HIT
CS&E

第二章

算法分析的数学基础

王宏志
计算机科学与工程系

DKE-LAB(2009)

提要

2.1 计算复杂性函数的阶
2.2 标准符号和通用函数
2.3 和式的估计与界限
2.4 递归方程

2.1 计算复杂性函数的阶

2.1.1 同阶函数集合
2.1.2 低阶函数集合
2.1.3 高阶函数集合
2.1.4 严格低阶函数集合
2.1.5 严格高阶函数集合
2.1.6 函数阶的性质

增长的阶

- 如何描述算法的效率?
  - 增长率
- 忽略低阶项，保留最高阶项
- 忽略常系数
- 利用$\Theta(n^2)$表示插入排序的最坏运行时间
- $\Theta(1)$, $\Theta(\lg n)$, $\Theta(\sqrt{n})$,$\Theta(n)$, $\Theta(n\lg n)$, $\Theta(n^2)$, $\Theta(n^3)$, $\Theta(2^n)$, $\Theta(n!)$

HIT
CS&E

增长函数

- 渐进效率:
  - 输入规模非常大
  - 忽略低阶项和常系数
  - 只考虑最高阶(增长的阶)
- 典型的增长阶:
  - $\Theta(1)$, $\Theta(\lg n)$, $\Theta(\sqrt{n})$,$\Theta(n)$, $\Theta(n\lg n)$, $\Theta(n^2)$, $\Theta(n^3)$, $\Theta(2^n)$, $\Theta(n!)$
- 增长的记号: O, Θ, Ω, o, ω.

2.1.1 同阶函数集合

定义2.1.1 (同阶函数集合) $\theta(f(n))=\{g(n) \mid \exists c_1, c_2>0, n_0, \forall n>n_0, c_1 f(n) \le g(n) \le c_2 f(n)\}$ 称为与$f(n)$同阶的函数集合。

*如果 $g(n) \in \theta(f(n))$，我们称个 $g(n)$ 与 $f(n)$ 同阶。

*$g(n) \in \theta(f(n))$ 常记作 $g(n)=\theta(f(n))$。

*$f(n)$必须是极限非负的，即当 n 充分大以后，$f(n)$是非负的，否则$\theta(f(n))$=空集。

Example

例1 证明 $6n^3 \neq \theta(n^2)$

证. 如果存在c_1、$c_2>0$, n_0使得当$n \geq n_0$

时，$c_1 n^2 \leq 6n^3 \leq c_2 n^2$。于是，当$n>c_2/6$时，

$n \leq c_2/6$，矛盾。

例 4 因任何常数 $c=\theta(n^0)=\theta(1)$，$c_1 \le c \le c_2$，如果令

$$c_1=\frac{1}{2}c, c_2=\frac{3}{2}c, n_0=1$$

。

2.1.2 低阶函数集合

定义 2.1.2(低阶函数集合). $O(f(n))=\{g(n)\mid\exists c>0, n_0$，当 $n\geq n_0$，$0\leq g(n)\leq cf(n)\}$称为比 $f(n)$低阶的函数集合。

*如果 $g(n)\in O(f(n))$，称 $f(n)$是 $g(n)$的上界。

*$g(n)\in O(f(n))$常记作 $g(n)=O(f(n))$。

Example

例 1 $\theta(g(n)) = f(n) \Rightarrow f(n) = O(g(n))$

$$\theta(g(n)) \subseteq O(g(n))$$

例 2 证明 $n=O(n^2)$.

证. 令 $c=1$, $n_0=1$, 则当 $n \geq n_0$ 时, $0 \leq n \leq cn^2$。

2.1.3 高阶函数集合

定义 2.1.3(高阶函数集合). $\Omega(f(n))=\{g(n) \mid \exists c>0, n_0$，当 $n \geq n_0$，$0 \leq cf(n) \leq g(n)\}$称为比 $f(n)$高阶的函数集合。

*如果 $g(n) \in \Omega(f(n))$，称 $f(n)$是 $g(n)$的下界。

*$g(n) \in \Omega(f(n))$常记作 $g(n)=\Omega(f(n))$。

定理 2.1.对于任意 f(n)和 g(n)，$f(n)=\theta(g(n))$ iff $f(n)=O(g(n))$ 而且 $f(n)=\Omega(g(n))$.

证. $\Rightarrow$ 如果 $f(n)=\theta(g(n))$，则 $\exists c_1,c_2>0, n_0>0$，当 $n \ge n_0$ 时，

$$c_1 g(n) \le f(n) \le c_2 g(n).$$

显然 $f(n)=\Omega(g(n))$ and $f(n)=O(g(n))$.

$\Leftarrow$ 如果 $f(n)=O(g(n))$ 且 $f(n)=\Omega(g(n))$，则由 $f(n)=O(g(n))$ 可知，

如果$f(n)=O(n^k)$，则称$f(n)$是多项式界限的。

$f(n)=\Omega(g(n))$ 可知，$\exists c_2, n_2 \ge 0$，使得当 $n \ge n_1$，$f(n) \le c_2 g(n)$

令 $n_0 \ge \max\{n_1, n_2\}$，则当 $n \ge n_0$，$c_2 f(n) \le f(n) \le c_1 g(n)$。

关于Ω 标记

- 用来描述运行时间的最好情况
- 对所有输入都正确
- 比如，对于插入排序
  - 最好运行时间是 $\Omega(n)$,
  - 或者说，运行时间是$\Omega(n)$.
  - 最坏运行时间是 $\Omega(n^2)$
  - 但是说运行时间是$\Omega(n^2)$则有误
- 可以用来描述问题
  - 排序问题的时间复杂性是$\Omega(n)$

2.1.4 严格低阶函数集合

定义 2.1.4(严格低阶函数集合). $o(g(n)) = \{f(n) | \forall c > 0, \exists n_0 > 0, 0 \le f(n) < cg(n)$ for all $n \ge n_0\}$称为严格比 $g(n)$低阶的函数集合。

*如果 $f(n) \in o(g(n))$,称 $g(n)$是 $f(n)$的严格上界。

* $f(n) \in o(g(n))$常记作 $f(n) = o(g(n))$。

例 1. 证明 $2n = o(n^2)$

证. 对 $\forall c>0$，欲 $2n < cn^2$，必 $2 < cn$，即 $\frac{2}{c} < n$。所以，当 $n_0 = \frac{2}{c}$ 时，$2n < cn^2$ 对 $\forall c > 0$，$n \geq n_0$。

例 2. 证明 $2n^2 \neq o(n^2)$

证. 当 $c=1>0$ 时，对于任何 n_0，当 $n \geq n_0$，$2n^2 < cn^2$ 都不成立

关于o标记

- O标记可能是或不是紧的
  - $2n^2 = O(n^2)$ 是紧的, 但$2n = O(n^2)$不是紧的.
- o标记用于标记上界但不是紧的情况
  - $2n = o(n^2)$, 但是$2n^2 \neq o(n^2)$.
- 区别: 某个正常数c在O标记中， 但所有正常数c在o标记中.

$$f(n) = o(g(n)) \Rightarrow \lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$$

证. 由于 $f(n)=o(g(n))$，对任意$\varepsilon>0$, 存在n_0， 当$n \geq n_0$时,

$$0 \leq f(n) < \varepsilon g(n),$$

即$0 \leq \frac{f(n)}{g(n)} \leq \varepsilon$. 于是, $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$.

2.1.5 严格高阶函数集合

定义 2.1.4(严格高阶函数集合). $w(g(n))=\{f(n)\mid\forall c>0,\exists n_0>0$, $0\le cg(n)<f(n)$ for all $n\ge n_0\}$称为严格比 $g(n)$高阶的函数集合。

命题 2.2. $f(n)\in w(g(n))$ iff $g(n)\in o(f(n))$.

证:

$\Rightarrow$ 对$\forall c>0$, $1/c>0$. 由$f(n)\in w(g(n))$知, 对 $1/c>0$, $\exists n_0$, 当$n\ge n_0$时, $(1/c)g(n)<f(n)$, 即 $g(n)<cf(n)$. 于是, $g(n)\in o(f(n))$.

$\Leftarrow$ 对于任意 $c>0$, $1/c>0$. 由$g(n)\in o(f(n))$可知, $\exists n_0\ge 0$, 当$n>n_0$时, $g(n)<(1/c)f(n)$, 即 $cg(n)<f(n)$. 于是, $f(n)\in w(g(n))$。

Example

例 1. 证明 $n^2/2=w(n)$.

证. 对$\forall c>0$, $cn<n^2/2$, 只需 $n>2c$. 令 $n_0=2c+1$, 则当 $n\geq n_0$, $cn<n^2/2$.

例 2. 证明 $n^2/2 \neq w(n^2)$

证. 若 $n^2/2=w(n^2)$, 则对于 $c=1/2$, 存在 n_0, 当 $n\geq n_0$时, $cn^2<n^2/2$, 即 $c<1/2$, 矛盾.

命题 2.3. $f(n)=w(g(n)) \Rightarrow \lim\limits_{n\to\infty}\dfrac{f(n)}{g(n)}=\infty$

证：对 $\forall c>0$，由于 $f(n)=w(g(n))$，必存在 n_0，使得当 $n \geq n_0$ 时，$f(n)>cg(n)$，即当 $n \geq n_0$ 时，$f(n)/g(n)>c$. 于是，$\lim\limits_{n\to\infty}\dfrac{f(n)}{g(n)}=\infty$.

2.1.6 函数阶的性质

A 传递性：

（a） $f(n)=\theta(g(n)) \wedge g(n)=\theta(h(n)) \Rightarrow f(n)=\theta(h(n))$

（b） $f(n)=O(g(n)) \wedge g(n)=O(h(n)) \Rightarrow f(n)=O(h(n))$

（c） $f(n)=\Omega(g(n)) \wedge g(n)=\Omega(h(n)) \Rightarrow f(n)=\Omega(h(n))$

（d） $f(n)=o(g(n)) \wedge g(n)=o(h(n)) \Rightarrow f(n)=o(h(n))$

（e） $f(n)=w(g(n)) \wedge g(n)=w(h(n)) \Rightarrow f(n)=w(h(n))$.

2.1.6 函数阶的性质(续)

B 自反性:

(a) $f(n)=\theta(f(n))$,

(b) $f(n)=O(f(n))$,

(c) $f(n)=\Omega(f(n))$.

C 对称性

$$f(n)=\theta(g(n))\ iff\ \ g(n)=\theta(f(n)).$$

D 反对称性:

$$f(n)=O(g(n))\quad iff\quad g(n)=\Omega(f(n))$$

$$f(n)=o(g(n))\quad iff\quad g(n)=w(f(n))$$

注意

!

*并非所有函数都是可比的，即对于函数 $f(n)$ 和 $g(n)$，可能 $f(n) \neq O(g(n)), f(n) \neq \Omega(g(n))$. 例如，n 和 $n^{1+\sin n}$.

2.2 标准符号和通用函数

- Flour 和 ceiling

2.2.1 Flour和ceiling

定义2.2.1(Flours和ceiling). $\lfloor x \rfloor$表示小于或等于x的最大整数. $\lceil x \rceil$表示大于等于x的最小整数.

命题 2.2.1 $x-1<\lfloor x \rfloor \le x \le \lceil x \rceil < x+1$

命题 2.2.2 对于任意函数 n, $\lceil n/2 \rceil + \lfloor n/2 \rfloor = n$

证. 若 $n = 2k$,则 $\left\lceil \frac{n}{2} \right\rceil = k$, $\lfloor n/2 \rfloor = k$. 于是 $\left\lceil \frac{n}{2} \right\rceil + \lfloor n/2 \rfloor = 2k = n$

若 n=2k+1, 则 $\left\lceil \frac{n}{2} \right\rceil + \lfloor n/2 \rfloor = k + 1 + k = 2k + 1 = n$.

命题 2.2.3 设 *n*、*a*、*b* 是任意整数， $a \neq 0, b \neq 0$， 则

(1) $\lceil \lceil n/a \rceil / b \rceil = \lceil n/ab \rceil$。

(2) $\lfloor \lfloor n/a \rfloor / b \rfloor = \lfloor n/ab \rfloor$

证. (1) 若 $n = kab$,则 $\left\lceil \frac{n/a}{b} \right\rceil = \left\lceil \frac{kb}{b} \right\rceil = k = \left\lceil \frac{kab}{ab} \right\rceil = \left\lceil \frac{n}{ab} \right\rceil$。

若 $n = kab + \alpha$， 0< $\alpha < ab$， 则

$$\left\lceil \left\lceil \frac{n}{a} \right\rceil \Big/ b \right\rceil = \left\lceil \frac{kb+1}{b} \right\rceil = k + 1 = \left\lceil \frac{n}{ab} \right\rceil = \left\lceil \frac{kab+\alpha}{ab} \right\rceil = k+1$$

(2) 类似于(1)的证法。

为什么需要和式的估计与界限

FOR $l=2$ **TO** n

  FOR $i=1$ **TO** $n-l+1$ **DO**

    $j=i+l-1;$

    $m[i, j]=\infty;$

    FOR $k \leftarrow i$ **To** $j-1$ **DO**

      $q=m[i, k]+m[k+1, j]+p_{i-1}p_kp_j$

      IF $q<m[i, j]$ **THEN** $m[i,j]=q;$

2.3 和式的估计与界限

1. 线性和

命题 2.4.5 $$\sum_{k=1}^{n}(ca_k+b_k)=c\sum_{k=1}^{n}a_k+\sum_{k=1}^{n}b_k$$

命题 2.4.6 $$\sum_{k=1}^{n}\theta(f(k))=\theta\left(\sum_{k=1}^{n}f(k)\right)$$

2. 级数

命题 2.4.7 $$\sum_{i=1}^{n} i = \frac{n(n+1)}{2} = \theta(n^2)$$

命题 2.4.8 $$\sum_{k=0}^{n} x^k = 1 + x + x^2 + \ldots\ldots + x^n = \frac{x^{n+1} - 1}{x - 1} \qquad (x \neq 1)$$

$$\sum_{k=0}^{\infty} x^k = \frac{1}{1-x} \qquad |x| < 1$$

命题 2.4.9 $$H_n = \sum_{k=1}^{n} \frac{1}{k} = \ln \mathbf{n} + O(1)$$

命题 2.4.10 $$\sum_{k=1}^{n}(a_k - a_{k-1}) = a_n - a_0 .$$

命题 2.4.11 $$\sum_{k=0}^{n-1}(a_k - a_{k+1}) = a_0 - a_n$$

命题 2.4.12 $$\sum_{k=1}^{n-1}\frac{1}{k(k+1)} = \sum_{k=1}^{n-1}\left(\frac{1}{k} - \frac{1}{k+1}\right) = 1 - \frac{1}{n}$$

命题 2.4.13 $$\lg(\prod_{k=1}^{n} a_k) = \sum_{k=1}^{n}\lg a_k$$

3. 和的界限

例 1. 证明 $\sum_{k=0}^{n} 3^k = O(3^n)$

证 证明对于 c≥ 3/2，存在一个 n_0，当 $n \ge n_0$ 时 $\sum_{k=0}^{n} 3^k \le c3^n$.

当 $n=0$ 时，$\sum_{k=0}^{n} 3^k = 1 \le c = c3^n$.

设 $n \le m$ 时成立. 令 $n = m+1$，则

$$\sum_{k=0}^{m+1} 3^k = \sum_{k=0}^{m} 3^k + 3^{m+1} \le c3^m + 3^{m+1} = c3^{m+1}\left(\frac{1}{3}+\frac{1}{c}\right) \le c3^{m+1} .$$

例 2. 错误证明：$\sum_{k=1}^{n} k = \theta(n)$

证. $n=1$时，$\sum_{k=1}^{1} k = 1 = O(1)$，$\sum_{k=1}^{n+1} k = \sum_{k=1}^{n} k + (n+1) = O(n) + (n+1) = O(n)$.

*错在 *O*(*n*)的常数 *c* 随 *n* 的增长而变化，不是常数。

*要证明 $\sum_{k=1}^{n} k = O(n)$ 需要证明：对某个 *c*>0，$\sum_{k=1}^{n} k = c(n)$.

直接求和的界限

例1. $\sum_{k=1}^{n} k \le \sum_{k=1}^{n} n = n^2$

例 2. $\sum_{k=1}^{n} a_i \le n \times \max\{a_k\}$.

例 3. 设对于所有 $k \ge 0$, $a_{k+1}/a_k \le r < 1$， 求 $\sum_{k=0}^{n} a_k$ 的上界.

解： $a_1/a_0 \le r \Rightarrow a_1 \le a_0 r$,

$a_2/a_1 \le r \Rightarrow a_2 \le a_1 r \le a_0 r^2$,

$a_3/a_2 \le r \Rightarrow a_3 \le a_2 r \le a_0 r^3$ ……

$a_k/a_{k-1} \le r \Rightarrow a_k \le a_{k-1} r \le a_0 r^k$

于是， $\sum_{k=0}^{n} a_k \le \sum_{k=0}^{\infty} a_0 r^k = a_0 \sum_{k=0}^{\infty} r^k = \frac{a_0}{1-r}$.

例 4．求 $\sum_{k=1}^{\infty}\left(k/3^{k}\right)$ 的界

解．使用例 3 的方法． $\frac{k+1}{3^{k+1}}\bigg/\frac{k}{3^{k}}=\frac{1}{3}\cdot\frac{k+1}{k}=\frac{1}{3}\left(1+\frac{1}{k}\right)\le\frac{2}{3}=r$． 于是

$$\sum_{k=1}^{\infty}\frac{k}{3^{k}}\le\sum_{k=1}^{\infty}a_1r^{k}=\sum_{k=1}^{\infty}\frac{1}{3}\left(\frac{2}{3}\right)^{k}=\frac{1}{3}\cdot\frac{1}{1-\frac{2}{3}}=1.$$

例 5．用分裂和的方法求 $\sum_{k=1}^{n}k$ 的下界．

解：

$$\sum_{k=1}^{n}k=\sum_{k=1}^{n/2}k+\sum_{k=n/2+1}^{n}k\ge\sum_{k=1}^{n/2}0+\sum_{k=n/2+1}^{n}n/2\ge\left(\frac{n}{2}\right)^{2}=\Omega\left(n^{2}\right).$$

例 6. 求 $\sum_{k=0}^{\infty}\frac{k^2}{2^k}$ 的上界

解：当 $k \geq 3$ 时，$\frac{(k+1)^2 / 2^{k+1}}{k^2 / 2^k} = \frac{(k+1)^2}{2k^2} \leq \frac{8}{9}$

于是 $\sum_{k=0}^{\infty}\frac{k^2}{2^k} = \sum_{k=0}^{2}\frac{k^2}{2^k} + \sum_{k=3}^{\infty}\frac{k^2}{2^k} \leq \theta(1) + \sum_{k=3}^{\infty}\frac{9}{8}\cdot\left(\frac{8}{9}\right)^k = O(1)$.

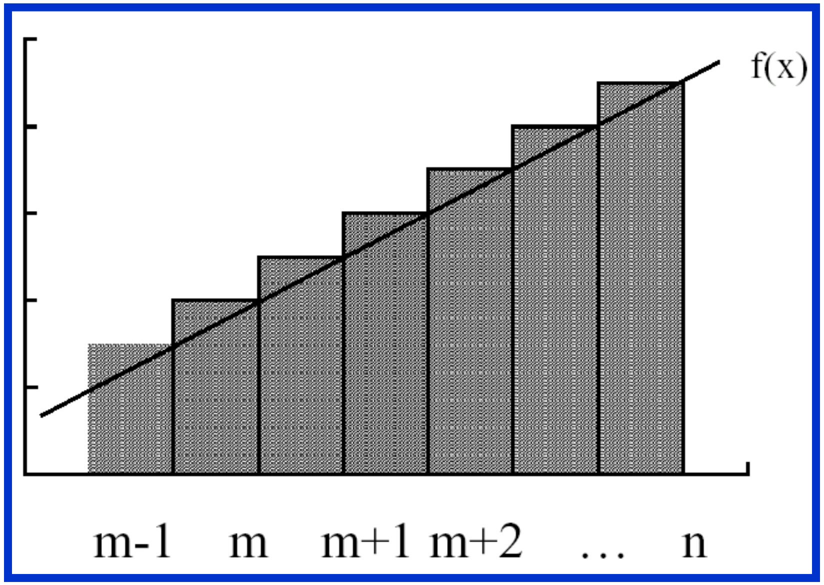
例 7. 求 $H_n = \sum_{k=1}^{n} \frac{1}{k}$ 的上界

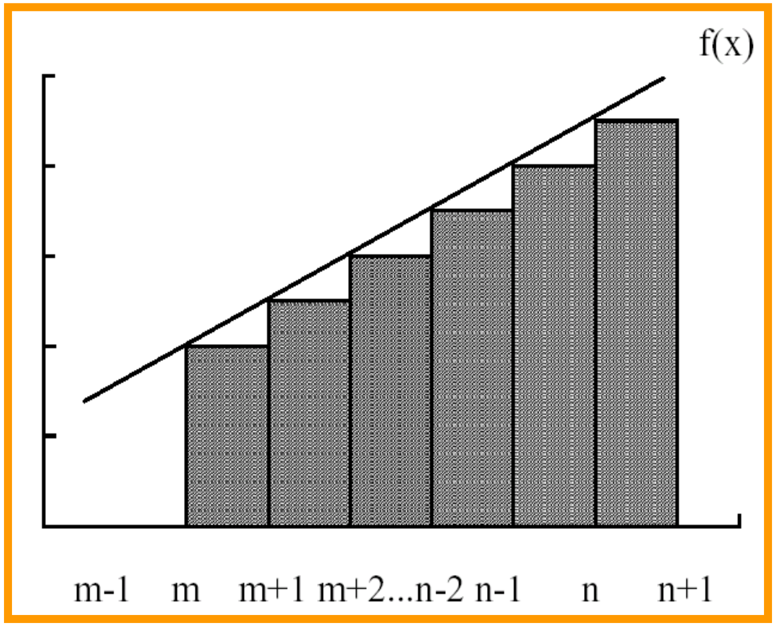
解： $$\sum_{k=1}^{n} \frac{1}{k} = \frac{1}{1} + \left(\frac{1}{2} + \frac{1}{3}\right) + \left(\frac{1}{4} + \frac{1}{5} + \frac{1}{6} + \frac{1}{7}\right)$$

$$+\left(\frac{1}{8} + \frac{1}{9} + \frac{1}{10} + \frac{1}{11} + \frac{1}{12} + \frac{1}{13} + \frac{1}{14} + \frac{1}{15}\right) + \ldots\ldots$$

$$\le \sum_{i=1}^{\lceil \log n \rceil} \sum_{j=0}^{2^i - 1} \frac{1}{2^i + j} \le \sum_{i=0}^{\lceil \log n \rceil} \sum_{j=0}^{2^i - 1} \frac{1}{2^i} \le \sum_{i=1}^{\lceil \log n \rceil} 1 \le \log n + 1 = O(\log n)$$

例 8. 如果 $f(k)$ 单调递增，则 $\int_{m-1}^{n} f(x)dx \le \sum_{k=m}^{n} f(k) \le \int_{m}^{n+1} f(x)dx$.





$$\sum_{k=m}^{n} f(k) \ge \sum_{k=m-1}^{n-1} f(k)\Delta x \ge \int_{m-1}^{n} f(x)dx \text{，} \quad f(m-1) < f(n) \text{，} \quad \Delta x = 1$$

$$\sum_{k=m}^{n} f(k) \le \sum_{k=m}^{n+1} f(k)\Delta x \le \int_{m}^{n+1} f(x)dx$$

例 9. 当 $f(x)$ 单调递减时，$\int_m^{n+1} f(x)dx \le \sum_{k=m}^{n} f(k) \le \int_{m-1}^{n} f(x)dx$.

例 10. $\sum_{k=1}^{n} \frac{1}{k} \ge \int_1^{n+1} \frac{dx}{x} = \ln(n+1)$ ， $\sum_{k=2}^{n} \frac{1}{k} \le \int_1^{n} \frac{dx}{x} = \ln n$.

2.4 递归方程

- 递归方程：递归方程是使用小的输入值来描述一个函数的方程或不等式.

- 递归方程例：Merge-sort排序算法的复杂性方程

$$T(n)=\theta(1) \qquad \text{if } n=1$$

$$T(n)=2T(n/2)+\theta(n) \qquad \text{if } n>1.$$

T(*n*)的解是$\theta(n\log n)$

求解递归方程的三个主要方法

- **Substitution方法:**
  - **Guess first,**
  - 然后用数学归纳法证明.
- **Iteration方法:**
  - 把方程转化为一个和式
  - 然后用估计和的方法来求解.
- **Master方法:**
  - 求解型为$T(n)=aT(n/b)+f(n)$的递归方程

2.4.1 Substitution方法

Substitution方法 I：联想已知的$T(n)$

例1. 求解$2T(n/2 + 17) + n$

解：猜测：$T(n)=2T\left(\frac{n}{2}+17\right)+n$ 与 $T(n)=2T\left(\frac{n}{2}\right)+n$ 只相差一个 17.

当 n 充分大时 $T\left(\frac{n}{2}+17\right)$ 与 $T\left(\frac{n}{2}\right)$ 的差别并不大，因为

$\frac{n}{2}+17$ 与 $\frac{n}{2}$ 相差小. 我们可以猜 $T(n)=O(n\lg n)$.

证明：用数学归纳法

Substitution方法 I：猜测上下界，减少不确定性范围

例 3. 求解 $T(n)=2T\left(\frac{n}{2}\right)+n$.

解： 首先证明 $T(n)=\Omega(n)$，$T(n)=O(n^2)$

然后逐阶地降低上界、提高下界。

$\Omega(n)$的上一个阶是$\Omega(n\log n)$,
$O(n^2)$的下一个阶是 $O(n\log n)$。

细微差别的处理

- 问题：猜测正确，数学归纳法的归纳步
  似乎证不出来
- 解决方法：从guess中减去一个低阶项，
  可能work.

例 **4**． 求解 $T(n)=T(\lfloor n/2\rfloor)+T\left(\left\lceil \frac{n}{2}\right\rceil\right)+1$

解：（1） 我们猜 $T(n)=O(n)$

证： $T(n)\le c\lfloor n/2\rfloor+c\lceil n/2\rceil+1=cn+1\ne cn$

证不出 $T(n)=O(cn)$

(2) 减去一个低阶项，猜 $T(n)\le cn-b$， $b\ge 0$ 是常数

证：设当 $\le n-1$ 时成立

$$T(n)=T(\lfloor n/2\rfloor)+T\left(\left\lceil \frac{n}{2}\right\rceil\right)+1\le c\left\lfloor \frac{n}{2}\right\rfloor-b+c\left\lceil \frac{n}{2}\right\rceil-b+1$$

$$=cn-2b+1=cn-b-b+1\le cn-b$$ （只要 $b\ge 1$ ）。

避免陷阱

例 5. 求解 $T(n)=2T(\lfloor n/2\rfloor)+n$ 。

解： 猜 $T(n)=O(n)$

证： 用数学归纳法证明 $T(n)\le cn$ 。 --错!!

$$T(n)\le 2(c\lfloor n/2\rfloor)+n\le cn+n=O(n)$$

错在那里： 过早地使用了 $O(n)$ 而陷入了陷阱应该在证明了 $T(n)\le cn$ 才可用。从 $T(n)\le cn+n$ 不可能得到 $T(n)\le cn$ 因为对于任何 c>0，我们都得不到 $cn+n\le cn$.

变量替换方法:

经变量替换把递归方程变换为熟悉的方程.

例 6. 求解 $T(n) = 2T(\sqrt{n}) + \lg n$

解: 令 $m = \lg n$, 则 $n = 2^m$, $T(2^m) = 2T\left(2^{\frac{m}{2}}\right) + m$.

令 $S(m) = T(2^m)$ 则 $T\left(2^{\frac{m}{2}}\right) = S\left(\frac{m}{2}\right)$. 于是, $S(m) = 2S\left(\frac{m}{2}\right) + m$.

显然, $S(m) = O(m\lg m)$, 即 $T(2^m) = \theta(m\lg m)$.

由于 $2^m = n$, $m = \lg n$, $T(n) = \theta(\lg n \times \lg(\lg n))$.

2.4.2 Iteration方法

方法：

循环地展开递归方程，
把递归方程转化为和式，
然后可使用求和技术解之。

例 1. $T(n)=n+3T\left(\left\lfloor n/4 \right\rfloor\right)$

$$=n+3\left(\left\lfloor n/4 \right\rfloor+3T\left(\left\lfloor n/16 \right\rfloor\right)\right)$$

$$=n+3\left(\left\lfloor n/4 \right\rfloor+3\left(\left\lfloor \frac{n}{16} \right\rfloor+3T\left(\left\lfloor n/64 \right\rfloor\right)\right)\right)$$

$$=n+3\left\lfloor n/4 \right\rfloor+9\left\lfloor \frac{n}{16} \right\rfloor+27T\left(\left\lfloor n/64 \right\rfloor\right)$$

$$=n+3\left\lfloor n/4 \right\rfloor+3^2\left\lfloor \frac{n}{4^2} \right\rfloor+3^3\left(\left\lfloor \frac{n}{4^3} \right\rfloor\right)+\ldots\ldots+3^i T\left(\left\lfloor n/4^i \right\rfloor\right)$$

$$\boxed{\text{令 } \frac{n}{4^i}=1 \Rightarrow 4^i=n \Rightarrow i=\log_4 n}$$

$$=n+3\left\lfloor n/4 \right\rfloor+3^2\left\lfloor \frac{n}{4^2} \right\rfloor+3^3\left(\left\lfloor \frac{n}{4^3} \right\rfloor\right)+\ldots\ldots+3^{\log_4 n}T\left(\left\lfloor 1 \right\rfloor\right)$$

$$\le \sum_{i=0}^{\log_4 n} 3^i \, n/4^i + O(n) \le n\sum_{i=0}^{\infty}\left(\frac{3}{4}\right)^i = n\times\frac{1}{1-\frac{3}{4}}=4n=O(n)$$

2.4.3 Master method

目的：求解 $T(n)=aT\left(\frac{n}{b}\right)+f(n)$ 型方程，$a\geq 1, b>0$ 是常数，$f(n)$ 是正函数

方法：记住三种情况，则不用笔纸即可求解上述方程.

Master 定理

定理 2.4.1 设$a \geq 1$和$b > 1$是常数，$f(n)$是一个函数，$T(n)$是定义在非负整数集上的函数$T(n) = aT\left(n/b\right) + f(n)$． $T(n)$可以如下求解：

(1)．若 $f(n) = O\left(n^{\log_b a-\varepsilon}\right)$, $\varepsilon > 0$是常数，则 $T(n) = \theta\left(n^{\log_b a}\right)$．

(2)．若 $f(n) = \theta\left(n^{\log_b a}\right)$，则 $T(n) = \theta\left(n^{\log_b a} \lg n\right)$．

(3)．若 $f(n) = \Omega\left(n^{\log_b a+\varepsilon}\right)$, $\varepsilon > 0$是常数，且对于所有充分大的n

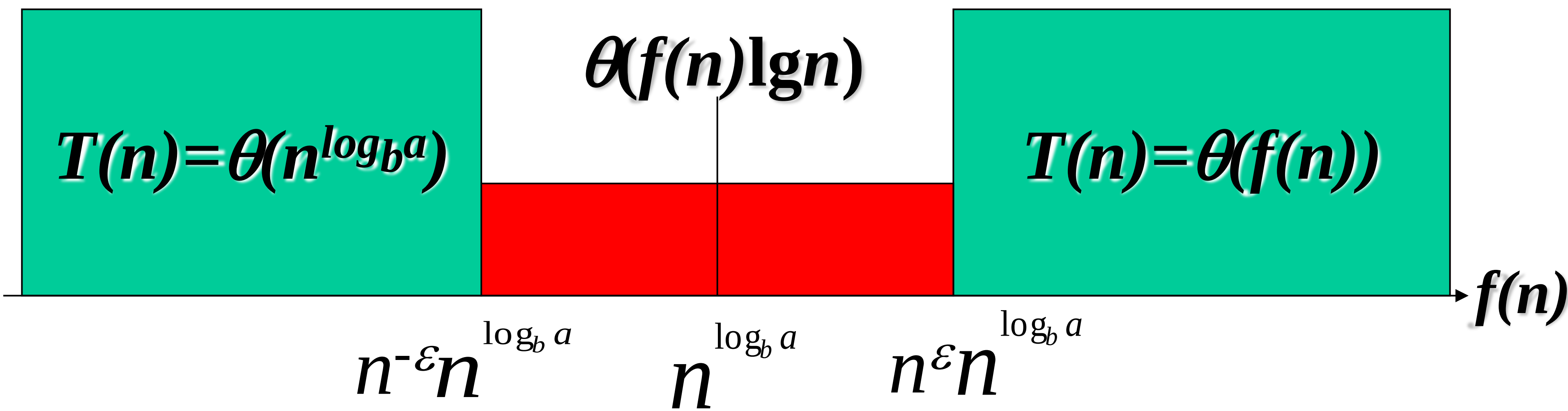
$af\left(n/b\right) \leq cf(n)$，c>1 是常数，则$T(n) = \theta(f(n))$．

*直观地：我们用 $f(n)$ 与 $n^{\log_b a}$ 比较

(1). 若 $n^{\log_b a}$ 大，则 $T(n)=\theta\left(n^{\log_b a}\right)$

(2). 若 $f(n)$ 大，则 $T(n)=\theta(f(n))$

(3). 若 $f(n)$ 与 $n^{\log_b a}$ 同阶，则 $T(n)=\theta\left(n^{\log_b a}\lg n\right)=\theta(f(n)\lg n)$.



对于红色部分，Master定理无能为力

更进一步:

(1). 在第一种情况, $f(n)$不仅小于$n^{\log_b a}$，必须多项式地小于, 即对于一个常数$\varepsilon>0$，$f(n)=O(\frac{n^{\log_b a}}{n^{\varepsilon}})$.

(2). 在第三种情况, $f(n)$不仅大于$n^{\log_b a}$，必须多项式地大于，即对一个常数$\varepsilon>0$，$f(n)=\Omega(n^{\log_b a}\cdot n^{\varepsilon})$.

Master定理的使用

例 1. 求解 $T(n)=9T\left(n/3\right)+n$.

解： $a=9$， $b=3$， $f(n)=n$， $n^{\log_b a}=\theta\left(n^2\right)$

$$\because f(n)=n=O\left(n^{\log_b a-\varepsilon}\right),\quad \varepsilon=1$$

$$\therefore T(n)=\theta\left(n^{\log_b a}\right)=\theta\left(n^2\right)$$

例 2. 求解 $T(n)=T\left(2n/3\right)+1$.

解： $a=1$， $b=\left(3/2\right)$， $f(n)=1$， $n^{\log_b a}=n^{\log_{3/2}1}=n^0=1$，

$$f(n)=1=\theta(1)=\theta\left(n^{\log_b a}\right),\quad T(n)=\theta\left(n^{\log_b a}\lg n\right)=\theta(\lg n)$$

Master定理的使用（续）

例 3. 求解 $T(n)=3T\left(n/4\right)+n\lg n$

解： $a=3$，$b=4$，$f(n)=n\lg n$，$n^{\log_b a}=n^{\log_4 3}=O\left(n^{0.793}\right)$

(1) $f(n)=n\lg n\ge n=n^{\log_b a+\varepsilon}$，$\varepsilon\approx 0.2$

(2) 对所有 **n**，$af\left(n/b\right)=3\times\frac{n}{4}\lg\frac{n}{4}=\frac{3}{4}n\lg\frac{n}{4}\le\frac{3}{4}n\lg n=cf(n)$，$c=\frac{3}{4}$.

于是，$T(n)=\theta(f(n))=\theta(n\lg n)$

例 4. 求解 $T(n)=2T\left(n/2\right)+n\lg n$.

解： $a=2$，$b=2$，$f(n)=n\lg n$， $n^{\log_b a}=n$. $f(n)=n\lg n$ 大于 $n^{\log_b a}=n$，但不是多项式**地**大于，Master 定理不**适**用于该 $T(n)$.

Master定理的证明

引理 1： 设 a≥1, b>1, $n=b^k$, k 是正整数， 则方程

$$T(n)=\begin{cases}\theta(1) & if \quad n=1\\ aT(n/b)+f(n) & if \quad n=\boldsymbol{b^i}\end{cases}$$

的解为:

$$T(n)=\theta(n^{\log_b a})+\sum_{j=0}^{\log_b n-1}a^j f(n/b^j)$$

证明: $T(n)=f(n)+aT(n/b)$

$$=f(n)+af(n/b)+a^2T(n/b^2)$$

$$=f(n)+af(n/b)+a^2f(n/b^2)+a^3T(n/b^3)+\ldots$$

$$+a^{\log_b n-1}f(n/b^{\log_b n-1})+a^{\log_b n}T(n/b^{\log_b n})$$

由于 $a^{\log_b n}=n^{\log_b a}$， $a^{\log_b n}T(n/b^{\log_b n})=a^{\log_b n}T(1)=\theta(n^{\log_b n})$． 于是

$$T(n)=\theta(n^{\log_b a})+\sum_{j=0}^{\log_b n-1}a^j f(n/b^j).$$

Master定理的证明(续)

引理 **2:** 设 a≥1, b>1, n=b^k, k 是正整数，$g(n)=\sum_{j=0}^{\log_b n-1} a^j f(n/b^j)$，则

(1) if $f(n)=O(n^{\log_b a-\varepsilon})$ for $\varepsilon>0$，则 $g(n)=O(n^{\log_b a})$

(2) if $f(n)=\theta(n^{\log_b a})$, then $g(n)=\theta(n^{\log_b a}\lg n)$

(3) if $af(n/b)\le cf(n)$ for some 0<c<1 and all n≥b, then $g(n)=\theta(f(n))$.

引理2的证明

(1) if $f(n)=O(n^{\log_b a-\varepsilon})$ for $\varepsilon>0$，则 $g(n)=O(n^{\log_b a})$

证明： (1) $g(n)=O(\sum_{j=0}^{\log_b n-1} a^j (\frac{n}{b^j})^{\log_b a-\varepsilon)})$

$$=n^{\log_b a-\varepsilon}\sum_{j=0}^{\log_b n-1}(\frac{ab^{\varepsilon}}{b^{\log_b a}})^j$$

$$=n^{\log_b a-\varepsilon}\sum_{j=0}^{\log_b n-1}(b^{\varepsilon})^j$$

$$=n^{\log_b a-\varepsilon}(\frac{b^{\varepsilon\log_b n}-1}{b^{\varepsilon}-1})$$

$$=n^{\log_b a-\varepsilon}(\frac{n^{\varepsilon}-1}{b^{\varepsilon}-1})$$

$$=n^{\log_b a-\varepsilon}O(n^{\varepsilon})=O(n^{\log_b a})$$

引理2的证明（续）

$$(2)\quad \text{if}\quad f(n)=\theta(n^{\log_b a}),\ \text{then}\quad g(n)=\theta(n^{\log_b a}\lg n)$$

证明:

$$(2)\ \text{由于}\ f(n/b^j)=\theta((n/b^j)^{\log_b a}),\quad g(n)=\theta\Big(\sum_{j=0}^{\log_b n-1} a^j\Big(\frac{n}{b^j}\Big)^{\log_b a}\Big).$$

$$\sum_{j=0}^{\log_b n-1} a^j\Big(\frac{n}{b^j}\Big)^{\log_b a}=n^{\log_b a}\sum_{j=0}^{\log_b n-1}\Big(\frac{a}{b^{\log_b a}}\Big)^j=n^{\log_b a}\sum_{j=0}^{\log_b n-1}1=n^{\log_b a}\log_b n$$

$$\Rightarrow g(n)=\theta(n^{\log_b a}\log_b n)=\theta(n^{\log_b a}\lg n).$$

(3) if $af(n/b) \le cf(n)$ for some 0<c<1 and all n≥b, then $g(n) = \theta(f(n))$.

(3) g(n)中的所有项皆为正. 由于对于 0<c<1 和 all n≥b,

$$af(n/b) \le cf(n),$$

$$af(\frac{n}{b^2}) \le cf(\frac{n}{b}),$$

$$af(\frac{n}{b^3}) \le cf(\frac{n}{b^2}),$$

$$\cdots$$

$$af(\frac{n}{b^j}) \le cf(\frac{n}{b^{j-1}})$$

我们有 $a^j f(n/b)\cdots \cancel{f(n/b^{j-1})} f(n/b^j) \le c^j f(n) f(n/b)\cdots \cancel{f(n/b^{j-1})}$

$$\Rightarrow a^j f(\frac{n}{b^j}) \le c^j f(n)$$

于是,

$$g(n) = \sum_{j=0}^{\log_b n-1} a^j f(\frac{n}{b^j}) \le \sum_{j=0}^{\log_b n-1} c^j f(n) \le f(n)\sum_{j=0}^{\infty} c^j = f(n)(\frac{1}{1-c}) = \theta(f(n))$$

Master定理的证明(续)

引理 3: a≥1, b>1, n=b^k, k 为正整数，则

$$T(n)=\begin{cases}\theta(1) & if \quad n=1\\ aT(n/b)+f(n) & if \quad n=b^k\end{cases}$$

的解为:

(1) if $f(n)=O(n^{\log_b a-\varepsilon})$ for some $\varepsilon>0$, then $T(n)=\theta(n^{\log_b a})$

(2) if $f(n)=\theta(n^{\log_b a})$, then $T(n)=\theta(n^{\log_b a}\lg n)$

(3) if $f(n)=\Omega(n^{\log_b a+\varepsilon})$ for some $\varepsilon>0$, and if

$af(n/b)\le cf(n)$ for some 0<c<1 and all 充分大的 n, then $T(n)=\theta(f(n))$

引理3的证明

证明： (1) 由引理 1 和引理 2:

$$T(n)=\theta(n^{\log_b a})+O(n^{\log_b a})=\theta(n^{\log_b a})$$

(2) $T(n)=\theta(n^{\log_b a})+\theta(n^{\log_b a}\lg n)=\theta(n^{\log_b a}\lg n)$

(3) $T(n)=\theta(n^{\log_b a})+\theta(f(n))=\theta(f(n))$

Master定理的证明(续)

当 n 不是 b 的幂时

思想：当 $T(n)$单调递增（单调递减类似）

- $aT\left(\left\lfloor \frac{n}{b} \right\rfloor\right)+f(n) \le aT\left(\frac{n}{b}\right)+f(n) \le aT\left(\left\lceil \frac{n}{b} \right\rceil\right)+f(n)$
- 求$T(n)=aT\left(\left\lceil \frac{n}{b} \right\rceil\right)+f(n)$的上界、$T(n)=aT\left(\left\lfloor \frac{n}{b} \right\rfloor\right)+f(n)$的下界

  可得到 $T(n)$的界限。
- 求$T(n)=aT\left(\left\lfloor \frac{n}{b} \right\rfloor\right)+f(n)$的下界类似于求$T(n)=aT\left(\left\lceil \frac{n}{b} \right\rceil\right)+f(n)$的上界，所以我们只求$T(n)=aT\left(\left\lceil \frac{n}{b} \right\rceil\right)+f(n)$的上界

Master定理的证明(续)

- 方法仍然是循环展开 $T(n)=aT(\lceil \frac{n}{b} \rceil)+f(n)$
- 需要处理序列：

$$n$$

$$\left\lceil \frac{n}{b} \right\rceil$$

$$\left\lceil \left\lceil \frac{n}{b} \right\rceil \Big/ b \right\rceil$$

$$\left\lceil \left\lceil \left\lceil \frac{n}{b} \right\rceil \Big/ b \right\rceil \Big/ b \right\rceil$$

$$\cdots$$

Master定理的证明(续)

定义： $n_i = \begin{cases} n & if \quad i = 0 \\ \lceil n_{i-1}/b \rceil & if \quad i > 0 \end{cases}$

引理 **4.** $n_0 \le n$， $n_1 \le \frac{n}{b}+1$， $n_2 \le \frac{n}{b^2}+\frac{1}{b}+1$， $n_3 \le \frac{n}{b^3}+\frac{1}{b^2}+\frac{1}{b}+1$，

$$\ldots, \quad n_i \le \frac{n}{b^i} + \sum_{j=0}^{i-1}\frac{1}{b^j} \le \frac{n}{b^i} + \frac{b}{b-1}$$ 。

证： 由 $\lceil x \rceil \le x+1$ 可证。

Master定理的证明(续)

引理 5：当$i=\lfloor \log_b n \rfloor$时，$n_i \le b+\frac{b}{b-1}=O(1)$

证：由于$n_i \le \frac{n}{b^i}+\frac{b}{b-1}$，我们有

$$n_{\lfloor \log_b n \rfloor} \le \frac{n}{b^{\lfloor \log_b n \rfloor}}+\frac{b}{b-1} \le \frac{n}{b^{(\log_b n)-1}}+\frac{b}{b-1}=b+\frac{b}{b-1}=O(1)\text{。}$$

Master定理的证明(续)

引理 6: $$T(n)=aT(\left\lceil \frac{n}{b} \right\rceil)+f(n)\le \theta(n^{\log_b a})+\sum_{j=0}^{\lfloor \log_b n\rfloor -1} a^j f(n_j)$$

证: $$\begin{aligned} T(n) &= f(n_0)+aT(n_1)=f(n_0)+af(n_1)+a^2f(n_2) \\ &\le f(n_0)+af(n_1)+a^2f(n_2)+\ldots+a^{\lfloor \log_b n\rfloor -1}f(n_{\lfloor \log_b n\rfloor -1}) \\ &\quad +a^{\lfloor \log_b n\rfloor}T(n_{\lfloor \log_b n\rfloor}) \\ &= \theta(n^{\log_b a})+\sum_{j=0}^{\lfloor \log_b n\rfloor -1} a^j f(n_j) \end{aligned}$$

Master定理的证明(续)

引理 7: $g(n)=\sum_{j=0}^{\lfloor \log_b n \rfloor -1} a^j f(n_j)$ 可以界限如下:

(1) If $f(n)=O(n^{\log_b a-\varepsilon})$ for $\varepsilon>0$, $g(n)=O(n^{\log_b a})$.

(2) If $f(n)=\theta(n^{\log_b a})$, then $g(n)=\theta(n^{\log_b a}\lg n)$.

(3) If $af(\lceil n/b \rceil)\le cf(n)$ for 0<c<1 and all 充分大的 n, then $g(n)=\theta(f(n))$.

引理7的证明

证明：（3）由 $af(\lceil n/b \rceil) \le cf(n)$ 有：

$$af\left(\left\lceil \frac{n}{b} \right\rceil\right) \le cf(n) \Leftrightarrow af(n_1) \le cf(n_0)$$

$$af\left(\left\lceil \left\lceil \frac{n}{b} \right\rceil \Big/ b \right\rceil\right) \le cf\left(\left\lceil \frac{n}{b} \right\rceil\right) \Leftrightarrow af(n_2) \le cf(n_1)$$

$$\cdots$$

$$af\left(\left\lceil \cdots \left\lceil \frac{n}{b} \right\rceil \Big/ b \quad \cdots \right\rceil\right) \le cf\left(\left\lceil \cdots \left\lceil \frac{n}{b} \right\rceil \quad \cdots \right\rceil\right) \Leftrightarrow af(n_j) \le cf(n_{j-1})$$

$$\Rightarrow a^j f(n_1) \cdots f(n_{j-1}) f(n_j) \le c^j f(n_0) f(n_1) \cdots f(n_{j-1})$$

$$\Rightarrow a^j f(n_j) \le c^j f(n_0) = c^j f(n)$$

证明的其余部分与引理 2 的（3）的证明类似。

引理7的证明

（2）只要证明 $f(n_j)=O(n^{\log_b a}/a^j)=O((\frac{n}{b^j})^{\log_b a})$，即可用引理 2 的（2）的证明完成本证明。

$$j\le\lfloor\log_b n\rfloor\Rightarrow b^j\le b^{\lfloor\log_b n\rfloor}=b^{\log_b n-\delta}=n\cdot\frac{1}{b^\delta}\qquad(0\le\delta<1)$$

$$\Rightarrow\frac{b^j}{n}\le\frac{1}{b^\delta}<1\ (\because b>1)\Rightarrow b^j/n<1\text{。}$$

由于 $f(n)=\Theta(n^{\log_b a})$，$\exists c>0$，使对于充分大的 n_j，

$$f(n_j)\le cn_j^{\log_b a}\le c(\frac{n}{b^j}+\frac{b}{b-1})^{\log_b a}$$

$$=c(\frac{n}{b^j})^{\log_b a}(1+\frac{b^j}{n}\cdot\frac{b}{b-1})^{\log_b a}$$

$$\le c(\frac{n^{\log_b a}}{a^j})(1+\frac{b}{b+1})^{\log_b a}\qquad(\because\frac{b^j}{n}<1)$$

$$\le O(\frac{n^{\log_b a}}{a^j})\qquad(\because c(1+\frac{b}{b+1})^{\log_b a}\text{是常数})$$

于是（2）被证明。

HIT
CS&E

引理7的证明

(1) 只要证明 $f(n_j) = O((\frac{n}{b^j})^{\log_b a-\varepsilon})$，则本证明的其余部分与引理 2 的（1）相同。类似（2）可证明 $f(n_j) = O((\frac{n}{b^j})^{\log_b a-\varepsilon})$。

至此，我们完成了 **Master** 定理的证明。