CSE 201 1-26-21

Master Method

finds asymptotic solutions to

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

where : $a \geq 1$, $b > 1$, $f(n)$ is a.p. . Also $\frac{n}{b}$ denotes either $\lfloor \frac{n}{b} \rfloor$ or $\lceil \frac{n}{b} \rceil$, note the asymp. soln. doesn't depend on whether its $\lfloor \cdot \rfloor$ or $\lceil \cdot \rceil$. Also asymp. soln. doesn't depend on initial values.

$$T(n) = \begin{cases} c & n = \dots \\ aT(\underbrace{\lfloor \tfrac{n}{b} \rfloor}_{\text{or } \lceil \frac{n}{b} \rceil}) + f(n) & n \geq \dots \end{cases}$$

Master Theorem

Let $T(n) = aT(\frac{n}{b}) + f(n)$ where $a \geq 1$, $b > 1$ and $f(n)$ is a.p. Then

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

then $\boxed{T(n) = \Theta(n^{\log_b(a)})}$.

(2) If $f(n) = \Theta(n^{\log_b(a)})$ then

$$\boxed{T(n) = \Theta(n^{\log_b a} \cdot \log n) = \Theta(f(n) \cdot \log n)}$$

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

and $\boxed{\text{if } a f(n/b) \leq c f(n) \text{ for some } 0 < c < 1 \text{ and for all suff. large } n}$, then

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

regularity condition

Rmks

In all cases we compare

$f(n)$ to $n^{\log_b(a)}$

Case 1: $n^{\log_b(a)}$ 'wins' by polynomial ~~factor~~ n^{ε}.

case 2: tie.

case 3: $f(n)$ 'wins' by poly ~~factor~~ n^{ε}

Ex. $T(n) = 8T\left(\frac{n}{2}\right) + n^3$

compare: n^3 to $n^{\log_2(8)} = n^3$.

case 2: $T(n) = \Theta\left(n^3 \log n\right)$

Ex. $T(n) = 5T(\frac{n}{4}) + n$

compare: n to $\underline{n^{\log_4(5)}}$

Pick: $\varepsilon = \log_4(5) - 1$. Then

$1 = \log_4 5 - \varepsilon$, and

$$n = n^1 = O(n^1) = O(n^{\log_4(5) - \varepsilon})$$

so by case 1:

$$T(n) = \Theta(n^{\log_4(5)}).$$

Ex. $T(n) = 5T(\frac{n}{4}) + n^2$

comp. $\underline{n^2}$ to $n^{\log_4(5)}$

let $\varepsilon = 2 - \log_4(5)$. Then $2 = \log_4(5) + \varepsilon$,

$$n^2 = \Omega(n^2) = \Omega(n^{\log_4(5) + \varepsilon})$$

we must find c s.t. $0 < c < 1$ and

$$5\left(\frac{n}{4}\right)^2 \le c n^2$$

for all suff. large n.

$$\frac{5}{16} \cancel{n^2} \le c \cancel{n^2}$$

$$\frac{5}{16} \le c < 1 \leftarrow \text{Pick any } c \text{ in this range.}$$

note: such c exist since $\frac{5}{16} < 1$

i.e. $5 < 16$. $\therefore$ reg. cond. holds.

$\therefore$ by Case 3:

$$T(n) = \Theta(n^2)$$

<u>Exercise</u>

Prove that if $f(n) = \Theta(n^d)$ and if $d > \log_b(a)$, then Case 3 applies and reg. cond. necessarily holds.

Ex. $f(n)$ not Polynomial

$$T(n) = \begin{cases} 6 & \text{if } n = 1 \\ 11 & \text{if } n = 2 \\ T(\lfloor \frac{n}{2} \rfloor) + 2T(\lceil \frac{n}{2} \rceil) + \log(n!) & \text{if } n \geq 3 \end{cases}$$

simplify: $T(n) = 3T(\frac{n}{2}) + \underbrace{\log(n!)}_{\Theta(n \log n)}$

simplify: $T(n) = 3T(\frac{n}{2}) + n \log n$

comp: $n \log n$ to $\underline{n^{\log_2(3)}}$

let $\varepsilon = \frac{1}{2}(\log_2(3) - 1) > 0$. Then

$2\varepsilon = \log_2(3) - 1$, $\therefore$ $1 + \varepsilon = \log_2(3) - \varepsilon$.

$$\frac{n \log n}{n^{\log_2(3) - \varepsilon}} = \frac{n \log n}{n^{1+\varepsilon}} = \frac{\log n}{n^{\varepsilon}} \longrightarrow 0$$

$\therefore \; n\log n = o\left(n^{\log_2(3)-\varepsilon}\right) \subseteq O\left(n^{\log_2(3)-\varepsilon}\right)$

$\therefore$ By Case 1: $T(n) = \Theta\left(n^{\log_2(3)}\right)$.

Remark: $\varepsilon = \log_2(3) - 1$ does not work in this example! why?

$$\frac{n\log n}{n^{\log_2 3 - \varepsilon}} = \frac{\cancel{n}\log n}{\cancel{n}} = \log n \longrightarrow \infty$$

$\therefore \; n\log n = \omega\left(n^{\log_2 3 - \varepsilon}\right)$

$\therefore \; n\log n \neq O\left(n^{\log_2 3 - \varepsilon}\right)$ for this ε.

Ex. $T(n) = 2T\left(\frac{n}{2}\right) + \frac{n}{\log n}$

Comp: $\frac{n}{\log n}$ to $\underbrace{n^{\log_2(2)} = n}_{\text{winner}}$

observe

$$\frac{n}{\log n} = \omega(n^{1-\varepsilon}) \text{ for any } \varepsilon > 0$$

since

$$\frac{\cancel{n}/\log n}{\cancel{n}/n^{\varepsilon}} = \frac{n^{\varepsilon}}{\log n} \longrightarrow \infty$$

and $\omega(n^{1-\varepsilon}) \cap O(n^{1-\varepsilon}) = \phi$.

$\therefore \frac{n}{\log n} \neq O(n^{1-\varepsilon})$ for any $\varepsilon > 0$.

$\therefore$ case 1 never applies.

$\therefore$ Master Thearm does not apply.

Exercise

find an example where $f(n)$ wins over $n^{\log_b(a)}$, but not by a poly. factor.

Exercise (hard)

find an example where

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

for some $\varepsilon > 0$ (so would be in case 3), but regularity cond. fails.

Iteration method

Ex.

$$T(n) = \begin{cases} 1 & 1 \le n < 3 \\ 2T\left(\left\lfloor \frac{n}{3} \right\rfloor\right) + n & n \ge 3 \end{cases}$$

re-write:

$$T(\text{☺}) = \text{☺} + 2T\left(\left\lfloor \frac{\text{☺}}{3} \right\rfloor\right)$$

$$\begin{aligned}
T(n) &= n + 2T\left(\left\lfloor \frac{n}{3} \right\rfloor\right) \\
&= n + 2\left(\left\lfloor \frac{n}{3} \right\rfloor + 2T\left(\left\lfloor \frac{\lfloor n/3 \rfloor}{3} \right\rfloor\right)\right) \\
&= n + 2\left\lfloor \frac{n}{3} \right\rfloor + 2^2 T\left(\left\lfloor \frac{n}{3^2} \right\rfloor\right) \\
&= n + 2\left\lfloor \frac{n}{3} \right\rfloor + 2^2\left(\left\lfloor \frac{n}{3^2} \right\rfloor + 2T\left(\left\lfloor \frac{\lfloor n/3^2 \rfloor}{3} \right\rfloor\right)\right) \\
&= n + 2\left\lfloor \frac{n}{3} \right\rfloor + 2^2\left\lfloor \frac{n}{3^2} \right\rfloor + 2^3 T\left(\left\lfloor \frac{n}{3^3} \right\rfloor\right) \\
&\;\vdots \\
&= \sum_{i=0}^{k-1} 2^i \left\lfloor \frac{n}{3^i} \right\rfloor + 2^k T\left(\left\lfloor \frac{n}{3^k} \right\rfloor\right)
\end{aligned}$$

recursion halts when

$$1 \le \left\lfloor \frac{n}{3^k} \right\rfloor < 3$$

$$\therefore \quad 1 \le \frac{n}{3^k} < 3$$

$$\therefore \quad 3^k \le n < 3^{k+1}$$

$$\therefore \quad k \le \log_3(n) < k+1$$

$$\therefore \quad \boxed{k = \lfloor \log_3(n) \rfloor}$$

depth at which rec. 'bottoms out'

For this k, $T\left(\left\lfloor \frac{n}{3^k} \right\rfloor\right) = 1$. Thus

$$\boxed{T(n) = \sum_{i=0}^{\lfloor \log_3 n \rfloor - 1} 2^i \left\lfloor \frac{n}{3^i} \right\rfloor + 2^{\lfloor \log_3 n \rfloor}}$$

iterative algorithm for computing $T(n)$.

asym. upper bound.

$$T(n) \le \sum_{i=0}^{k-1} 2^i \left(\frac{n}{3^i}\right) + 2^{\log_3 n}$$

$$\le n \cdot \sum_{i=0}^{\infty} \left(\frac{2}{3}\right)^i + n^{\log_3 2}$$

$$= n \left(\frac{1}{1-\frac{2}{3}}\right) + n^{\log_3 2}$$

$$= 3n + n^{\log_3 2}$$

$$= O(n)$$

$\therefore \quad T(n) = O(n).$