主定理证明

张腾

2022 年 3 月 11 日

定理 1 (主定理)**.** 令 $a \geq 1$ 和 $b > 1$ 是常数，$f(n)$ 是一个函数，$T(n)$ 是定义在**非负整数**上的递归式：

$$T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$$

其中我们将 n/b 解释为 $\lfloor n/b \rfloor$ 或 $\lceil n/b \rceil$，那么 $T(n)$ 有如下渐进界：

1. 若对某个常数 $\varepsilon > 0$ 有 $f(n) = O(n^{\log_b a - \varepsilon})$，则 $T(n) = \Theta(n^{\log_b a})$。
2. 若 $f(n) = \Theta(n^{\log_b a})$，则 $T(n) = \Theta(n^{\log_b a} \lg n)$。
3. 若对某个常数 $\varepsilon > 0$ 有 $f(n) = \Omega(n^{\log_b a + \varepsilon})$，且对某个常数 $c < 1$ 和所有足够大的 n 有 $af(n/b) \leq cf(n)$，则 $T(n) = \Theta(f(n))$。

证明. 若 n 是 b 的整数幂次，不妨设 $n = b^k$，此时 n/b 最后正好会递归成 1，即

$$\begin{aligned} T\left(\frac{n}{b^0}\right) &= a \cdot T\left(\frac{n}{b^1}\right) + f\left(\frac{n}{b^0}\right) \\ T\left(\frac{n}{b^1}\right) &= a \cdot T\left(\frac{n}{b^2}\right) + f\left(\frac{n}{b^1}\right) \\ &\vdots \\ T\left(\frac{n}{b^{k-1}}\right) &= a \cdot T\left(\frac{n}{b^k}\right) + f\left(\frac{n}{b^{k-1}}\right) \end{aligned}$$

注意上面共有 k 个递推式，又 $a^k = (b^{\log_b a})^k = (b^k)^{\log_b a} = n^{\log_b a}$、$n = b^k$，于是

$$T(n) = a^k \cdot T\left(\frac{n}{b^k}\right) + f\left(\frac{n}{b^0}\right) + af\left(\frac{n}{b^1}\right) + \cdots + a^{k-1} f\left(\frac{n}{b^{k-1}}\right) = n^{\log_b a} \cdot T(1) + \sum_{i=0}^{k-1} a^i f\left(\frac{n}{b^i}\right)$$

显然前一项 $n^{\log_b a} \cdot T(1) = \Theta(n^{\log_b a})$，故关键就是后面求和这一项，下面分情况讨论：

1. 由 $f(n) = O(n^{\log_b a - \varepsilon})$ 可知

$$a^i f\left(\frac{n}{b^i}\right) = a^i O\left(\left(\frac{n}{b^i}\right)^{\log_b a - \varepsilon}\right) = a^i O\left(\frac{n^{\log_b a - \varepsilon}}{a^i b^{-\varepsilon i}}\right) = O(n^{\log_b a - \varepsilon}) b^{\varepsilon i}$$

故

$$g(n) = \sum_{i=0}^{k-1} f\left(\frac{n}{b^i}\right) = O(n^{\log_b a - \varepsilon}) \sum_{i=0}^{k-1} b^{\varepsilon i} = O(n^{\log_b a - \varepsilon}) \frac{1 - b^{k\varepsilon}}{1 - b^{\varepsilon}} = O(n^{\log_b a - \varepsilon}) \frac{n^{\varepsilon} - 1}{b^{\varepsilon} - 1} = O(n^{\log_b a})$$

从而 $T(n) = \Theta(n^{\log_b a}) + O(n^{\log_b a}) = \Theta(n^{\log_b a})$，即此时 $T(n)$ 的复杂度由前一项决定。

2. 由 $f(n)=\Theta(n^{\log_b a})$ 可知

$$a^i f\left(\frac{n}{b^i}\right)=a^i\Theta\left(\left(\frac{n}{b^i}\right)^{\log_b a}\right)=a^i\Theta\left(\frac{n^{\log_b a}}{a^i}\right)=\Theta(n^{\log_b a})$$

于是

$$g(n)=\sum_{i=0}^{k-1}a^i f\left(\frac{n}{b^i}\right)=\sum_{i=0}^{k-1}\Theta(n^{\log_b a})=\Theta(n^{\log_b a})\log_b n=\Theta(n^{\log_b a}\lg n)$$

从而 $T(n)=\Theta(n^{\log_b a})+\Theta(n^{\log_b a}\lg n)=\Theta(n^{\log_b a}\lg n)$，即此时 $T(n)$ 的复杂度由求和这一项决定。

3. 由条件知存在 j 使得当 $n'\geq b^{k-j+1}=n/b^{j-1}$ 时递推式 $af(n'/b)\leq cf(n')$ 恒成立，于是

$$a^j f\left(\frac{n}{b^j}\right)\leq a^{j-1}cf\left(\frac{n}{b^{j-1}}\right)\leq\cdots\leq a^2c^{j-2}f\left(\frac{n}{b^2}\right)\leq ac^{j-1}f\left(\frac{n}{b}\right)\leq c^j f(n)$$

从而对 $\forall i\leq j$ 有 $a^i f(n/b^i)\leq c^i f(n)$，代入可得

$$\begin{aligned}g(n)&=\sum_{i=0}^{k-1}a^i f\left(\frac{n}{b^i}\right)=\sum_{i=0}^{j}a^i f\left(\frac{n}{b^i}\right)+\overbrace{a^{j+1}f\left(\frac{n}{b^{j+1}}\right)+\cdots+a^{k-1}f\left(\frac{n}{b^{k-1}}\right)}^{\text{不满足递推式的部分，复杂度为}O(1)}\\&\leq\sum_{i=0}^{j}c^i f(n)+O(1)<f(n)\sum_{i=0}^{\infty}c^i+O(1)\overset{c\leq 1}{=}\frac{f(n)}{1-c}+O(1)=O(f(n))\end{aligned}$$

又 $f(n)$ 是 $g(n)$ 中的一项且 $g(n)$ 所有求和项非负，故 $g(n)\geq f(n)$，从而

$$g(n)=\Theta(f(n))=\Omega(n^{\log_b a+\varepsilon})>\Theta(n^{\log_b a})$$

于是 $T(n)=\Theta(n^{\log_b a})+g(n)=\Theta(f(n))$，即此时 $T(n)$ 的复杂度由求和这一项决定。

若 n 不是 b 的整数幂次，则递推中某一轮 n/b 不一定为整数，又 $T(n)$ 是定义在非负整数上的，因此还需多做一步取整，此时递推式可写成

$$\begin{aligned}T(n_0)&=a\cdot T(n_1)+f(n_0),\quad n_1=[n_0/b]\\T(n_1)&=a\cdot T(n_2)+f(n_1),\quad n_2=[n_1/b]\\T(n_2)&=a\cdot T(n_3)+f(n_2),\quad n_3=[n_2/b]\\&\vdots\end{aligned}$$

不妨先考虑向上取整，即

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

根据 $x\leq\lceil x\rceil\leq x+1$ 易知

$$\frac{n}{b}\leq n_1\leq\frac{n}{b}+1$$

$$
\begin{aligned}
\frac{n}{b^2} &\le n_2 \le \frac{n}{b^2} + \frac{1}{b} + 1 \\
\frac{n}{b^3} &\le n_3 \le \frac{n}{b^3} + \frac{1}{b^2} + \frac{1}{b} + 1 \\
&\vdots
\end{aligned}
$$

于是

$$
\frac{n}{b^i} \le n_i \le \frac{n}{b^i} + \frac{1}{b^{i-1}} + \cdots + \frac{1}{b} + 1 < \frac{n}{b^i} + \frac{1}{1-1/b} = \frac{n}{b^i} + \frac{b}{b-1} \tag{1}
$$

令 $k = \lfloor \log_b n \rfloor \ge \log_b n - 1$，即 $n \le b^{k+1}$，代入式 (1) 可得

$$
n_k < \frac{n}{b^k} + \frac{b}{b-1} \le \frac{b^{k+1}}{b^k} + \frac{b}{b-1} = b + \frac{b}{b-1} = O(1)
$$

故递推到第 k 轮，子问题 $T(n_k)$ 已经是 $O(1)$ 复杂度了，同前面一样有

$$
T(n) = a^k T(n_k) + f(n_0) + af(n_1) + \cdots + a^{k-1} f(n_{k-1}) = \Theta(n^{\log_b a}) + \sum_{i=0}^{k-1} a^i f(n_i)
$$

关键依然是后面求和这一项，继续分情况讨论：

1. 由 $f(n) = O(n^{\log_b a - \varepsilon})$ 可知存在正常数 c_i 和 n_i 使得对 $\forall n \ge n_i$ 有

$$
\begin{aligned}
a^i f(n_i) &\le a^i c_i n_i^{\log_b a - \varepsilon} < a^i c_i \left(\frac{n}{b^i} + \frac{b}{b-1} \right)^{\log_b a - \varepsilon} = a^i c_i \left(\frac{n}{b^i} \right)^{\log_b a - \varepsilon} \left(1 + \frac{b^i}{n} \frac{b}{b-1} \right)^{\log_b a - \varepsilon} \\
&\overset{b^i \le n}{\le} a^i c_i n^{\log_b a - \varepsilon} \frac{b^{\varepsilon i}}{a^i} \left(1 + \frac{b}{b-1} \right)^{\log_b a - \varepsilon} = c_i n^{\log_b a - \varepsilon} b^{\varepsilon i} \left(\frac{2b-1}{b-1} \right)^{\log_b a - \varepsilon}
\end{aligned}
$$

   令 $\bar{c} = \max_i \{c_i\}$、$\bar{n} = \max_i \{n_i\}$，于是对 $\forall n \ge \bar{n}$ 有

$$
\begin{aligned}
g(n) &= \sum_{i=0}^{k-1} a^i f(n_i) < \sum_{i=0}^{k-1} c_i n^{\log_b a - \varepsilon} b^{\varepsilon i} \left(\frac{2b-1}{b-1} \right)^{\log_b a - \varepsilon} \le \bar{c} n^{\log_b a - \varepsilon} \left(\frac{2b-1}{b-1} \right)^{\log_b a - \varepsilon} \sum_{i=0}^{k-1} b^{\varepsilon i} \\
&\overset{b^k \le n}{\le} \bar{c} n^{\log_b a - \varepsilon} \left(\frac{2b-1}{b-1} \right)^{\log_b a - \varepsilon} \frac{n^\varepsilon - 1}{b^\varepsilon - 1} < \frac{\bar{c}}{b^\varepsilon - 1} \left(\frac{2b-1}{b-1} \right)^{\log_b a - \varepsilon} n^{\log_b a} = O(n^{\log_b a})
\end{aligned}
$$

   从而 $T(n) = \Theta(n^{\log_b a}) + O(n^{\log_b a}) = \Theta(n^{\log_b a})$，即此时 $T(n)$ 的复杂度由前一项决定。

2. 由 $f(n) = \Theta(n^{\log_b a})$ 可知存在正常数 c_i、d_i 和 n_i 使得对 $\forall n \ge n_i$ 有

$$
a^i f(n_i) \le a^i c_i n_i^{\log_b a} < a^i c_i \left(\frac{n}{b^i} \right)^{\log_b a} \left(1 + \frac{b^i}{n} \frac{b}{b-1} \right)^{\log_b a} \le c_i \left(1 + \frac{b}{b-1} \right)^{\log_b a} n^{\log_b a} = O(n^{\log_b a})
$$

   以及

$$
a^i f(n_i) \ge a^i d_i n_i^{\log_b a} \ge a^i d_i \left(\frac{n}{b^i} \right)^{\log_b a} = d_i n^{\log_b a} = \Omega(n^{\log_b a})
$$

   故 $a^i f(n_i) = \Theta(n^{\log_b a})$，于是

$$
g(n) = \sum_{i=0}^{k-1} a^i f(n_i) = \Theta(n^{\log_b a}) k = \Theta(n^{\log_b a} \lg n)
$$

   从而 $T(n) = \Theta(n^{\log_b a}) + \Theta(n^{\log_b a} \lg n) = \Theta(n^{\log_b a} \lg n)$，即此时 $T(n)$ 的复杂度由求和这一项决定。

3. 由条件知存在 j 使得当 $n \geq n_{j-1}$ 时递推式 $af(\lceil n/b \rceil) \leq cf(n)$ 恒成立，于是

$$a^j f(n_j) \leq a^{j-1} c f(n_{j-1}) \leq \cdots \leq a^2 c^{j-2} f(n_2) \leq a c^{j-1} f(n_1) \leq c^j f(n_0)$$

从而对 $\forall i \leq j$ 有 $a^i f(n_i) \leq c^i f(n)$，代入可得

$$\begin{aligned} g(n) = \sum_{i=0}^{k-1} a^i f(n_i) &= \sum_{i=0}^{j} a^i f(n_i) + \overbrace{a^{j+1} f(n_{j+1}) + \cdots + a^{k-1} f(n_{k-1})}^{\text{不满足递推式的部分，复杂度为}O(1)} \\ &\leq \sum_{i=0}^{j} c^i f(n) + O(1) < f(n) \sum_{i=0}^{\infty} c^i + O(1) \overset{c \leq 1}{=} \frac{f(n)}{1-c} + O(1) = O(f(n)) \end{aligned}$$

又 $f(n)$ 是 $g(n)$ 中的一项且 $g(n)$ 所有求和项非负，故 $g(n) \geq f(n)$，从而

$$g(n) = \Theta(f(n)) = \Omega(n^{\log_b a + \varepsilon}) > \Theta(n^{\log_b a})$$

于是 $T(n) = \Theta(n^{\log_b a}) + g(n) = \Theta(f(n))$，即此时 $T(n)$ 的复杂度由求和这一项决定。

最后考虑向下取整，即

$$n_i = \begin{cases} n & i = 0 \\ \lfloor n_{i-1}/b \rfloor & i > 0 \end{cases}$$

根据 $x - 1 \leq \lfloor x \rfloor \leq x$ 易知

$$\begin{aligned} \frac{n}{b} - 1 \leq n_1 \leq \frac{n}{b} \\ \frac{n}{b^2} - \frac{1}{b} - 1 \leq n_2 \leq \frac{n}{b^2} \\ \frac{n}{b^3} - \frac{1}{b^2} - \frac{1}{b} - 1 \leq n_3 \leq \frac{n}{b^3} \\ \vdots \end{aligned}$$

于是

$$\frac{n}{b^i} - \frac{b}{b-1} = \frac{n}{b^i} - \frac{1}{1-1/b} < \frac{n}{b^i} - \frac{1}{b^{i-1}} - \cdots - \frac{1}{b} - 1 \leq n_i \leq \frac{n}{b^i} \tag{2}$$

令 $k = \lfloor \log_b n(b-1) \rfloor - 2 \geq \log_b n(b-1) - 3$，即 $n(b-1) \leq b^{k+3}$，代入式 (2) 可得

$$n_k \leq \frac{n}{b^k} \leq \frac{b^{k+3}}{b^k} = \frac{b^3}{b-1} = O(1)$$

故递推到第 k 轮，子问题 $T(n_k)$ 已经是 $O(1)$ 复杂度了，同前面一样有

$$T(n) = a^k T(n_k) + f(n_0) + af(n_1) + \cdots + a^{k-1} f(n_{k-1}) = \Theta(n^{\log_b a}) + \sum_{i=0}^{k-1} a^i f(n_i)$$

关键依然是后面求和这一项，继续分情况讨论：

1. 由 $f(n)=O(n^{\log_b a-\varepsilon})$ 可知存在正常数 c_i 和 n_i 使得对 $\forall n\geq n_i$ 有

$$a^i f(n_i)\leq a^i c_i n_i^{\log_b a-\varepsilon}<a^i c_i\left(\frac{n}{b^i}\right)^{\log_b a-\varepsilon}=c_i n^{\log_b a-\varepsilon}b^{\varepsilon i}$$

令 $\bar{c}=\max_i\{c_i\}$、$\bar{n}=\max_i\{n_i\}$，于是对 $\forall n\geq\bar{n}$ 有

$$g(n)=\sum_{i=0}^{k-1}a^i f(n_i)<\sum_{i=0}^{k-1}c_i n^{\log_b a-\varepsilon}b^{\varepsilon i}\leq\bar{c}n^{\log_b a-\varepsilon}\sum_{i=0}^{k-1}b^{\varepsilon i}\leq\bar{c}n^{\log_b a-\varepsilon}\frac{n^{\varepsilon}-1}{b^{\varepsilon}-1}=O(n^{\log_b a})$$

从而 $T(n)=\Theta(n^{\log_b a})+O(n^{\log_b a})=\Theta(n^{\log_b a})$，即此时 $T(n)$ 的复杂度由前一项决定。

2. 由 $f(n)=\Theta(n^{\log_b a})$ 可知存在正常数 c_i、d_i 和 n_i 使得对 $\forall n\geq n_i$ 有

$$a^i f(n_i)\leq a^i c_i n_i^{\log_b a}\leq a^i c_i\left(\frac{n}{b^i}\right)^{\log_b a}\leq c_i n^{\log_b a}=O(n^{\log_b a})$$

以及

$$\begin{aligned}a^i f(n_i)\geq a^i d_i n_i^{\log_b a}\geq a^i d_i\left(\frac{n}{b^i}-\frac{b}{b-1}\right)^{\log_b a}&=a^i d_i\left(\frac{n}{b^i}\right)^{\log_b a}\left(1-\frac{b^i}{n}\frac{b}{b-1}\right)^{\log_b a}\\&=d_i n^{\log_b a}\left(1-\frac{b^i}{n}\frac{b}{b-1}\right)^{\log_b a}\end{aligned}$$

注意 $k=\lfloor\log_b n(b-1)\rfloor-2\leq\log_b n(b-1)-2$，即 $n(b-1)\geq b^{k+2}$，于是

$$1>1-\frac{b^i}{n}\frac{b}{b-1}\geq 1-\frac{b^{i+1}}{b^{k+2}}>0\Longrightarrow 1-\frac{b^i}{n}\frac{b}{b-1}\in(0,1)\Longrightarrow a^i f(n_i)=\Omega(n^{\log_b a})$$

故 $a^i f(n_i)=\Theta(n^{\log_b a})$，于是

$$g(n)=\sum_{i=0}^{k-1}a^i f(n_i)=\Theta(n^{\log_b a})k=\Theta(n^{\log_b a}\lg n)$$

从而 $T(n)=\Theta(n^{\log_b a})+\Theta(n^{\log_b a}\lg n)=\Theta(n^{\log_b a}\lg n)$，即此时 $T(n)$ 的复杂度由求和这一项决定。

3. 由条件知存在 j 使得当 $n\geq n_{j-1}$ 时递推式 $af(\lfloor n/b\rfloor)\leq cf(n)$ 恒成立，于是

$$a^j f(n_j)\leq a^{j-1}cf(n_{j-1})\leq\cdots\leq a^2c^{j-2}f(n_2)\leq ac^{j-1}f(n_1)\leq c^j f(n_0)$$

从而对 $\forall i\leq j$ 有 $a^i f(n_i)\leq c^i f(n)$，代入可得

$$\begin{aligned}g(n)=\sum_{i=0}^{k-1}a^i f(n_i)&=\sum_{i=0}^{j}a^i f(n_i)+\overbrace{a^{j+1}f(n_{j+1})+\cdots+a^{k-1}f(n_{k-1})}^{\text{不满足递推式的部分，复杂度为}O(1)}\\&\leq\sum_{i=0}^{j}c^i f(n)+O(1)<f(n)\sum_{i=0}^{\infty}c^i+O(1)\overset{c\leq 1}{=}\frac{f(n)}{1-c}+O(1)=O(f(n))\end{aligned}$$

又 $f(n)$ 是 $g(n)$ 中的一项且 $g(n)$ 所有求和项非负，故 $g(n)\geq f(n)$，从而

$$g(n)=\Theta(f(n))=\Omega(n^{\log_b a+\varepsilon})>\Theta(n^{\log_b a})$$

于是 $T(n)=\Theta(n^{\log_b a})+g(n)=\Theta(f(n))$，即此时 $T(n)$ 的复杂度由求和这一项决定。

♣