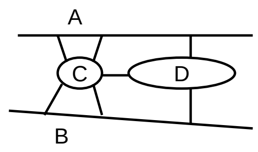
第三章 图论与网络模型

- 引子：哥尼斯堡七桥问题
- 图的概念
- 网络最短路径问题
- 最小生成树问题

- 网络最大流及最小费用流问题
- 关键路径分析与计划评审技术
- 应用举例

哥尼斯堡七桥问题

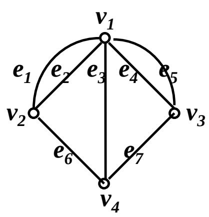
- 七桥分布如右图所示。
- 问题：从A、B、C、D中的某一处出发，通过每座桥恰好一次，再回到起点，是否可能？



哥尼斯堡七桥问题

欧拉于1736年解决这个问题。对上述问题的答案是”不可能“。

1、抽象为图
2、引入顶点度的概念：
即为与顶点相联的边的个数。



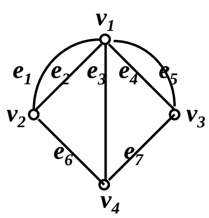
每个点的“度”为奇数(3 或 5), 必然先“出”后进，最后再”出“才能把与它相关联的边恰好经过一次，所以不可能回到出发点。

图的概念

- 图G＝(V,E)
  - 顶点集合V。顶点代表被研究的事物。
  - 边集合E。边代表事物之间的联系。每条边连接两个顶点。
  - 边的属性：边的权重和方向。
- 顶点的位置、边和长短形状都是无关紧要的，只要两个图的顶点及边是对应相同的，则两个图相同(同构)。

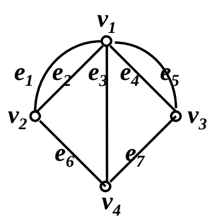
图的表示

$V = \{v_1, v_2, V_3, V_4\}$
$E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$
$e_1 = [v_2, v_1]$ $e_2 = [v_2, v_1]$
$e_3 = [v_1, v_4]$ $e_4 = [v_1, v_3]$
$e_5 = [v_1, v_3]$ $e_6 = [v_2, v_4]$
$e7 = [v_4, v_3]$



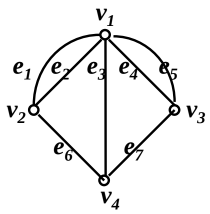
点边关系

- 若点u和v与同一条边相关联，则u和v为相邻点；若两条边ei和ej有同一个端点，则称ei与ej为相邻边。
- 例如在右图中v_1和v_2为相邻点， v_1和v_4不相邻；e_1与e_5为相邻边，e_1和e_7不相邻。



顶点的度数

- 度：点v作为边的端点的次数，记作d(v)。
- 端点度数为奇数的点称作奇点；度数为偶数的点称作偶点。
- 度数为0的点称为孤立点。



欧拉(Euler)定理

- 若图G中所有点都是孤立点，则称图G为空图。
- 欧拉定理 所有顶点的度数之和，等于所有边数的2倍。

$$\sum_{v \in V} d(v) = 2q$$

- 七桥问题中：度数和5+3+3+3=14，正好是桥数的2倍。

定理2

- 定理2：在任一图中，奇点的个数必为偶数。
- 设V_1和V_2分别是图G中度数为奇数和偶数的顶点集合。由欧拉定理有

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

简单图

- 若一条边的两个端点是同一个顶点，则称该边为环；又若两上端点之间有多于一条边，则称为多重边。
- 含有多重边的图称作多重图。
- 无环也无多重边的图称作简单图。
- 以下仅仅考虑简单图。

链的概念

- 由两两相邻的点及其相关联的边构成的点边序列称为链。

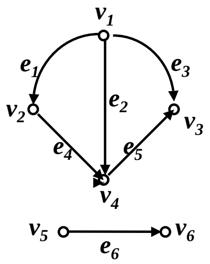
$$(v_0, e_1, v_1, \cdots, v_{n-1}, e_n, v_n)$$

- v_0称为链的起点， v_n称为链的终点。
- 若$v_0 \neq v_n$则称该链为开链，反之称为闭链或回路。

有向图

- 边没有方向的图称为无向图。
- 而有些关系是不对称的，例如父子关系、上下级关系、加工工序的先后顺序等都具有单向性，用图来表示这些关系时，得到的边是具有方向的，用带箭头的线来表示，称为弧。
- 从顶点u指向 υ 的弧a，记作＝a=(u,v),(u,v)≠(v,u)，其中u称为a的起点，v称为a的终点，这样的图称为有向图。仍以V表示点的集合，以A表示弧的集合，则有向图表示为D＝（V，A）

有向图例

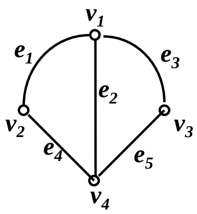


有向图的链路

- 有向图中，在不考虑边的方向时，也可以相同地定义链，若有向图D＝（V，A）中，P是一个从u到v的链，且对P中每一条弧而言，在序列中位于该弧前面的点恰好是其起点，而位于该弧后面的点恰好是其终点，这个链P就称为是D中从u到v的一条路径。
- 当路的起点与终点相同，即u=v时，称作一条回路。

圈的概念

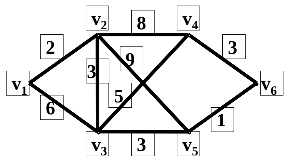
- 若链中所含的边均不相同，则称为简单链；若点均不相同，则称为初等链。
- 除起点和终点外点均不相同的闭链，称为初等回路或称为圈。
- 右图中有圈



$$(v_1, e_1, v_2, e_4, v_4, e_5, v_3, e_3, v_1)$$

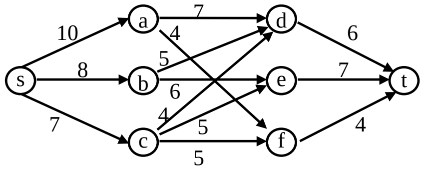
网络的概念

- 图只能用来研究事物之间有没有某种关系，而不能研究这种关系的强弱程度。
- 网络：赋权的图
- 权：程度的度量，数量描述。



网络最短路径问题

- 问题：给定连接若干城市的铁路网络，找一条给定两城市之间的最短路径。即所谓最短路径问题。比如说找从城市s到t的短路径。1959年，Dijkstra给出了算法。



狄克斯拉(Dijkstra)算法

1. 初始化。令

$$l(v_0) = 0, l(v) = \infty, \forall v \neq v_0$$
$$S_0 = \{v_0\}, i = 0$$

2. 迭代。

$$\mathrm{Min}\{l(v), l(u_i) + w(u_i v)\} \to l(v), \forall v \in \overline{S}_i$$
$$u_{i+1} = \operatorname*{Argmin}_{v \in \overline{S}_i} l(v)$$
$$S_{i+1} = S_i \bigcup u_{i+1}$$

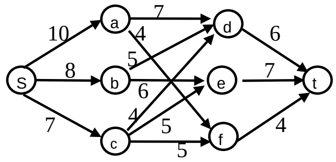
3. 若$i = n-1$ 停止；否则$i+1 \to i$，转(2).

Dijkstra算法说明

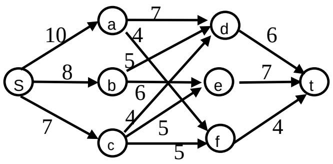
- 对于S_i中的每个点v，l(v)记录了从v到v_0的最短距离。当算法结束时，S_i=V(G)，则此时l(v)记录了v_0到每个点的最短距离。
- 算法本质上是一个动态规划算法。
- 算法复杂度为$O(n^2)$, 其中n为顶点个数。

Dijkstra算法示例

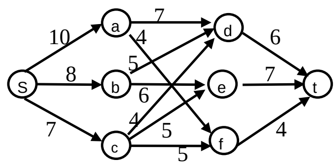
初始化 $l(v_s)=0$，$l(v_i)=\infty$, $i=1,\cdots,7$, $S_0=\{v_s\}$.



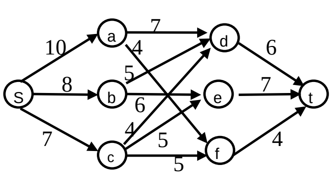
$l(a)=\min\{\infty,0+10\}=10$
$l(b)=\min\{\infty,0+8\}=8$
$l(c)=\min\{\infty,0+7\}=7$
$l(d)=l(e)=l(f)=l(t)=\infty$
$v_1=\underset{v\in \bar{S}_0}{\mathrm{Argmin}}\, l(v)=c$
$S_1=\{s,c\}$
$i=1$



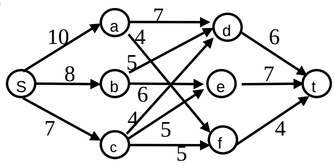
$l(a)=\min\{10,0+10\}=10$
$l(b)=\min\{8,0+8\}=8$
$l(d)=\min\{\infty,10+7,8+5,7+4\}=11$
$l(e)=\min\{\infty;8+6,7+5\}=12$
$l(f)=\min\{\infty,10+4,7+5\}=12$
$l(t)=\infty$
$v_2=\underset{v\in \bar{S}_1}{\mathrm{Argmin}}\, l(v)=b$
$S_2=\{s,c,b\}$
$i=2$



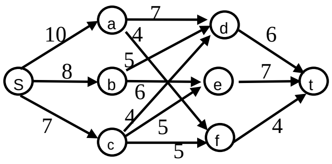
$l(a)=\min\{10;0+10\}=10$
$l(d)=\min\{11;10+7,8+5,7+4\}=11$
$l(e)=\min\{12;8+6,7+4\}=12$
$l(f)=\min\{12;10+4,7+5\}=12$
$l(t)=\min\{\infty;11+6,12+7,12+4\}=16$
$v_3=\underset{v\in \bar{S}_2}{\mathrm{Argmin}}\, l(v)=a$
$S_3=\{s,c,b,a\}$
$i=3$



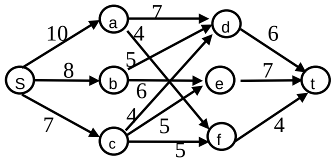
$l(d)=\min\{11;10+7,8+5,7+4\}=11$
$l(e)=\min\{12;8+6,7+5\}=12$
$l(f)=\min\{12;10+4,7+5\}=12$
$l(t)=\min\{16;11+6,12+7,12+4\}=16$
$v_4=\underset{v\in \bar{S}_3}{\mathrm{Argmin}}\, l(v)=d$
$S_4=\{s,c,b,a,d\}$
$i=4$



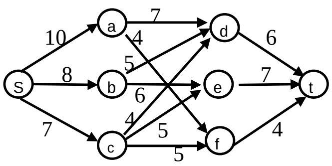
$l(e) = \min\{12; 8+6, 7+5\} = 12$
$l(f) = \min\{12; 10+4, 7+5\} = 12$
$l(t) = \min\{16; 11+6, 12+7, 12+4\} = 16$
$v_5 = \underset{v \in \bar{S}_4}{\mathrm{Argmin}}\, l(v) = e$
$S_5 = \{s, c, b, a, d, e\}$
$i = 5$



$l(f) = \min\{12; 10+4, 7+5\} = 12$
$l(t) = \min\{16; 11+6, 12+7, 12+4\} = 16$
$v_6 = \underset{v \in \bar{S}_5}{\mathrm{Argmin}}\, l(v) = f$
$S_6 = \{s, c, b, a, d, e, f\}$
$i = 6$

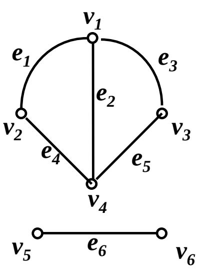


$l(t) = \min\{16; 11+6, 12+7, 12+4\} = 16$
$v_7 = \underset{v \in \bar{S}_6}{\mathrm{Argmin}}\, l(v) = t$
$S_7 = \{s, c, b, a, d, e, f, t\}$
$i = 7$停止



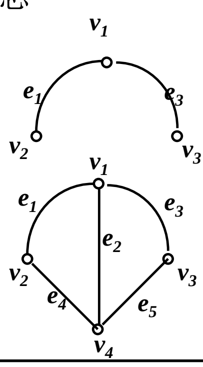
图的连通性

- 若一个图G的任意两点之间均至少有一条链连接起来，则称这个图G是一个连通图，否则称作不连通图。
- 例如图中，v_1和v_5之间没有通路，因此它不是连通图。
- 不连通图可以分解成连通的子图来分别研究。



子图的概念

- 子图的定义 设，$G_1=(V_1,E_1)$, $G_2=(V_2,E_2)$，如果$V_1 \subseteq V_2$，又$E_1 \subseteq E_2$，则称G_1是G_2的子图。
- 注意：并不是从图G_2中任选一些顶点和边在一起就组成G_2的子图G_1，而只有在G_2中的一条边以及连接该边的两个端点均选入G_1时，G_1才是G_2的子图。



特殊子图

- 当G_1中不包含G_2中所有的顶点和边，则称G_1是G_2的真子图。
- 部分图：若$V_1=V_2$，$E_1 \subset E_2$，则称G_1为G_2的一个部分图。
- 若$V_1 \subseteq V_2$，$E_1=\{ [u,v] \in E \mid u \in V_1, v \in V_1\}$，则称$G_1$是$G_2$中 由$V_1$导出的导出子图。

树的概念

- 一个没有圈的图称为一个无圈图或称为林。
- 一个连通的无圈图则称为树，一个林的每个连通子图都是一个树。
- 定理 以下关于树的六种不同描述是等价的：
  - 无圈连通图。
  - 无圈，q=p-1。
  - 连通，q=p-1。
  - 无圈，但若任意增加一条边，则可得到一个且仅一个圈。
  - 连通，但若任意舍弃一条边，图便不连通。
  - 每一对顶点之间有一条且仅有一条链。

生成树

- 若T是图G=(V, E)的部分图，且T是树，则称T为G的生成树。
- 若T是图G的部分树，则从G中去掉T中所有的边，所得到的子图称为G中的T的余树，也称为G的一个余树。
- 余树不一定是树！

网络最小生成树问题

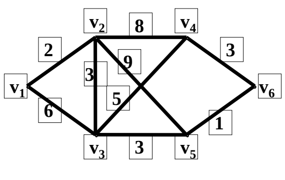
- 问题：要在几个城市之间架设光缆，已知城市之间的距离，试求应如何架设光缆，才能使任意两城市之间均由光缆相通，且使光缆的总长度最短。
- 数学模型：选取网络中的部分图，使得网络连通，且使总权数最小。
- 两种求解方法：一种称为破圈法，一种称为生长法 (Kruskal算法，1956年)

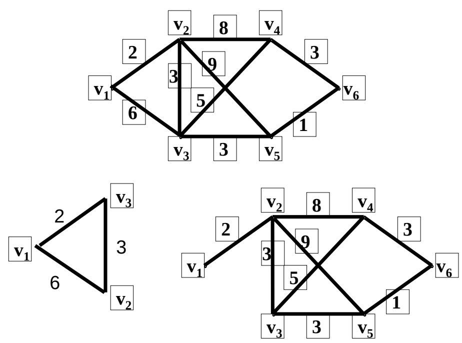
破圈法原理

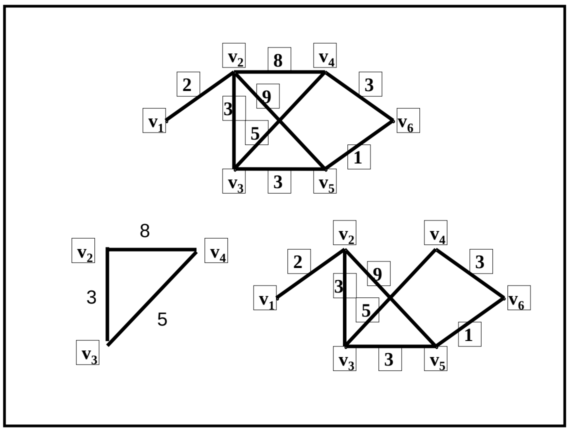
- 如果网络图中无圈并且q=p-1，则已经是树；
- 如果网络图中有圈，则截去该圈中权数最大的边；这样，并不影响网络图的连通性，且能使边数减少一个；
- 经过一定次数的截边，网络图中将再也没有圈，成为无圈图；
- 如果此时的网络满足q=p-1，则已经是树；
- 由于每次截去的边在圈中具有最大的权数，因此获得的树也是最短的树。

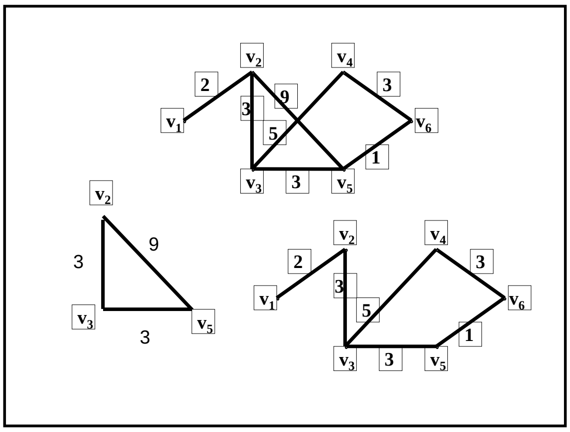
破圈法

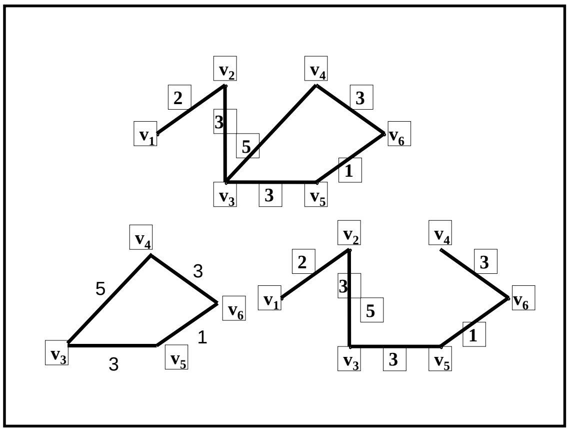
- ①在网络图中寻找一个圈，若已经无圈则转③。
- ②在圈中选取权数最大的边，从网络图中截去该边，对新的网络，转①。
- ③若q=p-1，则已找到最短树，否则网络图不连通，无最小生成树。











Krustral算法

1. 初始化。选$e_1 \in E(G)$，使得$w(e_1) = \min\limits_{e_j \in E(G)} w(e_j)$.

2. 迭代。若已经选好$\{e_1, \cdots, e_i\}$，从余下的边$E(G) - \{e_1, \cdots, e_i\}$ 中选取e_{i+1}，使得

   (i) $G[\{e_1, \cdots, e_i, e_{i+1}\}]$中无圈。

   (ii) $w(e_{i+1}) = \min\limits_{e_j \in E(G) - \{e_1, \cdots, e_i\}} w(e_j)$

3. 直到选到e_{n-1}为至，其中n为连通图G的节点数.