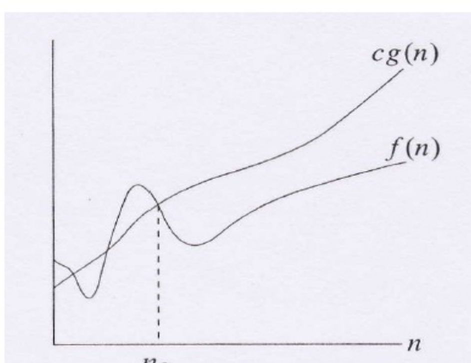
Growth of Functions

Asymptotic Notation: The order of growth of the running time of an algorithm gives a simple characterization of the algorithms's efficiency and also allows us to compare the relative performance of alternative algorithms. When we look at input sizes large enough to make only the order of growth of the running time relevant, we are studying the **asymptotic efficiency** of algorithms. In other words, we are really 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 (A way to describe the behavior of functions in the limit). Usually, an algorithm that is asymptotically more efficient will be the best choice for all but very small inputs.

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

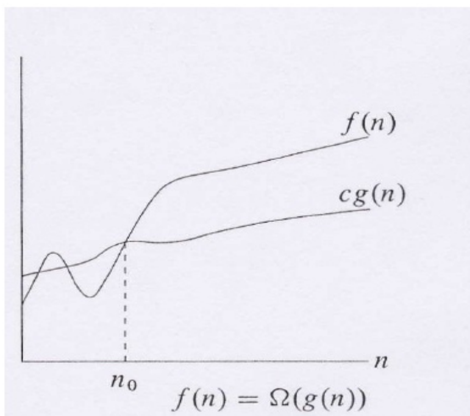


$f(n) = O(g(n))$

In other words $g(n)$ is an **asymptotic upper bound** for $f(n)$. If $f(n) \in O(g(n))$, we write $f(n) = O(g(n))$

Example: $2n^2 = O(n^2)$, with $c = 2$ and $n_0 = 1$. $n^2, n^2+n, n^2+1000n$ are functions in $O(n^2)$.

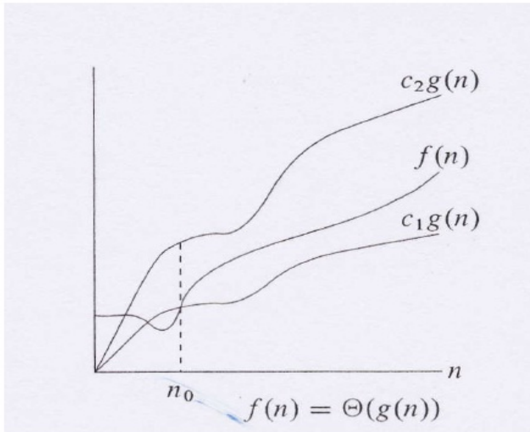
$\Omega-$ **Notation:** $\Omega(g(n)) = \{f(n)$: there exist positive constants c and n_0 such that $0 \leq cg(n) \leq f(n)$ for all $n \geq n_0\}$. In other



words $g(n)$ is an **asymptotic lower bound** for $f(n)$.

Example: $\sqrt{n} = \Omega(\lg n)$, with $c = 1$ and $n_0 = 16$. $n^2, n^2+n, n^2-n, 1000n^2+1000n$ are functions in $\Omega(n^2)$.

$\Theta-$ **Notation:** $\Theta(g(n)) = \{f(n)$: there exist positive constants c_1, c_2 and n_0 such that $0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$ for all $n \geq n_0\}$.



In other words, $g(n)$ is an **asymptotically tight bound** for $f(n)$.

Example: $\frac{n^2}{2} - 2n = \Theta(n^2)$, with $c_1 = \frac{1}{4}, c_2 = \frac{1}{2}$, and $n_0 = 8$.

Theorem: $f(n) = \Theta(g(n))$ if and only if $f = O(g(n))$ and $f = \Omega(g(n))$

Leading constants and low-order terms don't matter.

Asymptotic notation in equations

When on right-hand side: $O(n^2)$ stands for some anonymous function in the set $O(n^2)$. For example, we might write "$n = O(n^2)$" to mean $n \in O(n^2)$ and $2n^2 + 3n + 1 = 2n^2 + \Theta(n)$ means $2n^2 + 3n + 1 = 2n^2 + f(n)$ for some $f(n) \in \Theta(n)$. In particular, $f(n) = 3n + 1$

When on left-hand side: No matter how the anonymous functions are chosen on the left-hand side, there is a way to choose the anonymous functions on the right-hand side to make the equation valid. For example, we interpret $2n^2 + \Theta(n) = \Theta(n^2)$ as meaning for all functions $f(n) \in \Theta(n)$, there exists a function $g(n) \in \Theta(n^2)$ such that $2n^2 + f(n) = g(n)$. This could be chained together as: $2n^2 + 3n + 1 = 2n^2 + \Theta(n) = \Theta(n^2)$.

$o-$ **Notation:** $o(g(n)) = \{f(n)$: for all constants $c > 0$, there exists a constant $n_0 > 0$ such that $0 \leq f(n) < cg(n)$ for all $n > n_0\}$, or simply: $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$.

Example: $n^{1.9999} = o(n^2)$, $\frac{n^2}{\lg n} = o(n^2), n^2 \neq o(n^2)$

$\omega-$ **Notation:** $\omega(g(n)) = \{f(n)$: for all constants $c > 0$, there exists a constant $n_0 > 0$ such that $0 \leq cg(n) < f(n)$ for all $n > n_0\}$, or simply: $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$.

Example: $n^{2.0001} = \omega(n^2)$, $n^2 \lg n = \omega(n^2), n^2 \neq \omega(n^2)$

Comparing Functions Many of the relational properties of real numbers apply to asymptotic comparisons as well. If we assume that $f(n)$ and $g(n)$ are asymptotically positive, then

Transitivity

$f(n) = O(g(n))$ and $g(n) = O(h(n))$ imply $f(n) = O(h(n))$

$f(n) = \Omega(g(n))$ and $g(n) = \Omega(h(n))$ imply $f(n) = \Omega(h(n))$

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

Reflexivity

$f(n) = O(f(n))$

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

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

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

Hence, a way to compare "sizes" of functions f and g and the comparison of two real numbers a and b:

$f(n) = O(g(n))$ is like $a \leq b$

$f(n) = \Omega(g(n))$ is like $a \geq b$

$f(n) = \Theta(g(n))$ is like $a = b$

$f(n) = o(g(n))$ is like $a < b$

$f(n) = \omega(g(n))$ is like $a > b$

We say $f(n)$ is **asymptotically smaller** than $g(n))$ if $f(n) = o(g(n))$, and $f(n)$ is **asymptotically larger** than $g(n)$ if $f(n) = \omega(g(n))$.

Finally, one must notice that, although any two real numbers can be compared, not all functions are asymptotically comparable. For two functions $f(n)$ and $g(n)$, it may be the case that neither $f(n) = O(g(n))$ nor $f(n) = \Omega(g(n))$ holds. for example, we can-not compare the functions $f(n) = n$ and $g(n) = n^{1+\sin n}$ using asymptotic notation, since the value of the exponent in $n^{1+\sin n}$ oscillates between 0 and 2, taking on all values in between.