《近似算法》

*

December 5, 2022

近似算法PPT的章节简单总结以及部分经典例题的详细推导。

1 近似算法slides总结

1.1 章节总结

- 可计算理论 page1 − 6
- 问题形式化 page7 − 10
- P和NP问题 page11 − 20
- 规约规则和NP完全问题 page21 − 40
- 近似算法预备知识 page41 − 48
- 绝对近似算法 page49 − 56
- 相对近似算法 page84 − 91

1.2 经典证明题

- Graph Coloring顶点着色问题 page50
- Knapsack背包问题 page54
- Scheduling 多机调度问题 page62
- Enclidean Travelling Salesman Problem 旅行商问题 page77

2 Graph Coloring顶点着色问题

使用最少的颜色数来为图G的顶点上色，使得所有相邻的顶点均有不同的颜色，即使G是平面图，该问题的判定版本也是NP-hard的，它有1个绝对近似算法。具体判定问题：一个平面图是否可以满足n着色。

2.1 n=1

图G只有一个顶点可以满足1着色。

判定方式：$|V| = 1$ 时间复杂度为常数→P问题。

2.2 n=2

该问题是经典二分图判定问题（顶点集V可分割为两个互不相交的子集，并且图中每条边依附着的两个顶点分属于这两个互不相交的子集）

判定方式：假定一个节点数为n，边为m的图。从某节点开始染色，然后dfs函数中遍历与当前节点相邻的所有节点，若相邻节点没被染色则继续dfs，否则判断当前节点和相邻节点的颜色是否冲突时间复杂度为O（m+n）→ P问题。

2.3 n=3

判定一个平面图是否可3着色的问题是NPC的，所以需要近似算法。具体近似算法A（G）如下：

1. 检验G是否可以2着色（即是不是二分图），若是则G可2着色。
2. 否则多项式时间计算5着色（因为四色定理已证明任何G都是可以4着色的,少数可以3着色）。

该问题的OPT(G)=2时，A(G)=2，$|A(G) - OPT(G)| = 0$; OPT(G) = 3 or 4，A(G)= 5，$|A(G) - OPT(G)| = 2$, 所以 $|A(G) - OPT(G)| \leq 2$.

总结：图的3着色问题是NPH问题，以上算法A是解决该问题的绝对近似算法，绝对性能度量满足$\forall G \in D, |A(G) - OPT(G)| \leq 2$.

3 Knapsack背包问题

该问题不存在绝对近似算法，可以使用扩方法反证明，证明如下：

确定一个k，满足 $\forall I \in D, |A(I) - OPT(I)| \leq k$. 对于某个$I$, 可以构造新的实例$I'$使得每个物品的利润扩放k+1倍，其余参数不变。故$I$的可行解也是$I'$的可行解，反之亦然，只是解的值相差k+1倍。算法A可以解决I'得到$A(I')$, 设A在实例I上的解是σ:

$$|A(I') - OPT(I')| \leq k \Longrightarrow |(k+1)f(\sigma) - (k+1)OPT(I)| \leq k$$

$$\Longrightarrow |f(\sigma) - OPT(I)| \leq k/(k+1) \Longrightarrow |f(\sigma) - OPT(I)| = 0 \Longrightarrow f(\sigma) = OPT(I)$$

请注意，$f(\sigma)$和$OPT(I)$都是整数，而k/(k+1)是0-1的浮点数，所以小于k/(k+1)的非负整数只有0.

总结：背包问题是NPH问题，存在绝对近似算法A解决实例I的前提是$f(\sigma) = OPT(I)$,即多项式时间内找到了最优解OPT(I),多项式时间找到了NPH的解，又因为所有NP问题可以规约到NPH问题，所以多项式时间可以找到NP问题的解，即$P = NP$.

4 Scheduling 多机调度问题

考虑简单的多机调度问题：输入n个作业J1，J2，···，Jn，相应的运行时间为P1，P2，···，Pn，设每个Pi是有理数。将n个作业分配到m台同样的机器上，以使得完成时间最短。

完成时间定义为：所有机器上作业运行总时间最长的那一台机器的运行时间。

可行解集合：n个作业被划分为m个子集，一个解的值是所有子集中总运行时间最长的子集的运行时间。该问题即使在m=2时也是NP-hard的。

4.1 近似算法1：List调度

List调度算法(Graham)：将n个作业依次以online的方式分配到m台机器中的某一台上，规则是将当前作业分配到当时负载最小的机器上，而机器负载是分配给它的所有作业的总的运行时间。近似比为：$\frac{A(I)}{OPT(I)} \leq 2 - \frac{1}{m}$. 证明略。

4.2 近似算法2：LPT调度

LPT(Longest Processing Time)：将作业按其运行时间递减序排序, 然后用List策略调度。近似比为：$\frac{4}{3}-\frac{1}{3m}$. 证明如下：

1. 当m=1时，$\because A(I)=OPT(I) \therefore A(I)/OPT(I)=4/3-1/3$。

2. 当m¿1时，**先证明近似比上界**：反证法，假设该定理不成立。

   第一步：首先假设一个反例，可设违反该定理具有最少作业数的实例I是J1,J2,⋯,Jn, 不妨设$P1 \geq P2 \geq \cdots \geq Pn$。LPT的调度次序是J1,J2,⋯,Jn,其完成时间是A(I)。*设其中最迟完成的作业是Jk，则k=n*，证明如下：

   若k<n，算法A运行作业J1,J2,⋯,Jk(记为实例$I' \in I$)的完成时间仍是A(I)，即$A(I)=A(I')$，而对最优解显然$OPT(I') \leq OPT(I)$，故有：$\frac{A(I')}{OPT(I')} \geq \frac{A(I)}{OPT(I)} > \frac{4}{3}-\frac{1}{3m}$. 说明I'同样违反定理且作业数比I少，产生矛盾。

   第二步：证明上述实例I的最优调度，*在任何机器上分配的作业数不超过2，因此$n \leq 2m$*。证明如下：

   $\because$ Jn是LPT调度A中最迟完成的作业 $\therefore$ 在A中它开始于时刻A(I)-Pn，且此刻其它机器均无空余时间,即：

   $$A(I)-P_n \leq \frac{1}{m}\sum_{i=1}^{n-1}P_i \quad \Rightarrow \quad A(I) \leq \frac{1}{m}\sum_{i=1}^{n}P_i+\frac{m-1}{m}P_n$$

   另一方面，因为 $OPT(I) \geq \frac{1}{m}\sum_{i=1}^{n}P_i$ 可得：

   $$\frac{4}{3}-\frac{1}{3m} < \frac{A(I)}{OPT(I)} \leq \frac{OPT(I)+\frac{m-1}{m}P_n}{OPT(I)} = 1+\frac{m-1}{m}\frac{P_n}{OPT(I)}$$

   $$4m-1 < 3m+3(m-1)\frac{P_n}{OPT(I)} \Rightarrow OPT(I) < 3P_n$$

   $\because$ Pn是实例I中时间最短的作业，比$3P_n$小的值一定不会超过两个作业数的总时间。$\therefore$ 实例I的最优调度中任何机器上的作业数≤ 2.

   第三步：当最优调度在任何机器上*至多包含2个作业时，LPT也是最优的*。证明如下：

   不妨设n=2m，若$n < 2m$，则令Jn+1,⋯,J2m的时间均为0，将其加入I，不妨设$P1 \geq P2 \geq \cdots \geq Pn$. 设最优调度使得每台机器恰有2个作业：Ji和Jj，*则必有$i \leq m$,$j > m$*。证明[否则若某最优调度O有$i,j \leq m$，则定有某台机器上有Js和Jt，使得$s,t > m$. 因为Pi,Pj比Ps,Pt大，所以交换Pj和Pt后，无论Pi+Pt还是Ps+Pj都比Pi+Pj小，交换后的调度O'的最迟完成时间只能减少，所以O'也是最优调度。]

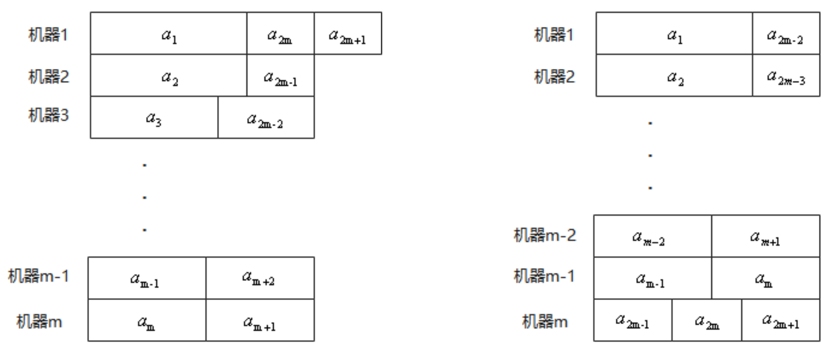
   必有最优调度使J1,⋯,Jm分别分配到M1,⋯,Mm上，当将Jm+1,⋯,J2m分配到M台机器上时，LPT是将长时间的作业分配到轻负载上，*必与该最优调度结果相同即$A(I)=OPT(I)$*。证明[假设机器 M_i 上运行的两个任务为 J_{i1} 和 J_{i2}, 不妨设 $i1 \leq m, i2 > m$ 。按照轻负载的分配原则,

$J_{m+1}, \ldots, J_{2m}$ 将会依次被分配到执行 $J_m, \ldots, J_1$ 的机器上。最终的分配结果满足: 对于任意两台机器 M_i 和 M_j, 若 $P_{i1} \geq P_{j1}$, 则有 $P_{i2} \leq P_{j2}$ 。假设存在一种调度策略 A', 它不同于 LPT, 使得 $OPT(I) = A'(I)$ 。则该策略中, 至少存在一对机器 M_i 和 M_j, 若 $P_{i1} \geq P_{j1}$, 则有 $P_{i2} \geq P_{j2}$ 。因为: $\max\{P_{i1}+P_{i2}, P_{j1}+P_{j2}\} \geq \max\{P_{i1}+P_{j2}, P_{j1}+P_{i2}\}$ 该不等式右侧即为 LPT 算法的调度策略。则此时的调度算法所用的时间 $OPT(I) = A'(I) \geq A(I)$ 。由 OPT 的定义可知, $OPT(I) \leq A(I)$, 所以 $A(I) = OPT(I)$ 。]

所以$\frac{A(I)}{OPT(I)} = 1 \leq \frac{4}{3} - \frac{1}{3m}(m \geq 2)$，产生矛盾。

综上所述，近似比上界证明完毕。

再证明近似比的紧确界：找出一个具体的实例来说明近似比是紧致的。



(a) 近似算法A^*　　(b) 最优算法OPT

Figure 1: 两个优化算法的图解

现在考虑一下这样一个输入实例 I^*, 设算法 A 表示 LPT 调度算法。(1) 设作业个数 n 和 m 台机器之间有如下关系: $\mathrm{n} = 2\,\mathrm{m} + 1$ 。

(2) 再设这 n 个作业的运行时间集合为 $A = \{\mathrm{a}_1, \mathrm{a}_2, \ldots, \mathrm{a}_{2\,\mathrm{m}+1}\}$, 给他们一组赋值: $A = \{\mathrm{a}_1, \mathrm{a}_2, \ldots, \mathrm{a}_{2\,\mathrm{m}+1}\} = \{2m-1, 2m-1, 2m-2, 2m-2, \ldots, m+1, m+1, m, m, m\}$

(3) 现在考虑使用 LPT 算法的运行情况, 用甘特图Fig1.a来表示:

很容易可以得到, $A(I^*) = a_1 + a_{2m} + a_{2m+1} = 2m - 1 + m + m = 4m - 1$

(4) 对于最优情况下我们可以知道应该是满足甘特图Fig1.b:

很容易可以得到, $OPT(I^*) = 3m$ 。综上所述, 此时 $\frac{A(I^*)}{OPT(I^*)} = \frac{4m-1}{3m} = \frac{4}{3} - \frac{1}{3m}$, 即 LPT 算法的近似性能比 $R_{LPT} = \frac{4}{3} - \frac{1}{3m}$ 得证。

5 Enclidean Travelling Salesman Problem 旅行商问题

设G是具有n个城市的地图，旅行商从其中某一城市出发周游，遍历每个城

市1次恰好1次后回到出发地，求其最短的周游路线。亦即，求一条最短的哈密顿回路(Hamiltonian Cycle，简称HC)。因为边的长度可以为无限大，因此不妨设G为**边加权的无向完全图**。下面只考虑TSP的特殊版本：满足三角不等式的ΔTSP（Metric TSP）问题,$d(i,k) \le d(i,j)+d(j,k)$ 即**去掉途中1个中转站不会使路径长度增加**。该问题是NPH问题。

5.1 近似算法1：近邻法

从任一顶点开始，在每一步从当前顶点出发去访问一个最近的尚未访问过的顶点，由此构造哈密尔顿圈。但这种方法性能差。设ΔTSP的顶点数为n，则近似比为：$R_{NN}^{\infty} = \Theta(lgn)$.

近似比是n的函数，当n无穷大时，近似比也无穷大，因此性能很差。

5.2 近似算法2：MST最小生成树启发

基于MST启发的ΔTSP的近似算法：

1. 求G的任一MST T
2. 重复T的边构造一欧拉环ET
3. 从ET产生一哈密尔顿圈

近似比为：$\frac{A_{MST}(G)}{OPT(G)} \le 2$. 请注意，连通的无向图G是欧拉环（存在欧拉回路）的充要条件是：G中每个顶点的度都是偶数。证明略。

5.3 近似算法3：CH启发(Christofides)

优化近似算法第二步：避免从MST T到欧拉环的双边（**不需要重复T所有的边**），*为T的每对奇度顶点加一条边使所有顶之度数为偶数*。因为T中所有顶点度数和为2—E—，所以**奇度顶点总数为偶数**。那么如何完成加边的操作？

最小权匹配问题(minimum weight perfect matching): V(G)表示完全图G的所有点，通过G生成的MST中，度数为奇数的点集合为O，则O是V(G)的子集，且$|O|$为偶数，该子集的各点之间都有边(因为G是完全图)，这些边为集合E(G)的子集E'，那么从E'中选取$|O|/2$条边(即集合E'的子集M)将集合S的所有点两两相连，并且保证总权重最小，子集M即最小权匹配集。已证明，可在多项式时间内找到S的一个最小权匹配(**P问题**)。

改变第二步的操作后，开始证明CH启发的近似比:

第一步：设M是T中奇度点集O上的最小权匹配集,断言$d(M) \le \frac{OPT(G)}{2}$,证明如下：

给定一个最优解（最短的哈密尔顿回路Hamilton Loop），然后对其做短路操作，将不在O中的顶点删去，剩下的点相连所得的圈X(O上的圈)必满足：$d(X) \le OPT(G)$ (由三不等式决定，去掉途中的1个中转站不会使路径长度增加)。

点集O上的圈必是O的两个匹配集的并。$\because$ T中奇度顶点是偶数个, $\therefore |O| = m$ 为偶数。O上圈有m条边，但是O上的一个匹配集只有m/2条边，故该圈是2个匹配集之并。这两个匹配集之一的权必定至多是圈X的一半，而M是O的最小权匹配集，故有：$d(M) \le d(X)/2 \le OPT(G)/2$, 证明完毕。

第二步：$\because$ 图$T \cup M$中所有顶点度为偶数 $\therefore$ 可从其构造欧拉环ET

$\because d(T) \leq OPT(G) and d(M) \leq OPT(G)/2$

$\therefore d(ET) = d(T \cup M) \leq 1.5OPT(G)$

用短路法从ET构造HC不会使权增加，故有：$A_{CH}(G) \leq d(ET) \leq 1.5OPT(G)$，即：$\frac{A_{CH}(G)}{OPT(G)} \leq 1.5$

总结：近似比1.5是目前最好的结果，但是要找一个最小权值匹配需要$O(n_3)$的时间，而MST启发的运行时间几乎是线性的(求MST除外)。