Theorem (Solving divide-and-conquer recurrences)

Let

$$T(n) := a\,T\left(\left[\tfrac{n}{b}\right]\right) + f(n),$$

where $a \geq 1$, $b > 1$ are constants, n is a non-negative integer, and $\left[\frac{n}{b}\right]$ denotes either $\left\lfloor\frac{n}{b}\right\rfloor$ or $\left\lceil\frac{n}{b}\right\rceil$.

Then

$$T(n) = \begin{cases} \Theta\left(n^{\log_b a}\right), & f(n) = O\left(n^{(\log_b a) - \epsilon}\right) \text{ for some } \epsilon > 0; \\ \Theta\left(n^{\log_b a} \log^{k+1} n\right), & f(n) = \Theta\left(n^{\log_b a} \log^k n\right) \text{ for some } k \geq 0; \\ \Theta\left(f(n)\right), & f(n) = \Omega\left(n^{(\log_b a) + \epsilon}\right) \text{ for some } \epsilon > 0, \text{ if } a\,f\left(\frac{n}{b}\right) < f(n) \text{ for all large } n. \end{cases}$$

$\square$

Theorem (Specialization for polynomial $f(n)$)

Let

$$T(n) := a\,T\left(\left[\tfrac{n}{b}\right]\right) + \Theta(n^c),$$

where $a, b, \left[\tfrac{n}{b}\right]$ are as before and c is a constant.

Then

$$T(n) = \begin{cases} \Theta\left(n^{\log_b a}\right), & c < \log_b a\,; \\ \Theta\left(n^c \log n\right), & c = \log_b a\,; \\ \Theta\left(n^c\right), & c > \log_b a\,. \end{cases}$$

$\square$

Iteration method for solving recurrences

- Keep expanding the recurrence until reaching the boundary condition.

Example

$$T(n) := n + 3\,T\left(\left\lfloor \frac{n}{4} \right\rfloor\right)$$

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

Expanding once.

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

since $\left\lfloor \frac{\lfloor \frac{n}{a} \rfloor}{b} \right\rfloor = \left\lfloor \frac{n}{ab} \right\rfloor$ for pos. int.s a,b

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

Expanding twice.

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

$$\vdots$$

Expanding arbitrary number of times.

$$= \sum_{0 \le i < k} 3^i \left\lfloor \frac{n}{4^i} \right\rfloor + 3^k\,T\left(\left\lfloor \frac{n}{4^k} \right\rfloor\right).$$

Example cont'd

Suppose the boundary condition is $T(c) = \Theta(1)$.
Then we hit the boundary condition when k is equal to:

$$
\begin{aligned}
k^* &:= \min\left\{ k : \left\lfloor \frac{n}{4^k} \right\rfloor \le c \right\} \\
&= \min\left\{ k : \frac{n}{4^k} < c+1 \right\} \\
&= \min\left\{ k : 4^k > \frac{n}{c+1} \right\} \\
&= \min\left\{ k : k > \log_4 \frac{n}{c+1} \right\} \\
&= \left\lfloor \log_4 \frac{n}{c+1} \right\rfloor + 1 .
\end{aligned}
$$

Substituting this value k^* for k,

$$
\begin{aligned}
T(n) &= \sum_{0 \le i \le \left\lfloor \log_4 \frac{n}{c+1} \right\rfloor} 3^i \left\lfloor \frac{n}{4^i} \right\rfloor + 3^{\left\lfloor \log_4 \frac{n}{c+1} \right\rfloor + 1} \Theta(1) \\
&= \sum_{0 \le i \le \lfloor \log_4 n - \Theta(1) \rfloor} 3^i \left\lfloor \frac{n}{4^i} \right\rfloor + 3^{\lfloor \log_4 n \rfloor} \Theta(1)
\end{aligned}
$$

(*)

Example, cont'd

Upper bounding $T(n)$ using equation $(*)$,

$$T(n) \leq \sum_{0 \leq i \leq \log_4 n} 3^i \left(\frac{n}{4^i}\right) + 3^{\log_4 n} \, \Theta(1)$$

$$\leq n \sum_{i \geq 0} \left(\frac{3}{4}\right)^i + n^{\log_4 3} \, \Theta(1)$$

(since $x^{\log_b y} = y^{\log_b x}$)

$$= n \frac{1}{1 - \frac{3}{4}} + o(n) \, \Theta(1)$$

(since $\log_4 3 < 1$)

$$= n \, O(1) + o(n)$$

$$= O(n) + o(n)$$

$$= O(n).$$

Example, cont'd

Lower bounding $T(n)$ using equation $(*)$,

$$T(n) = \sum_{0 \le i \le \lfloor \log_4 n - \Theta(1) \rfloor} 3^i \left\lfloor \frac{n}{4^i} \right\rfloor + 3^{\lfloor \log_4 n \rfloor} \Theta(1)$$

$$\ge \sum_{0 \le i \le \lfloor \log_4 n - \Theta(1) \rfloor} 3^i \left(\frac{n}{4^i} - 1 \right)$$

$$(\text{since } \lfloor x \rfloor > x-1)$$

$$\ge \sum_{i=0} 3^i \left(\frac{n}{4^i} - 1 \right)$$

$$= n - 1$$

$$= \Omega(n).$$

Combining bounds,

$$T(n) = \Theta(n). \qquad \square$$

Notes

The key steps in the iteration method are:

(1) Iterate the recurrence k times.
Find the form of the resulting expression (usually a sum).

(2) Determine how many iterations k^* are needed to hit the boundary condition.
(This usually amounts to finding the smallest value of k s.t. in the expression $T(f(n,k))$ occurring in (1), $f(n,k) \leq c$ for some convenient constant c.)

(3) Substitute this value k^* for k, replace $T(f(n,k))$ by $\Theta(1)$, and bound the resulting summation.

$\square$