COMPSCI 311: Introduction to Algorithms

Lecture 12: Divide and Conquer

Dan Sheldon

University of Massachusetts Amherst

A More General Recurrence

$$
\begin{aligned}
T(n) &\leq q \cdot T(n/2) + cn \\
T(2) &\leq c
\end{aligned}
$$

- What does the algorithm look like?
  - q recursive calls to itself on problems of half the size
  - $O(n)$ work outside of the recursive calls
- Exercises / board work: $q = 1$, $q > 2$ (recursion trees)

Review

Recursion Tree: $q = 1$

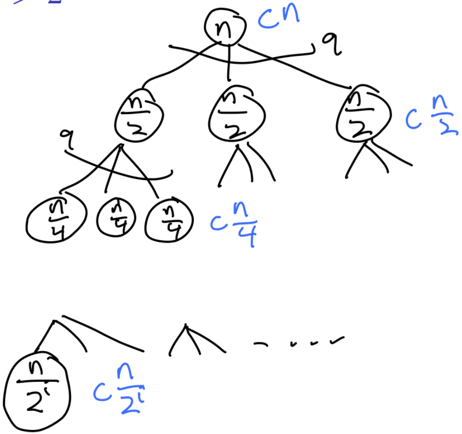
Level	# problems	Subproblem (node)	Work at node	work/level
0	1	n	cn	cn
1	1	$\frac{n}{2}$	$c\frac{n}{2}$	$c \cdot \frac{n}{2}$
2	1	$\frac{n}{4}$	$c \cdot \frac{n}{4}$	$c\frac{n}{4}$
$\vdots$	$\vdots$	$\vdots$		$\vdots$
i	1	$\frac{n}{2^i}$	$c\frac{n}{2^i}$	$c\frac{n}{2^i}$
$\vdots$				$\vdots$
$\log_2(n)-1$		2		

Review

Recursion Tree: $q > 2$

level	#problems
0	1
1	q
2	q^2
⋮	
i	q^i
⋮	

$d = \log_2(n) - 1$



work/level

$cn = cn$

$q \cdot c\frac{n}{2} = cn\left(\frac{q}{2}\right)$

$q^2 \cdot c\frac{n}{4} = cn\left(\frac{q^2}{2^2}\right)$

⋮ ⋮

$q^i \cdot c\frac{n}{2^i} = cn\left(\frac{q}{2}\right)^i$

General Case

Work at level j of recursion tree:

$$\underbrace{q^j}_{\text{num. subproblems}} \times \underbrace{cn/2^j}_{\text{work per subproblem}} = \left(\frac{q}{2}\right)^j cn$$

Total work:

$$T(n) = cn \cdot \sum_{j=0}^{d} \left(\frac{q}{2}\right)^j \qquad (d = \log_2 n - 1)$$

$q = 1$? Easy to see $T(n) \leq 2cn = O(n)$.

Useful Fact: Geometric Sum

If $r \neq 1$ then

$$1 + r + r^2 + \ldots + r^d = \frac{r^{d+1} - 1}{r - 1}$$

General Case $(q > 2)$

$$T(n) = cn \cdot \sum_{j=0}^{d} \left(\frac{q}{2}\right)^j \qquad (d = \log_2 n - 1)$$

Let $r = q/2 > 1$. Then

$$\begin{aligned}\sum_{j=0}^{d} r^j &= \frac{r^{d+1} - 1}{r - 1} \\ &\leq \tfrac{1}{r-1} r^{d+1} \\ &= \tfrac{1}{r-1} \left(\tfrac{q}{2}\right)^{\log_2 n} \\ &= \tfrac{1}{r-1} n^{\log_2 \frac{q}{2}} \\ &= O(n^{\log_2 q - 1})\end{aligned}$$

Therefore,

$$\begin{aligned}T(n) &= cn \cdot O(n^{\log_2 q - 1}) \\ &= O(n^{\log_2 q})\end{aligned}$$

E.g., $q = 3$, $T(n)$ is $O(n^{1.59})$

Summary

Useful general recurrence and its solutions:

$$\boxed{T(n) \le q \cdot T(n/2) + cn}$$

1. $q = 1$: $T(n) = O(n)$ dominated by root
2. $q = 2$: $T(n) = O(n \log n)$ same work every level
3. $q > 2$: $T(n) = O(n^{\log_2 q})$ dominated by leaves

Work at is either exponentially decreasing, staying same, or exponentially increasing with level

Algorithms with these recurrences?

1. ???
2. MSS, Mergesort
3. Integer multiplication. . .

Clicker Question

Which of the following is *not* true ?

A. $n \log n = O(n^2)$

B. $n \log n = O(n^{1.1})$

C. There exists some k such that $n \log n = \Theta(n^k)$.

D. $n \log n = \Omega(n \log \log n)$

Master Theorem

Consider the general recurrence:

$$T(n) \leq aT\left(\frac{n}{b}\right) + cn^d$$

Clicker. How many leaves are in the recursion tree?

A. $\Theta(a^{\log_b n})$

B. $\Theta(n^{\log_b a})$

C. $\Theta(b^{\log_a n})$

D. Both a and b

Master Theorem

Consider the general recurrence:

$$T(n) \le aT\left(\frac{n}{b}\right) + cn^d$$

Clicker. How much work is done outside recursion at the root of the recursion tree?

A. $\Theta(n^d)$

B. $\Theta(n^a)$

C. $\Theta(n^b)$

D. None of the above

Master Theorem

Consider the general recurrence:

$$T(n) \leq aT\left(\frac{n}{b}\right) + cn^d$$

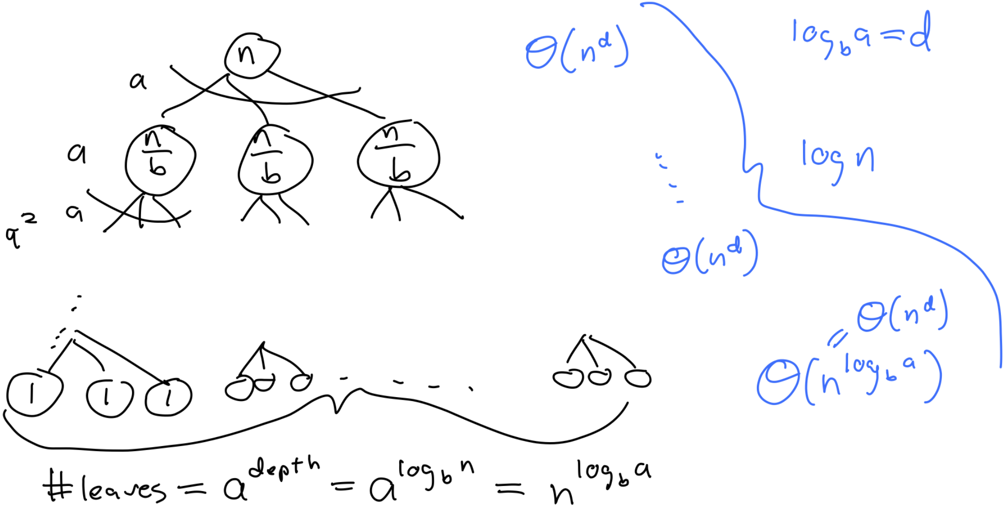
This solves to:

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

Intuition: work at each level of the recursion tree is (1) decreasing exponentially, (2) staying the same, (3) increasing exponentially.

Pick the largest of work at root vs. work at leaves; multiyply by $\log n$ if same.

Master Theorem Intuition Review



Clicker

Master Theorem:

$$T(n) \leq aT\left(\frac{n}{b}\right) + cn^d$$

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

Suppose $T(n) = 9T(n/3) + n^d$. What is the largest value for d below such that $T(n) = \Theta(n^2)$?

A. $d = 1$
B. $d = 1.5$
C. $d = 2$
D. $d = 3$