内购置费用和维修费用总的支付费用最小。已知：

设备每年年初的价格表

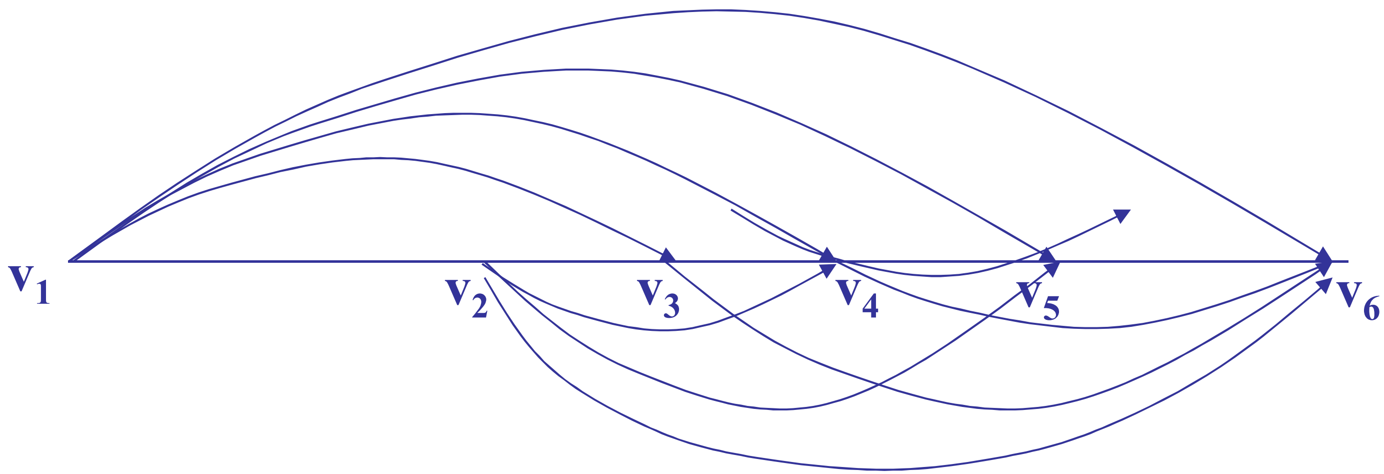
年份	1	2	3	4	5
年初价格	11	11	12	12	13

最短路问题

设备维修费如下表

使用年数	0-1	1-2	2-3	3-4	4-5
每年维修费用	5	6	8	11	18

解：将问题转化为最短路问题，如下图：用v_i表示“第i年年初购进一台新设备”,弧（v_i,v_j）表示第i年年初购进的设备一直使用到第j年年初。



最

表，把权数赋

$W_{12}=11+5=16$

$W_{13}=11+5+6=22$

$W_{14}=11+5+6+8=30$

$W_{15}=11+5+6+8+11=41$

$W_{16}=11+5+6+8+11+18=59$

$W_{34}=12+5=17$

$W_{35}=12+5+6=23$

$W_{36}=12+5+6+8=31$

						6
			22	30		59
2			16	22	3	41
3				17	23	31
4					17	23
						18

$W_{23}=11+5=16$

$W_{24}=11+5+6=22$

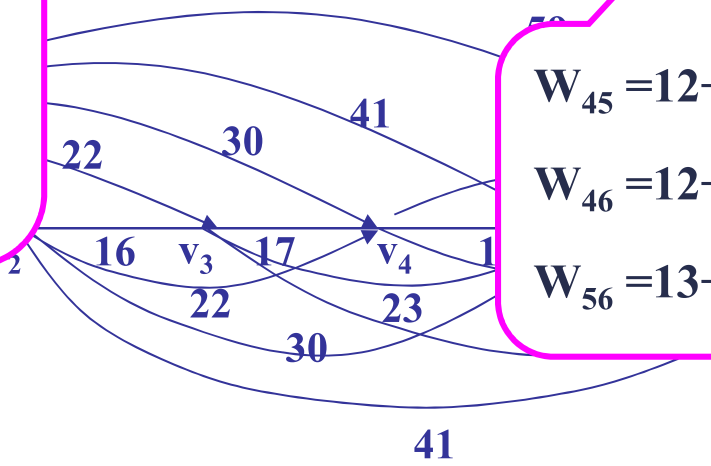
$W_{25}=11+5+6+8=30$

$W_{26}=11+5+6+8+11=41$

$W_{45}=12+5=17$

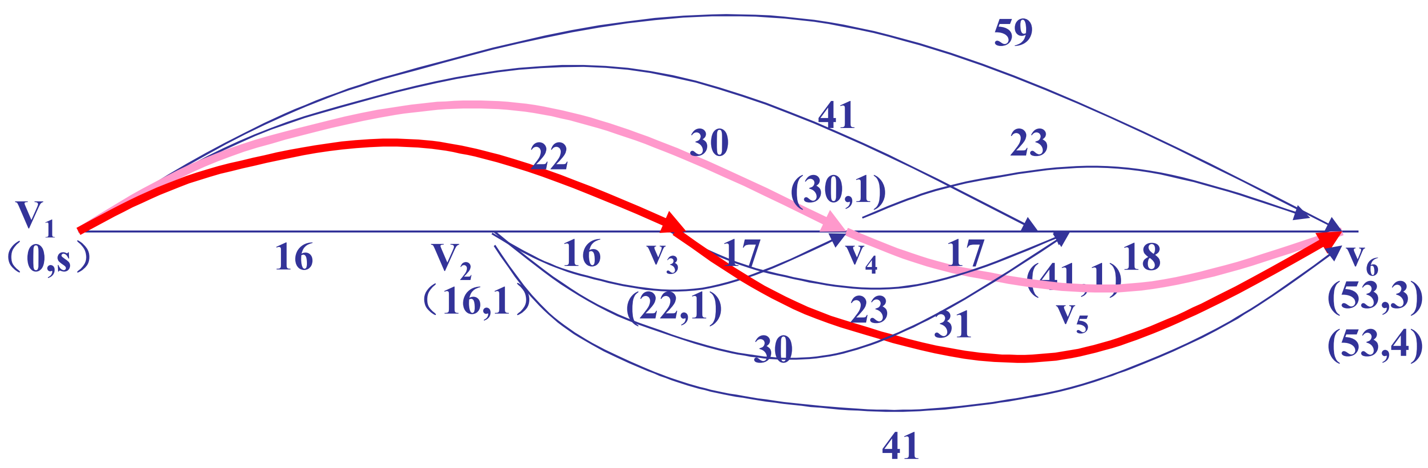
$W_{46}=12+5+6=23$

$W_{56}=13+5=18$



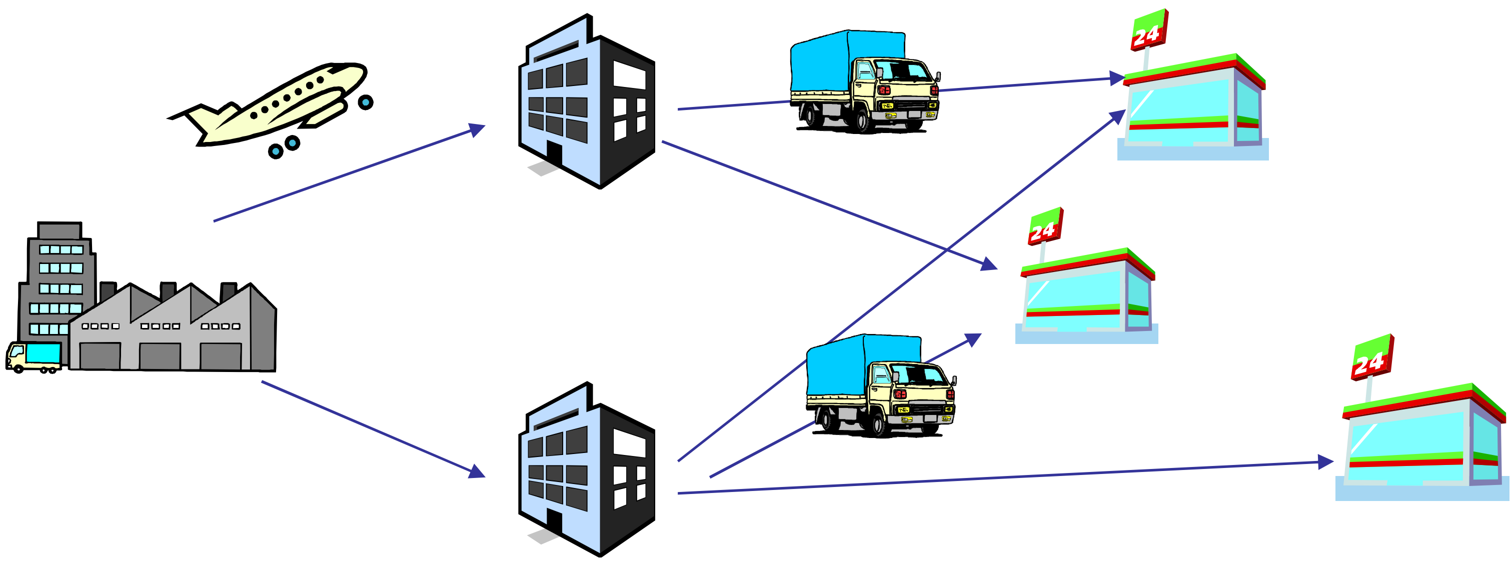
最短路问题

最终得到下图，可知，v_1到v_6的距离是53，最短路径有两条：$v_1 \to v_3 \to v_6$和 $v_1 \to v_4 \to v_6$



最大流问题

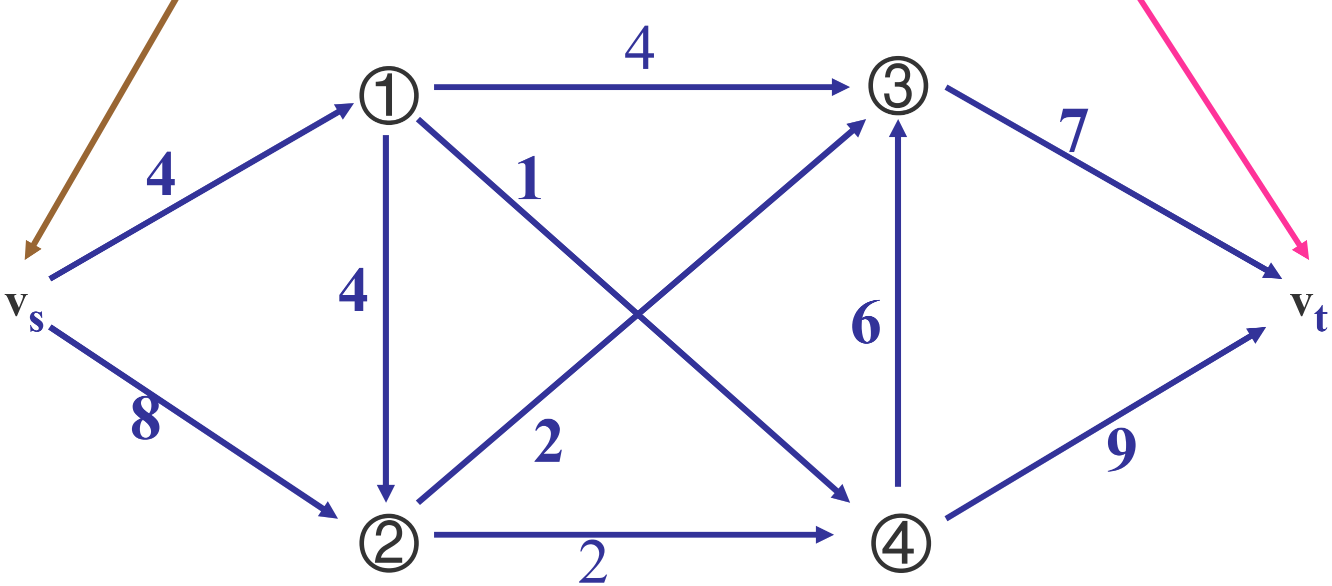
如何制定一个运输计划使生产地到销售地的产品输送量最大。这就是一个网络最大流问题。



最大流问题

基本概念：

1. 容量网络：队网络上的每条弧(v_i,v_j)都给出一个最大的通过能力，称为该弧的容量，简记为c_{ij}。容量网络中通常规定一个发点(也称源点，记为v_s)和一个收点(也称汇点，记为v_t)，网络中其他点称为中间点。



最大流问题

2. 网络的最大流

是指网络中从发点到收点之间允许通过的最大流量。

3. 流与可行流

流是指加在网络各条弧上的实际流量，对加在弧(v_i,v_j)上的载量记为f_{ij}。若$f_{ij}=0$，称为零流。

满足以下条件的一组流称为可行流。

- 容量限制条件。容量网络上所有的弧满足：$0 \le f_{ij} \le c_{ij}$
- 中间点平衡条件。

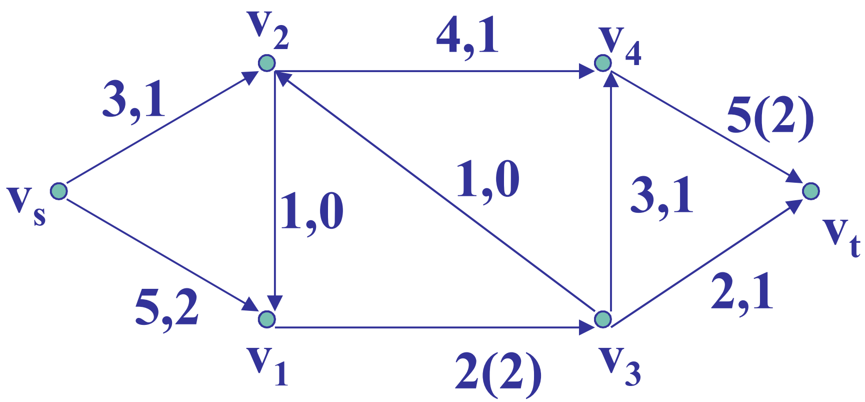
$$\sum f(v_i,v_j) - \sum f(v_j,v_i) = 0 \quad (i \ne s,t)$$

- 若以$v(f)$表示网络中从$v_s \to v_t$的流量，则有：

$$v(f) = \sum f(v_s,v_j) - \sum f(v_j,v_t) = 0$$

最大流问题

标示方式：每条边上标示两个数字，第一个是容量，第二是流量



结论：任何网络上一定存在可行流。（零流即是可行流）

网络最大流问题：

指满足容量限制条件和中间点平衡的条件下，使v(f)值达到最大。

最大流问题

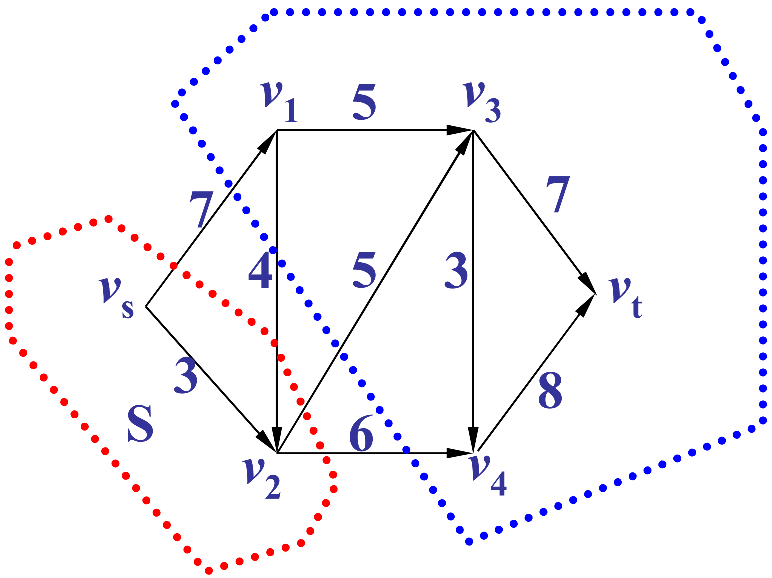
- 割集

容量网络G＝（V，E，C），v_s为始点，v_t为终点。如果把V分成两个非空集合 $S,\bar{S}$ ，使 $v_s \in S, v_t \in \bar{S}$ ，则所有始点属于S，而终点属于 $\bar{S}$ 的弧的集合，称为由S决定的割集,记作 $(S,\bar{S})$ 。割集中由S到$\bar{S}$所有弧的容量之和，称为这个割集的容量，记为 $C(S,\bar{S})$

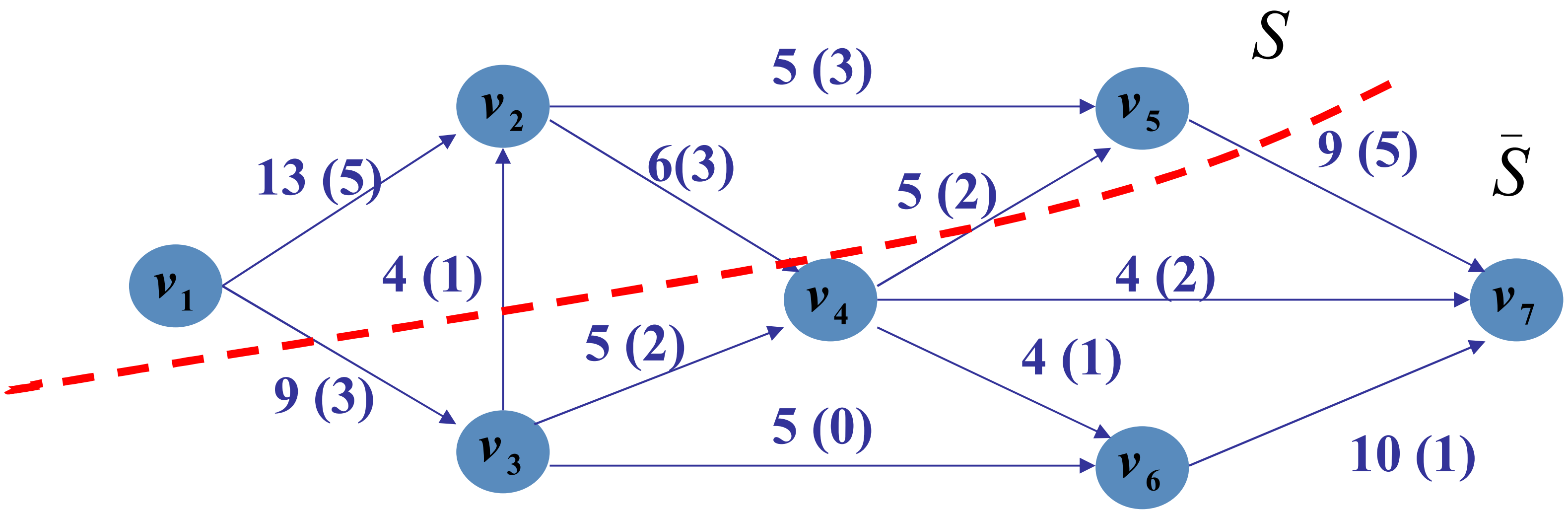
$$S=(v_s, v_2) \quad \bar{S}=(v_1, v_3, v_4, v_t)$$

$$(S,\bar{S})=\{(v_s,v_1),(v_2,v_4),(v_2,v_3)\}$$

$$C(S,\bar{S})=l_{s1}+l_{24}+l_{23}=7+6+5=18$$



最大流问题



设 $S=\{v_1,v_2,v_5\}$ $\bar{S}=\{v_3,v_4,v_6,v_7\}$

则 割集 $(S,\bar{S})=\{(v_1,v_3),(v_2,v_4),(v_5,v_7)\}$

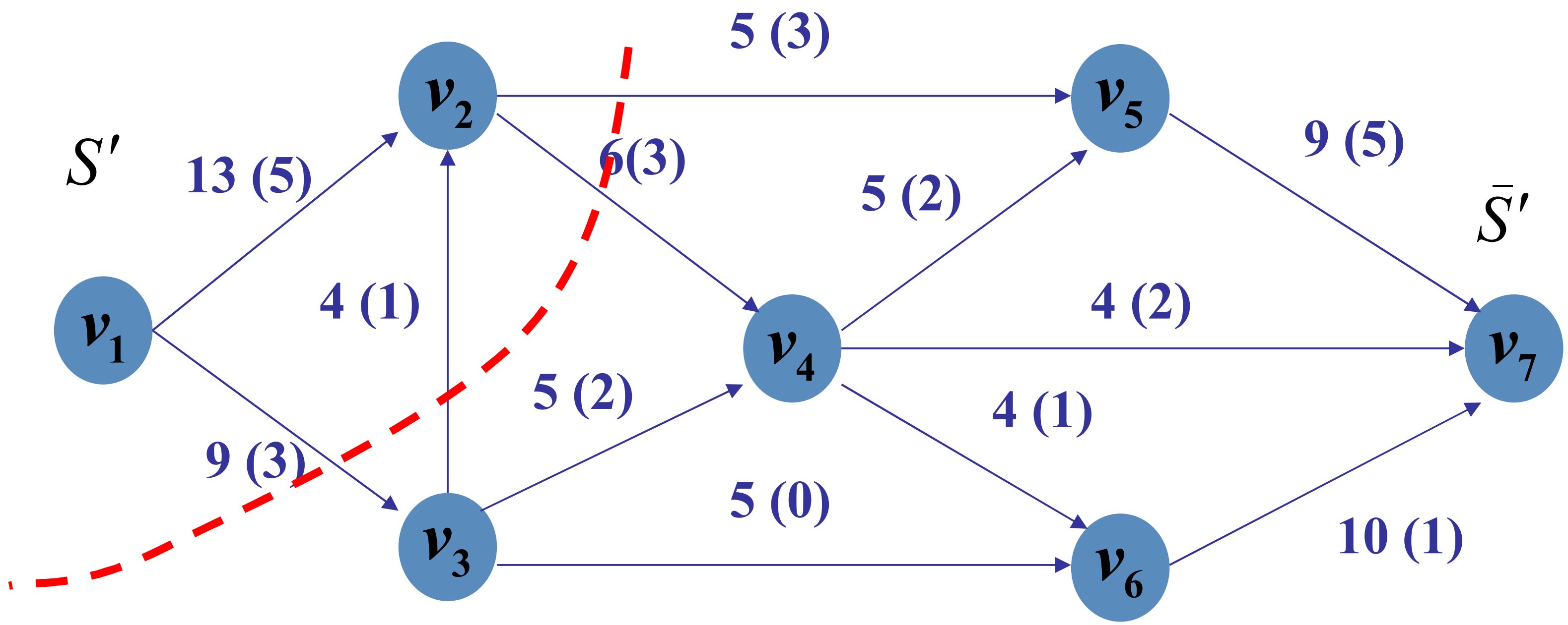
$$=l_{13}+l_{24}+l_{57}$$

$$=9+6+9=24$$

容量为24

而 (v_3,v_2) 和 (v_4,v_5) 不是该集中的弧

最大流问题



设 $S' = \{v_1, v_2\}$ $\bar{S}' = \{v_3, v_4, v_5, v_6, v_7\}$

则 割集 $(S', \bar{S}') = \{(v_1, v_3), (v_2, v_4), (v_2, v_5)\}$ 容量为**20**

$$= l_{13} + l_{24} + l_{25}$$
$$= 9 + 6 + 5 = 20$$

最大流问题

最大流—最小割定理

由发点v_s到收点v_t任一可行流量W显然必须受割集$(S,\bar{S})$容量的限制，即有：

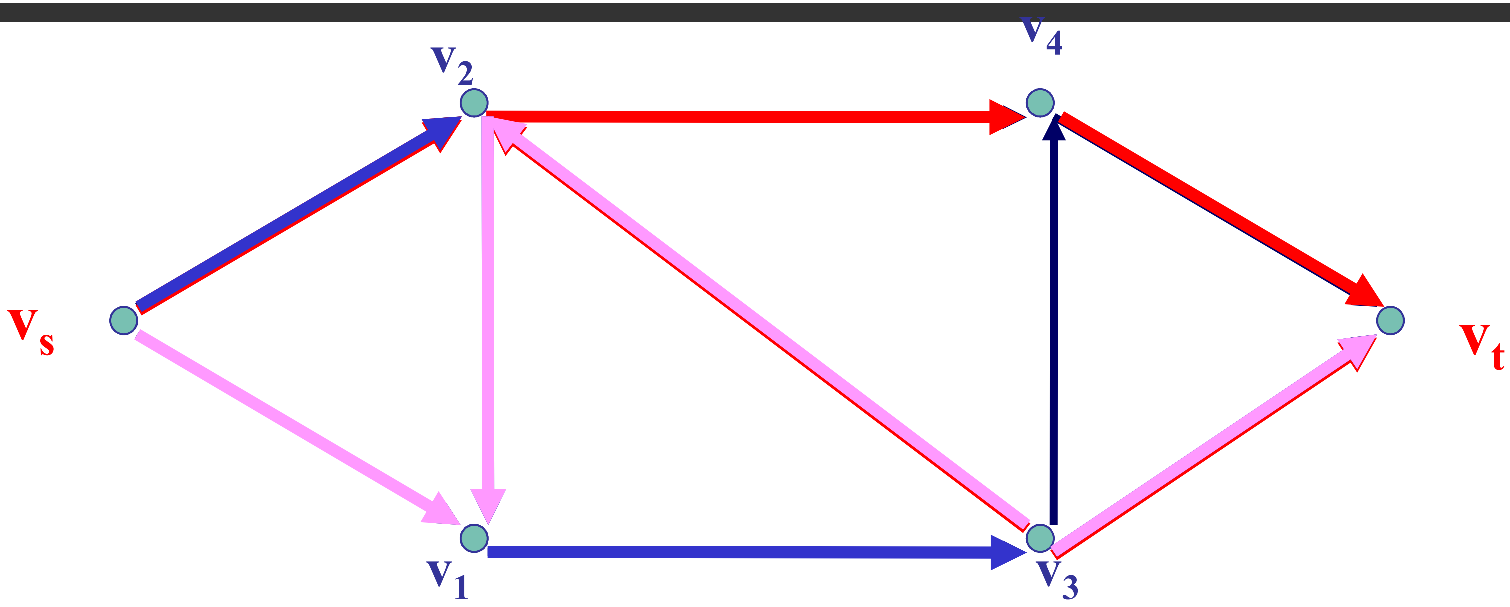
$$W \le C(S,\bar{S})$$

容量最小的割集称为最小割集

网络理论中著名的最大流最小割定理：

对于任一容量网络，从发点到收点的最大流量等于最小割量。

最大流问题



μ_1: $v_s \to v_1 \to v_2 \to v_3 \to v_t$

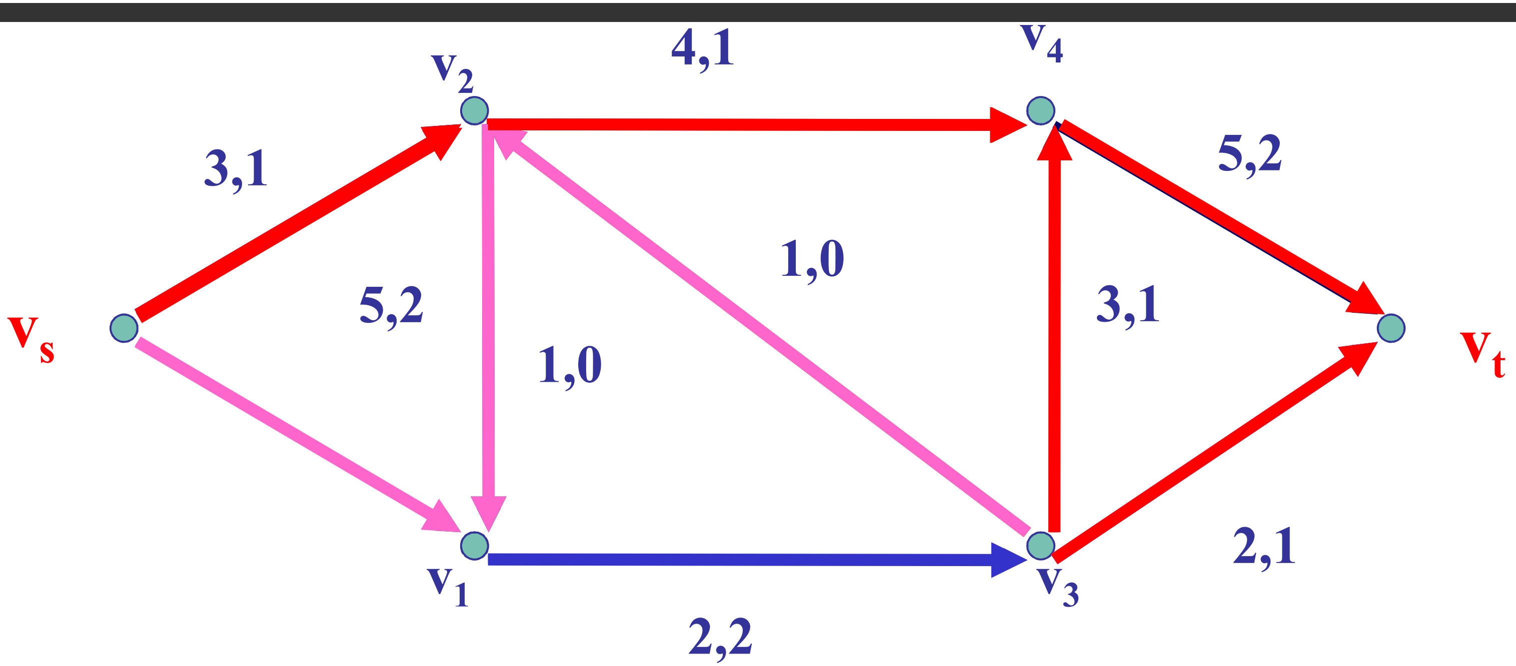
μ_2: $v_s \to v_2 \to v_1 \to v_3 \to v_t$

μ_3: $v_s \to v_2 \to v_3 \to v_t$

μ_4: $v_s \to v_2 \to v_4 \to v_t$

若 μ 是联结发点 v_s 和收点 v_t 的一条链，我们规定链的方向是从 v_s 到 v_t，则链上的弧被分成两类：前向弧、后向弧。

最大流问题



$\mu_5: v_s \to v_2 \to v_4 \to v_3 \to v_t$ × √ ×

设f 是一个可行流，μ是从v_s到v_t的一条链，若μ满足前向弧都是非饱和弧,后向弧都是非零流弧,则称μ是（可行流f 的）一条**增广链**。

最大流问题

求最大流的标号法

标号法思想是：先找一个可行流。对于一个可行流，经过标号过程得到从发点 v_s 到收点 v_t 的增广链；经过调整过程沿增广链增加可行流的流量，得新的可行流。重复这一过程，直到可行流无增广链，得到最大流。

从任一个可行流 f 出发（若网络中没有给定 初始可行流 f，可从零流开始），经历如下两过程：

- **标号过程**—用来找增广链的过程
- **调整过程**—用来增大增广链流量的过程

最大流问题

求网络最大流的标号算法：

[基本方法]

(1) 找出第一个可行流，（例如所有弧的流量f_{ij}=0。）

(2) 用标号的方法找一条增广链

- 首先给发点s标号(∞),标号中的数字表示允许的最大调整量。
- 选择一个点 v_i 已标号并且另一端未标号的弧沿着某条链向收点检查：

最大流问题

- 如果弧的起点为v_i，并且有$f_{ij}<C_{ij}$，则给v_j标号为$(C_{ij}-f_{ij})$
- 如果弧的方向指向v_i，并且有$f_{ji}>0$，则v_j标号(f_{ji})

(3) 重复第(2)步，可能出现两种结局：

- 标号过程中断，t 无法标号，说明网络中不存在增广链，目前流量为最大流。同时可以确定最小割集，<u>记已标号的点集为V,未标号的点集合为V',(V,V')为网络的最小割。</u>
- t 得到标号，反向追踪在网络中找到一条从s到t得由标号点及相应的弧连接而成的增广链。继续第(4)步

最大流问题

(4) 修改流量。设原图可行流为f，令

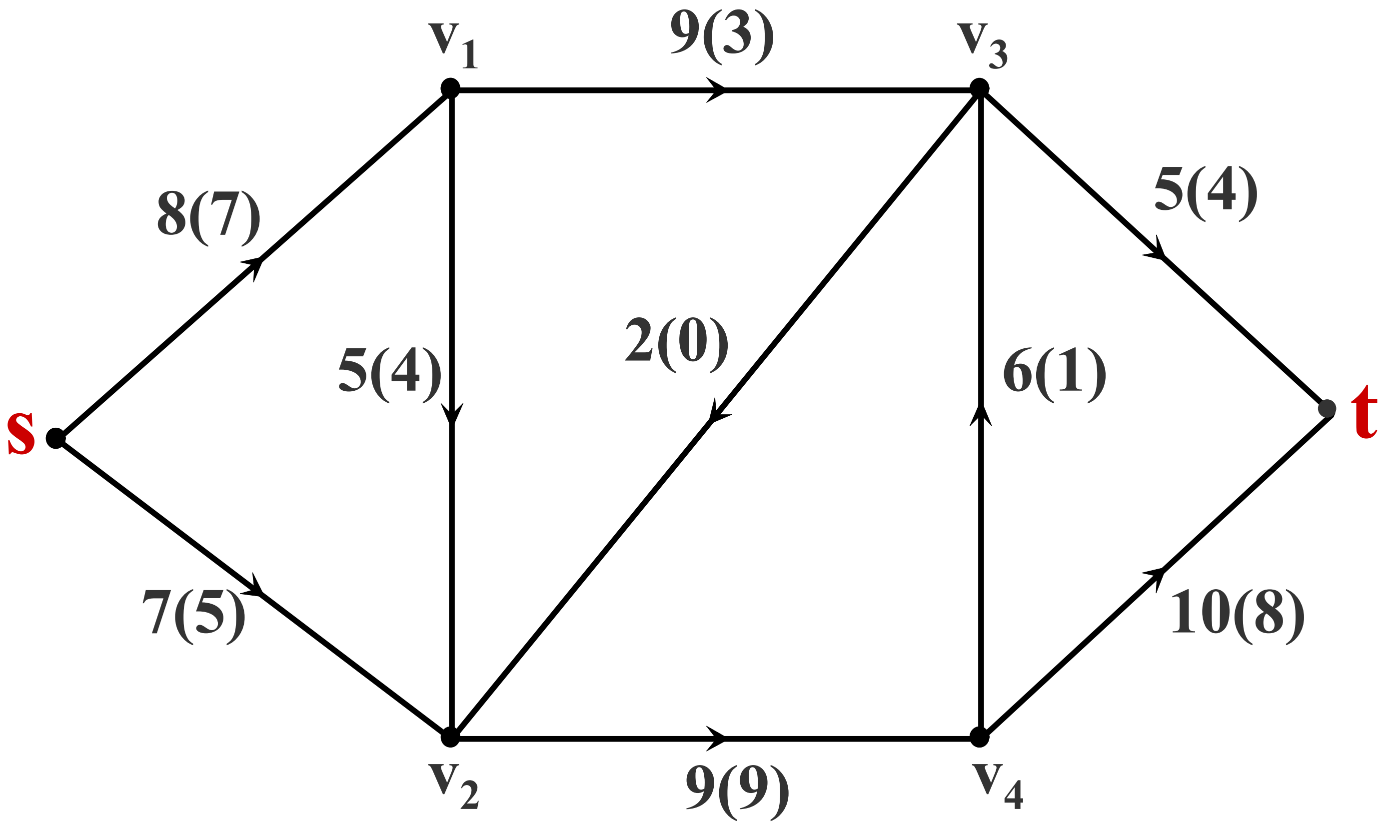
$$f' = \begin{cases} f + \varepsilon(t) & \text{对增广链上所有前向弧} \\ f - \varepsilon(t) & \text{对增广链上所有后向弧} \\ f & \text{所有非增广链上的弧} \end{cases}$$

得到网络上一个新的可行流f'。

(5) 擦除图上所有标号，重复(1)-(4)步，直到图中找不到任何增广链，计算结束。

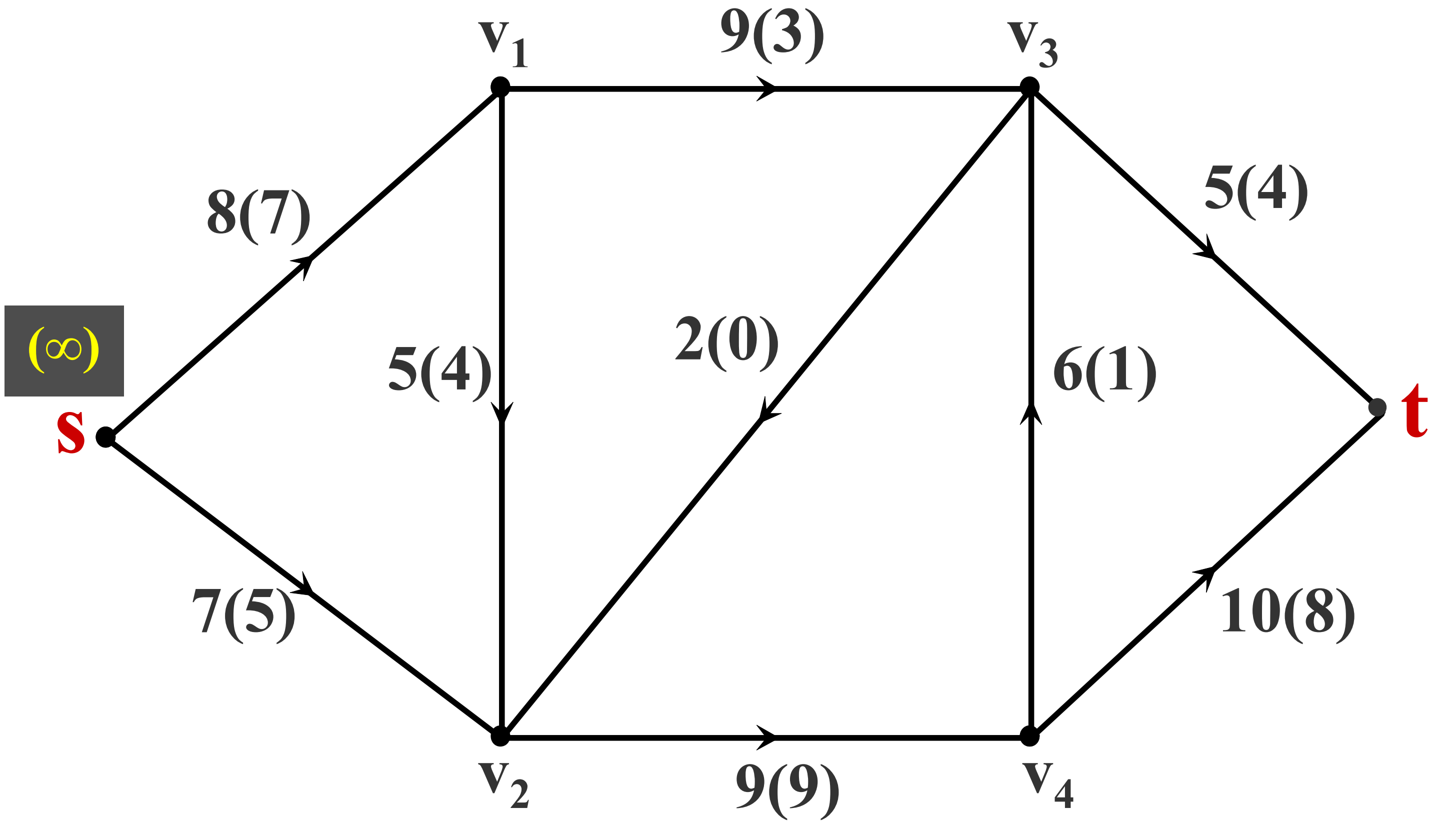
最大流问题

例6.10 用标号算法求下图中$s \to t$ 的最大流量，并找出最小割。



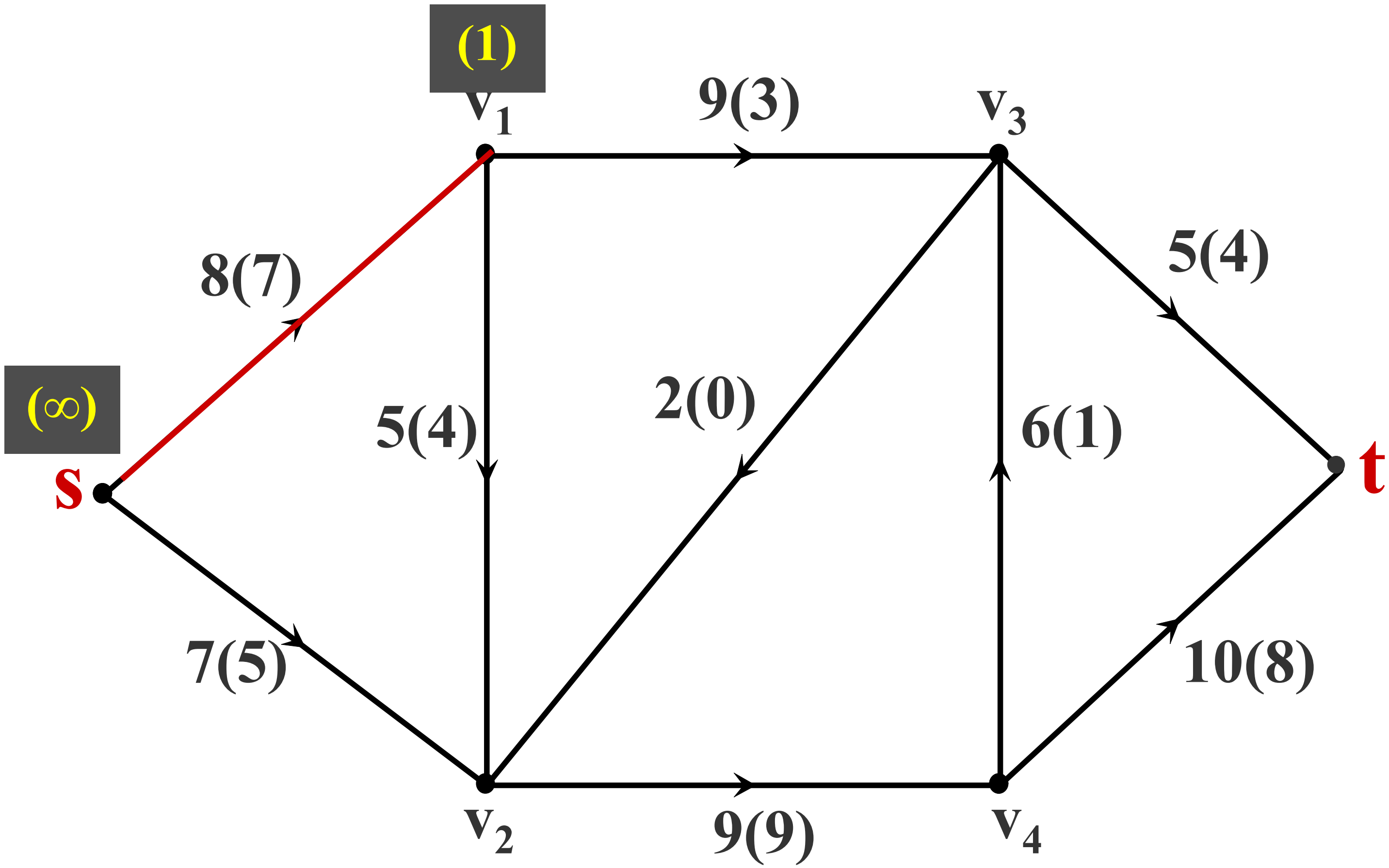
最大流问题

解： (1) 先给s标号(∞)



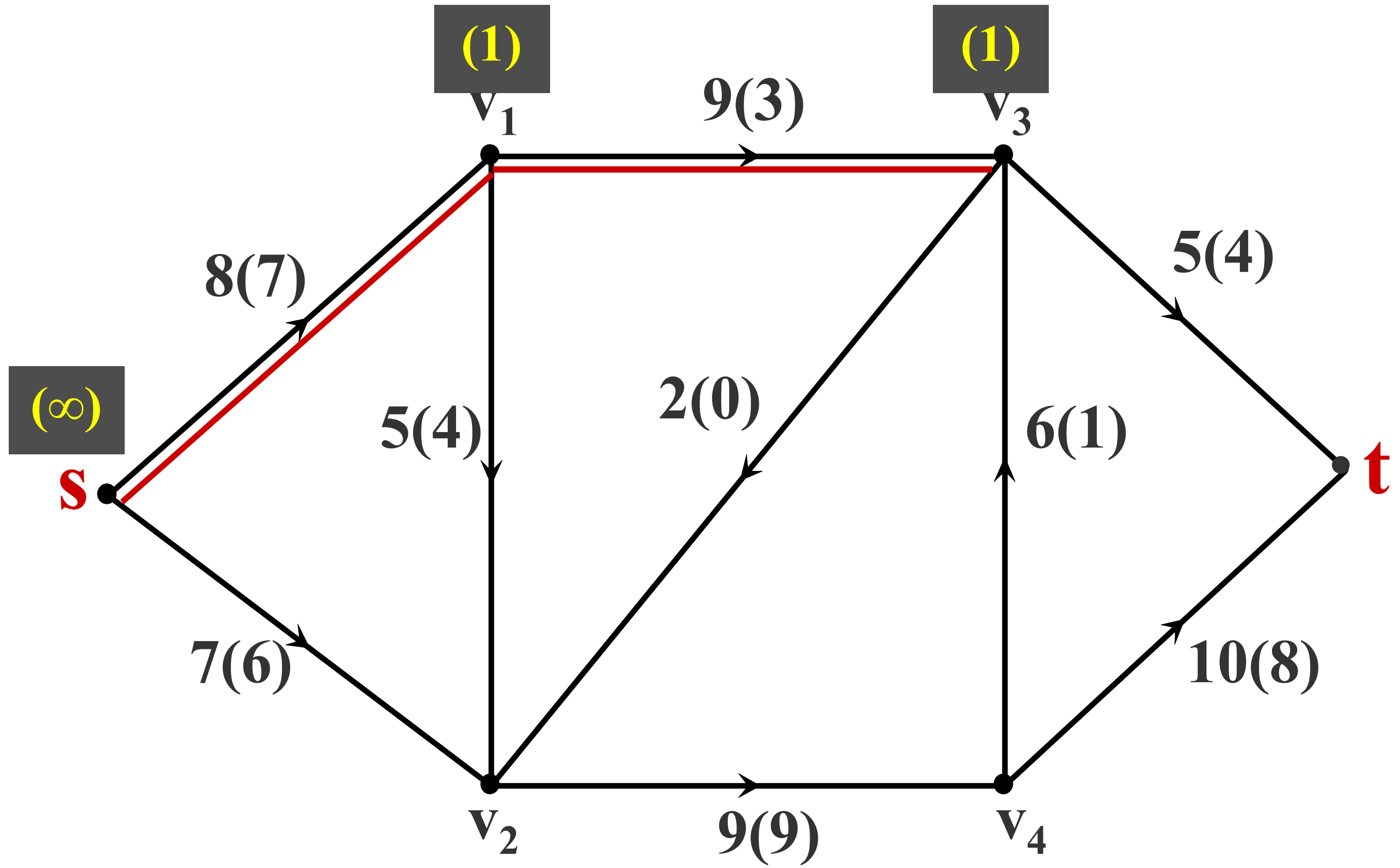
最大流问题

(2) 检查与s点相邻的未标号的点，因$f_{s1}<c_{s1}$，故对v_1标号 $\varepsilon(1)=\min\{\infty, c_{s1}-f_{s1}\}=1$,



最大流问题

(2) 检查与v_1点相邻的未标号的点，因$f_{13}<c_{13}$，故对v_3标号$\varepsilon(3)=\min\{1, c_{13}-f_{13}\}=\min\{1, 6\}=1$

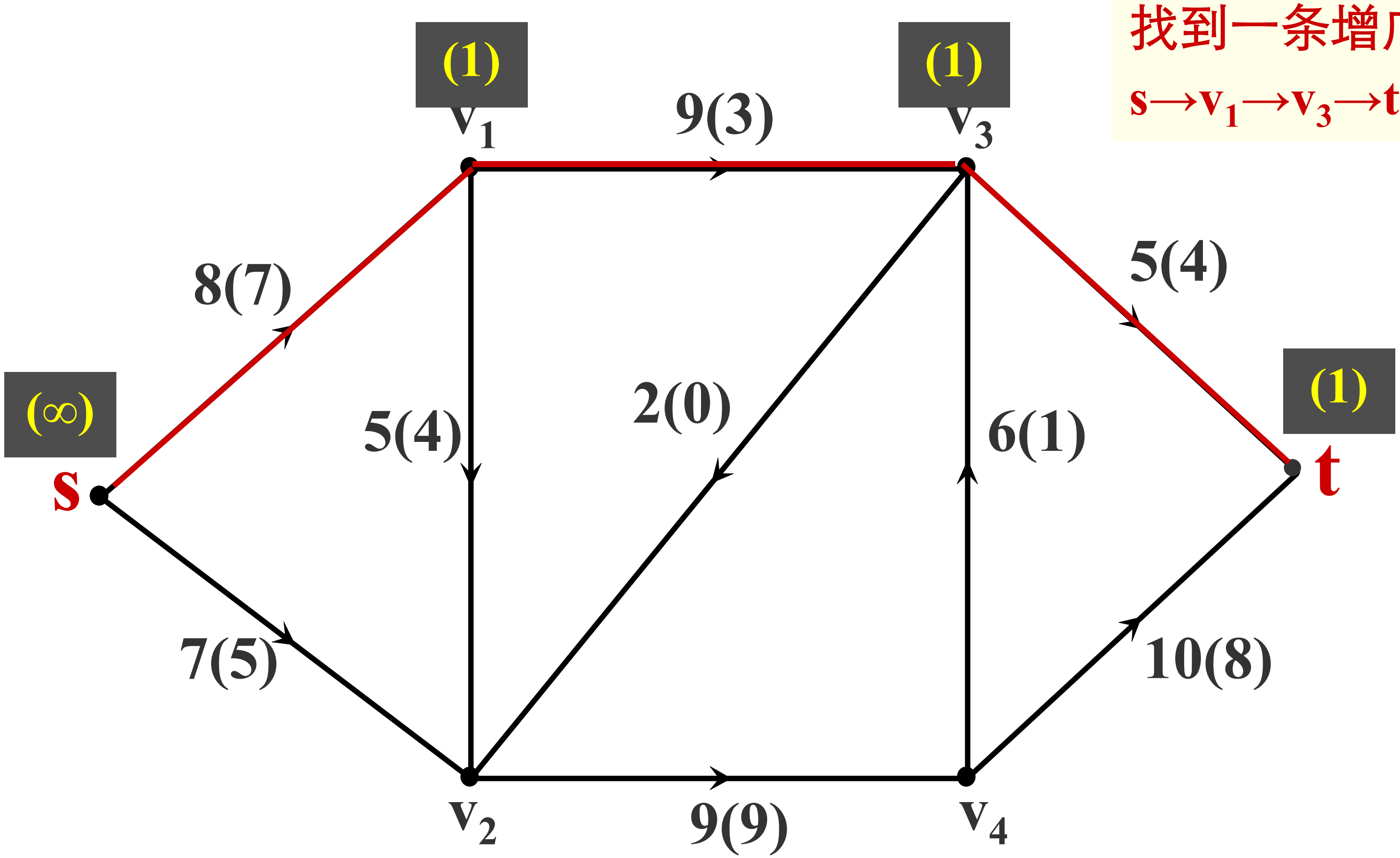


最大流问题

(3) 检查与v_3点相邻的未标号的点，因$f_{3t}<c_{3t}$，故对v_t标号 $\varepsilon(t) = \min\{1, c_{3t}-f_{3t}\} = \min\{1, 1\} = 1$

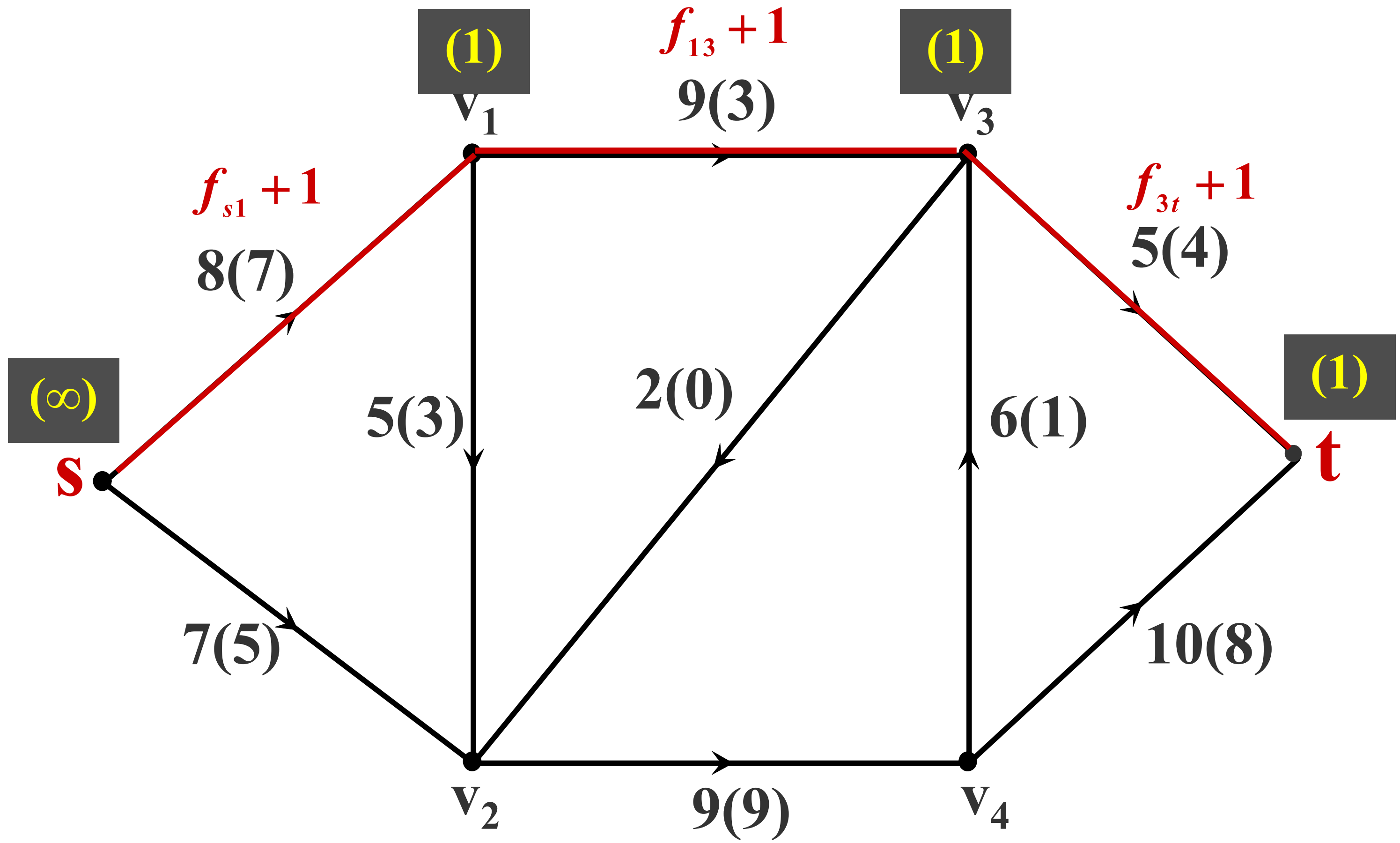
找到一条增广链

$s \to v_1 \to v_3 \to t$



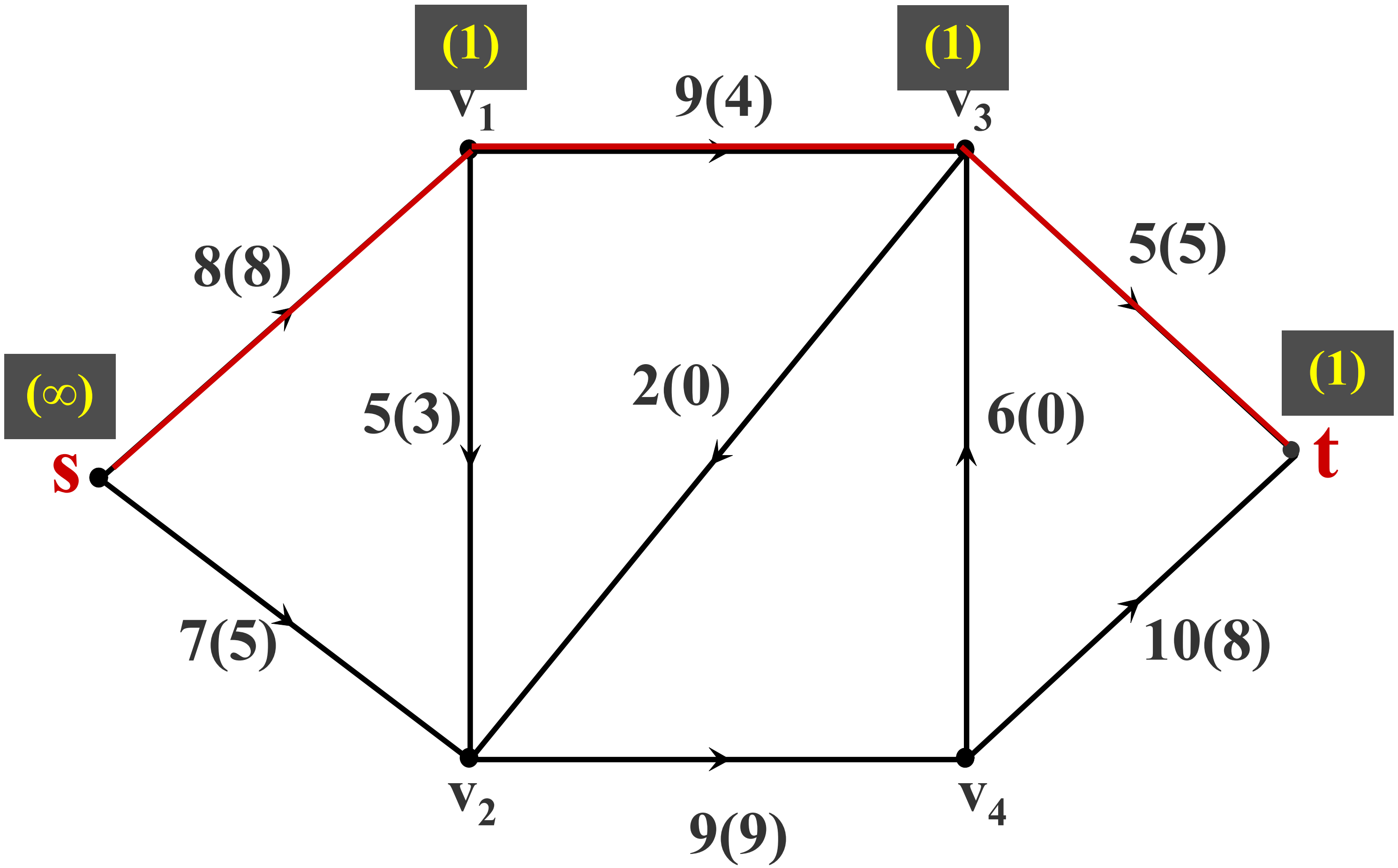
最大流问题

(4) 修改增广链上的流量，非增广链上的流量不变，得到新的可行流。



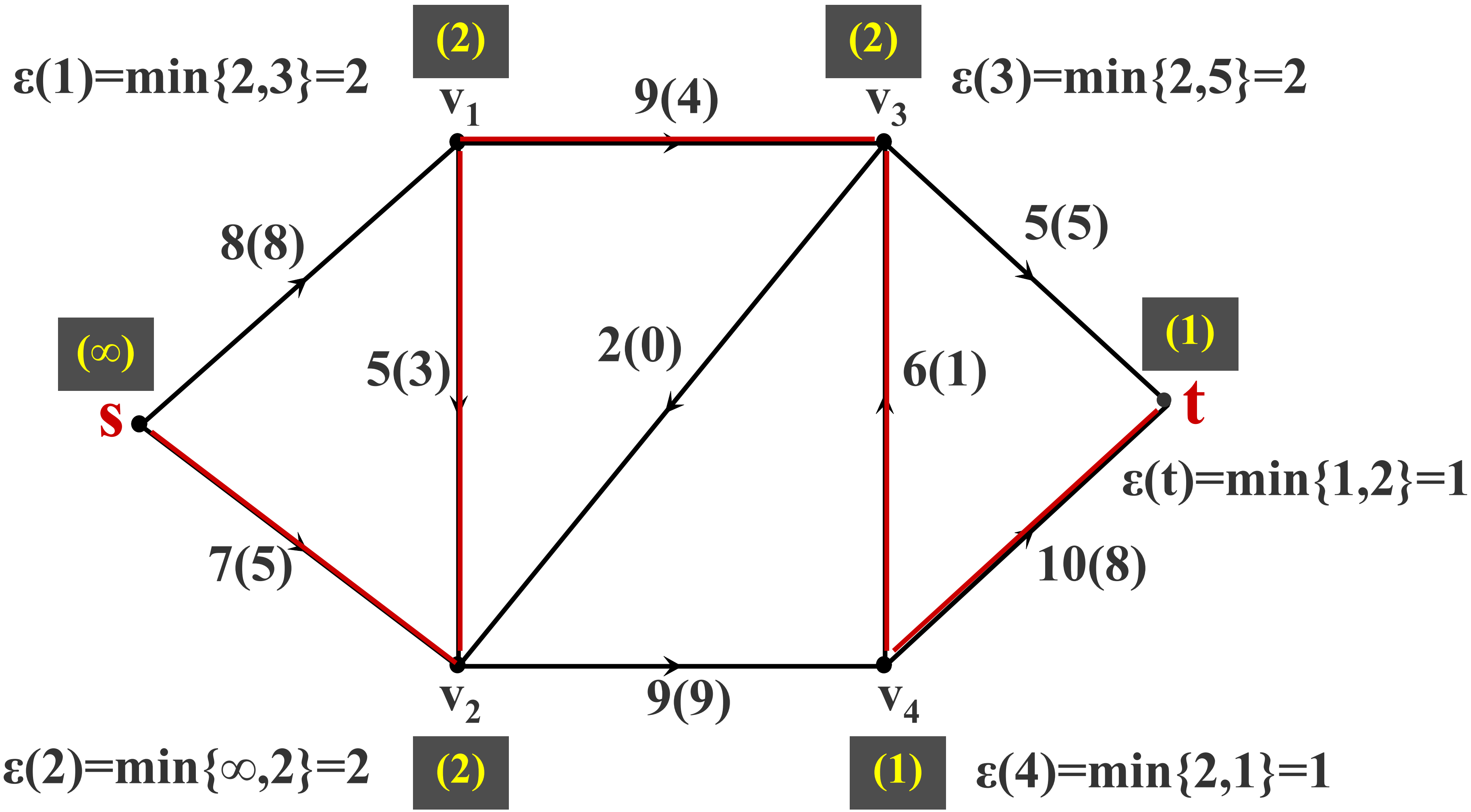
最大流问题

(5) 擦除所有标号，重复上述标号过程，寻找另外的增广链。



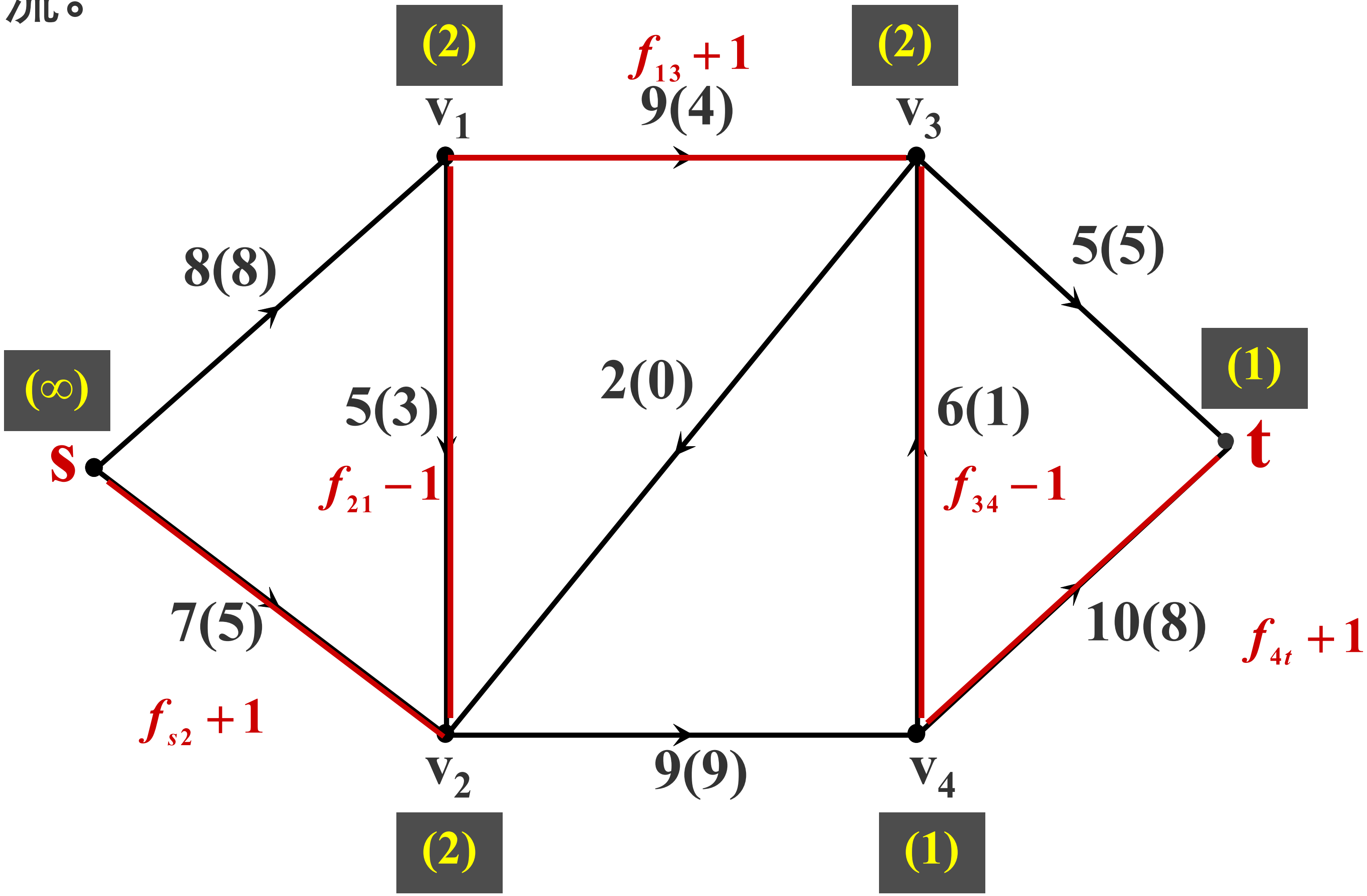
最大流问题

(5) 擦除所有标号，重复上述标号过程，寻找另外的增广链。



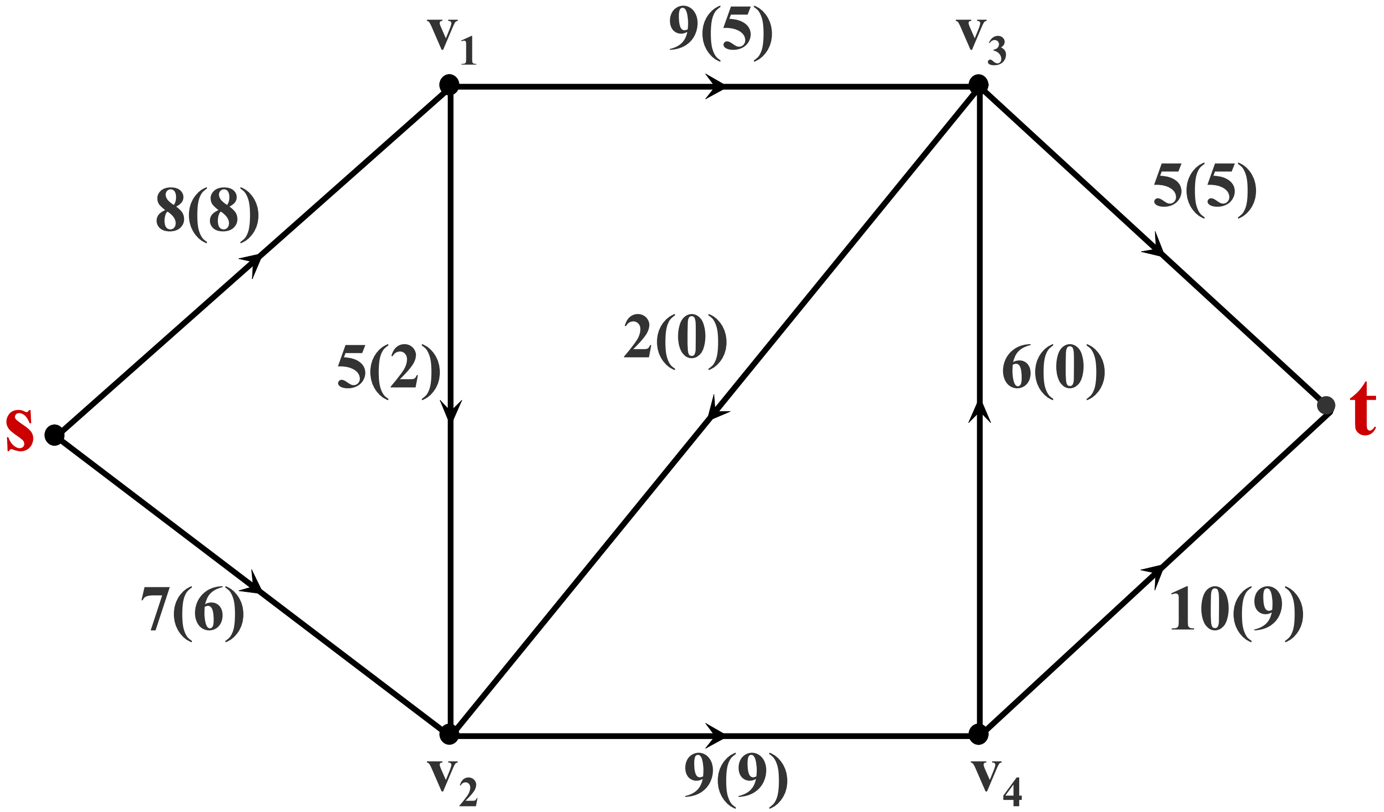
最大流问题

(6) 修改增广链上的流量，非增广链上的流量不变，得到新的可行流。



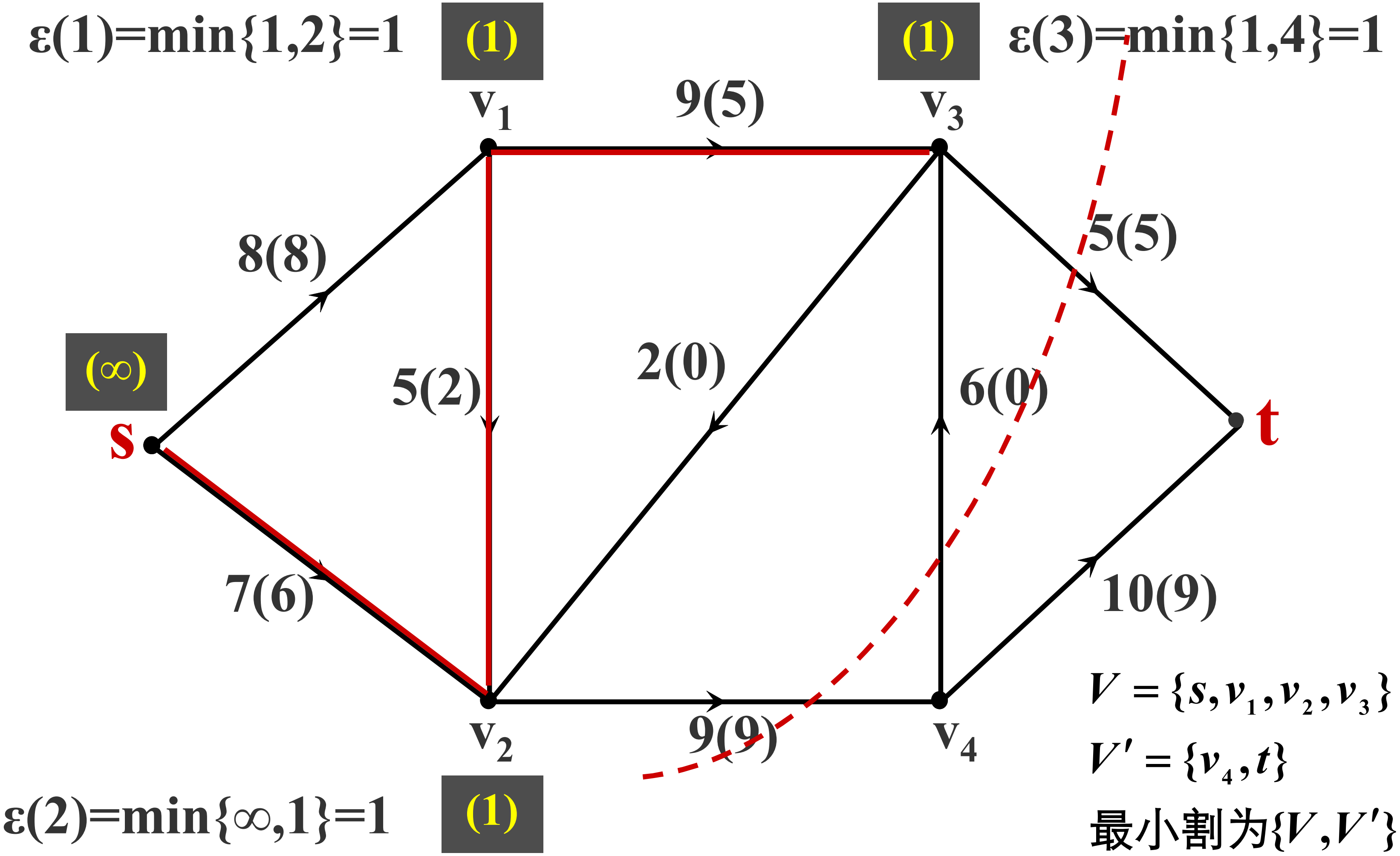
最大流问题

(7) 擦除所有标号，重复上述标号过程，寻找另外的增广链。



最大流问题

(7) 重复上述标号过程，寻找另外的增广链。



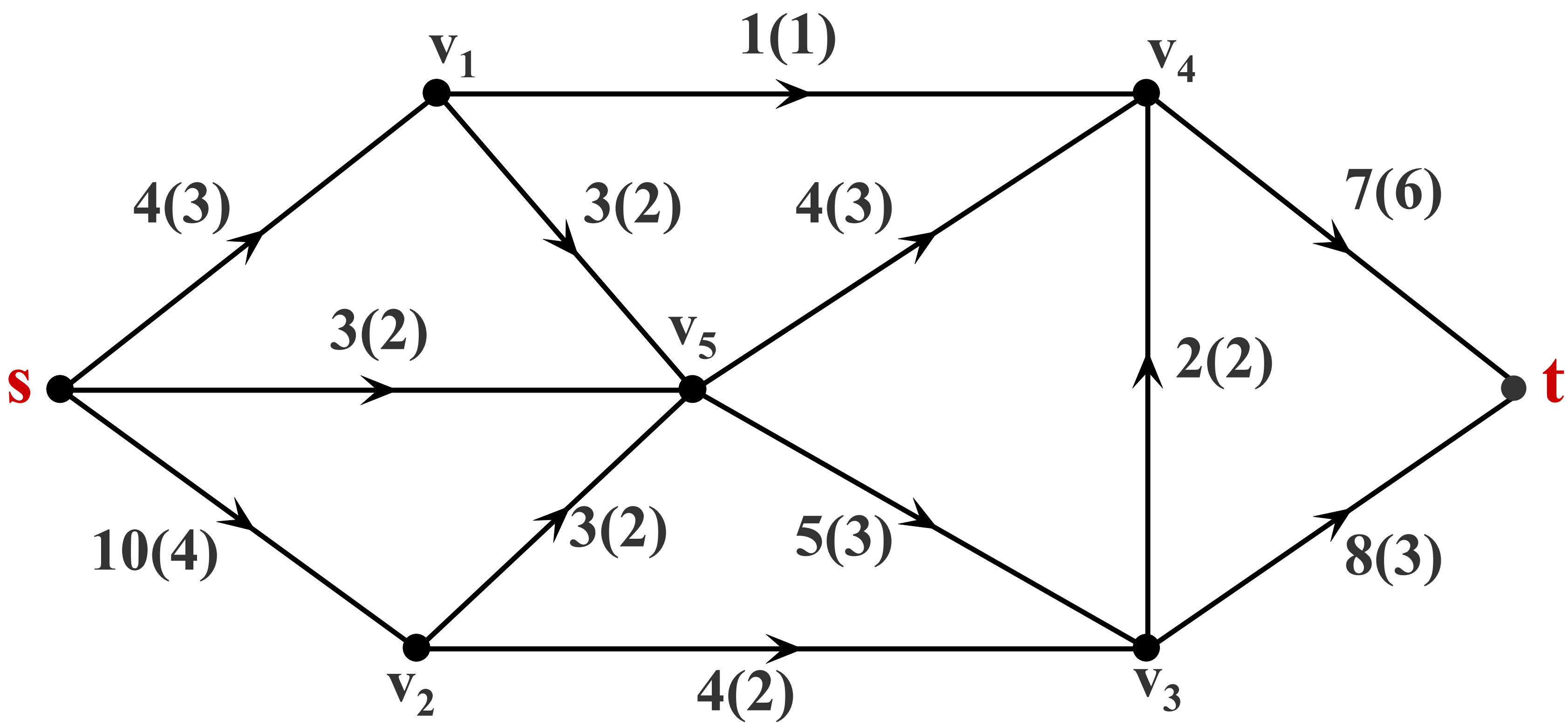
$V = \{s, v_1, v_2, v_3\}$

$V' = \{v_4, t\}$

最小割为$\{V, V'\}$

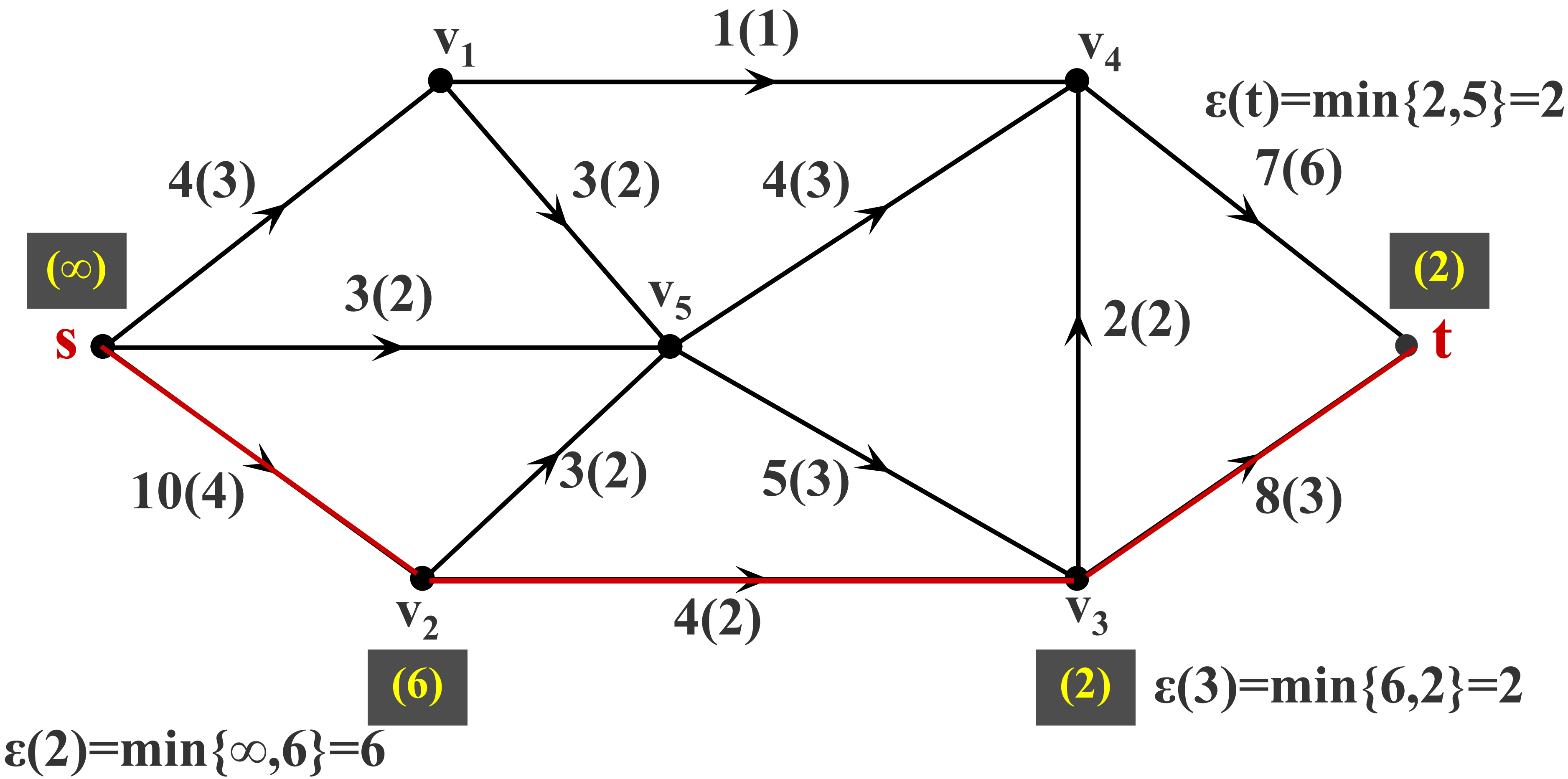
最大流问题

例6.9 求下图$s \to t$的最大流，并找出最小割



最大流问题

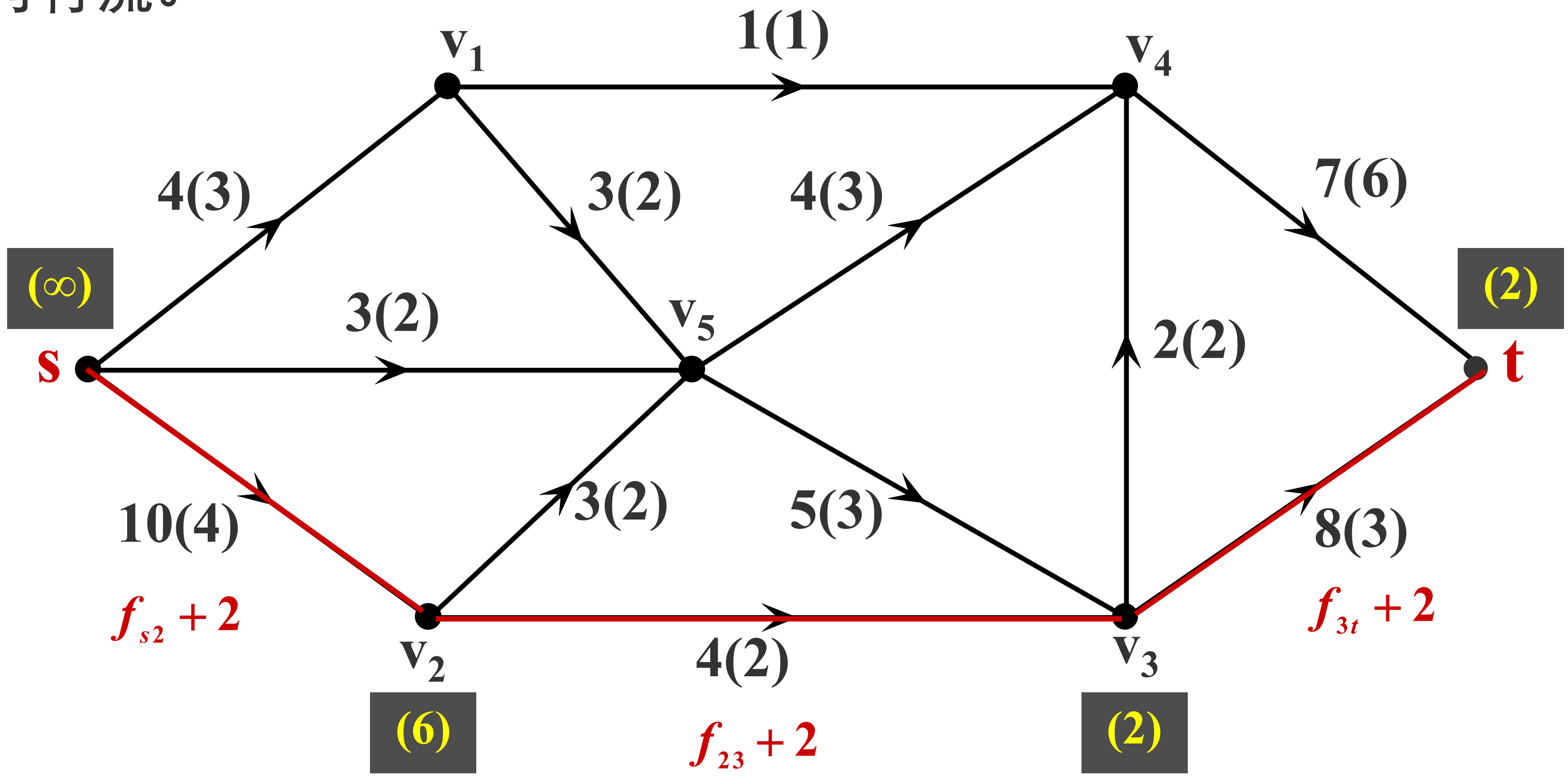
解: (1) 在已知可行流的基础上，通过标号寻找增广链。



存在增广链$s\rightarrow v_2\rightarrow v_3\rightarrow t$

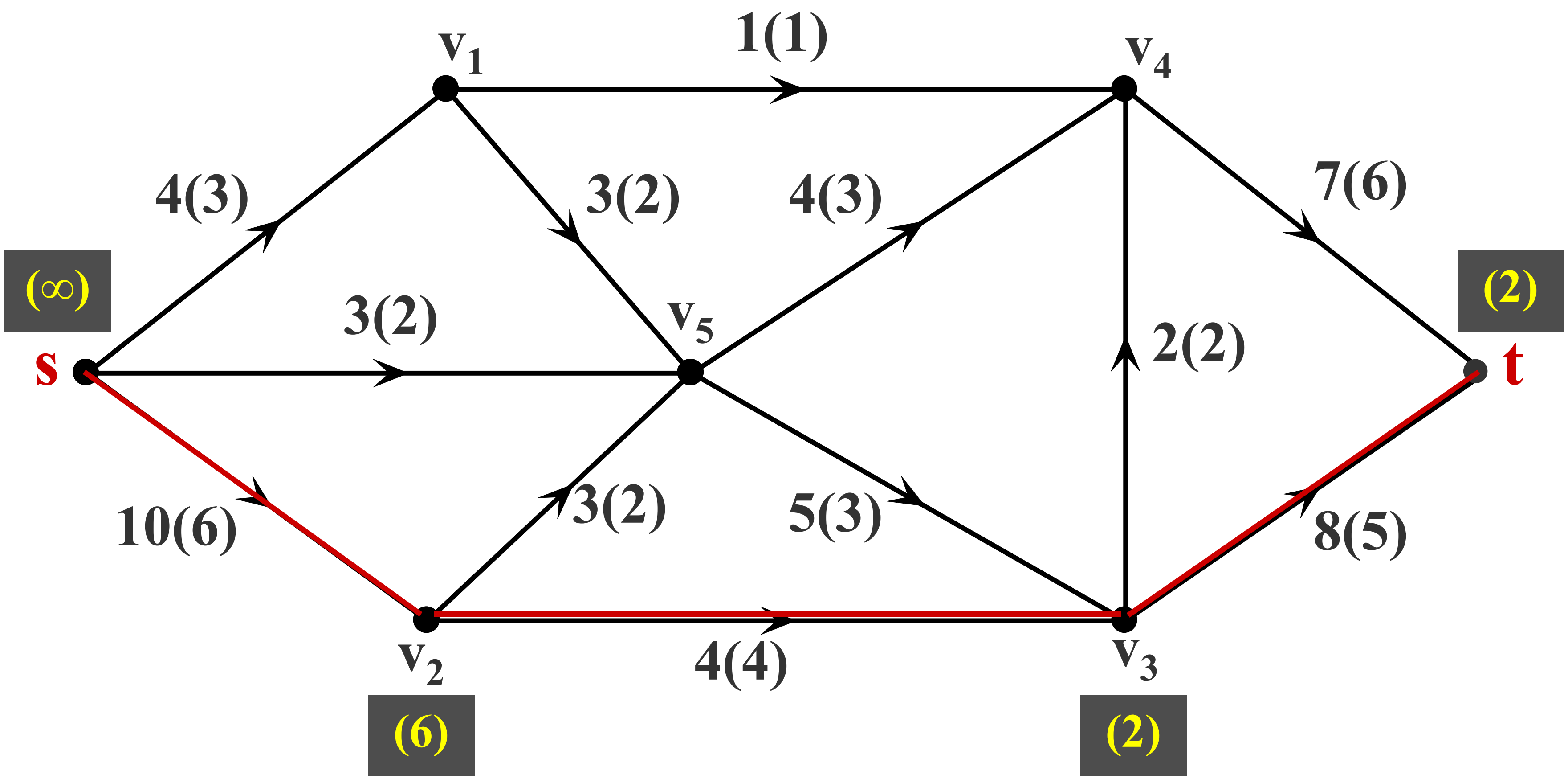
最大流问题

(2) 修改增广链上的流量，非增广链上的流量不变，得到新的可行流。



最大流问题

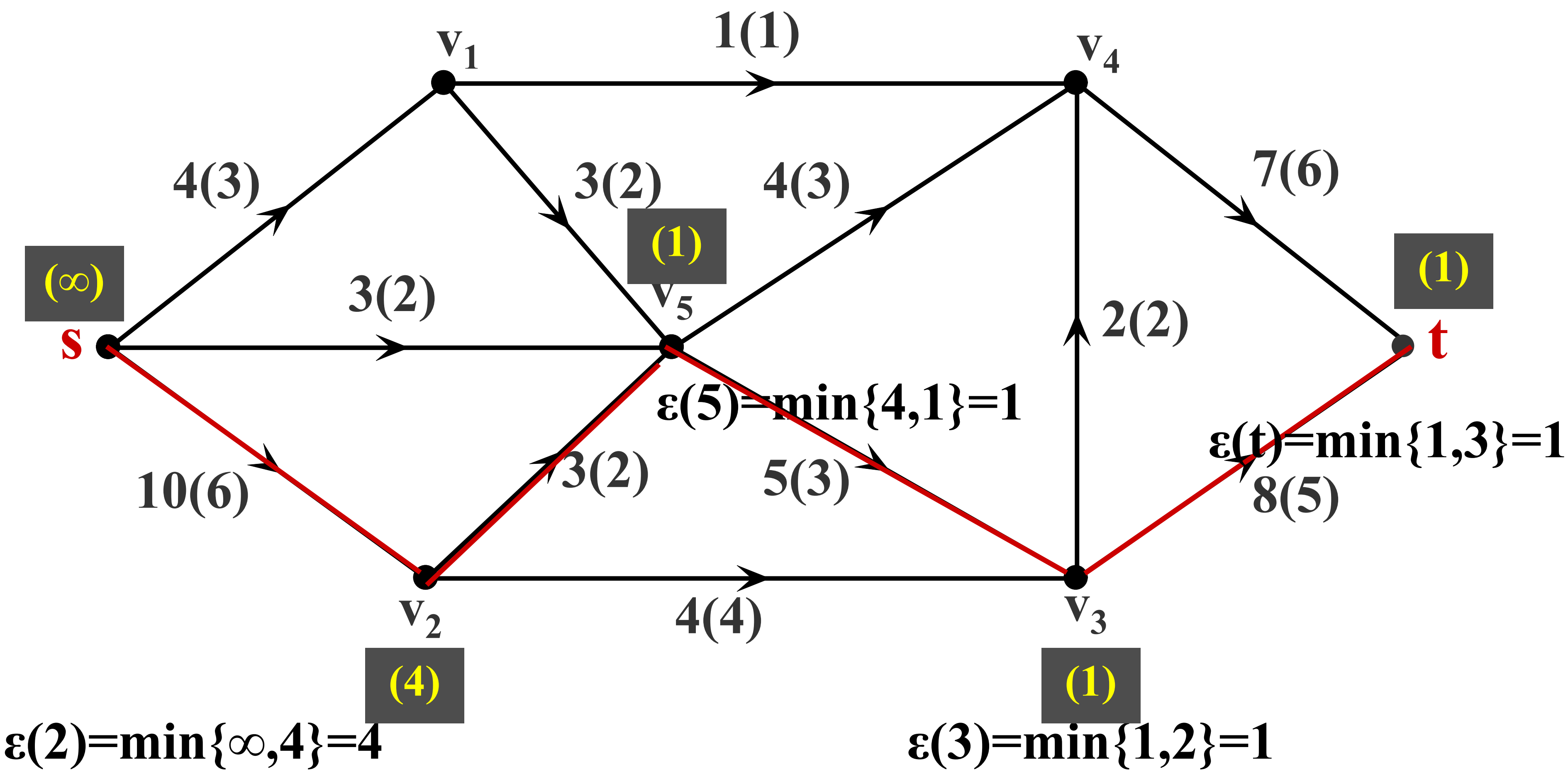
(3) 擦除原标号，重新搜寻增广链。



最大流问题

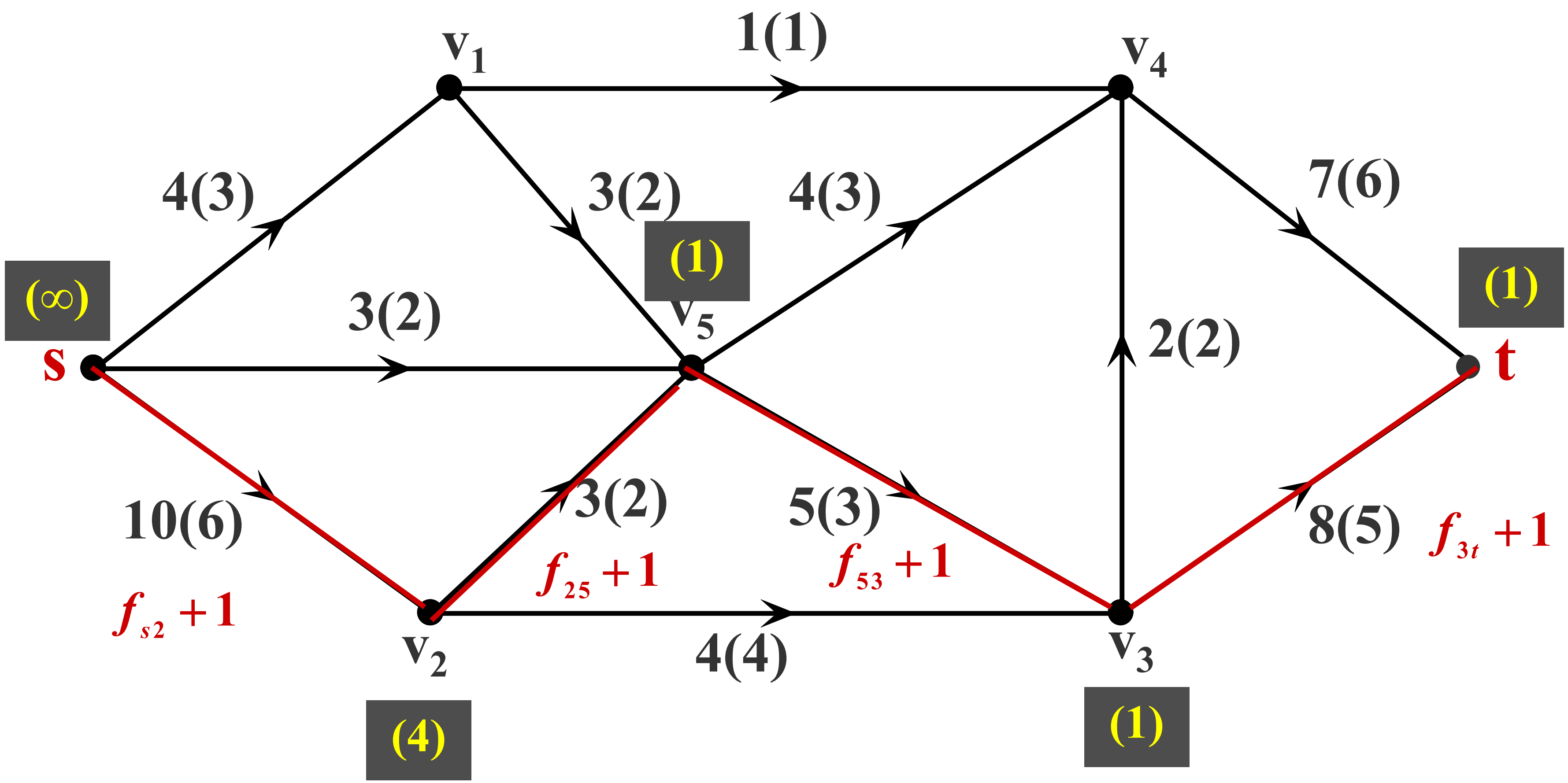
(4) 重新搜寻增广链。

存在增广链：$s \to v_2 \to v_5 \to v_3 \to t$



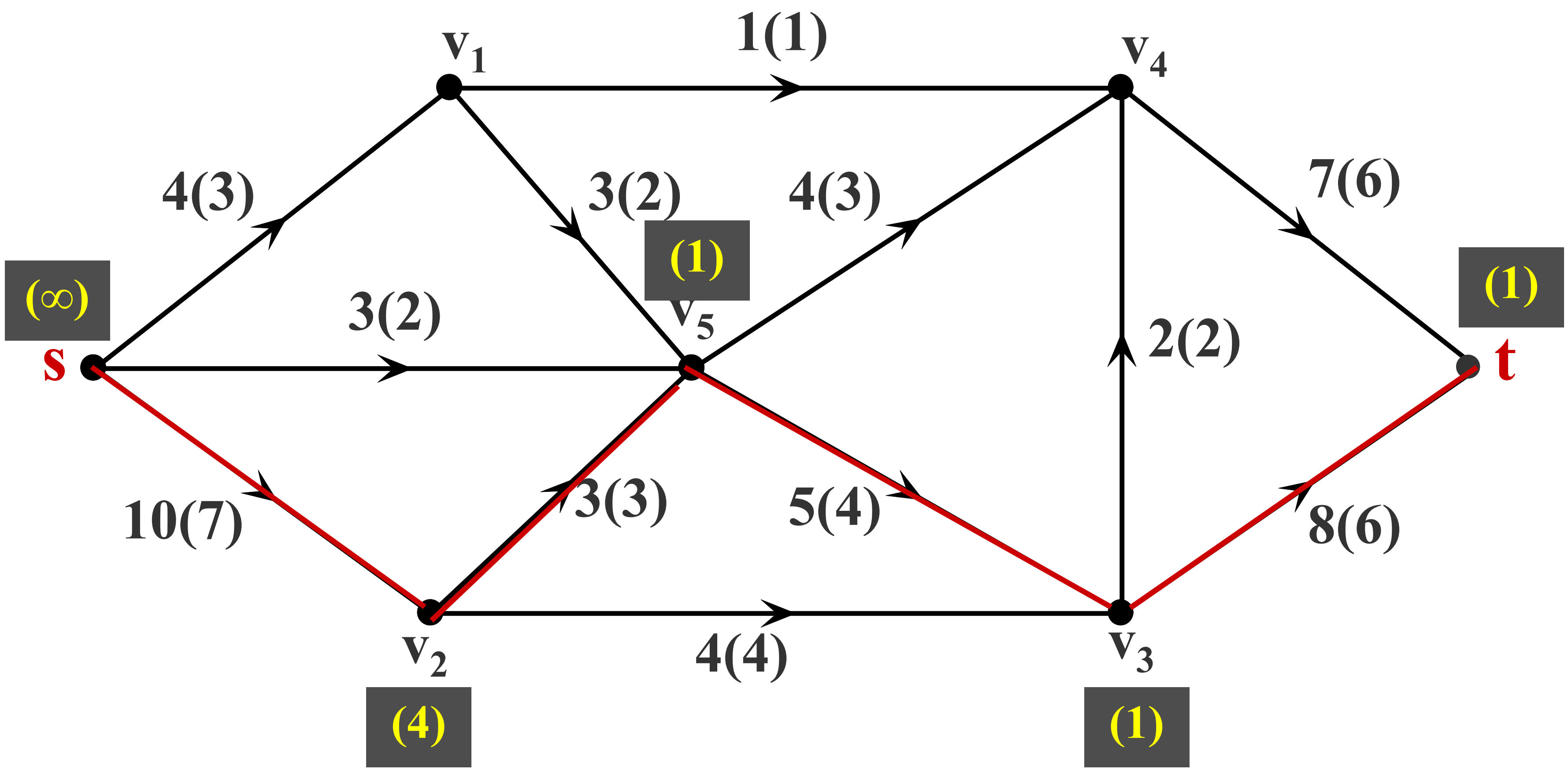
最大流问题

(5) 修改增广链上的流量，非增广链上的流量不变，得到新的可行流。



最大流问题

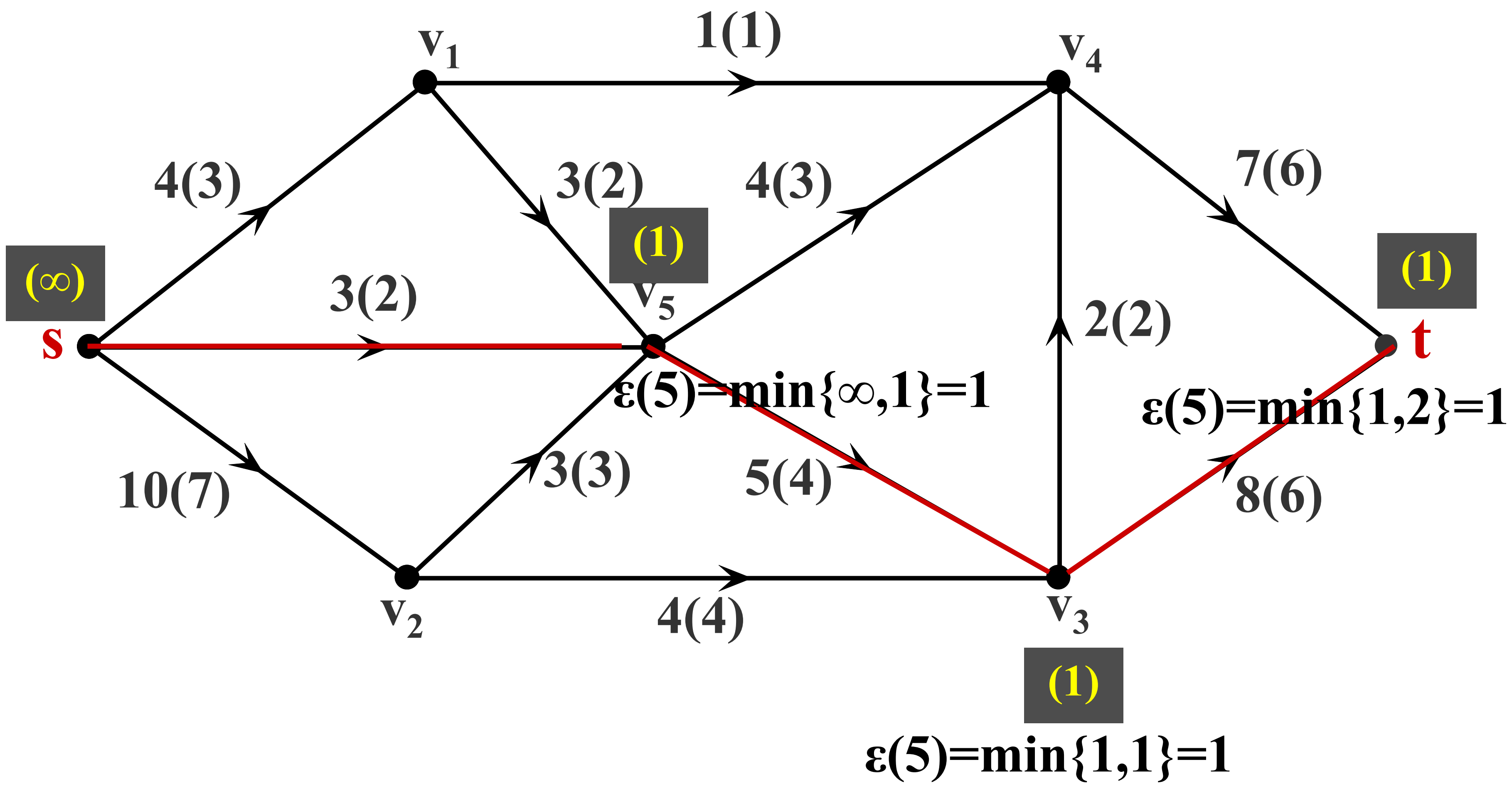
(6) 擦除原标号



最大流问题

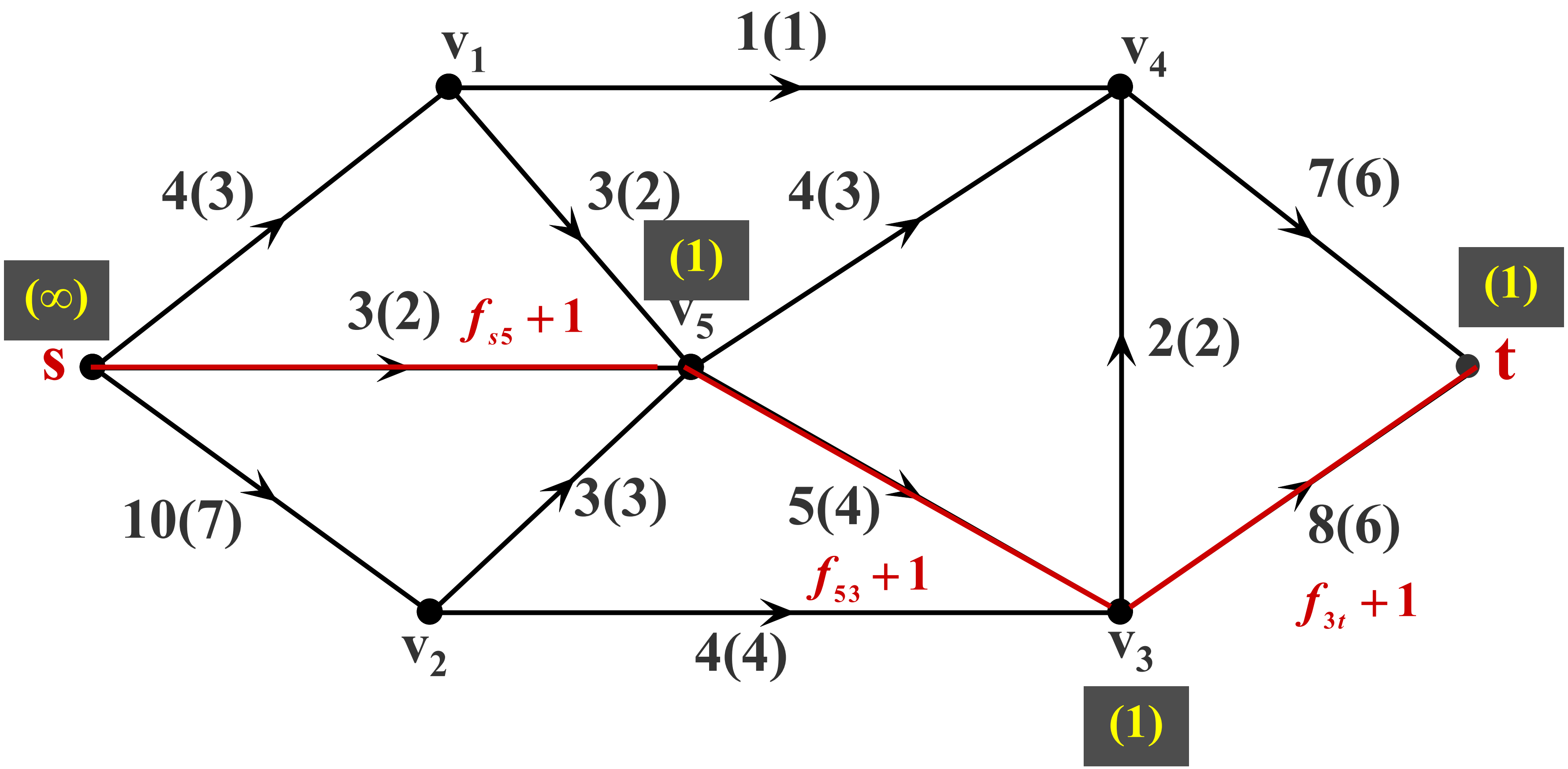
(7) 重新搜寻增广链。

存在增广链：$s \to v_5 \to v_3 \to t$



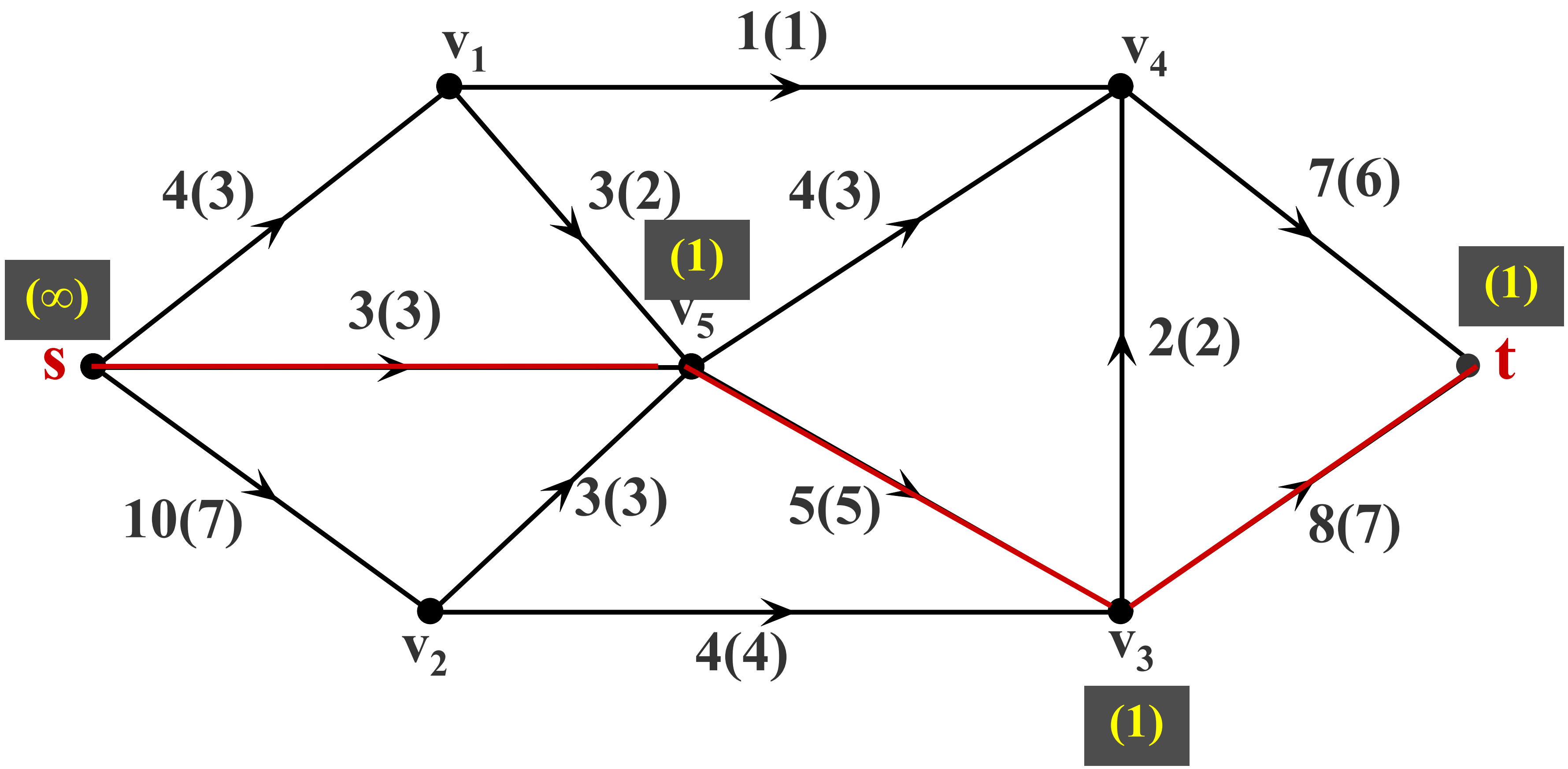
最大流问题

(8) 调整增广链上的流量，非增广链流量不变，得到新的可行流



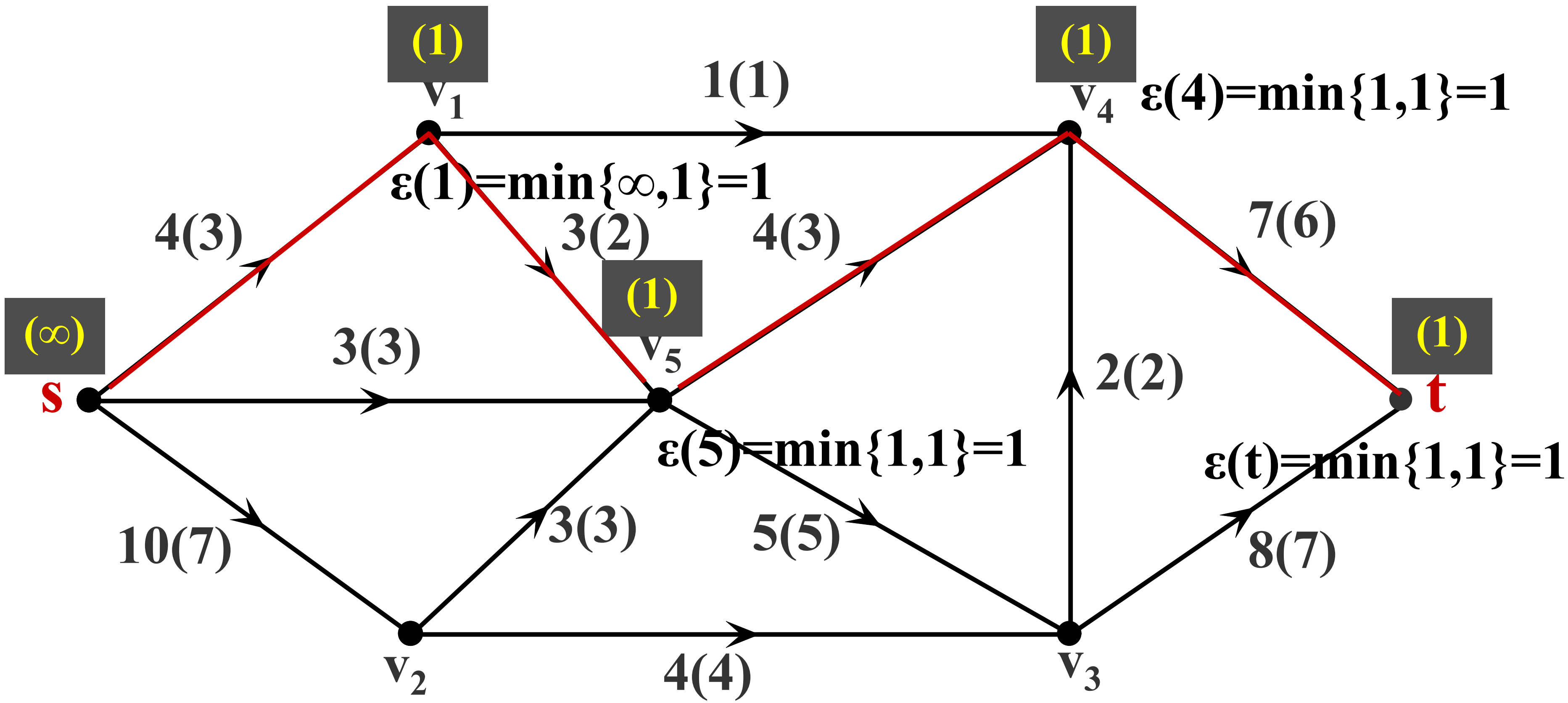
最大流问题

(9) 擦除原标号



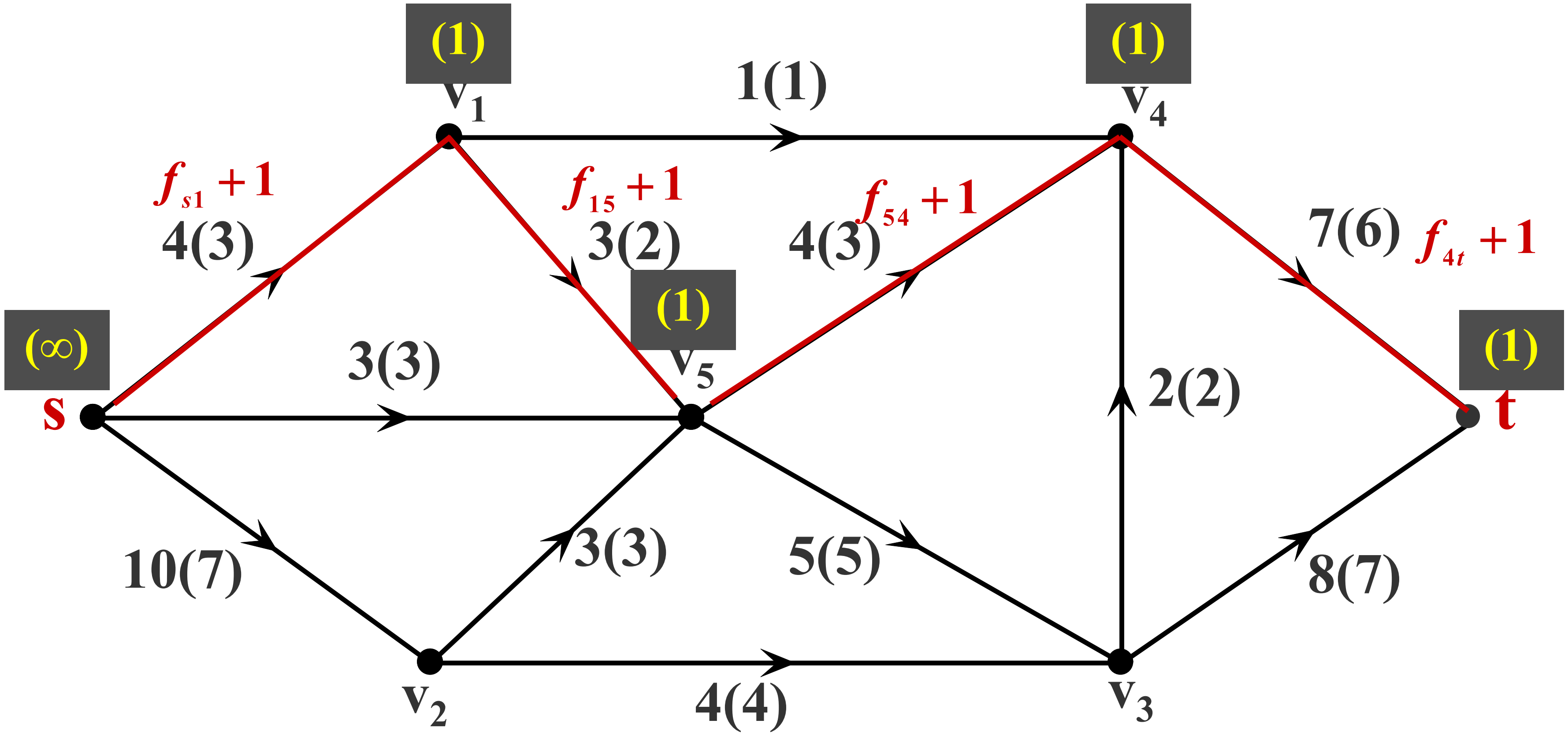
最大流问题

(10) 重新标号，搜索增广链　　存在增广链：$s\to v_1\to v_5\to v_4\to t$



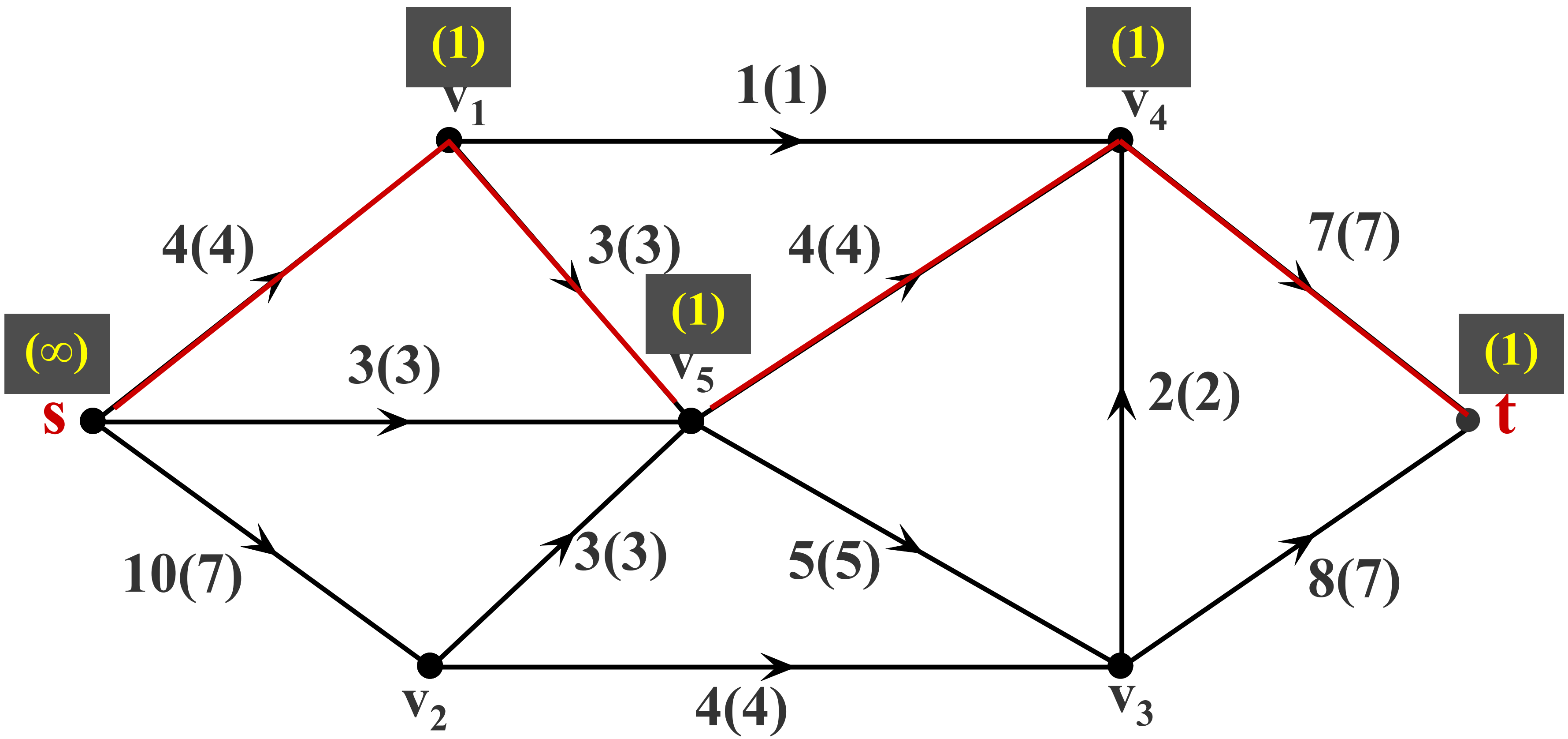
最大流问题

(11) 调整增广链上的流量，非增广链流量不变，得到新的可行流



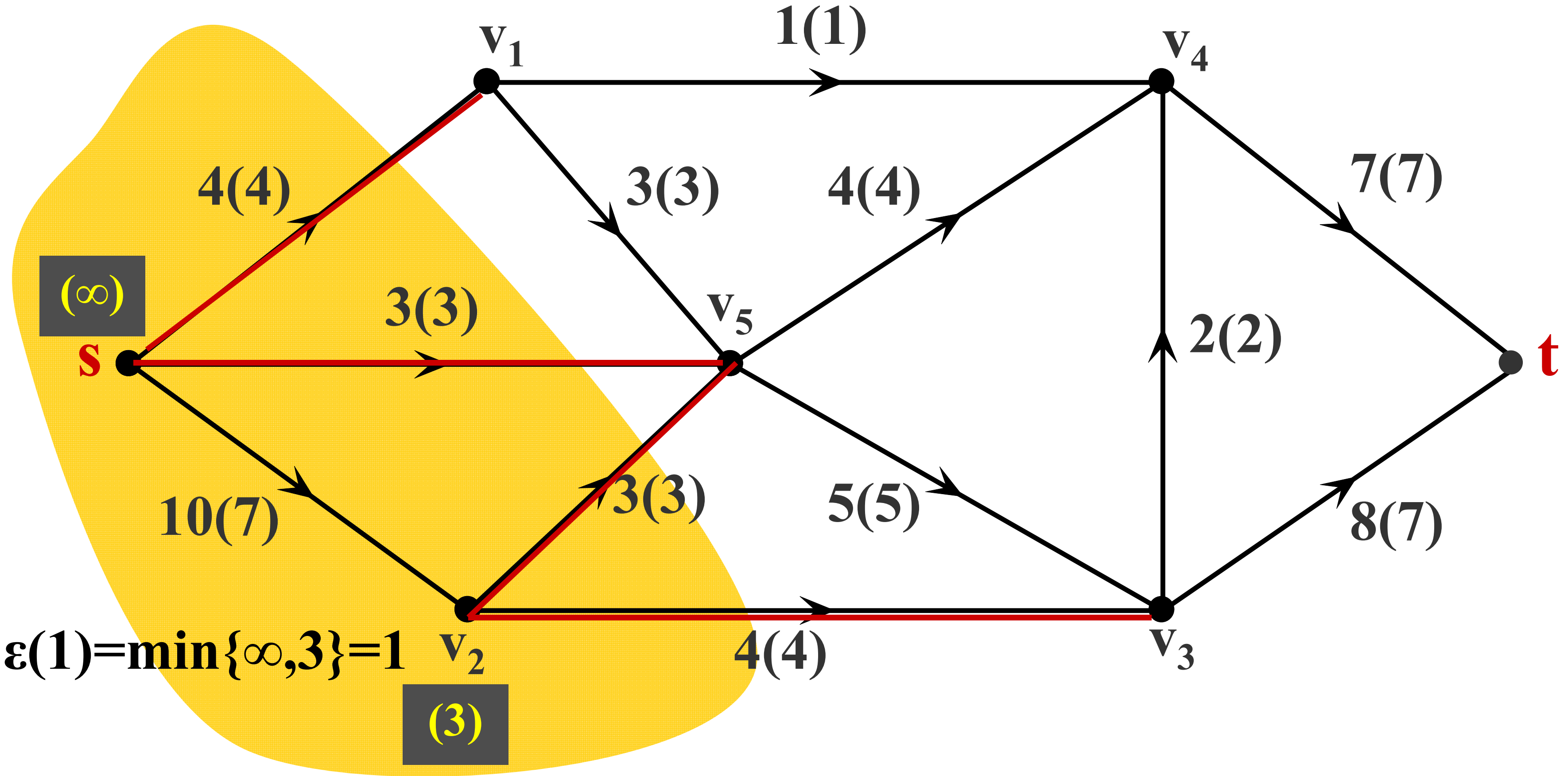
最大流问题

(11) 擦除标号，在新的可行流上重新标号。



最大流问题

(11) 擦除标号，在新的可行流上重新标号。



无法标号，不存在增广链，此可行流已为最大流。最大流量为14。

$V=\{s,v_2\},V'=\{v_1,v_3,v_4,v_5,t\}$, 最小割为 $\{V,V'\}$

最小费用流问题

除了考虑流量，还要考虑“费用”因素。

网络 $G=(V,A,c,b)$，其中权 $c_{ij}\geq 0$ 是弧 (v_i,v_j) 的容量，权 b_{ij} 是弧 (v_i,v_j) 上单位流量的费用。

线性规划模型

$$\min \quad b(f)=\sum_{(v_i,v_j)\in A} b_{ij} f_{ij}$$

$$s.t. \quad 0\leq f_{ij}\leq c_{ij}, \quad (v_i,v_j)\in A$$

$$\sum_{j:(v_i,v_j)\in A} f_{ij}-\sum_{j:(v_j,v_i)\in A} f_{ji}=\begin{cases}0, & i\neq s,t\\ v_0, & i=s\\ -v_0, & i=t\end{cases}$$

v_0 是指定的流值

最小费用流问题

给定流 f，定义增量网络 $G(f)=(V,A',c',b')$ 如下：

$A'=A^{+}\cup A^{-}$，其中，

若 $(v_i,v_j)\in A$ 且 $f_{ij}<c_{ij}$，则

$(v_i,v_j)\in A^{+}$，$c'_{ij}=c_{ij}-f_{ij}$，$b'_{ij}=b_{ij}$；

若 $(v_i,v_j)\in A$ 且 $f_{ij}>0$，则

$(v_j,v_i)\in A^{-}$，$c'_{ji}=f_{ij}$，$b'_{ji}=-b_{ij}$ 。

$G(f)$ 中的最小费用增广路：以 b'_{ij} 为权的 $s-t$ 最短路。

最小费用流算法：

i) 给初始流 $f=0$；

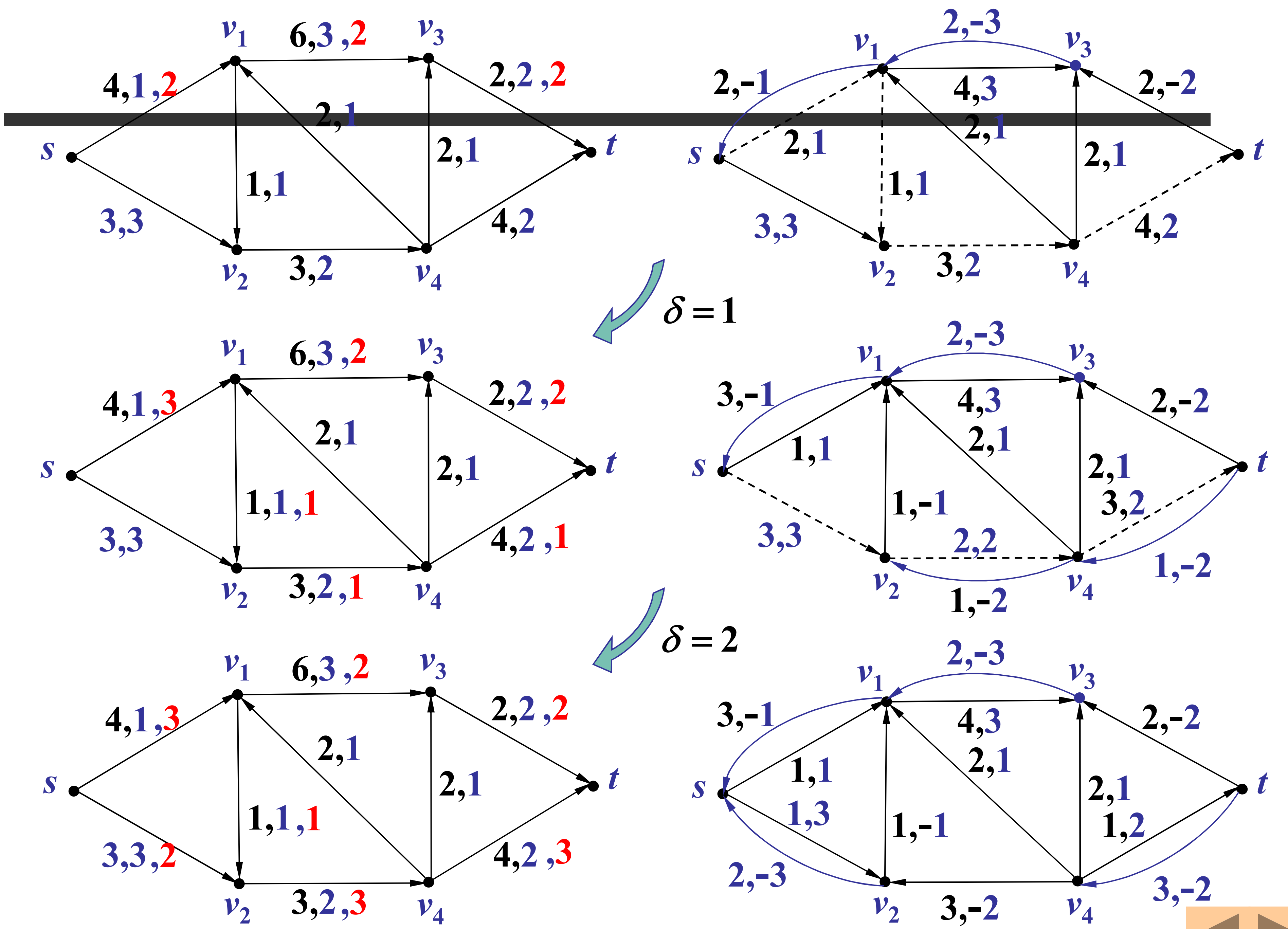
ii) 构造增量网络 $G(f)$；

iii) 在 $G(f)$ 中求最小费用增广路，若不存在，则 f 已是最小费用最大流；若求得增广路 P，则按以下方式修改 f：

$$f_{ij}=\begin{cases} f_{ij}+\delta\,, & (v_i,v_j)\in P\cap A^+ \\ f_{ij}-\delta\,, & (v_j,v_i)\in P\cap A^- \\ f_{ij}\,, & \text{否则} \end{cases}$$

其中 $\delta=\min\left\{v_0-v(f)\,,\ \min\limits_{(v_i,v_j)\in P} c'_{ij}\right\}$。

若 $v(f)=v_0$，则终止，否则返回 ii)。



v_1 v_2 v_3 v_4 s t
6,3 ,2
4,1 ,2
2,2 ,2
2,1
1,1
3,3
3,2
4,2
2,-3
2,-1
4,3
2,-2
$\delta = 1$
4,1 ,3
1,1 ,1
4,2 ,1
3,2 ,1
3,-1
1,-1
2,2
1,-2
$\delta = 2$
3,3 ,2
4,2 ,3
3,2 ,3
1,3
1,2
2,-3
3,-2

最小费用流算法（圈算法）：

i) 利用最大流算法求一个流值为 v_0 的流 f；

ii) 构造增量网络 $G(f)$；

iii) 在 $G(f)$ 中求负费用圈，若不存在，则 f 已是流值为 v_0 的最小费用流；若发现负费用圈 C，则按以下方式修改 f：

$$f_{ij} = \begin{cases} f_{ij} + \delta, & (v_i, v_j) \in C \cap A^+ \\ f_{ij} - \delta, & (v_j, v_i) \in C \cap A^- \\ f_{ij}, & \text{否则} \end{cases}$$

其中 $\delta = \min\limits_{(v_i, v_j) \in C} c'_{ij}$。

然后返回 ii)。

求流值为 3 的最小费用流：