Master Theorem

Jiman Hwang

This document is for self-study only.
Source: [THC09]

1 Statement

Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function, and define the complexity $T : \mathbb{N} \to \mathbb{Z}$ as

$$T(n) = aT(n/b) + f(n) \tag{1}$$

where we interpret n/b to mean either $\lceil n/b \rceil$ or $\lfloor n/b \rfloor$. Then $T(n)$ has the following asymptotic bounds:

1. If $f(n) = O\left(n^{\log_b a - \varepsilon}\right)$ for some constant $\varepsilon > 0$, then $T(n) = \Theta\left(n^{\log_b a}\right)$.
2. If $f(n) = \Theta\left(n^{\log_b a}\right)$, then $T(n) = \Theta\left(n^{\log_b a} \lg n\right)$.
3. If $f(n) = \Omega\left(n^{\log_b a + \varepsilon}\right)$ for some constant $\varepsilon > 0$, and if $af(n/b) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large n then $T(n) = \Theta\{f(n)\}$.

2 Derivation

Before we start, assume the complexity T is a monotonic function. This is plausible because the majority of complexity would be increased as the input size gets bigger. Using the assumption, the following is true.

$$T(\lfloor n/b \rfloor) \leq T(n/b) \leq T(\lceil n/b \rceil)$$

Modifying a little bit,

$$\begin{aligned} &T(\lfloor n/b \rfloor) \leq T(n/b) \leq T(\lceil n/b \rceil) \\ \Rightarrow\ &aT(\lfloor n/b \rfloor) + f(n) \leq aT(n/b) + f(n) \leq aT(\lceil n/b \rceil) + f(n) \end{aligned}$$

Let $T_1(n) = aT(\lfloor n/b \rfloor) + f(n)$, $T_2(n) = aT(\lceil n/b \rceil) + f(n)$, then

$$T_1(n) \leq T(n) \leq T_2(n)$$

That is, T_1 is a lower bound of T, while T_2 is an upper bound of T. If T_1 and T_2 are the same in the asymptotic sense, then so is T, which we will go through. To make this happen, we have to resolve the two recurrence respectively.

1. $T(n) = aT(\lfloor n/b \rfloor) + f(n)$
2. $T(n) = aT(\lceil n/b \rceil) + f(n)$

Then check if they match each other.

2.1 upper bound of T

The upper bound is the solution of

$$T(n) = aT(\lceil n/b \rceil) + f(n) \tag{2}$$

Assume

$$T(n) = \Theta(1) \quad \text{if } n \le s \tag{3}$$

where $s > b/(b-1)$. Since $b/(b-1)$ is a finite value, the assumption is legitimate. Let

$$n_i \stackrel{\text{def}}{=} \begin{cases} n & i = 0 \\ \lceil n_{i-1}/b \rceil & i = 1, 2, \cdots \end{cases}$$

Then, (2) is expressed as

$$T(n) = aT(n_1) + f(n)$$

By replacing n with n_1, and multiplying a on the both hand sides, we get

$$aT(n_1) = a^2 T(n_2) + af(n_1)$$

Again, do the same action on the above equation.

$$a^2 T(n_2) = a^3 T(n_3) + a^2 f(n_2)$$

In this manner, we get the series of equations. Add them up all.

$$\begin{aligned} T(n) &= aT(n_1) + f(n) \\ aT(n_1) &= a^2 T(n_2) + af(n_1) \\ &\vdots \\ a^{k-1} T(n_{k-1}) &= a^k T(n_k) + a^{k-1} f(n_{k-1}) \\ \hline T(n) &= a^k T(n_k) + g(n) \end{aligned} \tag{4}$$

where

$$n_k \le s < n_{k-1}, \text{ and } g(n) = \sum_{r=0}^{k-1} a^r f(n_r) \tag{5}$$

To resolve k, let's take a look at (5)

$$\begin{aligned}
s \geq n_k = \left\lceil \frac{n_{k-1}}{b} \right\rceil \geq \frac{n_{k-1}}{b} \\
\Rightarrow sb \geq n_{k-1} \\
\vdots \\
\Rightarrow sb^k \geq n_0 = n \\
\Rightarrow \frac{n}{s} \leq b^k
\end{aligned}$$

$$\therefore\ b^k = \Omega(n) \tag{6}$$

Again from (5),

$$\begin{aligned}
s < n_{k-1} = \left\lceil \frac{n_{k-2}}{b} \right\rceil < \frac{n_{k-2}}{b} + 1 \\
\Rightarrow sb < n_{k-2} + b = \left\lceil \frac{n_{k-3}}{b} \right\rceil + b < \frac{n_{k-3}}{b} + 1 + b \\
\Rightarrow sb^2 < n_{k-3} + b + b^2 \\
\vdots \\
\Rightarrow sb^{k-1} < n_0 + b + \cdots + b^{k-1} = n + b\frac{b^{k-1} - 1}{b - 1} \\
\Rightarrow \left(s - \frac{b}{b-1}\right) b^{k-1} < n - \frac{b}{b-1} \\
\Rightarrow b^k < \frac{b}{s - \frac{b}{b-1}} \left(n - \frac{b}{b-1}\right)
\end{aligned}$$

$$\therefore b^k = O(n) \tag{7}$$

From (6), (7)

$$b^k = \Theta(n) \tag{8}$$

or for some constant c_1 and c_2,

$$\begin{aligned}
c_1 n \leq b^k \leq c_2 n \\
\Rightarrow \log_b c_1 + \log_b n \leq k \leq \log_b c_2 + \log_b n
\end{aligned}$$

Here, note that

$$k = \Theta(\lg n) \tag{9}$$

and keep continue to process the inequality.

$$\begin{aligned}
\log_b c_1 + \log_b n \leq k \leq \log_b c_2 + \log_b n \\
\Rightarrow a^{\log_b c_1 + \log_b n} \leq a^k \leq a^{\log_b c_2 + \log_b n} \\
\Rightarrow a^{\log_b c_1} n^{\log_b a} \leq a^k \leq a^{\log_b c_2} n^{\log_b a}
\end{aligned}$$

$$\therefore\ a^k = \Theta\left(n^{\log_b a}\right) \tag{10}$$

By (3) and (10), (4) becomes

$$T(n) = a^k = \Theta\left(n^{\log_b a}\right) + g(n) \tag{11}$$

Here, we distinguish $g(n)$ into three cases.

1. $f(n) = O\left(n^{\log_b a-\varepsilon}\right)$ for some constant $\varepsilon > 0$
2. $f(n) = \Theta\left(n^{\log_b a}\right)$
3. $f(n) = \Omega\left(n^{\log_b a+\varepsilon}\right)$ for some constant $\varepsilon > 0$, and if $af(\lceil n/b\rceil) \le cf(n)$ for some constant $c < 1$ and all sufficiently large n

2.1.1 Case 1

Note that

$$\begin{aligned} n_r^{\log_b a-\varepsilon} &= \left\lceil \frac{n_{r-1}}{b} \right\rceil^{\log_b a-\varepsilon} \\ &\le \left(\frac{n_{r-1}}{b} + 1\right)^{\log_b a-\varepsilon} \\ &\le \left(\frac{1}{b}\right)^{\log_b a-\varepsilon} (n_{r-1} + b)^{\log_b a-\varepsilon} \end{aligned}$$

There is a constant $c_{r-1} > 0$ such that

$$n_r^{\log_b a-\varepsilon} \le \left(\frac{1}{b}\right)^{\log_b a-\varepsilon} c_{r-1} n_{r-1}^{\log_b a-\varepsilon} \quad \text{for all sufficiently large } n$$

Applying this action recursively, there is a constant $c_0 > 0$ such that

$$n_r^{\log_b a-\varepsilon} \le \left(\frac{1}{b}\right)^{r(\log_b a-\varepsilon)} c_0 n_0^{\log_b a-\varepsilon} = \left(\frac{b^\varepsilon}{a}\right)^r c_0 n^{\log_b a-\varepsilon} \tag{12}$$

for all sufficiently large n. From the case assumption,

$$f(n_r) = O\left(n_r^{\log_b a-\varepsilon}\right)$$

There is a constant $c > 0$ such that

$$f(n_r) \le c n_r^{\log_b a-\varepsilon}$$

for all sufficiently large n. Using (31),

$$\begin{aligned} & f(n_r) \le \left(\frac{b^\varepsilon}{a}\right)^r c c_0 n^{\log_b a-\varepsilon} \\ \Rightarrow\ & a^r f(n_r) \le c c_0 b^{\varepsilon r} n^{\log_b a-\varepsilon} \\ \Rightarrow\ & \sum_{r=0}^{k-1} a^r f(n_r) \le \sum_{r=0}^{k-1} c c_0 b^{\varepsilon r} n^{\log_b a-\varepsilon} \\ \Rightarrow\ & g(n) \le \sum_{r=0}^{k-1} c c_0 b^{\varepsilon r} n^{\log_b a-\varepsilon} \end{aligned} \tag{13}$$

Since $b > 1$ and $\varepsilon > 0$,

$$g(n) \le cc_0 \frac{b^{\varepsilon k} - 1}{b^{\varepsilon} - 1} n^{\log_b a - \varepsilon}$$

Using (8),

$$g(n) \le \Theta(n^{\varepsilon}) n^{\log_b a - \varepsilon} = \Theta(n^{\varepsilon}) n^{\log_b a - \varepsilon} = \Theta\left(n^{\log_b a}\right)$$

$$\therefore\ g(n) = O\left(n^{\log_b a}\right)$$

From (11),

$$T(n) = \Theta\left(n^{\log_b a}\right) + O\left(n^{\log_b a}\right) = \Theta\left(n^{\log_b a}\right)$$

2.1.2 Case 2

Comparing to *Case 1*, the only difference is ε, and there is no necessary condition during the process of *Case 1* until (13). Hence, by putting $\varepsilon = 0$, (13) becomes

$$g(n) \le \sum_{r=0}^{k-1} cc_0 n^{\log_b a} = cc_0 k n^{\log_b a}$$

By (9),

$$g(n) \le \Theta(\lg n) n^{\log_b a} = \Theta\left(n^{\log_b a} \lg n\right)$$

$$\therefore\ g(n) = O\left(n^{\log_b a} \lg n\right) \tag{14}$$

Meanwhile,

$$\begin{aligned} n_r^{\log_b a} &= \left\lceil \frac{n_{r-1}}{b} \right\rceil^{\log_b a} \\ &\ge \left(\frac{n_{r-1}}{b}\right)^{\log_b a} \\ &= \left(\frac{1}{b}\right)^{\log_b a} n_{r-1}^{\log_b a} \\ &= \frac{1}{a} n_{r-1}^{\log_b a} \end{aligned}$$

Repeating recursively,

$$\begin{aligned} & n_r^{\log_b a} \ge \left(\frac{1}{a}\right)^r n^{\log_b a} \\ \Rightarrow\ & a^r n_r^{\log_b a} \ge n^{\log_b a} \\ \Rightarrow\ & \sum_{r=0}^{k-1} a^r n_r^{\log_b a} \ge \sum_{r=0}^{k-1} n^{\log_b a} = k n^{\log_b a} \\ \Rightarrow\ & g(n) \ge k n^{\log_b a} \end{aligned}$$

Using (9),

$$g\left(n\right) \geq \Theta\left(\lg n\right) n^{\log_b a} = \Theta\left(n^{\log_b a} \lg n\right)$$

$$g\left(n\right) = \Omega\left(n^{\log_b a} \lg n\right) \tag{15}$$

From (14) and (15),

$$g\left(n\right) = \Theta\left(n^{\log_b a} \lg n\right)$$

Consequently, (11) becomes

$$T\left(n\right) = \Theta\left(n^{\log_b a}\right) + \Theta\left(n^{\log_b a} \lg n\right) = \Theta\left(n^{\log_b a} \lg n\right)$$

2.1.3 Case 3

Since $f\left(n\right) = \Omega\left(n^{\log_b a+\varepsilon}\right)$, from (11),

$$\begin{aligned} T\left(n\right) &= \Theta\left(n^{\log_b a}\right) + \sum_{r=0}^{k-1} a^r f\left(n_r\right) \\ &= \Theta\left(n^{\log_b a}\right) + f\left(n\right) \\ \therefore\ T\left(n\right) &= \Omega\left\{f\left(n\right)\right\} \end{aligned} \tag{16}$$

Back to the summation at the early stage, apply $af\left(\lceil n/b \rceil\right) \leq cf\left(n\right)$.

$$\begin{array}{rll} T\left(n\right) &= aT\left(n_1\right) + f\left(n\right) & \leq aT\left(n_1\right) + f\left(n\right) \\ aT\left(n_1\right) &= a^2 T\left(n_2\right) + af\left(n_1\right) & \leq a^2 T\left(n_2\right) + cf\left(n\right) \\ & \vdots & \vdots \\ a^{k-1} T\left(n_{k-1}\right) &= a^k T\left(n_k\right) + a^{k-1} f\left(n_{k-1}\right) & \leq a^k T\left(n_k\right) + c^{k-1} f\left(n\right) \\ \hline T\left(n\right) &= & \leq a^k T\left(n_k\right) + \sum_{r=0}^{k-1} c^r f\left(n\right) \end{array}$$

From (10),

$$T\left(n\right) \leq \Theta\left(n^{\log_b a}\right) + \sum_{r=0}^{k-1} c^r f\left(n\right)$$

Since $c < 1$, $\sum_{r=0}^{\infty} c^r < \infty$. Thus,

$$\begin{aligned} T\left(n\right) &\leq \Theta\left(n^{\log_b a}\right) + \frac{1}{1-c} f\left(n\right) \\ &\leq \Theta\left(n^{\log_b a}\right) + \Theta\left\{f\left(n\right)\right\} \\ &\leq \Theta\left\{f\left(n\right)\right\} && \text{since } f\left(n\right) = \Omega\left(n^{\log_b a+\varepsilon}\right) \end{aligned}$$

$$\therefore\ T(n) = O\{f(n)\} \tag{17}$$

From (16) and (17),

$$\therefore\ T(n) = \Theta\{f(n)\}$$

2.2 lower bound of T

The lower bound is the solution of

$$T(n) = aT(\lfloor n/b \rfloor) + f(n) \tag{18}$$

and is solvable in a similar manner with that of upper bound. Assume

$$T(n) = \Theta(1) \quad \text{if } n \leq s \tag{19}$$

where s is some constant. Let

$$n_i \stackrel{\text{def}}{=} \begin{cases} n & i = 0 \\ \lfloor n_{i-1}/b \rfloor & i = 1, 2, \cdots \end{cases}$$

Then, (18) is expressed as

$$T(n) = aT(n_1) + f(n)$$

By replacing n with n_1, and multiplying a on the both hand sides, we get

$$aT(n_1) = a^2 T(n_2) + af(n_1)$$

Again, do the same action on the above equation.

$$a^2 T(n_2) = a^3 T(n_3) + a^2 f(n_2)$$

In this manner, we get the series of equations. Add them up all.

$$\begin{aligned} T(n) &= aT(n_1) + f(n) \\ aT(n_1) &= a^2 T(n_2) + af(n_1) \\ &\vdots \\ a^{k-1} T(n_{k-1}) &= a^k T(n_k) + a^{k-1} f(n_{k-1}) \\ \hline T(n) &= a^k T(n_k) + g(n) \end{aligned} \tag{20}$$

where

$$n_k \leq s < n_{k-1}, \text{ and } g(n) = \sum_{r=0}^{k-1} a^r f(n_r) \tag{21}$$

To resolve k, let's take a look at (21)

$$\begin{aligned} & s \geq n_k = \left\lfloor \frac{n_{k-1}}{b} \right\rfloor > \frac{n_{k-1}}{b} - 1 \\ \Rightarrow\ & sb > n_{k-1} - b \\ \Rightarrow\ & sb > \frac{n_{k-2}}{b} - 1 - b \\ \Rightarrow\ & sb^2 > n_{k-2} - b - b^2 \\ & \vdots \\ \Rightarrow\ & sb^k > n_0 - b - \cdots - b^k = n - b\frac{b^k - 1}{b - 1} \\ \Rightarrow\ & \left(s + \frac{b}{b-1}\right) b^k > n + \frac{b}{b-1} \\ \Rightarrow\ & b^k > \frac{1}{s + \frac{b}{b-1}} \left(n + \frac{b}{b-1}\right) \end{aligned}$$

$$\therefore\ b^k = \Omega(n) \tag{22}$$

Again from (21),

$$\begin{aligned} & s < n_{k-1} = \left\lfloor \frac{n_{k-2}}{b} \right\rfloor \leq \frac{n_{k-2}}{b} \\ \Rightarrow\ & sb \leq n_{k-2} \\ & \vdots \\ \Rightarrow\ & sb^{k-1} \leq n_0 = n \\ \Rightarrow\ & b^k \leq \frac{b}{s} n \end{aligned}$$

$$\therefore b^k = O(n) \tag{23}$$

From (22), (23)

$$b^k = \Theta(n) \tag{24}$$

or for some constant c_1 and c_2,

$$\begin{aligned} & c_1 n \leq b^k \leq c_2 n \\ \Rightarrow\ & \log_b c_1 + \log_b n \leq k \leq \log_b c_2 + \log_b n \end{aligned}$$

Here, note that

$$k = \Theta(\lg n) \tag{25}$$

and keep continue to process the inequality.

$$\begin{aligned} & \log_b c_1 + \log_b n \leq k \leq \log_b c_2 + \log_b n \\ \Rightarrow\ & a^{\log_b c_1 + \log_b n} \leq a^k \leq a^{\log_b c_2 + \log_b n} \\ \Rightarrow\ & a^{\log_b c_1} n^{\log_b a} \leq a^k \leq a^{\log_b c_2} n^{\log_b a} \end{aligned}$$

$$\therefore\ a^k = \Theta\left(n^{\log_b a}\right) \tag{26}$$

By (19) and (26), (20) becomes

$$T(n) = a^k = \Theta\left(n^{\log_b a}\right) + g(n) \tag{27}$$

Here, we distinguish $g(n)$ into three cases.

1. $f(n) = O\left(n^{\log_b a-\varepsilon}\right)$ for some constant $\varepsilon > 0$
2. $f(n) = \Theta\left(n^{\log_b a}\right)$
3. $f(n) = \Omega\left(n^{\log_b a+\varepsilon}\right)$ for some constant $\varepsilon > 0$, and if $af(\lfloor n/b \rfloor) \le cf(n)$ for some constant $c < 1$ and all sufficiently large n

2.2.1 Case 1

Note that

$$\begin{aligned} n_r^{\log_b a-\varepsilon} &= \left\lfloor \frac{n_{r-1}}{b} \right\rfloor^{\log_b a-\varepsilon} \\ &\le \left(\frac{n_{r-1}}{b}\right)^{\log_b a-\varepsilon} \\ &\le \left(\frac{1}{b}\right)^{\log_b a-\varepsilon} n_{r-1}^{\log_b a-\varepsilon} \end{aligned}$$

Applying this action recursively,

$$n_r^{\log_b a-\varepsilon} \le \left(\frac{1}{b}\right)^{r(\log_b a-\varepsilon)} n_0^{\log_b a-\varepsilon} = \left(\frac{b^{\varepsilon}}{a}\right)^r n^{\log_b a-\varepsilon} \tag{28}$$

From the case assumption,

$$f(n_r) = O\left(n_r^{\log_b a-\varepsilon}\right)$$

There is a constant $c > 0$ such that

$$f(n_r) \le cn_r^{\log_b a-\varepsilon}$$

for all sufficiently large n. Using (28),

$$\begin{aligned} & f(n_r) \le \left(\frac{b^{\varepsilon}}{a}\right)^r cn^{\log_b a-\varepsilon} \\ \Rightarrow\ & a^r f(n_r) \le cb^{\varepsilon r} n^{\log_b a-\varepsilon} \\ \Rightarrow\ & \sum_{r=0}^{k-1} a^r f(n_r) \le \sum_{r=0}^{k-1} cb^{\varepsilon r} n^{\log_b a-\varepsilon} \\ \Rightarrow\ & g(n) \le \sum_{r=0}^{k-1} cb^{\varepsilon r} n^{\log_b a-\varepsilon} \end{aligned} \tag{29}$$

Since $b > 1$ and $\varepsilon > 0$,

$$g(n) \le c\frac{b^{\varepsilon k} - 1}{b^{\varepsilon} - 1} n^{\log_b a - \varepsilon}$$

Using (24),

$$g(n) \le \Theta(n^{\varepsilon}) n^{\log_b a - \varepsilon} = \Theta(n^{\varepsilon}) n^{\log_b a - \varepsilon} = \Theta\left(n^{\log_b a}\right)$$

$$\therefore \; g(n) = O\left(n^{\log_b a}\right)$$

From (27),

$$T(n) = \Theta\left(n^{\log_b a}\right) + O\left(n^{\log_b a}\right) = \Theta\left(n^{\log_b a}\right)$$

2.2.2 Case 2

Comparing to *Case 1*, the only difference is ε, and there is no necessary condition during the process of *Case 1* until (29). Hence, by putting $\varepsilon = 0$, (29) becomes

$$g(n) \le \sum_{r=0}^{k-1} c n^{\log_b a} = ckn^{\log_b a}$$

By (25),

$$g(n) \le \Theta(\lg n) n^{\log_b a} = \Theta\left(n^{\log_b a} \lg n\right)$$

$$\therefore \; g(n) = O\left(n^{\log_b a} \lg n\right) \tag{30}$$

Meanwhile,

$$\begin{aligned} n_r^{\log_b a} &= \left\lfloor \frac{n_{r-1}}{b} \right\rfloor^{\log_b a} \\ &> \left(\frac{n_{r-1}}{b} - 1\right)^{\log_b a} \\ &= \left(\frac{1}{b}\right)^{\log_b a} (n_{r-1} - b)^{\log_b a} \\ &= \frac{1}{a} (n_{r-1} - b)^{\log_b a} \end{aligned}$$

There is a constant $c_{r-1} > 0$ such that

$$n_r^{\log_b a} > \frac{1}{a} c_{r-1} n_{r-1}^{\log_b a}$$

for all sufficiently large n. Applying this action recursively, there is a constant $c_0 > 0$ such that

$$n_r^{\log_b a} > \left(\frac{1}{a}\right)^r c_0 n_0^{\log_b a} = \left(\frac{1}{a}\right)^r c_0 n^{\log_b a} \tag{31}$$

for all sufficiently large n. From the case assumption,

$$f(n_r) = \Theta\left(n_r^{\log_b a}\right)$$

There is a constant c such that

$$f(n_r) \geq c n_r^{\log_b a}$$

for all sufficiently large n. Using (31),

$$\begin{aligned} f(n_r) &> \left(\frac{1}{a}\right)^r c c_0 n^{\log_b a} \\ \Rightarrow a^r f(n_r) &> c c_0 n^{\log_b a} \\ \Rightarrow \sum_{r=0}^{k-1} a^r f(n_r) &> \sum_{r=0}^{k-1} c c_0 n^{\log_b a} \\ \Rightarrow g(n) &> c c_0 k n^{\log_b a} \end{aligned}$$

Using (25),

$$g(n) > \Theta(\lg n) n^{\log_b a} = \Theta\left(n^{\log_b a} \lg n\right)$$

$$g(n) = \Omega\left(n^{\log_b a} \lg n\right) \tag{32}$$

From (30) and (32),

$$g(n) = \Theta\left(n^{\log_b a} \lg n\right)$$

Consequently, (27) becomes

$$T(n) = \Theta\left(n^{\log_b a}\right) + \Theta\left(n^{\log_b a} \lg n\right) = \Theta\left(n^{\log_b a} \lg n\right)$$

2.2.3 Case 3

Since $f(n) = \Omega\left(n^{\log_b a + \varepsilon}\right)$, from (27),

$$\begin{aligned} T(n) &= \Theta\left(n^{\log_b a}\right) + \sum_{r=0}^{k-1} a^r f(n_r) \\ &= \Theta\left(n^{\log_b a}\right) + f(n) \end{aligned}$$

$$\therefore\ T(n) = \Omega\{f(n)\} \tag{33}$$

Back to the summation at the early stage, apply $af(\lfloor n/b \rfloor) \leq cf(n)$.

$$
\begin{aligned}
T(n) &= aT(n_1) + f(n) && \leq aT(n_1) + f(n) \\
aT(n_1) &= a^2T(n_2) + af(n_1) && \leq a^2T(n_2) + cf(n) \\
&\vdots && \vdots \\
a^{k-1}T(n_{k-1}) &= a^kT(n_k) + a^{k-1}f(n_{k-1}) && \leq a^kT(n_k) + c^{k-1}f(n) \\
\hline
T(n) &= && \leq a^kT(n_k) + \sum_{r=0}^{k-1} c^r f(n)
\end{aligned}
$$

From (26),

$$T(n) \leq \Theta\left(n^{\log_b a}\right) + \sum_{r=0}^{k-1} c^r f(n)$$

Since $c < 1$, $\sum_{r=0}^{\infty} c^r < \infty$. Thus,

$$
\begin{aligned}
T(n) &\leq \Theta\left(n^{\log_b a}\right) + \frac{1}{1-c} f(n) \\
&\leq \Theta\left(n^{\log_b a}\right) + \Theta\{f(n)\} \\
&\leq \Theta\{f(n)\} && \text{since } f(n) = \Omega\left(n^{\log_b a + \varepsilon}\right)
\end{aligned}
$$

$$\therefore\ T(n) = O\{f(n)\} \tag{34}$$

From (33) and (34),

$$\therefore\ T(n) = \Theta\{f(n)\}$$

2.3 Sandwich

We've checked the upper and lower bounds are coincident, hence the solutions of each cases are valid for all n.

3 Rationale

For the *Case 3* at Sec (2.1) and Sec (2.2), one condition is differ from the other one.

1. $af(\lceil n/b \rceil) \leq cf(n)$
2. $af(\lfloor n/b \rfloor) \leq cf(n)$

I setup both assumptions to deduce the result. However, given $f(n) = \Omega\left(n^{\log_b a + \varepsilon}\right)$, $af(n/b) \leq cf(n)$ doesn't imply $af(\lceil n/b \rceil) \leq cf(n)$ and $af(\lfloor n/b \rfloor) \leq cf(n)$ generally. This is a hole in this paper.

References

[THC09] Ronald L. Rivest Thomas H. Cormen, Charles E. Leiserson. *Introduction to Algorithms*, page 94. The MIT Press, 3rd edition, 2009.