The Master Theorem

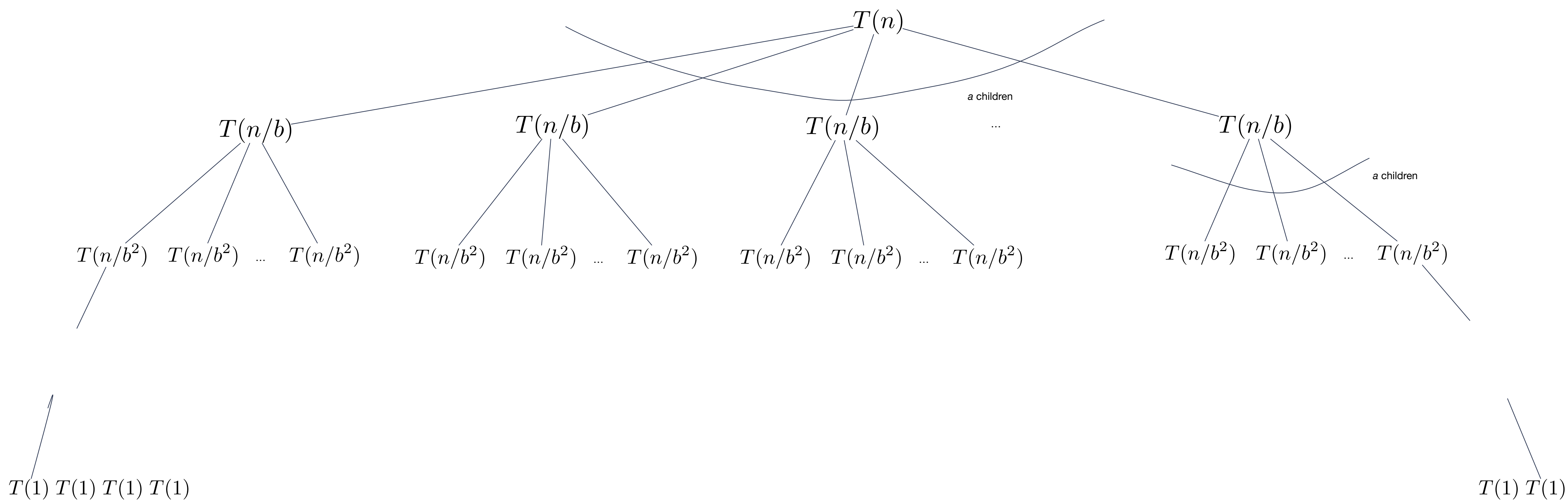
A family of Recursion Equations

- Divide and conquer frequently lead to recursions of the form

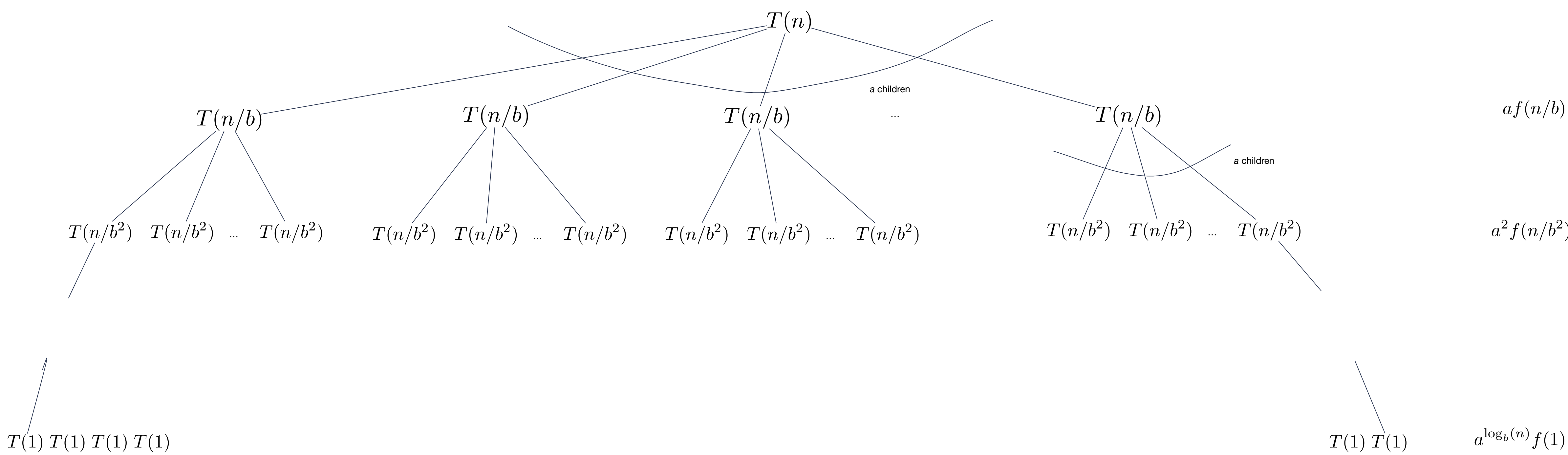
$$T(n) = aT(n/b) + f(n)$$

A family of Recursion Equations

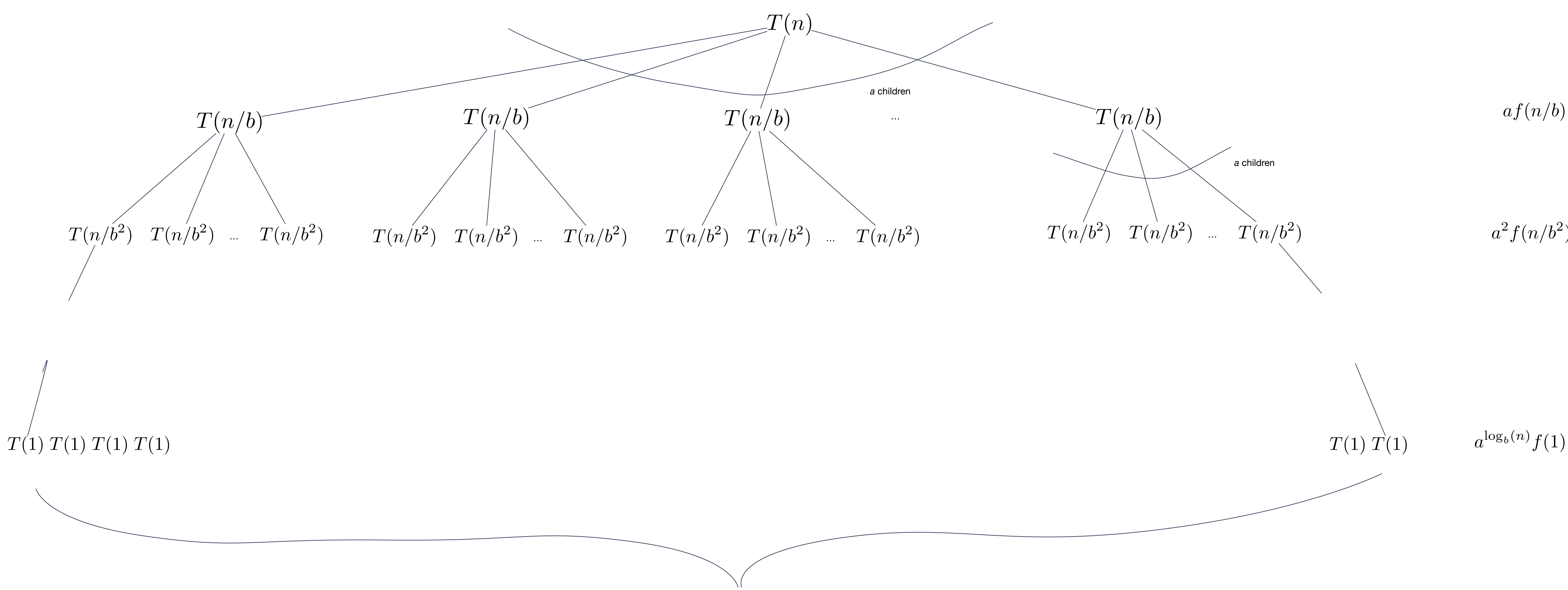
- Solve Recurrence:



A family of Recursion Equations



A family of Recursion Equations



$$a^{\log_b n} = \exp\log(a^{\log_b n}) = \exp(\log_b(n)\log a)) =$$

$$\exp(\log(n)\log(b)^{-1}\log(a)) = \exp(log_b(a)\log(n)) = \log(n^{\log_b(a)})$$

A family of Recursion Equations

- Total is

$$\left(\sum_{j=0}^{\log_b n-1} a^j f(n/b^j)\right) + cn^{\log_b a}$$

- Need to compare *f* with power of *n* in order to see what dominates

A family of Recursion Equations

$$f(n) = O(n^{\log_b a - \epsilon}) \implies T(n) = \Theta(n^{\log_b a})$$

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

$f(n) = \Omega(n^{\log_b a + \epsilon})$ **and** $af(n/b) \leq cf(n)$ **eventually** $\implies T(n) = \Theta(f(n))$

A family of Recursion Equations

- There are gaps between the three cases, where the master theorem does not apply

Examples

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

$$n^{\log_2 2} = n = f(n)$$ **Case 2**

$$T(n) = \Theta(n \log n)$$

Examples

$$T(n) = 3T(n/2) + n$$

$$\log_2 3 = 1.58496$$

$$n = O(n^{\log_2 3 - 0.1})$$

$$\implies T(n) = \Theta(n^{\log_2 3})$$

Examples

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

$$a = 1,\ b = 2$$

$$\log_2 1 = 0$$

$$n = \Omega(n^{0+1/2})$$

$$\implies T(n) = \Theta(n)$$

Examples

$$T(n) = 3T(n/3) + n \log n$$

$a = 3$ $b = 3$ **so compare with** n

$n \log n \notin \Theta(n)$ $\quad$ $n \log n \notin \Omega(n^{1+\epsilon})$

Falls into the gap between Case 2 and Case 3