Recurrence Relations

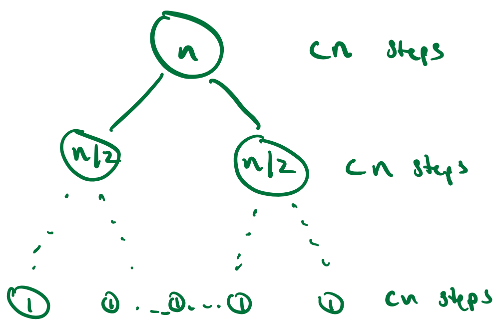
Given recurrence relation for $T(n)$, find closed-form expression for $T(n)$

Example: runtime of mergesort

→ Let $T(n)$ be runtime of Mergesort on n-length arr.

Note $T(n) = O(n \log n)$

$$T(n) = \underbrace{2T\left(\tfrac{n}{2}\right)}_{\text{calling ms recursively}} + \underbrace{cn}_{\text{merging halves}}$$



$$T(n) = \sum_{i=1}^{\log n} cn = cn \log n = O(n \log n)$$

Other examples:

Karatsuba Multiplication

$$T(n) = 3T\left(\tfrac{n}{2}\right) + O(n)$$

$$= O(n^{\log 3}) \approx O(n^{1.6})$$

The Master Theorem

Suppose $a \geq 1$, $b \geq 1$, d are constants
Suppose $T(n) = aT(\frac{n}{b}) + O(n^d)$. Then

$$T(n) = \begin{cases} O(n^d \log n) & \text{if } a = b^d \\ O(n^d) & \text{if } a < b^d \\ O(n^{\log_b a}) & \text{if } a > b^d \end{cases}$$

Intution: reduce one problem to a subproblems of size $\frac{n}{b}$ and do n^d steps of additional work

Substitution Method

1. Guess answer
2. Prove it (probably using induction)
3. Present proof w/o guess steps

ex. $T(n) \leq T(\frac{n}{5}) + T(\frac{7n}{10}) + n$ for $n \geq 10$
$T(n) = 1 \quad 1 \leq n \leq 10$

Inductive hypothesis: $T(n) \leq cn$ → not $O(n)$
Base case: $T(n) = 1 \leq cn$ for all $1 \leq n \leq 10$
Inductive step: Assume IH holds for $1 \leq n < k$, $k > 10$

$$T(k) \leq k + T\left(\frac{k}{5}\right) + T\left(\frac{7k}{10}\right)$$
$$\leq k + \frac{ck}{5} + \frac{7ck}{10} \leq ck$$
$$\Rightarrow \boxed{c = 10}$$

now, show proof w/ c = 10 replacing c

Therefore $T(n) \leq 10n \Rightarrow T(n) = O(n)$.