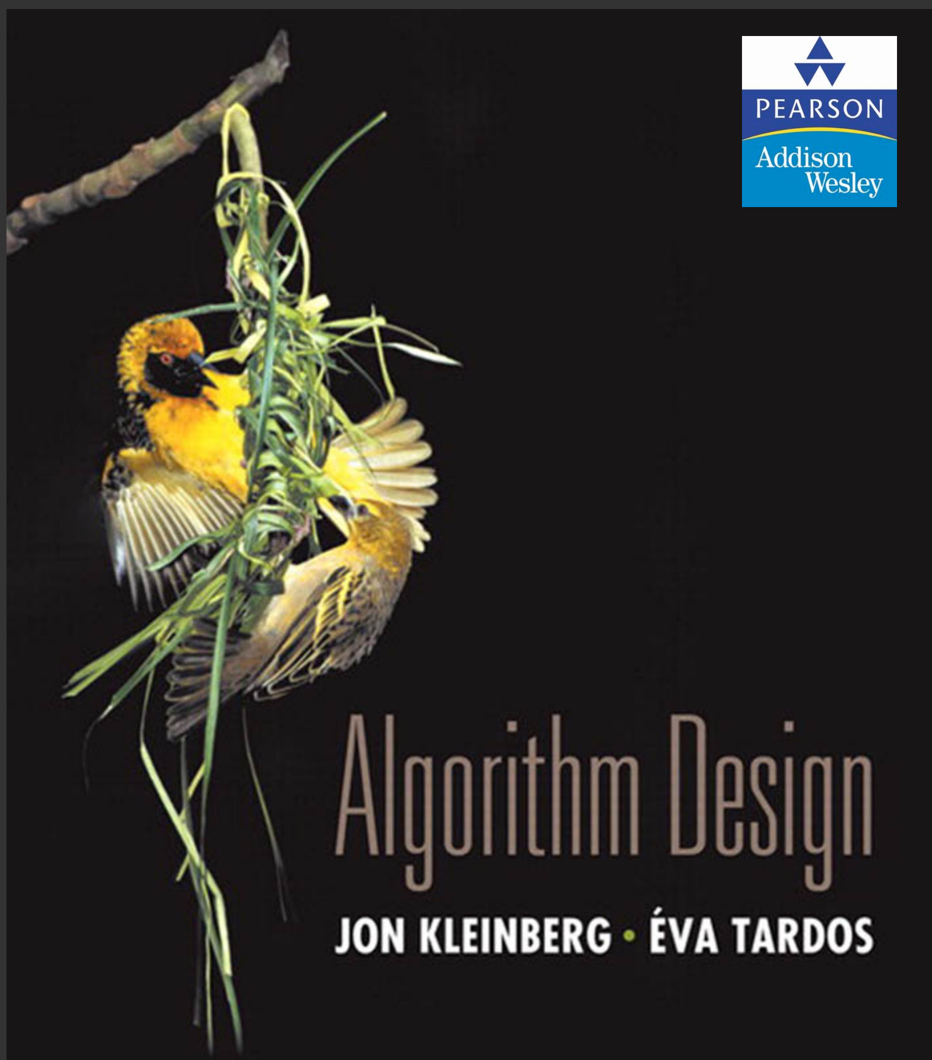


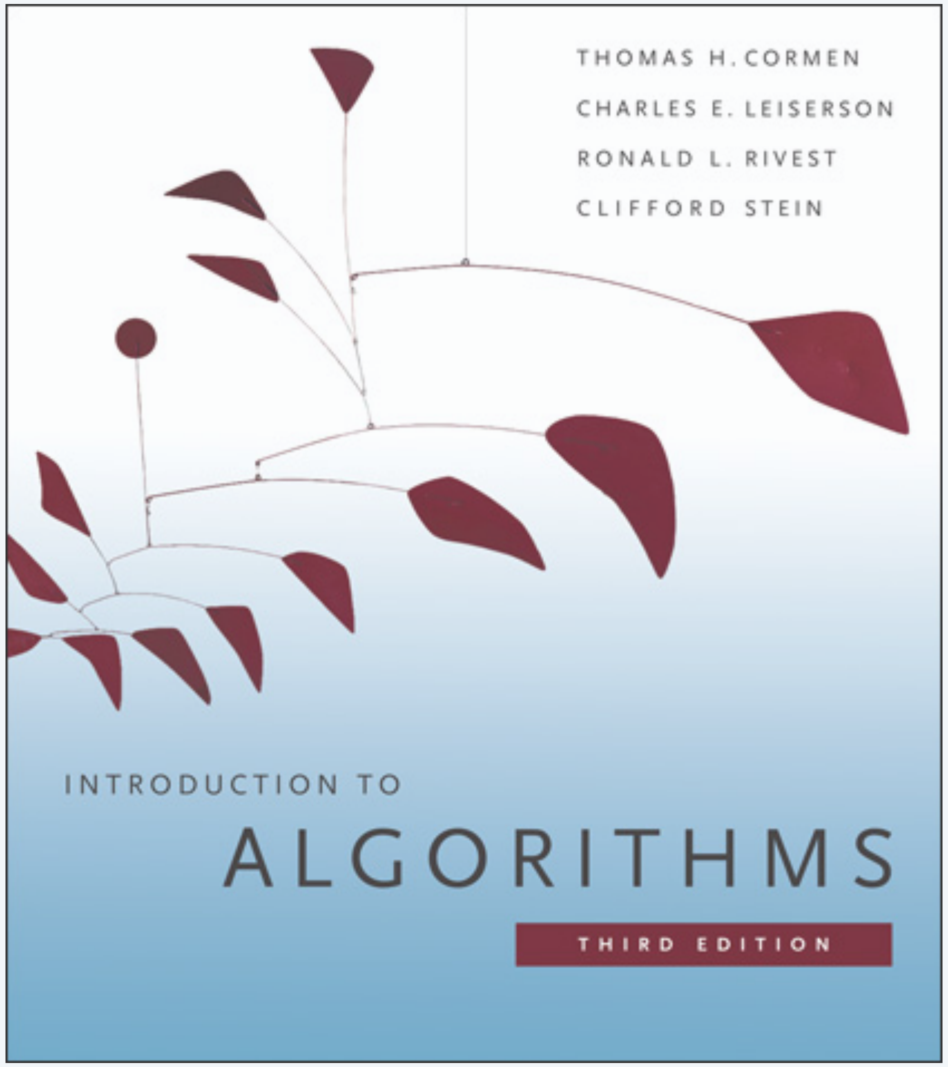
Lecture slides by Kevin Wayne

http://www.cs.princeton.edu/~wayne/kleinberg-tardos

Divide and Conquer II

- *master theorem*
- *integer multiplication*
- *matrix multiplication*
- *convolution and FFT*

Last updated on 2/6/21 8:29 PM



SECTIONS 4.4–4.6

DIVIDE AND CONQUER II

- *master theorem*
- *integer multiplication*
- *matrix multiplication*
- *convolution and FFT*

Divide-and-conquer recurrences

Goal. Recipe for solving common divide-and-conquer recurrences:

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

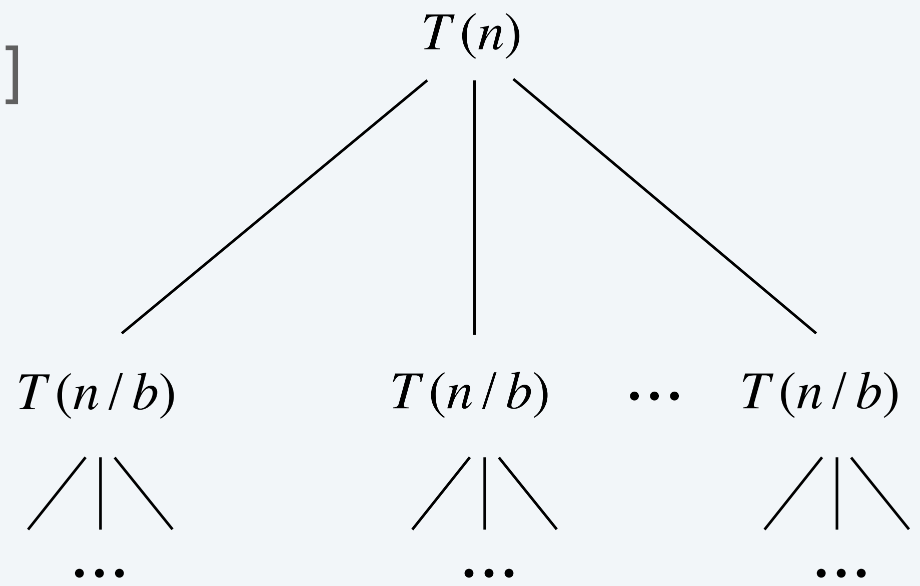
with $T(0) = 0$ and $T(1) = \Theta(1)$.

Terms.

- $a \ge 1$ is the number of subproblems.
- $b \ge 2$ is the factor by which the subproblem size decreases.
- $f(n) \ge 0$ is the work to divide and combine subproblems.

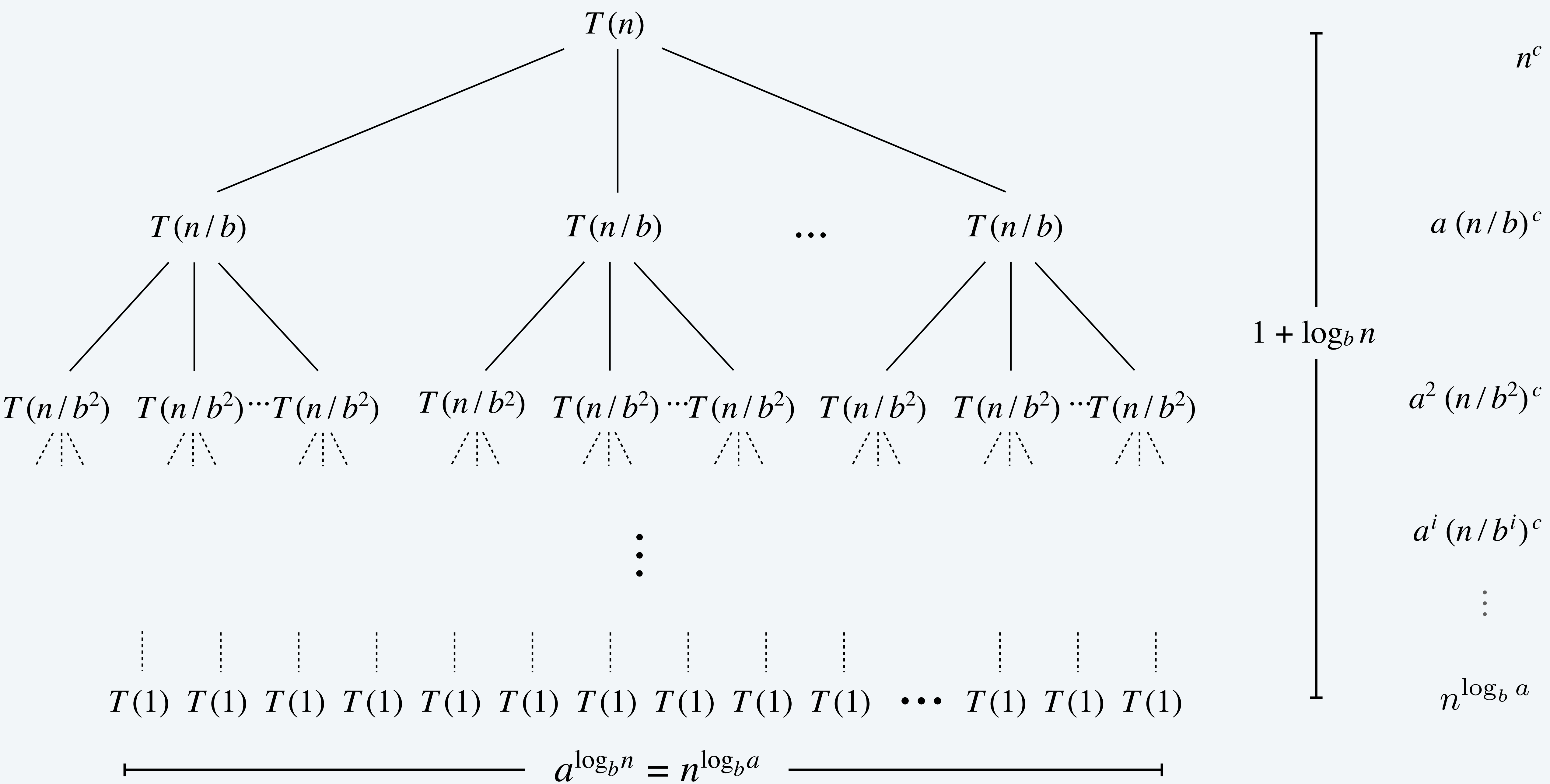
Recursion tree. [assuming n is a power of b]

- a = branching factor.
- a^i = number of subproblems at level i.
- $1 + \log_b n$ levels.
- n / b^i = size of subproblem at level i.



Divide-and-conquer recurrences: recursion tree

Suppose $T(n)$ satisfies $T(n) = a\,T(n/b) + n^c$ with $T(1) = 1$, for n a power of b.



$r = a / b^c \qquad T(n) \;=\; n^c \sum_{i=0}^{\log_b n} r^i$

Divide-and-conquer recurrences: recursion tree analysis

Suppose $T(n)$ satisfies $T(n) = a\,T(n / b) + n^c$ with $T(1) = 1$, for n a power of b.

Let $r = a / b^c$. Note that $r < 1$ iff $c > \log_b a$.

$$T(n) \;=\; n^c \sum_{i=0}^{\log_b n} r^i \;=\; \begin{cases} \Theta(n^c) & \text{if } r < 1 \\ \Theta(n^c \log n) & \text{if } r = 1 \\ \Theta(n^{\log_b a}) & \text{if } r > 1 \end{cases}$$

- $c > \log_b a$ ← cost dominated by cost of root
- $c = \log_b a$ ← cost evenly distributed in tree
- $c < \log_b a$ ← cost dominated by cost of leaves

Geometric series.

- If $0 < r < 1$, then $1 + r + r^2 + r^3 + \ldots + r^k \le 1 / (1 - r)$.
- If $r = 1$, then $1 + r + r^2 + r^3 + \ldots + r^k = k + 1$.
- If $r > 1$, then $1 + r + r^2 + r^3 + \ldots + r^k = (r^{k+1} - 1) / (r - 1)$.

Divide-and-conquer recurrences: master theorem

Master theorem. Let $a \geq 1$, $b \geq 2$, and $c \geq 0$ and suppose that $T(n)$ is a function on the non-negative integers that satisfies the recurrence

$$T(n) \;=\; a\,T\left(\frac{n}{b}\right) \;+\; \Theta(n^c)$$

with $T(0) = 0$ and $T(1) = \Theta(1)$, where n / b means either $\lfloor n / b \rfloor$ or $\lceil n / b \rceil$. Then,

Case 1. If $c > \log_b a$, then $T(n) = \Theta(n^c)$.

Case 2. If $c = \log_b a$, then $T(n) = \Theta(n^c \log n)$.

Case 3. If $c < \log_b a$, then $T(n) = \Theta(n^{\log_b a})$.

Pf sketch.

- Prove when b is an integer and n is an exact power of b.
- Extend domain of recurrences to reals (or rationals).
- Deal with floors and ceilings. ← at most 2 extra levels in recursion tree

$$\lceil \lceil \lceil n/b \rceil /b \rceil /b \rceil \;<\; n/b^3 + (1/b^2 + 1/b + 1)$$

$$\leq\; n/b^3 + 2$$

Divide-and-conquer recurrences: master theorem

Master theorem. Let $a \ge 1$, $b \ge 2$, and $c \ge 0$ and suppose that $T(n)$ is a function on the non-negative integers that satisfies the recurrence

$$T(n) \;=\; a\,T\left(\frac{n}{b}\right) \;+\; \Theta(n^c)$$

with $T(0) = 0$ and $T(1) = \Theta(1)$, where n/b means either $\lfloor n/b \rfloor$ or $\lceil n/b \rceil$. Then,

Case 1. If $c > \log_b a$, then $T(n) = \Theta(n^c)$.

Case 2. If $c = \log_b a$, then $T(n) = \Theta(n^c \log n)$.

Case 3. If $c < \log_b a$, then $T(n) = \Theta(n^{\log_b a})$.

Extensions.

- Can replace Θ with O everywhere.
- Can replace Θ with Ω everywhere.
- Can replace initial conditions with $T(n) = \Theta(1)$ for all $n \le n_0$ and require recurrence to hold only for all $n > n_0$.

Divide-and-conquer recurrences: master theorem

Master theorem. Let $a \geq 1$, $b \geq 2$, and $c \geq 0$ and suppose that $T(n)$ is a function on the non-negative integers that satisfies the recurrence

$$T(n) = a\,T\left(\frac{n}{b}\right) + \Theta(n^c)$$

with $T(0) = 0$ and $T(1) = \Theta(1)$, where n / b means either $\lfloor n / b \rfloor$ or $\lceil n / b \rceil$. Then,

Case 1. If $c > \log_b a$, then $T(n) = \Theta(n^c)$.

Case 2. If $c = \log_b a$, then $T(n) = \Theta(n^c \log n)$.

Case 3. If $c < \log_b a$, then $T(n) = \Theta(n^{\log_b a})$.

Ex. [Case 1] $T(n) = 3\,T(\lfloor n / 2 \rfloor) + 5\,n$.

- $a = 3,\ b = 2,\ c = 1 < \log_b a = 1.5849....$
- $T(n) = \Theta(n^{\log_2 3}) = O(n^{1.58})$.

Divide-and-conquer recurrences: master theorem

Master theorem. Let $a \geq 1$, $b \geq 2$, and $c \geq 0$ and suppose that $T(n)$ is a function on the non-negative integers that satisfies the recurrence

$$T(n) \;=\; a\,T\left(\frac{n}{b}\right) \;+\; \Theta(n^c)$$

with $T(0) = 0$ and $T(1) = \Theta(1)$, where n / b means either $\lfloor n / b \rfloor$ or $\lceil n / b \rceil$. Then,

Case 1. If $c > \log_b a$, then $T(n) = \Theta(n^c)$.

Case 2. If $c = \log_b a$, then $T(n) = \Theta(n^c \log n)$.

Case 3. If $c < \log_b a$, then $T(n) = \Theta(n^{\log_b a})$.

ok to intermix floor and ceiling

Ex. [Case 2] $T(n) = T(\lfloor n / 2 \rfloor) + T(\lceil n / 2 \rceil) + 17\,n$.

- $a = 2,\; b = 2,\, c = 1 = \log_b a$.
- $T(n) = \Theta(n \log n)$.

Divide-and-conquer recurrences: master theorem

Master theorem. Let $a \ge 1$, $b \ge 2$, and $c \ge 0$ and suppose that $T(n)$ is a function on the non-negative integers that satisfies the recurrence

$$T(n) \;=\; a\,T\left(\frac{n}{b}\right) \;+\; \Theta(n^c)$$

with $T(0) = 0$ and $T(1) = \Theta(1)$, where n / b means either $\lfloor n / b \rfloor$ or $\lceil n / b \rceil$. Then,

Case 1. If $c > \log_b a$, then $T(n) = \Theta(n^c)$.

Case 2. If $c = \log_b a$, then $T(n) = \Theta(n^c \log n)$.

Case 3. If $c < \log_b a$, then $T(n) = \Theta(n^{\log_b a})$.

Ex. [Case 3] $T(n) = 48\, T(\lfloor n / 4 \rfloor) + n^3$.

- $a = 48,\; b = 4,\; c = 3 > \log_b a \;=\; 2.7924...$.
- $T(n) = \Theta(n^3)$.

Master theorem need not apply

Gaps in master theorem.

- Number of subproblems is not a constant.

$$T(n) \ = n T(n/2) \ + \ n^2$$

- Number of subproblems is less than 1.

$$T(n) \ = \frac{1}{2} T(n/2) \ + \ n^2$$

- Work to divide and combine subproblems is not $\Theta(n^c)$.

$$T(n) \ = \ 2\,T(n/2) \ + n \log n$$

Consider the following recurrence. Which case of the master theorem?

$$T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \\ 3T(\lceil n/2 \rceil) + \Theta(n) & \text{if } n > 1 \end{cases}$$

A. Case 3: $T(n) = \Theta(n)$.

B. Case 2: $T(n) = \Theta(n \log n)$.

C. Case 1: $T(n) = \Theta(n^{\log_2 3}) = O(n^{1.585})$.

D. Master theorem not applicable.

Consider the following recurrence. Which case of the master theorem?

$$T(n) = \begin{cases} 0 & \text{if } n \le 1 \\ T(\lfloor n/5 \rfloor) + T(n - 3\lfloor n/10 \rfloor) + \frac{11}{5}n & \text{if } n > 1 \end{cases}$$

A. Case 1: $T(n) = \Theta(n)$.

B. Case 2: $T(n) = \Theta(n \log n)$.

C. Case 3: $T(n) = \Theta(n)$.

D. Master theorem not applicable.