Algorithms 02

CS201

Kaustuv Nag

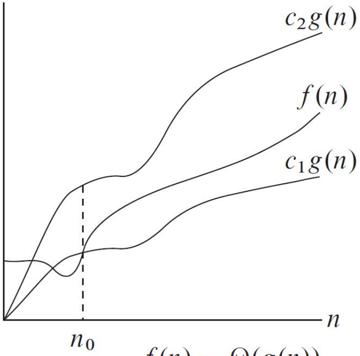
Order of Growth and Asymptotic Efficiency

- The *rate of growth*, or the *order of growth*, of the running time is a simplifying abstraction.
  - We consider only the leading term of a formula (e.g., an^2), since the lower-order terms are relatively insignificant for large values of n.
  - We also ignore the leading term's constant coefficient.
  - We usually consider one algorithm to be more efficient than another if its worstcase running time has a lower order of growth.
- When we look at input sizes large enough to make only the order of growth of the running time relevant, we study the *asymptotic efficiency* of algorithms.
  - We are concerned with how the running time of an algorithm increases with the size of the input in the limit, as the size of the input increases without bound.

Θ-notation

- For a given function $g(n)$, we denote by $\Theta(g(n))$ the set of functions

$$\Theta(g(n)) = \{f(n) : \text{there exist positive constants } c_1, c_2, \text{ and } n_0 \text{ such that } 0 \le c_1 g(n) \le f(n) \le c_2 g(n) \text{ for all } n \ge n_0\}$$



$f(n) = \Theta(g(n))$

$\Theta(g(n))$

- A function $f(n)$ belongs to the set $\Theta(g(n))$ if there exist positive constants c_1 and c_2 such that it can be "sandwiched" between $c_1 g(n)$ and $c_2 g(n)$ for sufficiently large n.
- Because $\Theta(g(n))$ is a set, we could write "$f(n) \in \Theta(g(n))$" to indicate that $f(n)$ is a member of $\Theta(g(n))$.
- Instead, we usually write "$f(n) = \Theta(g(n))$" to express the same notion.
  - This might be confusing because we *abuse* equality in this way, but doing so has its advantages.
- $g(n)$ is an *asymptotically tight bound* for $f(n)$.
- The definition of $\Theta(g(n))$ requires that every member $f(n) \in \Theta(g(n))$ be *asymptotically nonnegative*, that is, that $f(n)$ be nonnegative whenever n is sufficiently large.
  - An *asymptotically positive* function is one that is positive for all sufficiently large n.
  - the function $g(n)$ itself must be asymptotically nonnegative, or else the set $\Theta(g(n))$ is empty.

Θ-notation: An Example

$\frac{1}{2}n^2 - 3n = \Theta(n^2)$

To justify this, We must determine positive constants c_1, c_2, and n_0 such that

$$c_1 n^2 \leq \frac{1}{2}n^2 - 3n \leq c_2 n^2, \text{ for all } n \geq n_0$$

Dividing by n^2 yields

$$c_1 \leq \frac{1}{2} - \frac{3}{n} \leq c_2$$

For right-hand inequality $n \geq 1$ and $c_2 \geq \frac{1}{2}$. For $n \geq 7$ and $c_1 \leq \frac{1}{14}$. So, we choose $c_1 = \frac{1}{14}$, $c_2 = \frac{1}{2}$, and $n_0 = 7$.

Θ-notation: An Example

$6n^3 \neq \Theta(n^2)$

- Suppose, c_2 and n_0 exist such that $6n^3 \leq c_2 n^2$ for all $n \geq n_0$.
- Dividing by n^2 yields $n \leq \frac{c_2}{6}$, which cannot possibly hold for arbitrarily large n, since c_2 is constant.
- Hence, by contradiction it is proved that $6n^3 \neq \Theta(n^2)$.

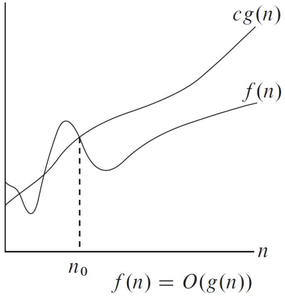
O-notation

- When we have only an *asymptotic upper bound*, we use O-notation.
- For a given function $g(n)$ we denote by $O(g(n))$ (pronounced "big-oh of g of n" or sometimes just "oh of g of n") the set of functions

$$O(g(n)) = \{f(n) : \text{there exist positive constants } c \text{ and } n_0 \text{ such that } 0 \leq f(n) \leq c(g(n)) \text{ for all } n \geq n_0\}$$

- We write $f(n) = O(g(n))$ to indicate that a function $f(n)$ is a member of the set $O(g(n))$.
- $f(n) = \Theta(g(n))$ implies $f(n) = O(g(n))$.
- $\Theta(g(n)) \subseteq O(g(n))$

O-notation



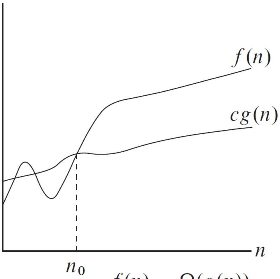
Ω-notation

- When we have only an *asymptotic lower bound*, we use Ω-notation.
- For a given function $g(n)$ we denote by $\Omega(g(n))$ (pronounced "big-omega of g of n" or sometimes just "omega of g of n") the set of functions

$$\Omega(g(n)) = \{f(n) : \text{there exist positive constants } c \text{ and } n_0 \text{ such that } 0 \leq c(g(n)) \leq f(n) \text{ for all } n \geq n_0\}$$

- We write $f(n) = \Omega(g(n))$ to indicate that a function $f(n)$ is a member of the set $\Omega(g(n))$.
- $f(n) = \Theta(g(n))$ implies $f(n) = \Omega(g(n))$.
- $\Theta(g(n)) \subseteq \Omega(g(n))$
- **Theorem:** For any two functions $f(n)$ and $g(n)$, we have $f(n) = \Theta(g(n))$ if and only if $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.

Ω-notation



$f(n) = \Omega(g(n))$

o-notation

- We define $o(g(n))$ ("little-oh of g of n") as the set

$$o(g(n)) = \{f(n) : \text{there exist positive constants } c \text{ and } n_0 \text{ such that } 0 \leq f(n) < c(g(n)) \text{ for all } n \geq n_0\}$$

- For example, $2n = o(n^2)$, but $2n^2 \neq o(n^2)$.
- $$\lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$$

ω-notation

- $f(n) \in \omega(g(n))$ if and only if $g(n) \in o(f(n))$.
- We define $\omega(g(n))$ ("little-omega of g of n") as the set

$$\omega(g(n)) = \{f(n) : \text{there exist positive constants } c \text{ and } n_0 \text{ such that}$$
$$0 \leq c(g(n)) < f(n) \text{ for all } n \geq n_0\}$$

- For example, $n^2/2 = \omega(n)$, but $n^2/2 \neq \omega(n^2)$.
-

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$$

Properties

Assume $f(n)$ and $g(n)$ are asymopotically positive.

Transitivity

- $f(n) = \Theta(g(n))$ and $g(n) = \Theta(h(n))$ imply $f(n) = \Theta(h(n))$
- $f(n) = O(g(n))$ and $g(n) = O(h(n))$ imply $f(n) = O(h(n))$
- $f(n) = \Omega(g(n))$ and $g(n) = \Omega(h(n))$ imply $f(n) = \Omega(h(n))$
- $f(n) = o(g(n))$ and $g(n) = o(h(n))$ imply $f(n) = o(h(n))$
- $f(n) = \omega(g(n))$ and $g(n) = \omega(h(n))$ imply $f(n) = \omega(h(n))$

Reflexivity

- $f(n) = \Theta(f(n))$
- $f(n) = O(f(n))$
- $f(n) = \Omega(f(n))$

Properties

Symmetry

- $f(n) = \Theta(g(n))$ if and only if $g(n) = \Theta(f(n))$

Transpose Symmetry

- $f(n) = O(g(n))$ if and only if $g(n) = \Omega(f(n))$
- $f(n) = o(g(n))$ if and only if $g(n) = \omega(f(n))$

Note

- We say that $f(n)$ is asymptotically smaller than $g(n)$ if $f(n) = o(g(n))$, and $f(n)$ is asymptotically larger than $g(n)$ if $f(n) = \omega(g(n))$.
- Although any two real numbers can be compared, not all functions are asymptotically comparable.

White Board

White Board