CSE 613: Parallel Programming

Lecture 10
(The Master Theorem)

Rezaul A. Chowdhury
Department of Computer Science
SUNY Stony Brook
Fall 2013

A Useful Recurrence

Consider the following recurrence:

$$T(n) = \begin{cases} \Theta(1), & if\ n \leq 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise; \end{cases}$$

where, $a \geq 1$ and $b > 1$.

Arises frequently in the analyses of *divide-and-conquer* algorithms.

Consider the following recurrences from my CSE548 (Fall'12) lectures.

Karatsuba's Algorithm: $T(n) = 3T\left(\frac{n}{2}\right) + \Theta(n)$

Strassen's Algorithm: $T(n) = 7T\left(\frac{n}{2}\right) + \Theta(n^2)$

Fast Fourier Transform: $T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$

How the Recurrence Unfolds

$$T(n) = \begin{cases} \Theta(1), & if\ n \leq 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$

How the Recurrence Unfolds

$$T(n) = \begin{cases} \Theta(1), & if\ n \leq 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$

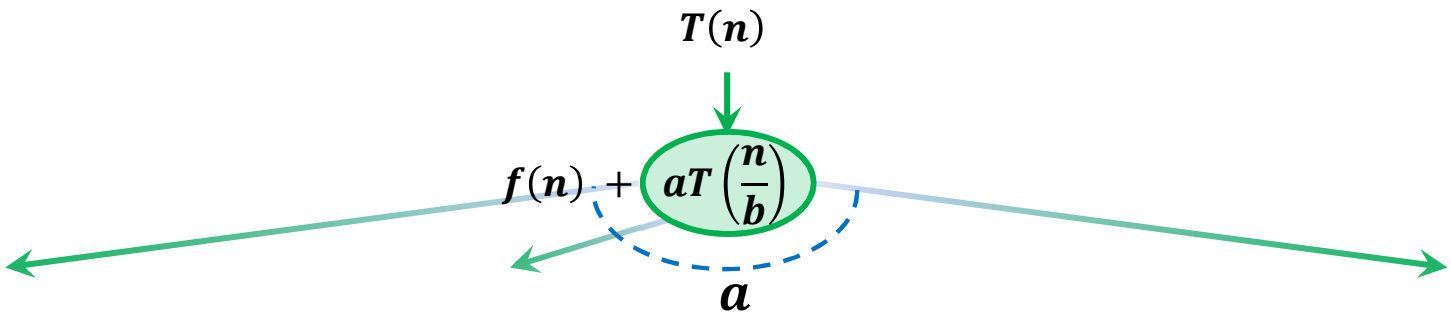
$$\boldsymbol{T(n)}$$

$$\downarrow$$

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

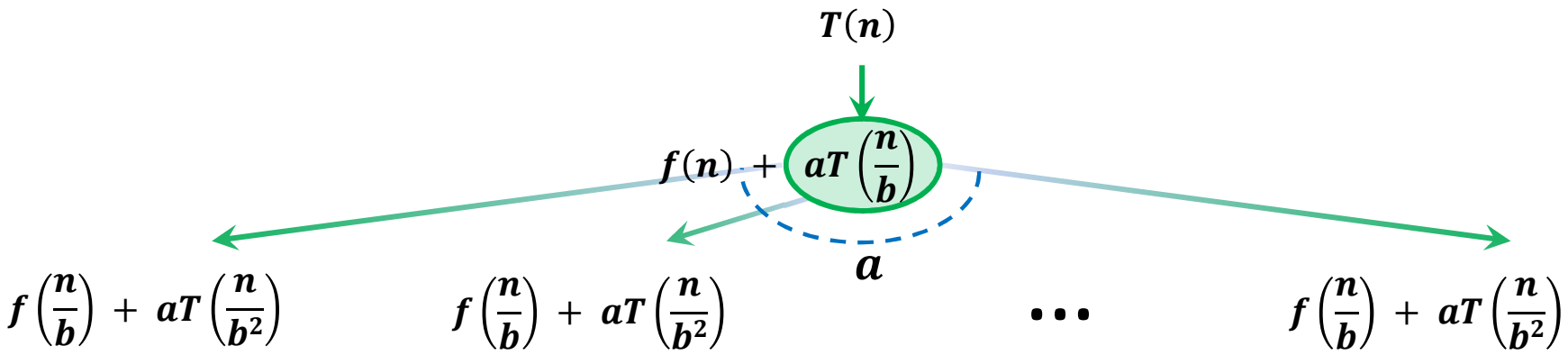
How the Recurrence Unfolds

$$T(n) = \begin{cases} \Theta(1), & if\ n \leq 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



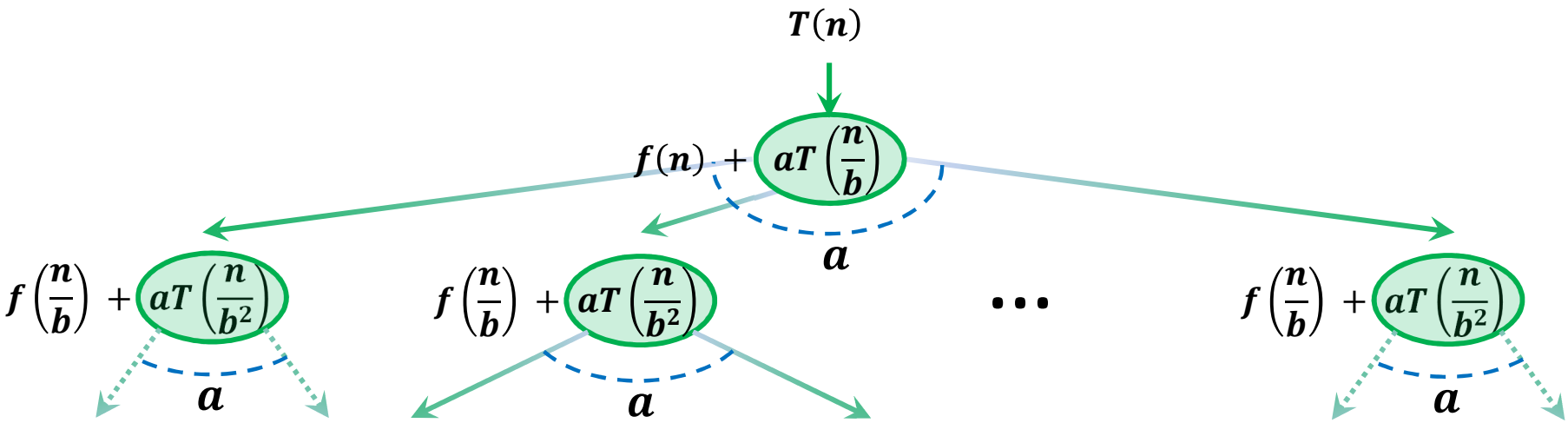
How the Recurrence Unfolds

$$T(n) = \begin{cases} \Theta(1), & if\ n \leq 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



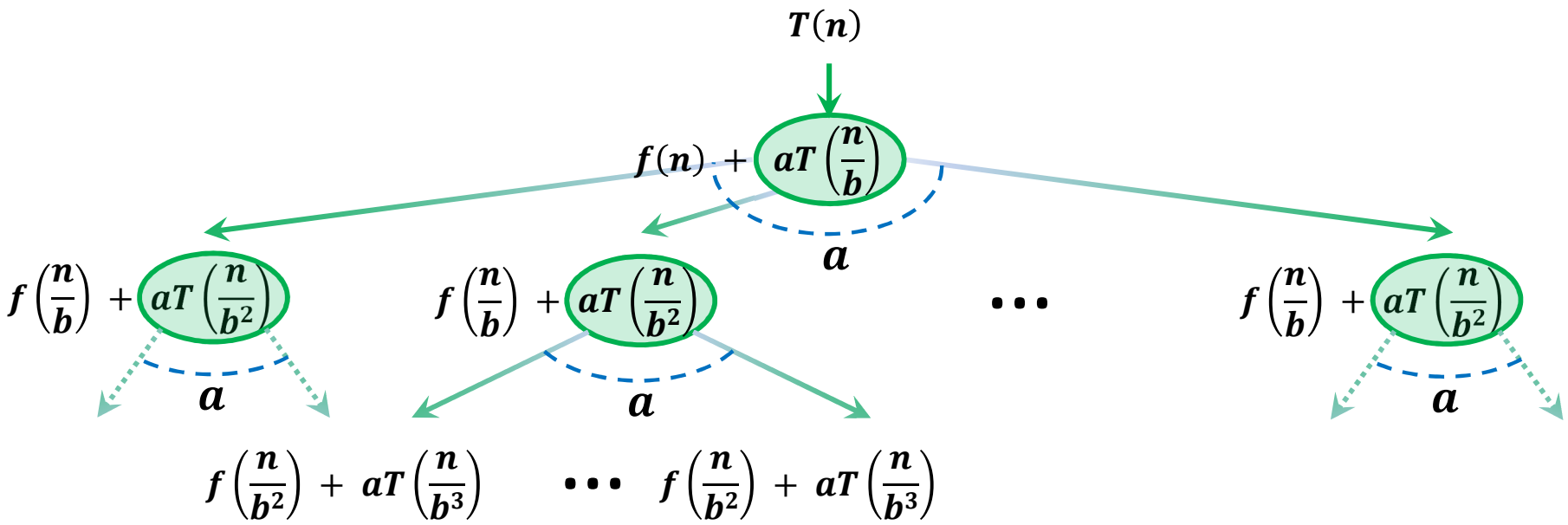
How the Recurrence Unfolds

$$T(n) = \begin{cases} \Theta(1), & if\ n \le 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



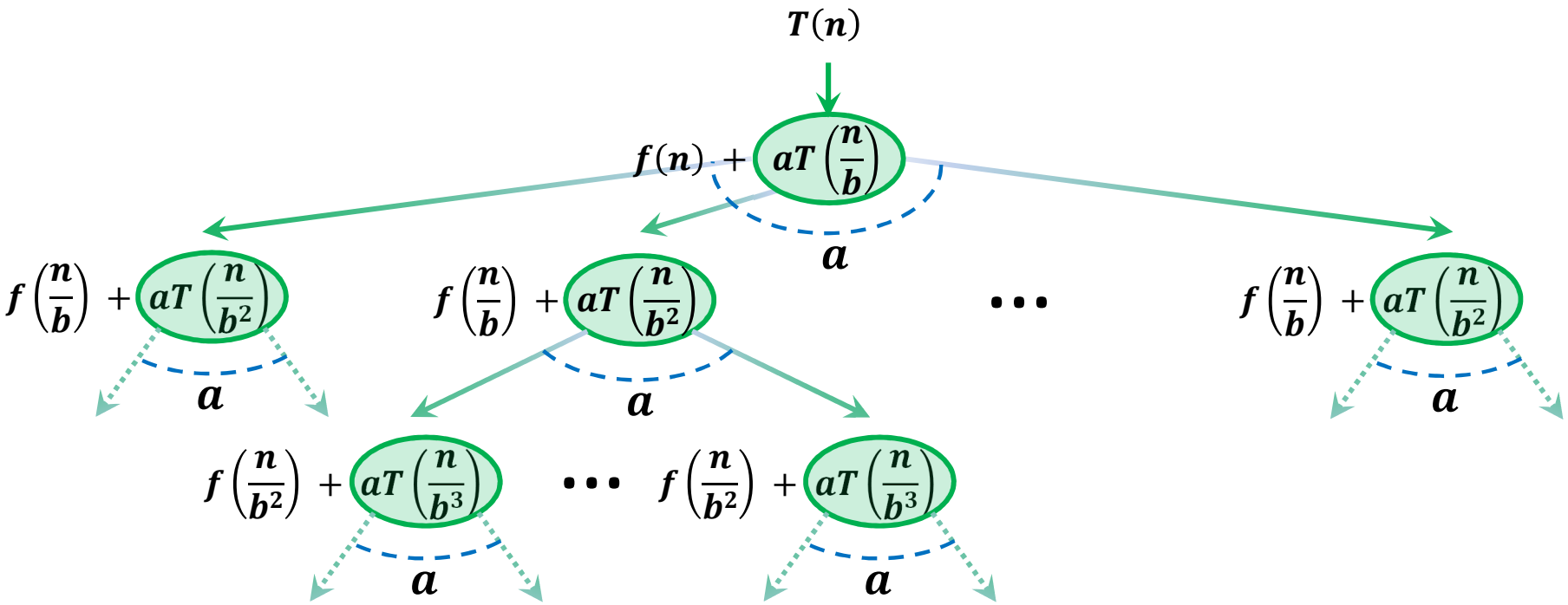
How the Recurrence Unfolds

$$T(n) = \begin{cases} \Theta(1), & if\ n \le 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



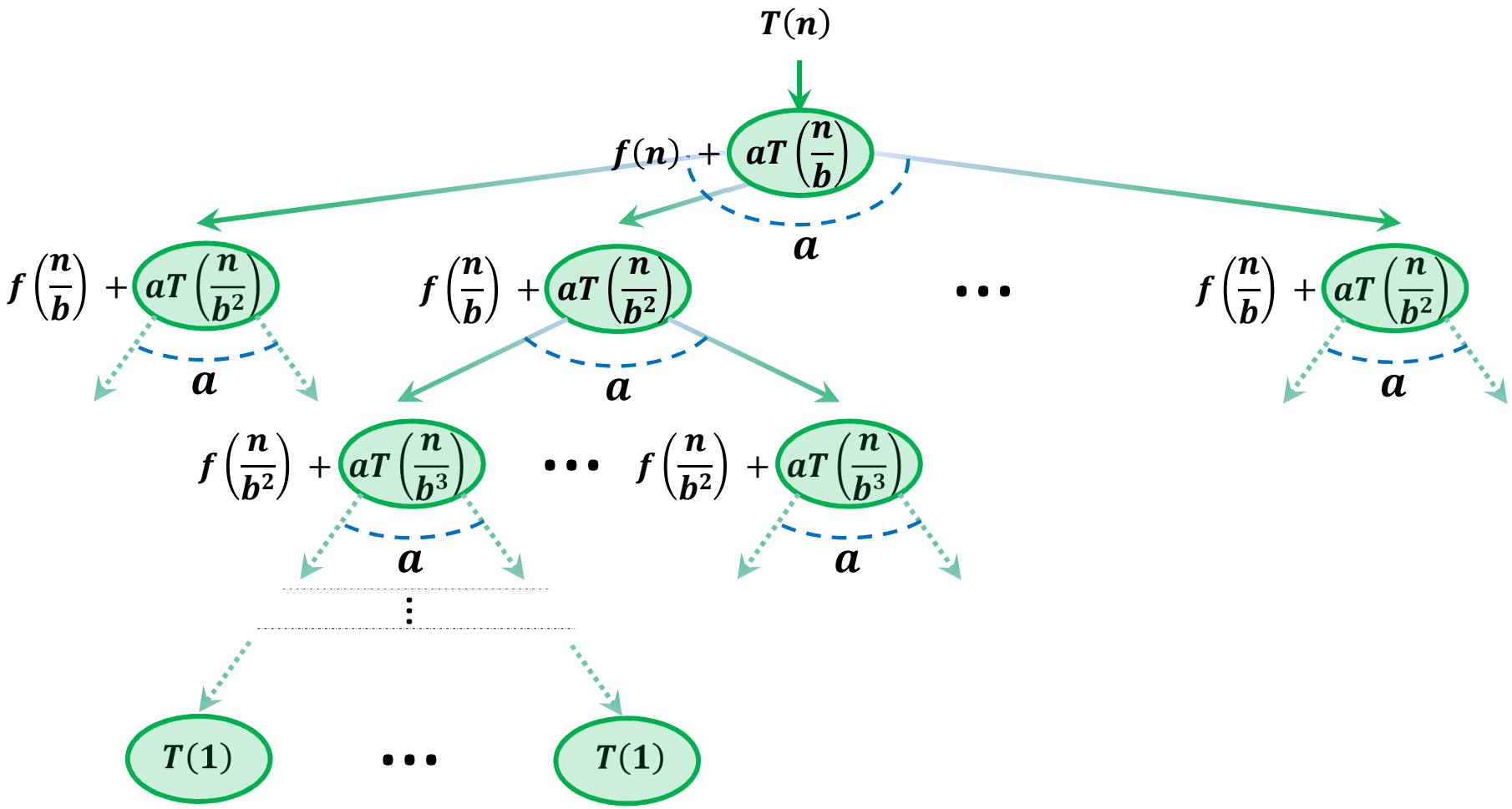
How the Recurrence Unfolds

$$T(n) = \begin{cases} \Theta(1), & if\ n \le 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



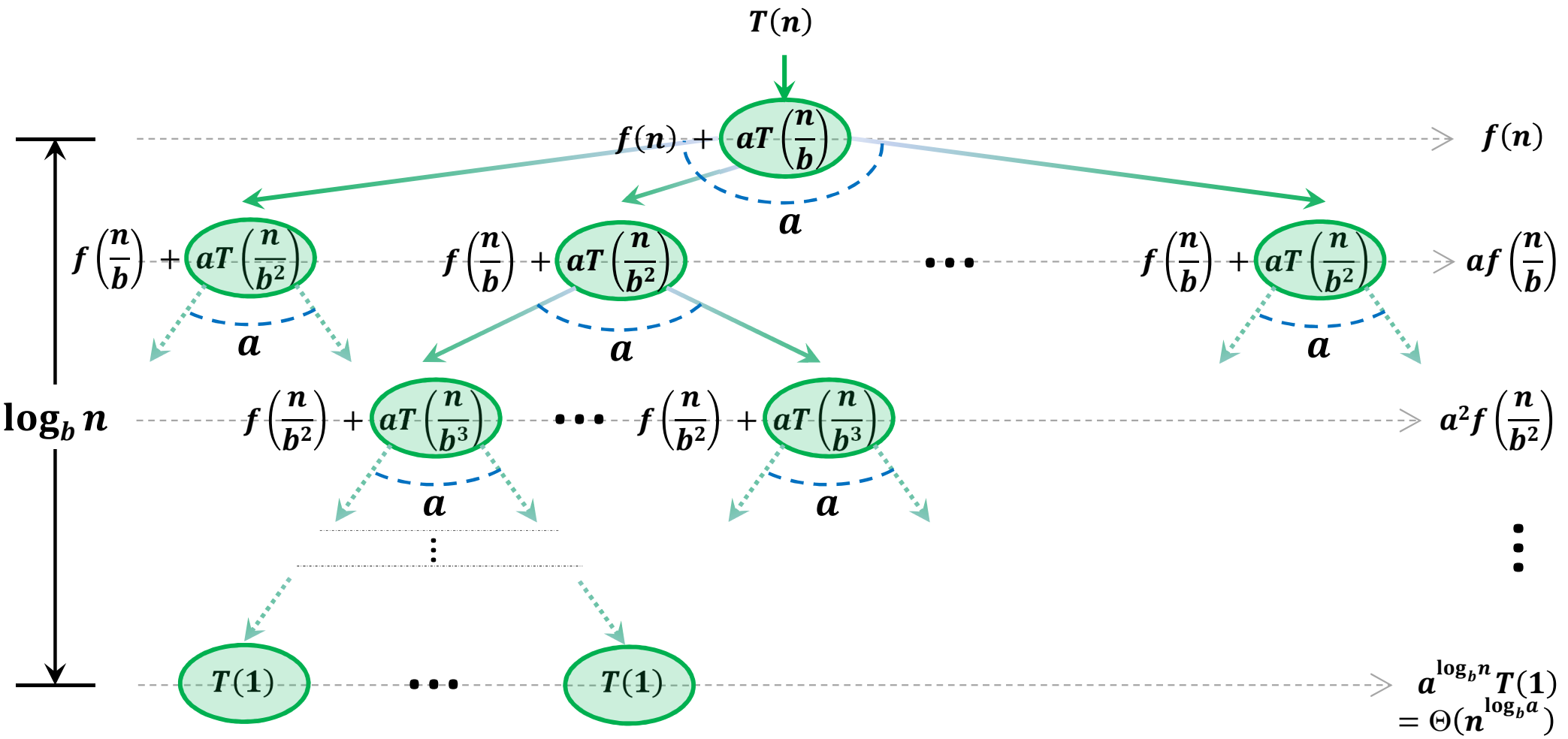
How the Recurrence Unfolds

$$T(n) = \begin{cases} \Theta(1), & if\ n \le 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



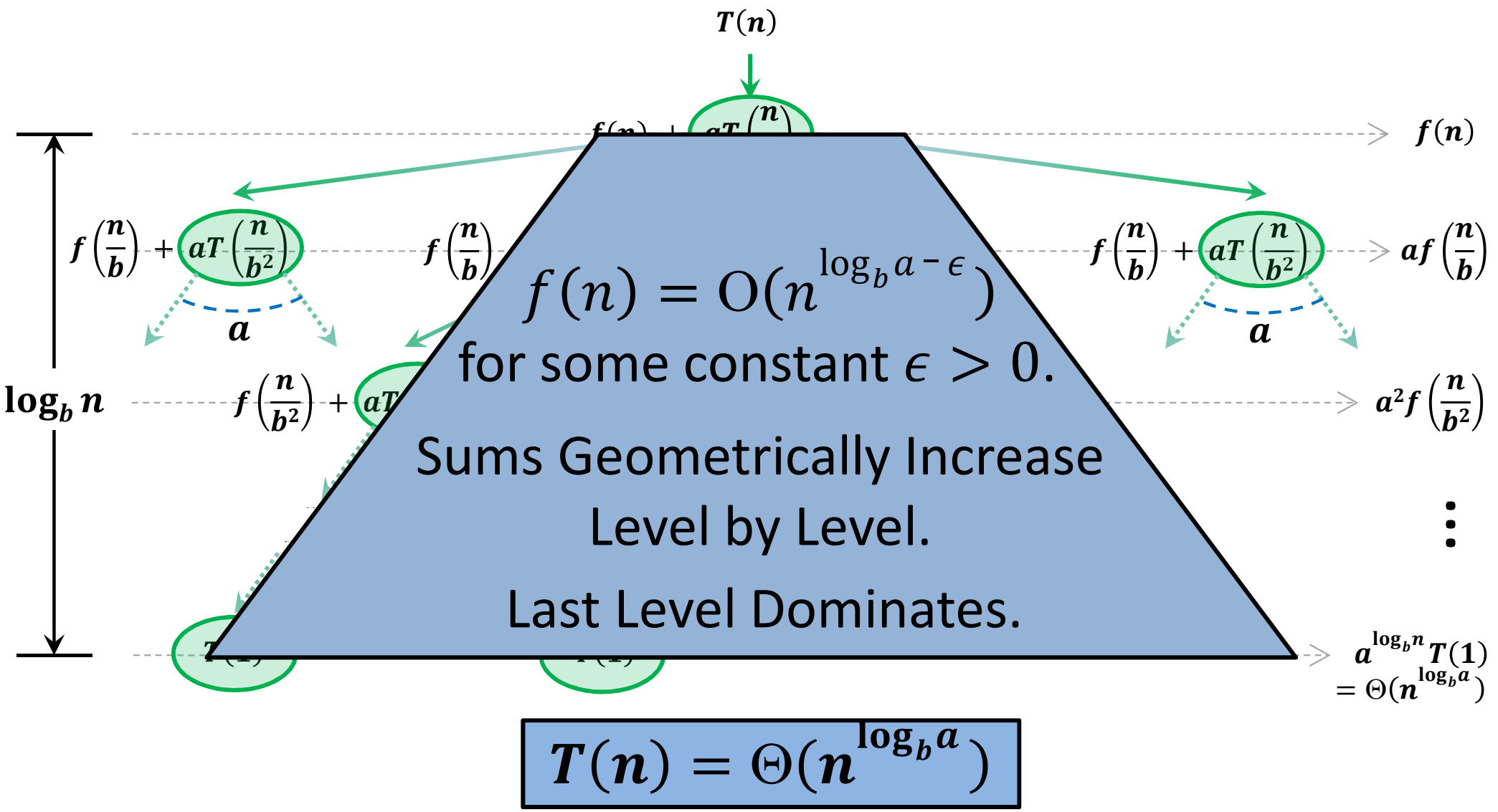
How the Recurrence Unfolds

$$T(n) = \begin{cases} \Theta(1), & if\ n \le 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



How the Recurrence Unfolds: Case 1

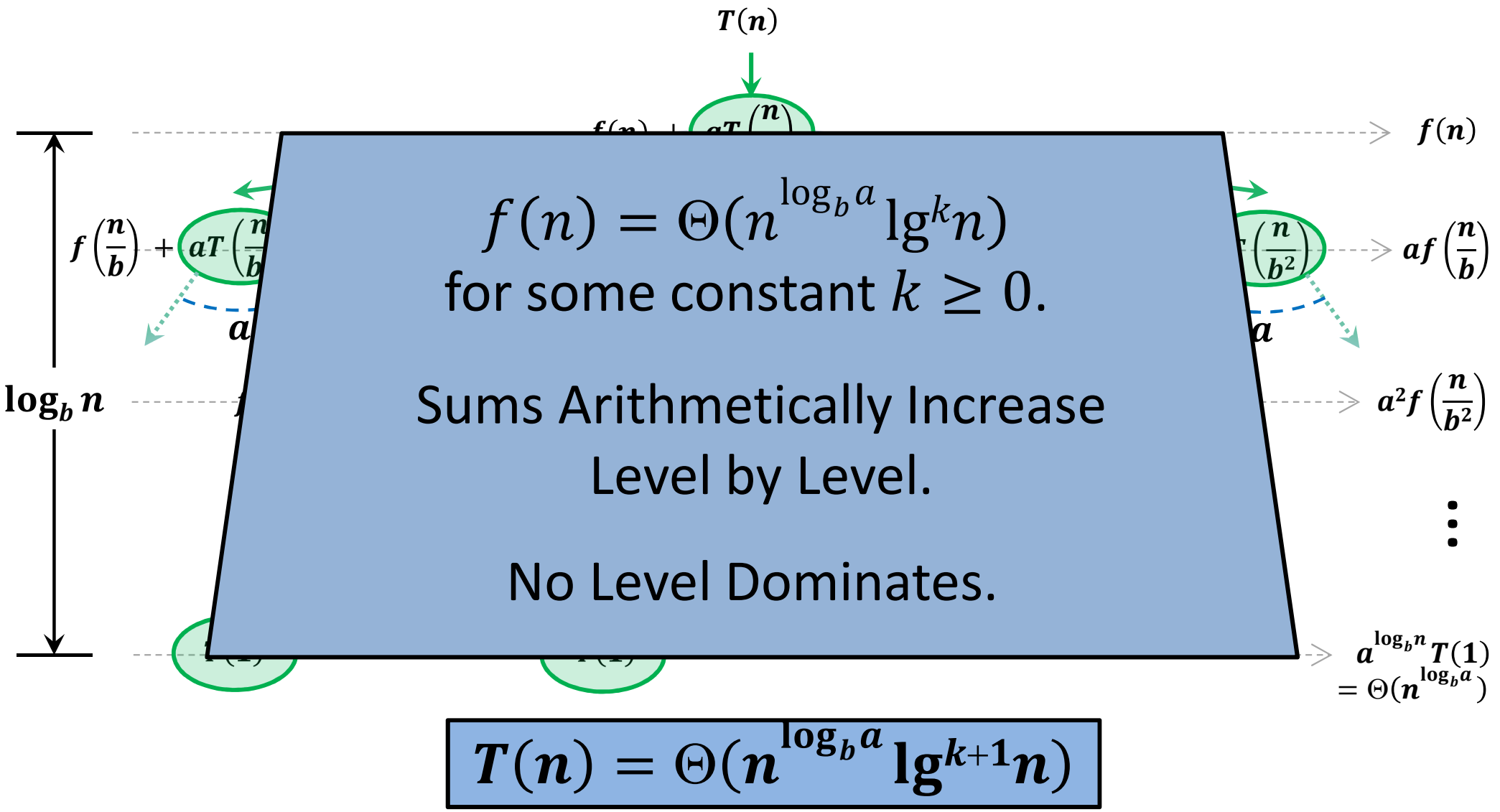
$$T(n) = \begin{cases} \Theta(1), & if\ n \le 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



$$\boldsymbol{T(n) = \Theta(n^{\log_b a})}$$

How the Recurrence Unfolds: Case 2

$$T(n) = \begin{cases} \Theta(1), & if\ n \le 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



$$f(n) = \Theta(n^{\log_b a} \lg^k n)$$
for some constant $k \ge 0$.

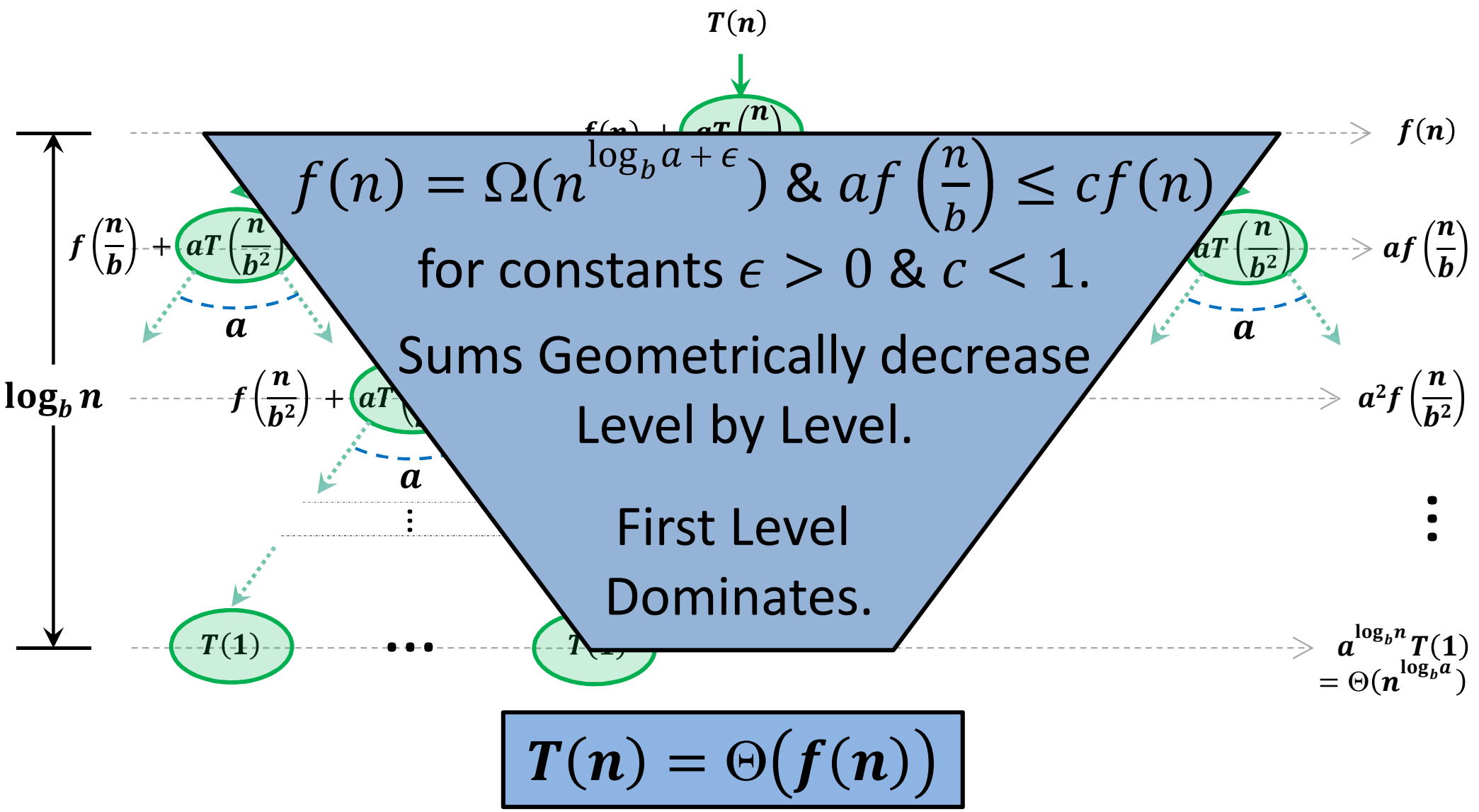
Sums Arithmetically Increase Level by Level.

No Level Dominates.

$$\boldsymbol{T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)}$$

How the Recurrence Unfolds: Case 3

$$T(n) = \begin{cases} \Theta(1), & if\ n \le 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise. \end{cases}$$



$$\boldsymbol{T(n) = \Theta\big(f(n)\big)}$$

The Master Theorem

$$T(n) = \begin{cases} \Theta(1), & if\ n \le 1, \\ aT\left(\frac{n}{b}\right) + f(n), & otherwise\ (a \ge 1, b > 1). \end{cases}$$

Case 1: $f(n) = \mathrm{O}(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$

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

Case 2: $f(n) = \Theta(n^{\log_b a} \lg^k n)$ for some constant $k \ge 0$.

$$T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$$

Case 3: $f(n) = \Omega(n^{\log_b a + \epsilon})$ and $af\left(\frac{n}{b}\right) \le cf(n)$

for constants $\epsilon > 0$ and $c < 1$.

$$T(n) = \Theta\big(f(n)\big)$$

Example Applications of Master Theorem

Example 1: $T(n) = 3T\left(\frac{n}{2}\right) + \Theta(n)$

Master Theorem Case 1: $T(n) = \Theta\left(n^{\log_2 3}\right)$

Example 2: $T(n) = 7T\left(\frac{n}{2}\right) + \Theta(n^2)$

Master Theorem Case 1: $T(n) = \Theta\left(n^{\log_2 7}\right)$

Example 3: $T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$

Master Theorem Case 2: $T(n) = \Theta(n\log n)$

Assuming that we have an infinite number of processors, and each recursive call in example 2 above can be executed in parallel:

Example 4: $T(n) = T\left(\frac{n}{2}\right) + \Theta(n^2)$

Master Theorem Case 3: $T(n) = \Theta(n^2)$