…近似算法.

1. 分治法

将 X 向量拆分对半为 $\vec{x}_1, \vec{x}_2$，则 $H_k x = \frac{1}{\sqrt{2}}\begin{bmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & -H_{k-1} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$$= \frac{1}{\sqrt{2}}\begin{bmatrix} H_{k-1}\vec{x}_1 + H_{k-1}\vec{x}_2 \\ H_{k-1}\vec{x}_1 - H_{k-1}\vec{x}_2 \end{bmatrix}$$

记 $H_k x$ 复杂度 $\cancel{\Theta(n)}$ $T(n)$，n 为 x 维数，则

有 $T(n) = 2T(\frac{n}{2}) + \Theta(n)$，即先计算 $H_{k-1}\vec{x}_1$ 与 $H_{k-1}\vec{x}_2$，

再 $\Theta(n)$ 时间进行向量加法。

```
FJLT(H, x):
    res1 = FJLT(H_{k-1}, x1)
    res2 = FJLT(H_{k-1}, x2)
    return 1/√2 [ res1 + res2
                  res1 - res2 ]
```

时间复杂度由主定理 $n^{\log_b a} = n^{\log_2 2} = n = \Theta(f(n))$，故 $T(n) = \Theta(n\log n)$

2. (Hash 函数运用)

$\forall a, \hat{f}_a = g(a)\,C[h(a)]$. 记 X_p 为 $h(i_p)=h(a)$ 这一事件的示性函数，则

$$\hat{f}_a = g(a)\left(\sum_{p=1}^{n} X_p g(i_p)\cdot c_p\right)$$

$$= \sum_{p=1}^{n} X_p c_p \cdot g(a)\cdot g(i_p)$$

这其中 c_p 为常数，X_p 与 $g(a)$, $g(i_p)$ 相互独立（h, g 随机选取）.

$$\text{故 } E(X) = E(\hat{f}_a) = \sum_{p=1}^{n} E(X_p \cdot c_p \cdot g(a)\cdot g(i_p))$$

$$= \sum_{p=1}^{n} c_p \cdot E(X_p)\cdot E(g(a)\cdot g(i_p))$$

$$= \sum_{i_p \neq a} c_p \cdot E(X_p)\cdot E(g(a)\cdot g(i_p)) + c_a\cdot 1\cdot 1$$

而 $g(a)\cdot g(i_p)$ 取 $1/-1$ 的概率相同，故 $E(g(a)\cdot g(i_p)) = 0$，即前半部分为0，故有：

$$E(X) = c_a.$$

2) 快速傅立叶变换(FFT)

4APD问题：n个整数构成数组A，元素A[i]都在 2n-1 之内，给定整数t，应用FFT在 $O(n\log n)$ 时间内求解判断是否存在 $i_1 \cdots i_4$ 使 $A[i_1]+A[i_2]+A[i_3]+A[i_4]=t$（无需返回 $i_1 \cdots i_4$）

解：·构造多项式 $p(x)=\sum_{i=0}^{n-1} x^{A[i]}$ 并扩展至8n-1次，以系数0扩展。(点表示)

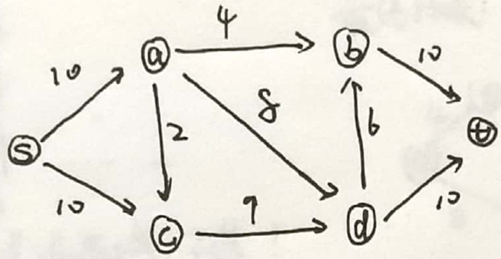
· 使用FFT变换将p(x)变换到频域表示；计算(频域上) $p^4(x)$；
使用IFFT将频域表示变换回时域，得到 Q(x)；

· 考察 Q(x)中是否存在 x^t 项，若存在则有，否则无。

1与3步均为O(n) 而2步 FFT与IFFT均为 $O(n\log n)$，故总复杂度 $O(n\log n)$

3. 网络流与最小割

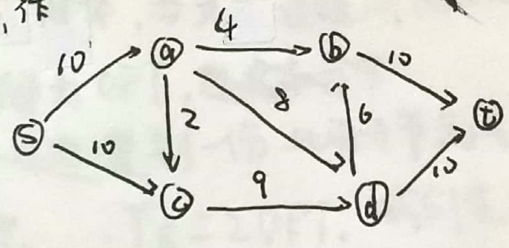
给定流网络：



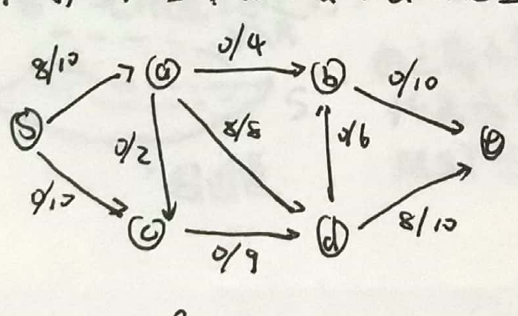
a) 找到 s→t 最大流并在各边上标出，求最大流量值。

b) 流分解。

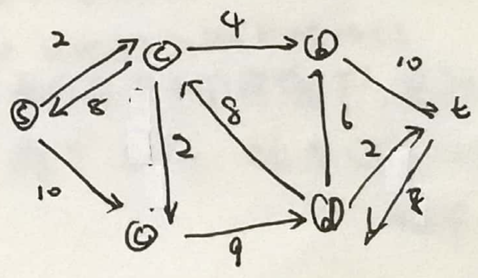
解 a) 初始化f=0，作残差 G_f 网络



有增广路 s→a→d→t，流量为8。

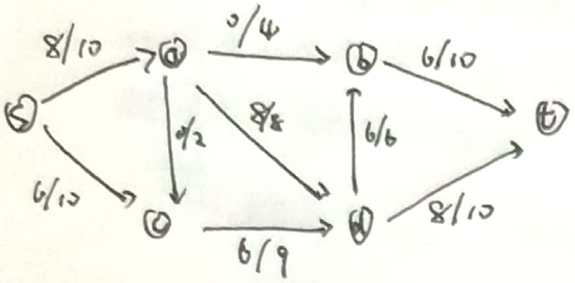


G

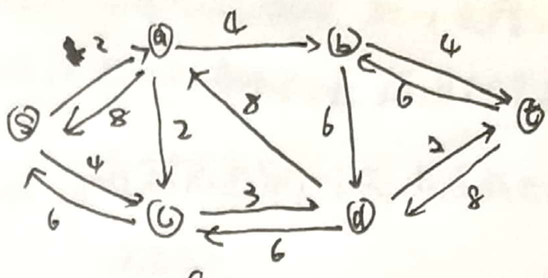


G_f

有增广路 s→c→d→b→t，流量为6

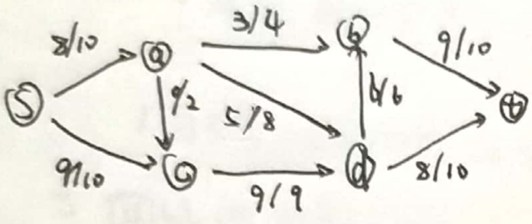


G

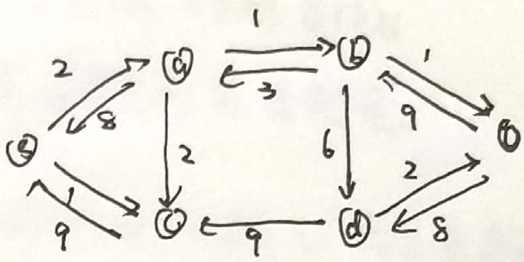


Gf

有增广路 S→c→d→a→b→t，流量为3

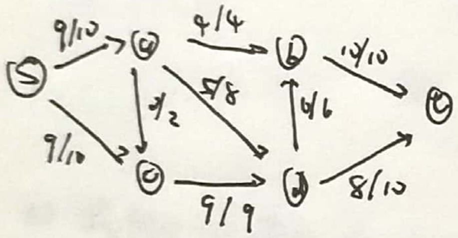


G

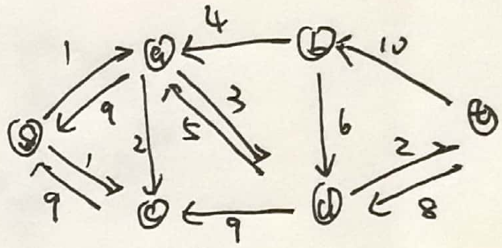


Gf

有增广路 S→a→b→t，流量为1

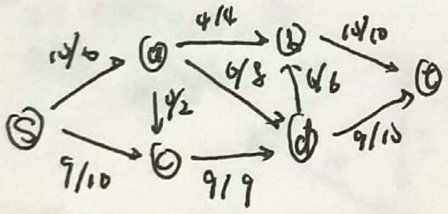


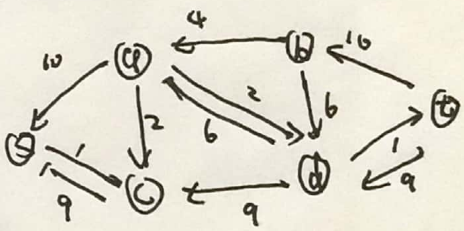
G



Gf.

有增广路 S→a→d→t，流量为1





Gf

无增广路，已达最大流.

2) ~~[illegible]~~法，可分解为

s→a→d→t，流量为6；s→a→b→t，流量为4；

s→c→d→t，流量为9

a) $P[S,i] = \begin{cases} 0 & |S| = 1 \\ \max\limits_{j} \{ P[S-\{i\}, j] + pref(j,i) ;\ j \neq i ;\ j \in S \} & |S| > 1 \end{cases}$

b)

```
MAX_PREFERENCE:
    ~~for all i in {1,...,n}:~~              // 初始化边界条件
        ~~P[{i}, i] = 0~~   for all cell in P:
                                P[][] = 0
    for s = 2 to n:                        // s 表示 |S|
        for all subset of {1...n} S whose size is s:
            for all j ∈ S:                 // 视情况更新 P[S,i], 取 max_j ...
                P[S,i] = max{P[S,i], P[S-{j}, j] + pref(j,i)}
    return max_i {S, i}
```

c) 共有子问题数 $2^n * n$ 个. 每个子问题花费时间 $O(n)$ 求最大值, 故总时间复杂度 $O(n^2 \cdot 2^n)$

5. LP与NPC

击中集问题：给定集合$E=\{e_1,e_2\cdots e_n\}$与m个子集$S_1,\cdots S_m\subseteq E$. 若$H\subseteq E$与每个$S_i$交非空，称H为击中集，求最小击中集.

(1) 用整数规划(IP)建模.

(2) 松弛为线性规划(LP)后求对偶形式

(3) 写出击中集判定版本，简单说明其为NP

解：(1) 决策变量：$x_i=\begin{cases}1 & e_i\in H\\ 0 & e_i\notin H\end{cases}$

为每个E中元素构造一个变量，代表其是否为击中集选中.

目标函数：最小化 $\sum_{i=1}^{n} x_i$

约束：$x_i\in\{0,1\}$ $i=1\cdots n$

$\sum_{i:e_i\in S_j} x_i\geqslant 1$ $\forall j=1\cdots m$

(2) 松弛后约束1变为

$x_i\geqslant 0$ $i=1\cdots n$

对偶形式：

最大化 $\sum_{i=1}^{m} y_i$

约束为 $y_i\geqslant 0$ $i=1\cdots m$

$\sum_{i:e_j\in S_i} y_i\leqslant 1$ $\forall j=1\cdots n$

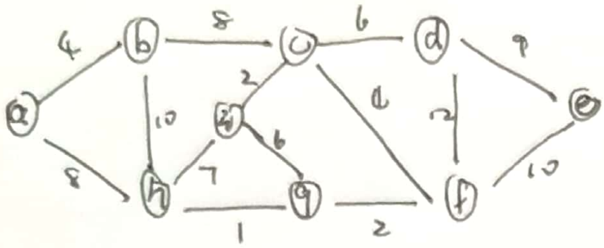
(3) 是否存在集合基数小等于k的击中集

判定版本：

需证其多项式时间可“验证”，若给出一种击中集H的取法，将H分别与$S_1\cdots S_m$取交集，观察是否为空. 这与m个子集合的基数之和成正比，多项式时间可验证，故其为NP

(扩展：亦可证击中集为NPC. 由顶点覆盖归约. E取顶点V，$S_i=\{u,v\}$,$(u,v)\in E$)

6. 最小生成树与近似算法.



1) Kruskal算法过程，给出边序列

2) Steiner树：给定边上带非负权重无向完全图 $G=(V,E,W)$，顶点子集 $R\subseteq V$ 称终点集，找权重最小树将R中顶点连接——可使用 V/R 中顶点。
近似算法：返回R的导出子图 $G[R]$ 上一棵最小生成树. 证边权满足三角不等式时. 近似度为2.

解：

(1) 边权由小到大选择. 初始各点自成连通域

① (h,g) h与g分别不同连通域. 选择.

② (i,c). 选择

③ (g,f). 选择

④ (a,b). 选择

⑤ (c,f) 选择

⑥ (c,d) 选择

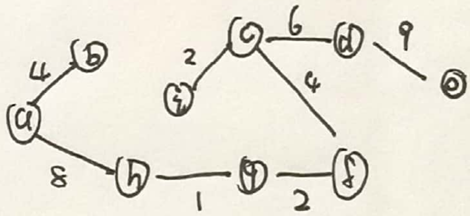
⑦ (h,i). 同属连通域. 不选!

⑧ (a,h) 选择

⑨ (b,c) 不选择!

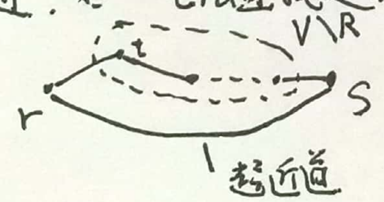
⑩ (d,e) 选择，各点均连通. 算法结束.

最终结果：



(2) 证 设最优解为OPT，~~边权和为下~~
将OPT中各边复制一份. 此时各节点度数为偶，可找出一条欧拉回路. 记为 T_E. 显然 $T_E = 2OPT$. 我们在其中任取一点 $r\in R$. 沿欧拉回路行进. 若无法直接走到下一个R中的点S，则直接跳，以"超近道".

如图：



由三角不等式知 $w(r,s) > w(r\to t\to s)$
结束后得到一条仅在R中的回路 T'，显然 $MST \le T'$. 而 $T' \le T_E$. 故 $MST \le 2OPT$

证毕