# Lecture Notes for Chapter 3: Growth of Functions

## Chapter 3 overview

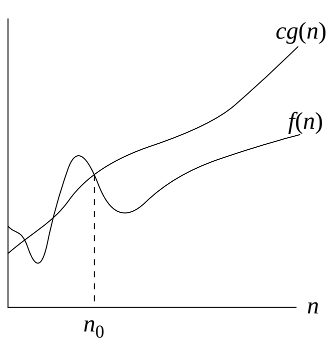
- A way to describe behavior of functions *in the limit*. We're studying ***asymptotic*** efficiency.
- Describe *growth* of functions.
- Focus on what's important by abstracting away low-order terms and constant factors.
- How we indicate running times of algorithms.
- A way to compare "sizes" of functions:

$$\begin{array}{ccc} O & \approx & \leq \\ \Omega & \approx & \geq \\ \Theta & \approx & = \\ o & \approx & < \\ \omega & \approx & > \end{array}$$

## Asymptotic notation

### *O*-notation

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




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

If $f(n) \in O(g(n))$, we write $f(n) = O(g(n))$ (will precisely explain this soon).

***Example***

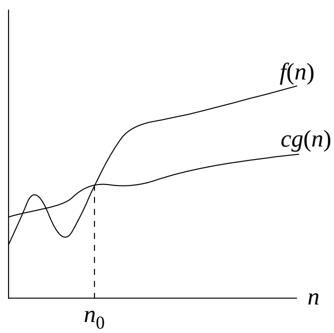
$2n^2 = O(n^3)$, with $c = 1$ and $n_0 = 2$.

Examples of functions in $O(n^2)$:

$n^2$
$n^2 + n$
$n^2 + 1000n$
$1000n^2 + 1000n$
Also,
$n$
$n/1000$
$n^{1.99999}$
$n^2 / \lg\lg\lg n$

### $\Omega$-notation

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




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

***Example***

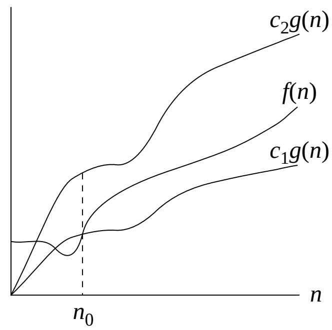
$\sqrt{n} = \Omega(\lg n)$, with $c = 1$ and $n_0 = 16$.

Examples of functions in $\Omega(n^2)$:

$n^2$
$n^2 + n$
$n^2 - n$
$1000n^2 + 1000n$
$1000n^2 - 1000n$
Also,
$n^3$
$n^{2.00001}$
$n^2 \lg\lg\lg n$
$2^{2^n}$

### $\Theta$-notation

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




$g(n)$ is an ***asymptotically tight bound*** for $f(n)$.

***Example***

$n^2/2 - 2n = \Theta(n^2)$, with $c_1 = 1/4$, $c_2 = 1/2$, 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 $O(n^2)$.

$2n^2 + 3n + 1 = 2n^2 + \Theta(n)$ means $2n^2 + 3n + 1 = 2n^2 + f(n)$ *for some* $f(n) \in \Theta(n)$. In particular, $f(n) = 3n + 1$.

By the way, we interpret # of anonymous functions as = # of times the asymptotic notation appears:

| | |
|---|---|
| $\sum_{i=1}^{n} O(i)$ | OK: 1 anonymous function |
| $O(1) + O(2) + \cdots + O(n)$ | not OK: $n$ hidden constants $\Rightarrow$ no clean interpretation |

***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.

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 $2n^2 + f(n) = g(n)$.

Can chain together:

$$
\begin{aligned}
2n^2 + 3n + 1 &= 2n^2 + \Theta(n) \\
&= \Theta(n^2) .
\end{aligned}
$$

Interpretation:

- First equation: There exists $f(n) \in \Theta(n)$ such that $2n^2 + 3n + 1 = 2n^2 + f(n)$.
- Second equation: For all $g(n) \in \Theta(n)$ (such as the $f(n)$ used to make the first equation hold), there exists $h(n) \in \Theta(n^2)$ such that $2n^2 + g(n) = h(n)$.

### $o$-notation

$$
o(g(n)) = \{f(n) : \text{ for all constants } c > 0, \text{ there exists a constant } n_0 > 0 \text{ such that } 0 \le f(n) < cg(n) \text{ for all } n \ge n_0\} .
$$

Another view, probably easier to use: $\lim_{n\to\infty} \dfrac{f(n)}{g(n)} = 0$.

$n^{1.9999} = o(n^2)$
$n^2/\lg n = o(n^2)$
$n^2 \ne o(n^2)$ (just like $2 \not< 2$)
$n^2/1000 \ne o(n^2)$

### $\omega$-notation

$$
\omega(g(n)) = \{f(n) : \text{ for all constants } c > 0, \text{ there exists a constant } n_0 > 0 \text{ such that } 0 \le cg(n) < f(n) \text{ for all } n \ge n_0\} .
$$

Another view, again, probably easier to use: $\lim_{n\to\infty} \dfrac{f(n)}{g(n)} = \infty$.

$n^{2.0001} = \omega(n^2)$
$n^2 \lg n = \omega(n^2)$
$n^2 \ne \omega(n^2)$

### Comparisons of functions

Relational properties:

**Transitivity:**
$f(n) = \Theta(g(n))$ and $g(n) = \Theta(h(n)) \Rightarrow f(n) = \Theta(h(n))$.
Same for $O, \Omega, o$, and $\omega$.

**Reflexivity:**
$f(n) = \Theta(f(n))$.
Same for $O$ and $\Omega$.

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

Comparisons:

- $f(n)$ is ***asymptotically smaller*** than $g(n)$ if $f(n) = o(g(n))$.
- $f(n)$ is ***asymptotically larger*** than $g(n)$ if $f(n) = \omega(g(n))$.

No trichotomy. Although intuitively, we can liken $O$ to $\le$, $\Omega$ to $\ge$, etc., unlike real numbers, where $a < b$, $a = b$, or $a > b$, we might not be able to compare functions.

Example: $n^{1+\sin n}$ and $n$, since $1 + \sin n$ oscillates between 0 and 2.

## Standard notations and common functions

*[You probably do not want to use lecture time going over all the definitions and properties given in Section 3.2, but it might be worth spending a few minutes of lecture time on some of the following.]*

### Monotonicity

- $f(n)$ is ***monotonically increasing*** if $m \le n \Rightarrow f(m) \le f(n)$.
- $f(n)$ is ***monotonically decreasing*** if $m \ge n \Rightarrow f(m) \ge f(n)$.
- $f(n)$ is ***strictly increasing*** if $m < n \Rightarrow f(m) < f(n)$.
- $f(n)$ is ***strictly decreasing*** if $m > n \Rightarrow f(m) > f(n)$.

### Exponentials

Useful identities:

$$\begin{aligned} a^{-1} &= 1/a \,, \\ (a^m)^n &= a^{mn} \,, \\ a^m a^n &= a^{m+n} \,. \end{aligned}$$

Can relate rates of growth of polynomials and exponentials: for all real constants $a$ and $b$ such that $a > 1$,

$$\lim_{n\to\infty} \frac{n^b}{a^n} = 0 \,,$$

which implies that $n^b = o(a^n)$.

A suprisingly useful inequality: for all real $x$,

$$e^x \ge 1 + x \,.$$

As $x$ gets closer to 0, $e^x$ gets closer to $1 + x$.

### Logarithms

Notations:

$$\begin{aligned} \lg n &= \log_2 n && \text{(binary logarithm)} \ , \\ \ln n &= \log_e n && \text{(natural logarithm)} \ , \\ \lg^k n &= (\lg n)^k && \text{(exponentiation)} \ , \\ \lg\lg n &= \lg(\lg n) && \text{(composition)} \ . \end{aligned}$$

Logarithm functions apply only to the next term in the formula, so that $\lg n + k$ means $(\lg n) + k$, and *not* $\lg(n + k)$.

In the expression $\log_b a$:

- If we hold $b$ constant, then the expression is strictly increasing as $a$ increases.
- If we hold $a$ constant, then the expression is strictly decreasing as $b$ increases.

Useful identities for all real $a > 0$, $b > 0$, $c > 0$, and $n$, and where logarithm bases are not 1:

$$\begin{aligned} a &= b^{\log_b a} \ , \\ \log_c(ab) &= \log_c a + \log_c b \ , \\ \log_b a^n &= n \log_b a \ , \\ \log_b a &= \frac{\log_c a}{\log_c b} \ , \\ \log_b(1/a) &= -\log_b a \ , \\ \log_b a &= \frac{1}{\log_a b} \ , \\ a^{\log_b c} &= c^{\log_b a} \ . \end{aligned}$$

Changing the base of a logarithm from one constant to another only changes the value by a constant factor, so we usually don't worry about logarithm bases in asymptotic notation. Convention is to use lg within asymptotic notation, unless the base actually matters.

Just as polynomials grow more slowly than exponentials, logarithms grow more slowly than polynomials. In $\lim_{n\to\infty} \frac{n^b}{a^n} = 0$, substitute $\lg n$ for $n$ and $2^a$ for $a$:

$$\lim_{n\to\infty} \frac{\lg^b n}{(2^a)^{\lg n}} = \lim_{n\to\infty} \frac{\lg^b n}{n^a} = 0 \ ,$$

implying that $\lg^b n = o(n^a)$.

### Factorials

$n! = 1 \cdot 2 \cdot 3 \cdot n$. Special case: $0! = 1$.

Can use ***Stirling's approximation***,

$$n! = \sqrt{2\pi n} \left(\frac{n}{e}\right)^n \left(1 + \Theta\left(\frac{1}{n}\right)\right) \ ,$$

to derive that $\lg(n!) = \Theta(n \lg n)$.