近似算法

东南大学计算机学院 方效林

本章内容

- 近似算法的基本概念
- 顶点覆盖问题
- 集合覆盖问题
- 旅行商问题
- 子集和问题
- 线性规划

近似算法的基本概念

- 实际应用中很多问题都是NP-完全问题
- 求解NP-完全问题很难
  - 若NP-完全问题输入规模很小，可指数级穷举搜索
  - 否则，用多项式算法近似

近似算法的基本概念

- 近似算法
  - 能够给出一个优化问题的近似优化解的算法
  - 主要解决优化问题(最大化、最小化)
- 近似算法的时间复杂度分析
  - 分析方法与传统算法一致
- 近似算法的近似度(近似解与优化解的差距)
  - Ratio Bound
  - 相对误差

近似算法的基本概念

- Ratio Bound
  - 设A是一个优化问题的近似解，A的Ratio Bound为$B(n)$，则
  - $\max\left\{\frac{A}{OPT},\frac{OPT}{A}\right\}\le B(n)$
  - 若问题是最大化问题，则$\max\left\{\frac{A}{OPT},\frac{OPT}{A}\right\}=\frac{OPT}{A}$
  - 若问题是最小化问题，则$\max\left\{\frac{A}{OPT},\frac{OPT}{A}\right\}=\frac{A}{OPT}$
  - Ratio Bound越大，近似解越坏

近似算法的基本概念

- 相对误差
  - 设A是一个优化问题的近似解，A的相对误差为 $\frac{|A-OPT|}{OPT}$
- 相对误差界
  - $\frac{|A-OPT|}{OPT} \leq \varepsilon(n)$

近似算法的基本概念

- 相对误差与Ratio Bound关系
  - $\varepsilon(n) \le B(n) - 1$
  - 对于最小化问题
    - $\varepsilon(n) = \frac{|A-OPT|}{OPT} = \frac{A-OPT}{OPT} = \frac{A}{OPT} - 1 = B(n) - 1$
  - 对于最大化问题
    - $\varepsilon(n) = \frac{|A-OPT|}{OPT} = \frac{OPT-A}{OPT} = \frac{OPT/A-1}{OPT/A} = \frac{B(n)-1}{B(n)} \le B(n) - 1$
- 只要求出了Ratio Bound，就求出了相对误差

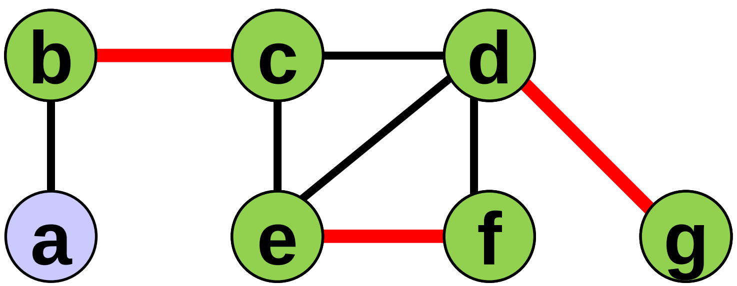
顶点覆盖问题

- 输入：无向图 $G=(V,E)$
- 输出：$A\subseteq V$，满足
  - $\forall(u,v)\in E$，其中 $u\in A$或者$v\in A$
  - 且$|A|$大小最小

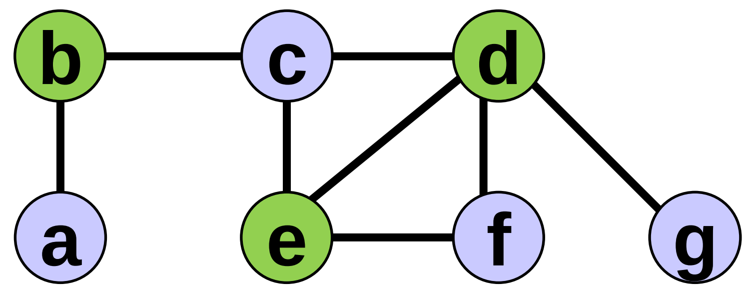
顶点覆盖是NP-完全问题

顶点覆盖问题

- 近似算法A的基本思想
  - 随机选一条边(u,v)，删除与u或v相连的边
  - 重复，直到无边，将选中的边的端点作为结果



算法解{b,c,e,f,d,g}



最优解{b,c,d}

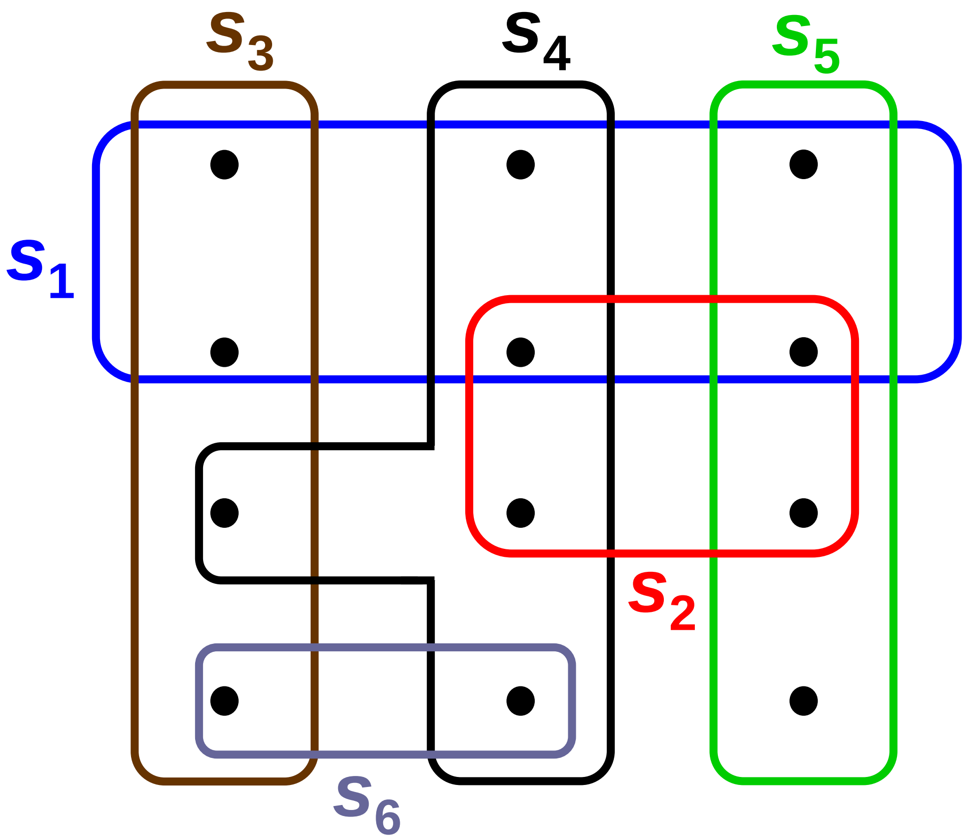
顶点覆盖问题

- 近似算法A性能分析
  - 时间复杂度为O(|E|)
  - Ratio Bound为2
    - 令E'是选中的边的集合，若$(u, v) \in E'$，则与(u, v)相邻的边都被删除，因此E'中无相邻边
    - 每次选择一条边，即每次有两个顶点加入近似解A，$|A| = 2|E'|$
    - 设OPT是最优解，OPT必须覆盖E'，由于E'中无邻接边，OPT至少包含E'中每条边的一个顶点
    - $|E'| \le |OPT|$，$|A| = 2|E'| \le 2|OPT|$，即$\frac{|A|}{|OPT|} \le 2$

集合覆盖问题

- 输入：有限集$U=\{1,2,\ldots,n\}$，子集合的集合$S=\{s_1,s_2,\cdots,s_m\}$，$s_i\subseteq U$
- 输出：$A\subseteq S$，满足
  - $U=\bigcup_{s\in A} s$
  - 且$|A|$最小

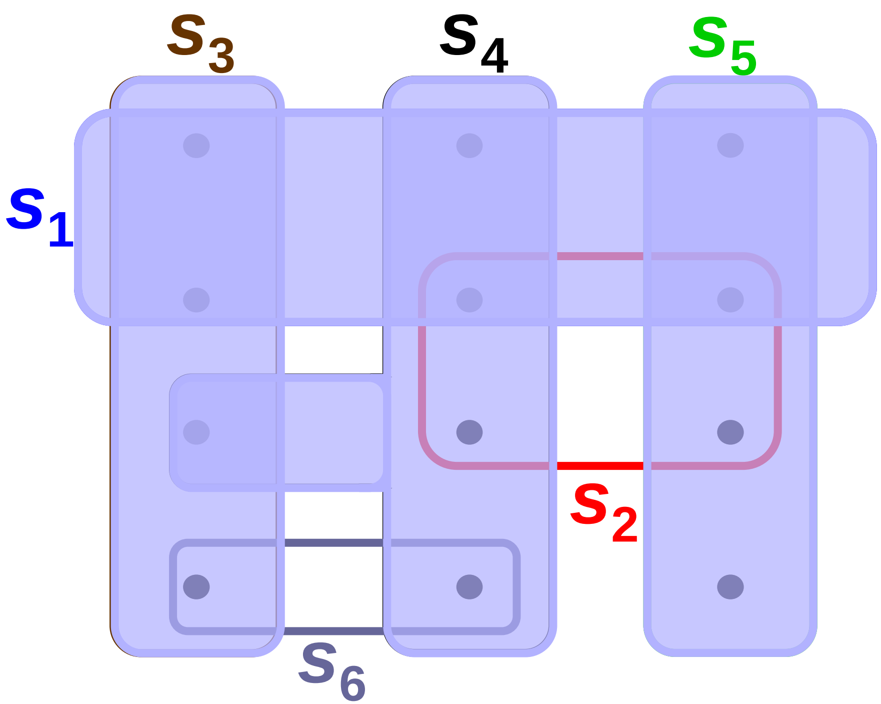
集合覆盖问题



最优解$\{s_3, s_4, s_5\}$

集合覆盖问题

- 近似算法A的基本思想
  - 每次迭代选择能覆盖最多未被覆盖元素的子集



近似解$\{s_1, s_4, s_5, s_3\}$

集合覆盖问题

- 近似算法A性能分析
  - 时间复杂度
    - 每次选能覆盖最多未被覆盖元素的子集
    - $|S|$个子集，判断覆盖未被覆盖元素个数，一次选择要计算$|S||U|$次
    - 共选择$\min\{|S|, |U|\}$次
    - 总计算复杂度$|S||U|\min\{|S|, |U|\}$

集合覆盖问题

- 近似算法A性能分析
  - Ratio Bound为 $\ln n+1$
    - 令OPT为最优解，其每个元素平均覆盖代价为$\frac{|OPT|}{n}$
    - 算法A第一次迭代时，必然有一个集合s_1，其平均代价$\frac{1}{n_1}\le\frac{|OPT|}{n}$，不然$OPT$不存在的
    - 同理，设第i次迭代时剩余k_i个元素未被覆盖，对这k_i个元素覆盖最优解为OPT_i，则必然存在一个集合s_2，$\frac{1}{n_i}\le\frac{|OPT_i|}{k_i}\le\frac{|OPT|}{k_i}$。算法A的代价$\frac{1}{n_1}n_1+\frac{1}{n_2}n_2+\cdots+\frac{1}{n_i}n_i\le\sum\frac{|OPT|}{k_i}n_i\le|OPT|\sum_{x=1}^{\infty}\frac{1}{x}\le|OPT|(\ln n+1)$

$\frac{n_1}{k_1}=\frac{n_1}{n}$ $\frac{n_2}{k_2}=\frac{n_2}{n-n_1}$ $\frac{n_3}{k_3}=\frac{n_3}{n-n_1-n_2}$

旅行商问题(Hamilton环)

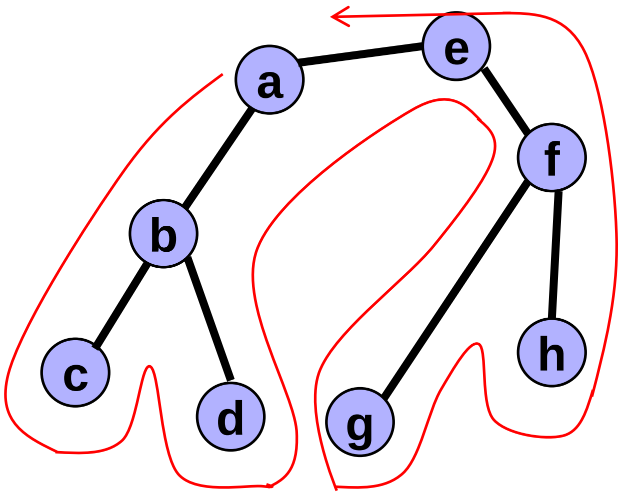
- 给定一个图，Hamilton环是包含图中每个顶点一次的简单环

旅行商问题(Hamilton环)

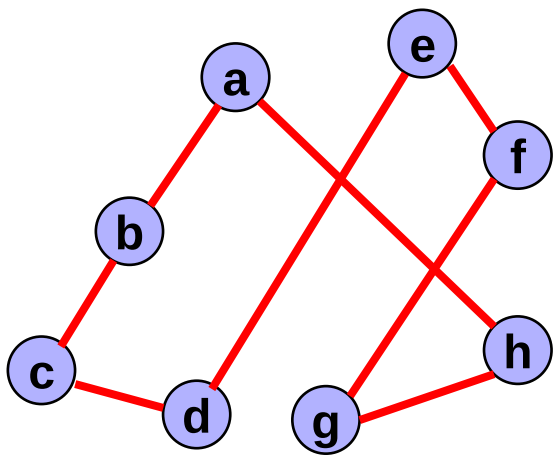
- 输入：完全无向图$G = (V, E)$，
- 每条边(a, b)存在一个权值$C(a, b)$，
- 边满足三角不等式$C(a, b) + C(b, c) \geq C(a, c)$
- 输出：边权值和最小的Hamilton环

旅行商问题(Hamilton环)

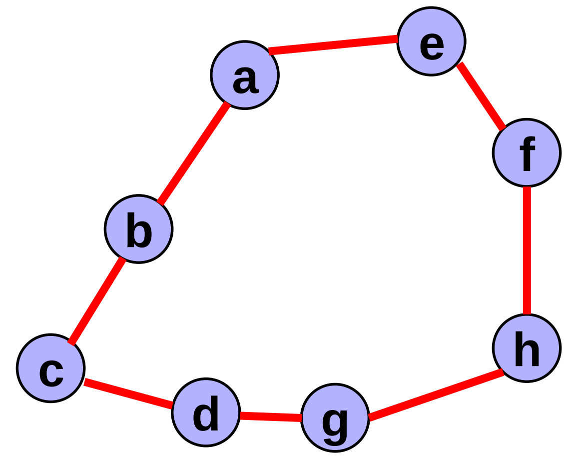
- 近似算法A的基本思想
  - 先构造最小生成树，先序遍历的顺序构造环



构造最小生成树
先序遍历



近似解

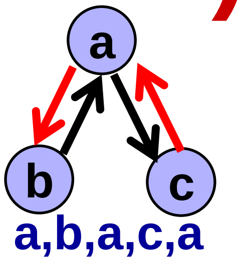


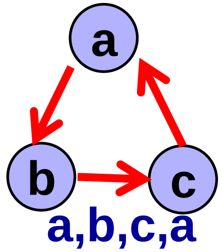
最优解

旅行商问题(Hamilton环)

- 近似算法A性能分析
  - 时间复杂度
    - 构造最小生成树$O(|V|^2)$
    - 先序遍历$O(|V|)$
    - 总时间复杂度$O(|V|^2)$

旅行商问题(Hamilton环)





- 近似算法A性能分析
  - Ratio Bound为2
    - 令OPT为最优解，路径权值和即代价为$C(OPT)$
    - OPT环中删除一条边可得一棵生成树T，代价$C(T)$
    - 构造的最小生成树T_{min}，$C(T_{min}) \leq C(T) \leq C(OPT)$
    - 先序遍历实际上对T_{min}中所有的边走了2次，即代价为$2C(T_{min})$
    - 按先序遍历的节点第一次访问的顺序构造的Hamilton环为A，是先序遍历2次访问满足三角不等式的简化
    - $C(A) \leq 2C(T_{min}) \leq 2C(OPT)$

线性规划

- 给定
  - n个变量
  - m个约束
  - 以及线性目标函数
- 目标
  - 在满足约束条件下求解最优目标函数

线性规划

- **示例**

Objective: min $5x_1+4x_2$

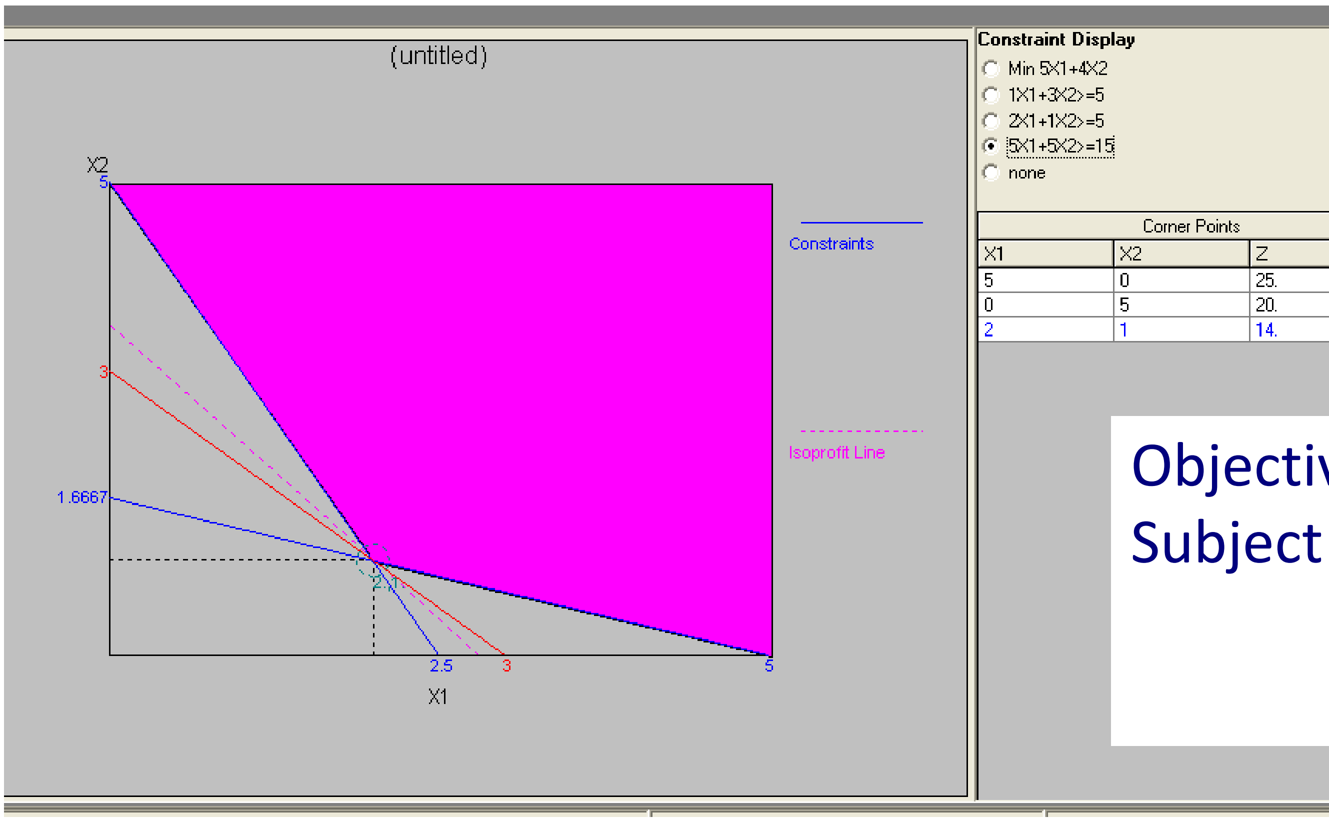
Subject to: $5x_1+5x_2 \geq 15$

and $x_1+3x_2 \geq 5$

and $2x_1+x_2 \geq 5$

线性规划

- 示例



Objective: min $5x_1+4x_2$

Subject to: $5x_1+5x_2 \geq 15$

and $x_1+3x_2 \geq 5$

and $2x_1+x_2 \geq 5$

线性规划

- 求解线性规划方法
  - Simplex algorithm (Dantzig – 1947) – practical, widely used, exponential time in worst case
  - Ellipsoid algorithm (Khachiyan – 1979) – impractical, polynomial time
  - Interior point algorithm (Kharmarkar – 1984) – practical, polynomial time

线性规划

- 一般表示

  - **Min** $\sum_{j=1}^{n} c_j x_j$
  - **满足** $\sum_{j=1}^{n} a_{ij} x_j \geq b_i$ i = 1,…, m

    $x_j \geq 0$ j = 1,…, n

顶点覆盖问题(线性规划)

- 输入：无向图 $G=(V,E)$，顶点v有权值$\omega(v)$
- 输出：$A \subseteq V$，满足
  - $\forall(u,v) \in E$，其中 $u \in A$或者$v \in A$
  - 且$\omega(A)=\sum_{v \in A} \omega(v)$ 最小
- 将问题换另一种描述
  - 假设A为一个覆盖，
    - 若$v \in A$，令$x(v)=1$，
    - 否则$x(v)=0$
  - 要覆盖所有的边，则对任意边$\forall(u,v) \in E$，需满足 $x(u)+x(v) \geq 1$

顶点覆盖问题(线性规划)

- 问题可描述为一个0-1整数规划问题
  - 目标：最小化$\sum_{v\in V}\omega(v)x(v)$
  - 约束： $\forall(u,v)\in E, x(u)+x(v)\geq 1$，且$x(v)=\{0,1\}$

整数规划是NP完全的
线性规划是多项式可解的

可放宽限制条件，设$x(v)$取值可为浮点数，问题转换为线性规划问题
使用线性规划求解，得到线性最优解$x(v)$
若$x(v)\geq\frac{1}{2}$，令$x(v)=1$
否则令$x(v)=0$

顶点覆盖问题(线性规划)

- **近似算法A**

Approx(G, w)

1. $C=0$;
2. **计算线性规划问题的最优解**x;
3. **For each** $v \in V$ **Do**
4. **If** $x(v) \geq 1/2$ **Then** $C=C \cup \{v\}$;

   /*用四舍五入法把线性规划的解近似为0-1规划的解 */

5. **Return** C.

顶点覆盖问题(线性规划)

- 近似算法A分析
  - A的近似比为2
    - $\forall(u,v)\in E, x(u)+x(v)\geq 1$，故$x(u)$和$x(v)$至少有一个大于等于1/2，即 u 或 v 至少有一个在覆盖中
    - 令线性规划最优解代价为 z
    - $z=\sum_{v\in V}\omega(v)x(v)\geq\sum_{v\in V,x(v)\geq 1/2}\omega(v)x(v)$
    - $\geq\sum_{v\in V,x(v)\geq 1/2}\omega(v)\frac{1}{2}$
    - $\geq\sum_{v\in A}\omega(v)\frac{1}{2}\geq\frac{1}{2}\omega(A)$
    - 因为OPT是线性规划可能解, 故$\omega(OPT)\geq z\geq\frac{1}{2}\omega(A)$

集合覆盖问题

- 输入：有限集$U=\{1,2,\ldots,n\}$，子集合的集合$S=\{s_1,s_2,\cdots,s_m\}$，$s_i\subseteq U$, 集合s有权值c_s
- 输出：$A\subseteq S$，满足
  - $U=\bigcup_{s\in A} s$
  - 且$\sum_{s\in A} c_s$最小
- 将问题换另一种描述
  - 为每一个集合S增加一个变量x_s, $\min\sum_{s\in A} c_s x_s$
  - 满足：$\sum_{e\in s} x_s\geq 1, \forall e\in U$
  - 且$x_s=\{0,1\}$

集合覆盖问题

- 输入：有限集$U=\{1,2,\ldots,n\}$，子集合的集合$S=\{s_1,s_2,\cdots,s_m\}$，$s_i\subseteq U$, 集合s有权值c_s
- 输出：$A\subseteq S$，满足
  - $U=\bigcup_{s\in A} s$
  - 且$\sum_{s\in A} c_s$最小
- 将问题换另一种描述
  - 为每一个集合S增加一个变量x_s, $\min\sum_{s\in A} c_s x_s$
  - 满足：$\sum_{e\in s} x_s\geq 1, \forall e\in U$
  - 且$1\geq x_s\geq 0$

集合覆盖问题

- **将问题换另一种描述**
  - 为每一个集合S增加一个变量x_s，
  - $\min \sum_{s\in A} c_s x_s$
  - 满足： $\sum_{e\in s} x_s \geq 1, \forall e \in U$
  - 且$1 \geq x_s \geq 0$

整数规划是NP完全的

线性规划是多项式可解的

可放宽限制条件，设x_s取值可为浮点数，问题转换为线性规划问题

使用线性规划求解，得到线性最优解x_s

令f为元素e在子集合中出现的最大频次

若$x_s \geq \frac{1}{f}$，令$x_s = 1$

否则令$x_s = 0$

顶点覆盖问题(线性规划)

- 近似算法A分析
  - A的近似比为f
    - $\forall e \in U$, 令$s_1, s_2, \ldots s_f$包含了元素e,
    - 则 $x_{s1} + x_{s2} + \cdots + x_{sf} \geq 1$
    - 令线性规划最优解代价为 z
    - $z = \sum_{s \in S} c_s x_s \geq \sum_{s \in S, x_s \geq 1/f} c_s x_s$
    - $\geq \sum_{s \in S, x_s \geq 1/f} c_s \frac{1}{f}$
    - $\geq \sum_{s \in A} c_s \frac{1}{f} \geq \frac{1}{f} c(A)$
    - 因为OPT是线性规划可能解, 故$c(OPT) \geq z \geq \frac{1}{f} c(A)$

子集和问题

- 输入
  - 给定正整数集合$S=\{x_1,x_2,\cdots x_n\}$和一个正整数T
- 输出
  - 最大化$\sum_{x\in A} x$
  - 其中$A\subseteq S$，满足$\sum_{x\in A} x\le T$

$S=\{3,4,5,7,10\}, T=9$

$A=\{4,5\}$，$\sum_{x\in A} x$=9最大不超过T的结果

子集和问题

- 穷举方法
  - 给定正整数集合$S=\{x_1,x_2,\cdots x_n\}$和一个正整数T
  - $P_0=\{0\}$
  - $P_1=\{0,x_1\}$
  - $P_2=\{0,x_1,x_2,x_1+x_2\}$
  - $P_3=\{0,x_1,x_2,x_1+x_2,x_3,x_1+x_3,x_2+x_3,x_1+x_2+x_3\}$
  - ...
  - $P_i=P_{i-1}\cup\{P_{i-1}+x_i\}$ 时间复杂度是指数级的
  - 每步删除大于T的结果
  - 在P_n中找最大的即为最终结果

子集和问题

- 近似算法A
  - 对穷举方法进行修改，减少P_i的长度
  - 穷举法中：$P_i = P_{i-1} \cup \{P_{i-1} + x_i\}$ **长度是指数级的**
  - 给定一误差参数δ，每次将P_i中相差不超过δ的数只用一个表示

```
Trim(P, δ)
m=|P|;
L=P;
last=P[0];
For i=2  To  m  Do
    If  last*(1+δ) < P[i]  Then
        P[i]加入到L末尾;
        last=P[i];
Return L
```

$P = \langle 10, 11, 12, 15, 20, 21, 22, 23, 24, 29 \rangle$

$\delta = 0.1$

$L = \langle 10, 12, 15, 20, 23, 29 \rangle$

子集和问题

- 近似算法A
  - 对穷举方法进行修改，减少P_i的长度
  - 穷举法中：$P_i = P_{i-1} \cup \{P_{i-1} + x_i\}$长度是指数级的
  - 给定一误差参数δ，每次将P_i中相差不超过δ的数只用一个表示

```
Trim(P, δ)
m=|P|;
L=P;
last=P[0];
For i=2  To  m  Do
    If  last*(1+δ) < P[i]  Then
        P[i]加入到L末尾;
        last=P[i];
Return L
```

```
Approx(S, T, ε)
n=|S|;
L_0={0}
For  i=1  To  n  Do
    L_i = L_{i-1} ∪ {L_{i-1}+x_i}
    L_i =Trim(L_i, ε/n)
    从L_i中删除大于T的元素;
Return L_n中最大值.
```

子集和问题

近似算法A分析

- 时间复杂度，与修剪后L_i的长度相关，而L_i的长度是$\frac{1}{\varepsilon}$的多项式
  - 令修剪后的$L_i = \{0, z_1, z_2, \cdots z_k, z_{k+1}\}$，则有$z_{i+1} > z_i(1+\frac{\varepsilon}{n})$
  - 假设L_i中$k+2$个元素，$L_i[0]=0$，$L_i[1]=z_1$，
  - $L_i[2] > z_1(1+\frac{\varepsilon}{n})$，$L_i[3] > z_1(1+\frac{\varepsilon}{n})^2$，$L_i[4] > z_1(1+\frac{\varepsilon}{n})^3$，…，$L_i[k+1] > z_1(1+\frac{\varepsilon}{n})^k$，
  - $\because L_i$中所有元素都小于T，$\therefore z_1(1+\frac{\varepsilon}{n})^k \le T$，
  - $\because z_1$是正整数，$\therefore (1+\frac{\varepsilon}{n})^k \le T$，有 $k \le \log_{1+\frac{\varepsilon}{n}} T = \frac{\ln T}{\ln(1+\frac{\varepsilon}{n})}$
  - $|L_i| = k+2 \le 2 + \frac{\ln T}{\ln(1+\frac{\varepsilon}{n})}$，$\because \lim_{x\to 0} \frac{\ln(1+x)}{x} = 1$，
  - $\therefore |L_i| \le 2 + \frac{n \ln T}{\varepsilon}$

子集和问题

近似算法A分析

- **近似比为 $1+2\varepsilon$**
  - 令y属于穷举法(修剪前)的P_i， z属于修剪后的L_i
  - 对于P_i中任意y，在L_i中存在z，使得$\frac{y}{(1+\frac{\varepsilon}{n})^i} \le z \le y$
    - $i=0$时， $P_i=\{0\}$， $L_i=\{0\}$，显然成立
    - 假设$i=k$时有$\frac{y}{(1+\frac{\varepsilon}{n})^k} \le z \le y$
    - 现证当$i=k+1$时也成立。 $P_{k+1}=P_k \cup \{P_k+x_{k+1}\}$，
    - $\frac{y}{(1+\frac{\varepsilon}{n})^k}+x_{k+1} \le z+x_{k+1} \le y+x_{k+1}$
    - $\because \frac{y+x_{k+1}}{(1+\frac{\varepsilon}{n})^{k+1}} < \frac{y}{(1+\frac{\varepsilon}{n})^k}+x_{k+1}$，
    - $\therefore \frac{y+x_{k+1}}{(1+\frac{\varepsilon}{n})^{k+1}} \le z+x_{k+1} \le y+x_{k+1}$，
  - P_n中最优解y^*满足$\frac{y^*}{(1+\frac{\varepsilon}{n})^n} \le z \le y^*$， L_n最优解z^*有$\frac{y^*}{z^*} \le \frac{y^*}{z} \le \left(1+\frac{\varepsilon}{n}\right)^n \le e^{\varepsilon} \le 1+2\varepsilon$，当$\varepsilon<1$

$$\frac{y}{\left(1+\frac{\varepsilon}{n}\right)^k}\left(1-\frac{1}{1+\frac{\varepsilon}{n}}\right)+x_{k+1}\left(1-\frac{1}{\left(1+\frac{\varepsilon}{n}\right)^{k+1}}\right)>0$$

$$\lim_{n\to\infty}\left(1+\frac{1}{n}\right)^n=e$$

- **考虑这样一个应用问题，市场上销售的某型钢材的长度为1，现需要一些长度分别为$a_1, a_2, \dots, a_N$的小钢条（$a_i \le 1, 1 \le i \le N$），问至少要买多少根钢条进行切割？请给出一个多项式时间算法，使得买的钢条尽可能少，并分析该算法的近似比。**

- **给定*n*个任务以及两机器，每个任务所需的处理时间为$p_1, p_2, \dots, p_n$。一个任务可由任一台机器执行，但不能拆开分配给多台机器执行。一台机器在一时间只能处理一个任务。求两台机器的最短最长执行时间。有一算法，它将这*n*个任务依次分配给这两机器。算法在分配一任务时，总是将任务分配给到目前为止分配了最少工作时间的机器。请给出该算法的近似比。**