Master method

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.

- $a \geq 1$ is the (integer) number of subproblems.
- $b \geq 2$ is the (integer) factor by which the subproblem size decreases.
- $f(n)$ = work to divide and merge subproblems.

Recursion tree.

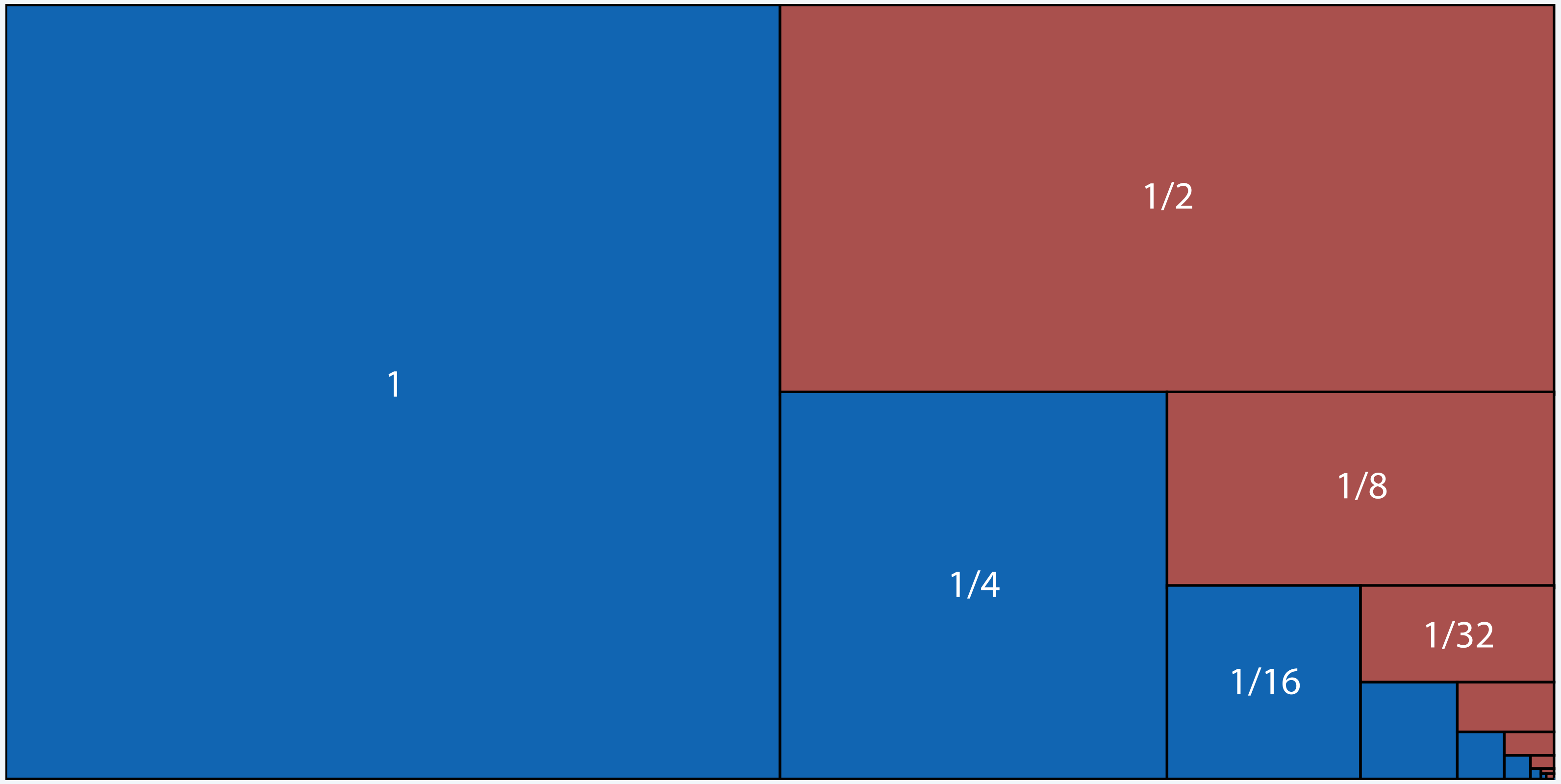
- $k = \log_b n$ levels.
- a^i = number of subproblems at level i.
- n / b^i = size of subproblem at level i.

Geometric series

Fact 1. For $r \neq 1$, $1 + r + r^2 + r^3 + \ldots + r^{k-1} = \dfrac{1 - r^k}{1 - r}$

Fact 2. For $r = 1$, $1 + r + r^2 + r^3 + \ldots + r^{k-1} = k$

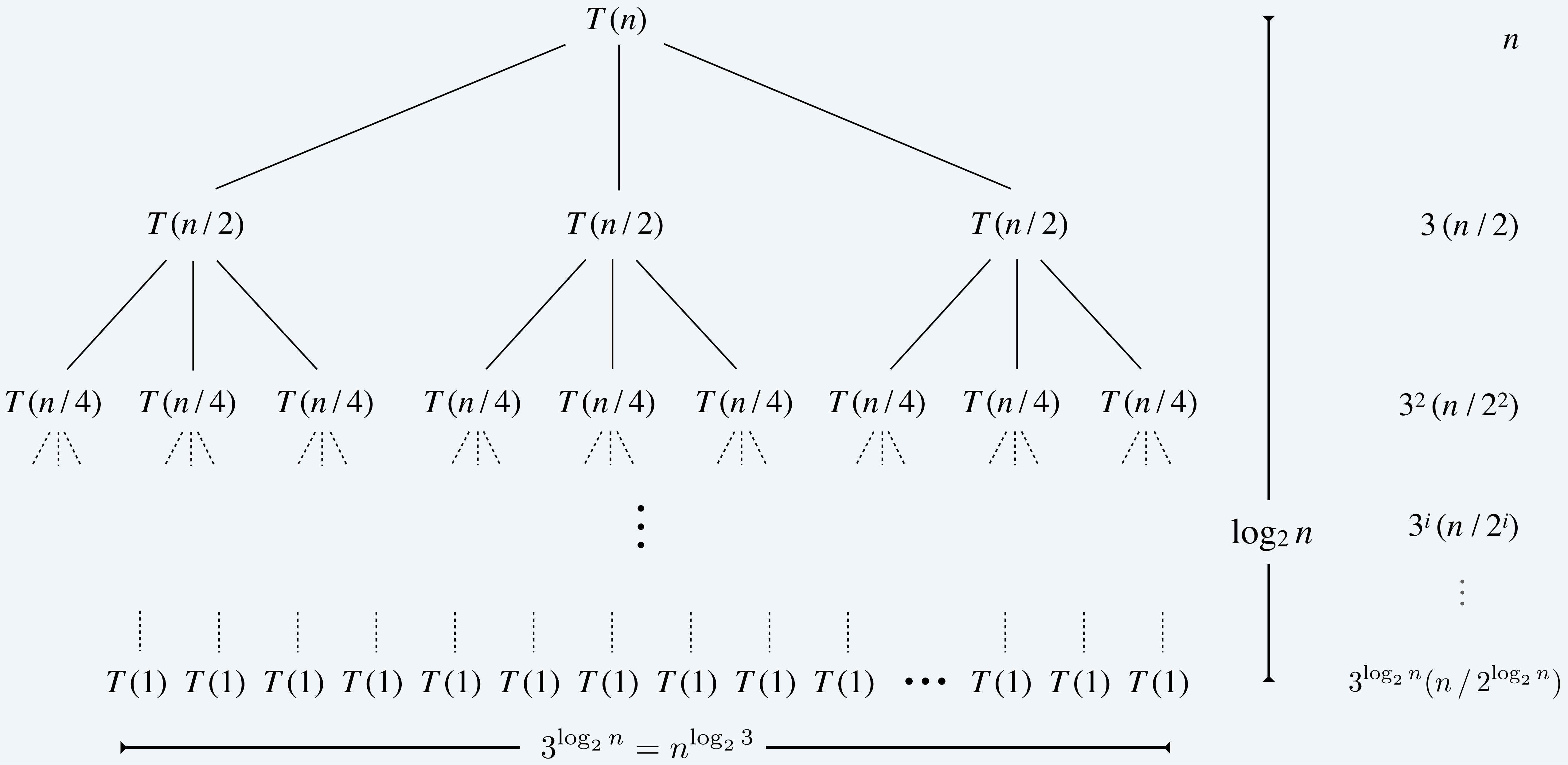
Fact 3. For $r < 1$, $1 + r + r^2 + r^3 + \ldots = \dfrac{1}{1 - r}$



1 + 1/2 +1/4 + 1/8 + ... = 2

Case 1: total cost dominated by cost of leaves

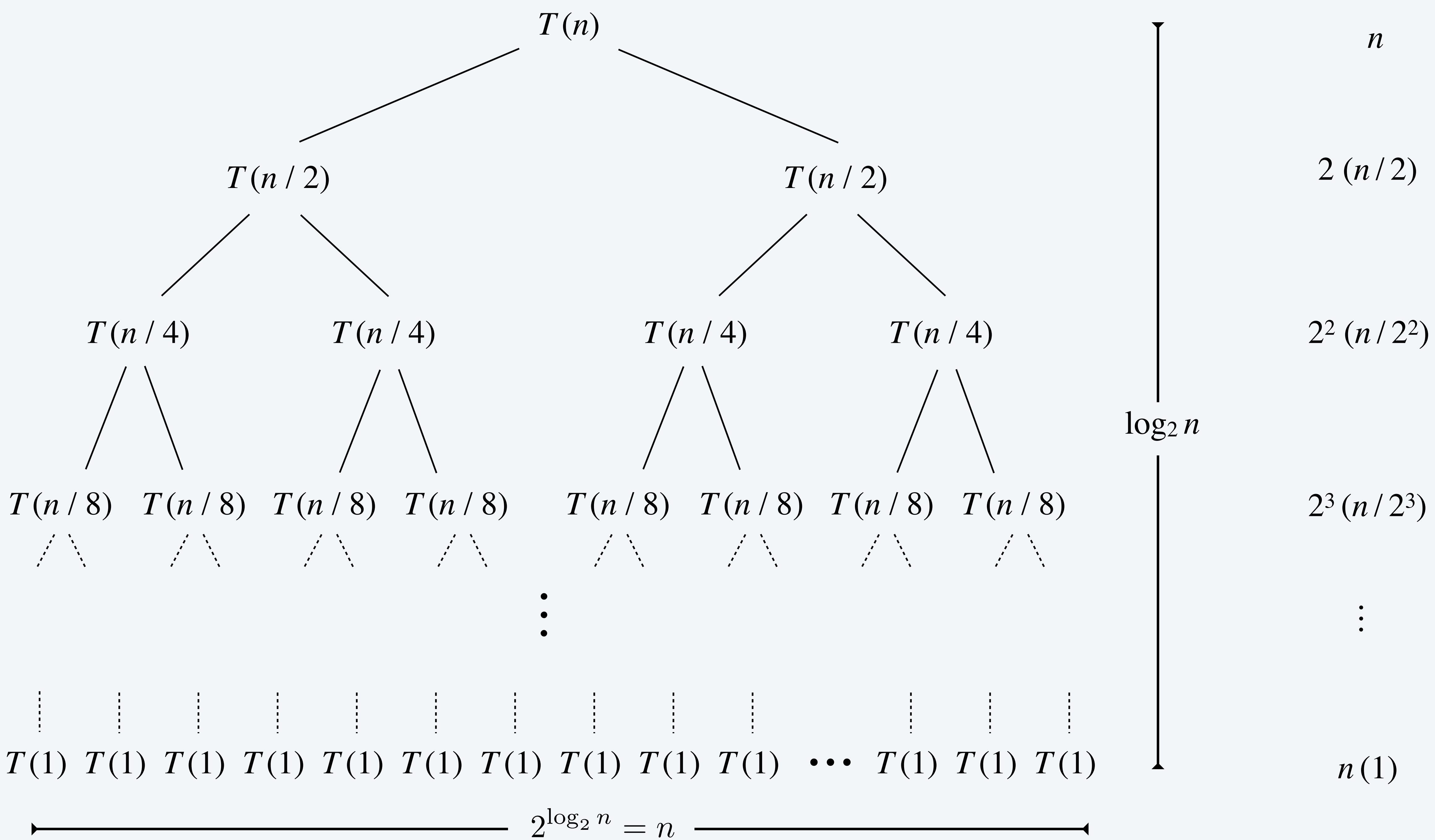
Ex 1. If $T(n)$ satisfies $T(n) = 3\,T(n/2) + n$, with $T(1) = 1$, then $T(n) = \Theta(n^{\log_2 3})$.



$r = 3/2 > 1$ $\quad T(n) = (1 + r + r^2 + r^3 + \ldots + r^{\log_2 n})\, n \;=\; \dfrac{r^{1+\log_2 n} - 1}{r - 1}\, n \;=\; 3n^{\log_2 3} - 2n$

Case 2: total cost evenly distributed among levels

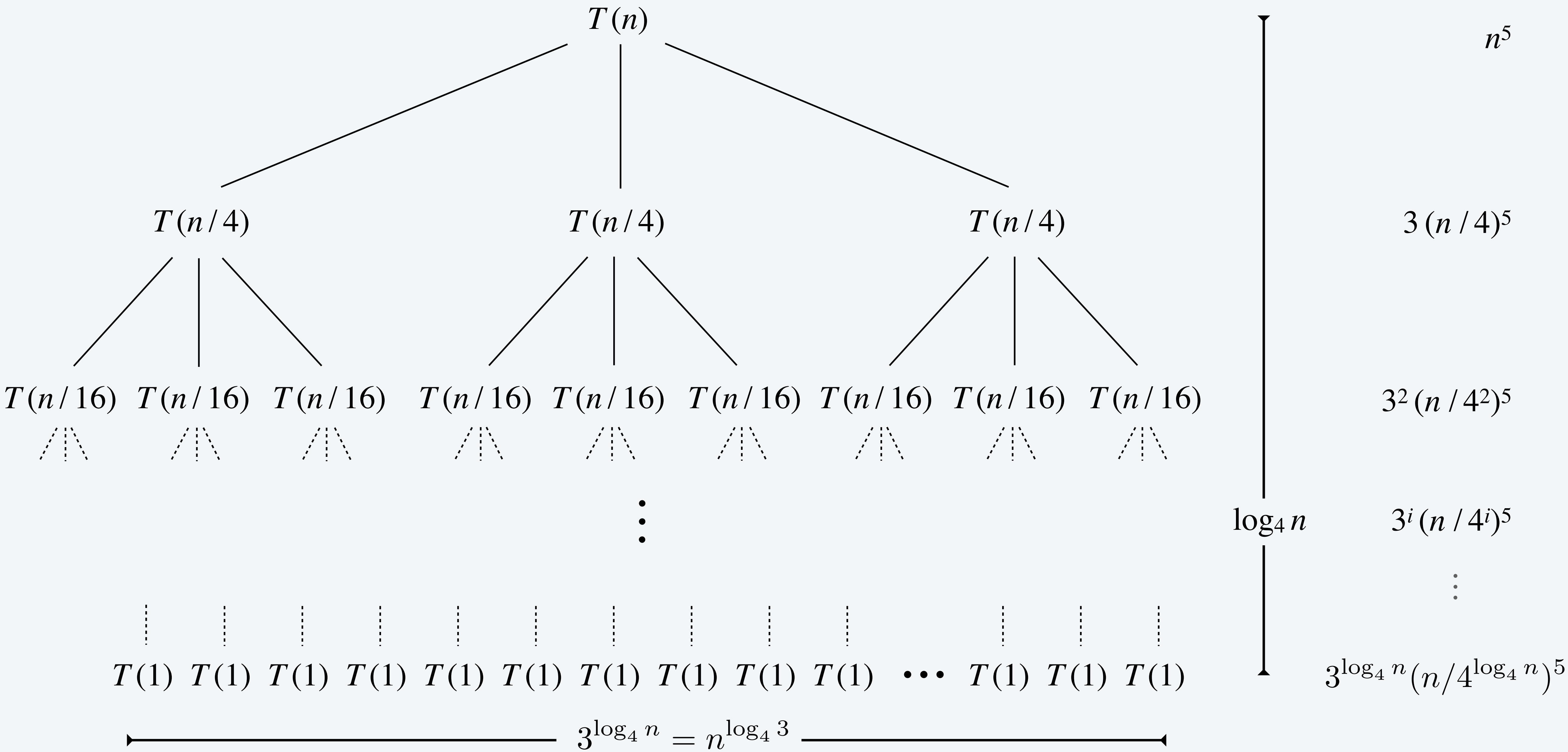
Ex 2. If $T(n)$ satisfies $T(n) = 2\,T(n/2) + n$, with $T(1) = 1$, then $T(n) = \Theta(n \log n)$.



$r = 1$ $\qquad T(n) = (1 + r + r^2 + r^3 + \ldots + r^{\log_2 n})\, n \;=\; n\,(\log_2 n + 1)$

Case 3: total cost dominated by cost of root

Ex 3. If $T(n)$ satisfies $T(n) = 3\,T(n/4) + n^5$, with $T(1) = 1$, then $T(n) = \Theta(n^5)$.



$r = 3/4^5 < 1 \qquad n^5 \le T(n) \le (1 + r + r^2 + r^3 + \ldots)\,n^5 \le \dfrac{1}{1-r}\,n^5$

Master theorem

Master theorem. Suppose that $T(n)$ is a function on the non-negative integers that satisfies the recurrence

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

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 $f(n) = O(n^k)$ for some constant $k < \log_b a$, then $T(n) = \Theta\left(n^{\log_b a}\right)$

Ex. $T(n) = 3\,T(n / 2) + 5\,n$.

- $a = 3,\ b = 2,\ f(n) = 5\,n,\ k = 1,\ \log_b a = 1.58\ldots$
- $T(n) = \Theta(n^{\log_2 3})$.

Master theorem

Master theorem. Suppose that $T(n)$ is a function on the non-negative integers that satisfies the recurrence

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

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 2. If $f(n) = \Theta(n^k \log^p n)$ for $k = \log_b a$, then $T(n) = \Theta(n^k \log^{p+1} n)$.

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

- $a = 2,\ b = 2,\ f(n) = 17\,n,\ \ k = 1, \log_b a = 1,\ p = 1$.
- $T(n) = \Theta(n \log^2 n)$.

Master theorem

Master theorem. Suppose that $T(n)$ is a function on the non-negative integers that satisfies the recurrence

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

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 3. If $f(n) = \Omega(n^k)$ for some constant $k > \log_b a$, and if $a\,f(n/b) \le c\,f(n)$ for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$.

"regularity condition" holds if f(n) = Θ(n^k)

Ex. $T(n) = 3\,T(n/2) + n^2$.

- $a = 3,\ b = 2,\ f(n) = n^2,\ k = 2,\ \log_b a = 1.58\ldots$
- Regularity condition: $3\,(n/2)^2 \le c\,n^2$ for $c = 3/4$.
- $T(n) = \Theta(n^2)$.

Master theorem

Master theorem. Suppose that $T(n)$ is a function on the non-negative integers that satisfies the recurrence

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

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 $f(n) = O(n^k)$ for some constant $k < \log_b a$, then $T(n) = \Theta(n^k)$.

Case 2. If $f(n) = \Theta(n^k \log^p n)$ for $k = \log_b a$, then $T(n) = \Theta(n^k \log^{p+1} n)$.

Case 3. If $f(n) = \Omega(n^k)$ for some constant $k > \log_b a$, and if $a\,f(n/b) \le c\,f(n)$ for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$.

Pf sketch.

- Use recursion tree to sum up terms (assuming n is an exact power of b).
- Three cases for geometric series.
- Deal with floors and ceilings.

Master theorem need not apply

Gaps in master theorem.

- Number of subproblems must be a constant.

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

- Number of subproblems must be ≥ 1.

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

- Non-polynomial separation between $f(n)$ and $n^{\log_b a}$.

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

- $f(n)$ is not positive.

$$T(n) = 2T(n/2) - n^2$$

- Regularity condition does not hold.

$$T(n) = T(n/2) + n(2 - \cos n)$$

Akra–Bazzi theorem

Desiderata. Generalizes master theorem to divide-and-conquer algorithms where subproblems have substantially different sizes.

Theorem. [Akra–Bazzi] Given constants $a_i > 0$ and $0 < b_i \le 1$, functions $h_i(n) = O(n / \log^2 n)$ and $g(n) = O(n^c)$, if the function $T(n)$ satisfies the recurrence:

$$T(n) = \sum_{i=1}^{k} a_i \, T\left(b_i n + h_i(n)\right) + g(n)$$

a_i subproblems of size b_i n

small perturbation to handle floors and ceilings

Then $T(n) = \Theta\left(n^p \left(1 + \int_1^n \frac{g(u)}{u^{p+1}} du\right)\right)$ where p satisfies $\sum_{i=1}^{k} a_i \, b_i^p = 1$.

Ex. $T(n) = 7/4 \; T(\lfloor n/2 \rfloor) + T(\lceil 3/4 \, n \rceil) + n^2$, with $T(0) = 0$ and $T(1) = 1$.

- $a_1 = 7/4, \; b_1 = 1/2, \; a_2 = 1, \; b_2 = 3/4 \Rightarrow p = 2$.
- $h_1(n) = \lfloor 1/2 \, n \rfloor - 1/2 \, n, \; h_2(n) = \lceil 3/4 \, n \rceil - 3/4 \, n$.
- $g(n) = n^2 \Rightarrow T(n) = \Theta(n^2 \log n)$.