3. Growth of Functions

3.1 Asymptotic notation

$$\Theta(g(n)) = \{f(n) \mid \exists \text{ positive } c_1, c_2, n_0 \text{ s.t.}$$
$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \quad \forall n \geq n_0\}$$

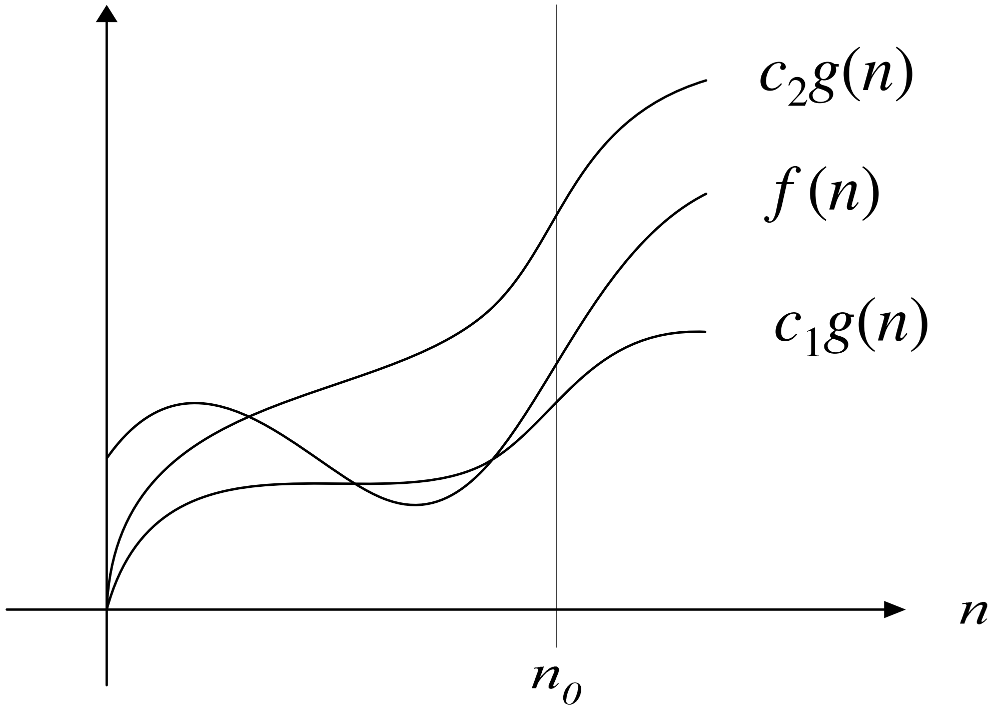
We write $f(n) = \Theta(g(n))$ or $f(n) \in \Theta(g(n))$

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

Note: "=" abused, e.g.,

$f(n) = \Theta(g(n))$ and $h(n) = \Theta(g(n))$, but $f(n) = h(n)$?

- The definition requires every member to be asymptotically nonnegative.



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

Example:

$$\frac{n^2}{14} \le \frac{n^2}{2} - 3n \le \frac{n^2}{2} \; if \;\; n \ge 7.$$

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

$$f(n) = an^2 + bn + c, a, b, c \text{ constants}, a>0.$$

$$\Rightarrow f(n) = \Theta(n^2).$$

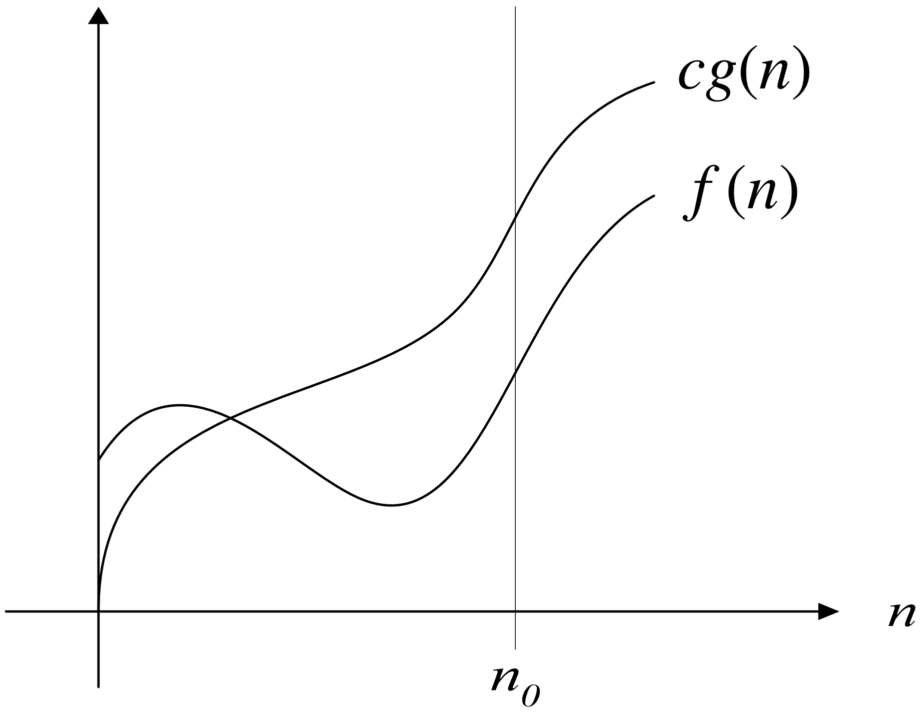
- In general,

$$p(n) = \sum_{i=0}^{d} a_i n^i \text{ where } a_i \text{ are constant with } a_d > 0.$$

Then $p(n) = \Theta(n^d)$.

asymptotic upper bound

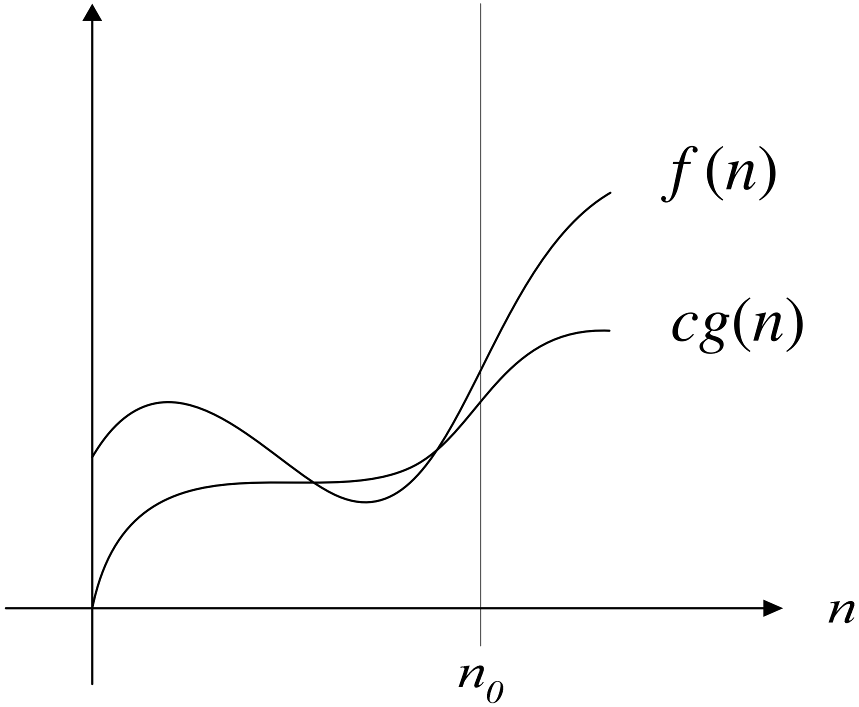
$$O(g(n)) = \{ f(n) \mid \exists c, n_0 > 0 \text{ s.t. } 0 \le f(n) \le cg(n) \; \forall n \ge n_0 \}$$



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

asymptotic lower bound

$$\Omega(g(n)) = \{ f(n) \mid \exists c, n_0 > 0 \text{ s.t. } 0 \le cg(n) \le f(n)\ \forall n \ge n_0 \}$$



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

Theorem 3.1.

- For any two functions $f(n)$ and $g(n)$,
  $f(n) = \Theta(g(n))$ if and only if
  $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.

Asymptotic notation in equations and inequalities

- $2n^2 + 3n + 1 = 2n^2 + \Theta(n)$
- $T(n) = 2T(n / 2) + \Theta(n)$
- $2n^2 + 3n + 1 = 2n^2 + \Theta(n) = \Theta(n^2)$
  - *Interpretation: no matter how the anonymous functions are chosen on the left of the equal sign, there is a way to choose the anonymous functions on the right of the equal sign to make the equation valid.*

o-notation, ω-notation

- $o(g(n)) = \{ f(n) \mid \forall c > 0, \exists n_0 \forall n \geq n_0, 0 \leq f(n) \leq cg(n) \}$
- $f(n) = o(g(n)) \Leftrightarrow \lim_{n\to\infty} \dfrac{f(n)}{g(n)} = 0$
- $\omega(g(n)) = \{ f(n) \mid \forall c > 0, \exists n_0 \forall n \geq n_0, 0 \leq cg(n) \leq f(n) \}$
- $f(n) = \omega(g(n)) \Leftrightarrow \lim_{n\to\infty} \dfrac{f(n)}{g(n)} = \infty$

Some rules

- Transitivity

$$f(n)=\Theta(g(n)) \wedge g(n)=\Theta(h(n)) \Rightarrow f(n)=\Theta(h(n))$$
$$f(n)=O(g(n)) \wedge g(n)=O(h(n)) \Rightarrow f(n)=O(h(n))$$
$$f(n)=\Omega(g(n)) \wedge g(n)=\Omega(h(n)) \Rightarrow f(n)=\Omega(h(n))$$
$$f(n)=o(g(n)) \wedge g(n)=o(h(n)) \Rightarrow f(n)=o(h(n))$$
$$f(n)=\omega(g(n)) \wedge g(n)=\omega(h(n)) \Rightarrow f(n)=\omega(h(n))$$

- Reflexivity

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

- Symmetry

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

Some rules (cont.)

- Transpose symmetry

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

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

- How to remember them?

$$f(n) = O(g(n)) \quad \approx \quad a \le b$$

$$f(n) = \Omega(g(n)) \quad \approx \quad a \ge b$$

$$f(n) = \Theta(g(n)) \quad \approx \quad a = b$$

$$f(n) = o(g(n)) \quad \approx \quad a < b$$

$$f(n) = \omega(g(n)) \quad \approx \quad a > b$$

Some rules: trichotomy

- For real numbers, $a < b$, $a = b$, or $a > b$.
- For functions, can we say either

  $f(n) = O(g(n))$ or $f(n) = \Omega(g(n))$? No!

  e.g., compare $n^0, n^1, n^2, n^{1+\sin n}$

2.2 Standard notations and common functions

- Monotonicity:
  - A function f is *monotonically increasing* if $m \leq n$ implies $f(m) \leq f(n)$.
  - A function f is *monotonically decreasing* if $m \leq n$ implies $f(m) \geq f(n)$.
  - A function f is *strictly increasing* if $m < n$ implies $f(m) < f(n)$.
  - A function f is *strictly decreasing* if $m > n$ implies $f(m) > f(n)$.

Floor and ceiling

$$x-1<\lfloor x\rfloor\leq x\leq\lceil x\rceil<x+1$$

$$\lceil n/2\rceil+\lfloor n/2\rfloor=n$$

$$\lceil\lceil n/a\rceil/b\rceil=\lceil n/ab\rceil$$

$$\lfloor\lfloor n/a\rfloor/b\rfloor=\lfloor n/ab\rfloor$$

$$\lceil a/b\rceil\leq(a+(b-1))/b$$

$$\lfloor a/b\rfloor\geq(a-(b-1))/b$$

Modular arithmetic

- For any integer a and any positive integer n, the value a mod n is the **remainder** (or **residue**) of the quotient a/n :

$$a \bmod n = a - \lfloor a/n \rfloor n.$$

- If (a mod n) = (b mod n). We write $a \equiv b \pmod{n}$ and say that a is **equivalent** to b, modulo n.

- We write $a \not\equiv b \pmod{n}$ if a is not equivalent to b, modulo n.

Polynomials vs. Exponentials

- Polynomials: $P(n) = \sum_{i=0}^{d} a_i n^i$
  - A function is *polynomial bounded* if $f(n) = n^{O(1)}$.
- Exponentials:
  - $n^b = o(a^n)$ (a > 1)
  - Any positive exponential function with a base strictly greater than 1 grows faster than any polynomial function.

$$e^x = \sum_{i=0}^{\infty} \frac{x^i}{i!}$$

$$1 + x \le e^x \le 1 + x + x^2 \quad \text{if} \quad |x| \le 1$$

$$\lim_{n \to \infty} \left(1 + \frac{x}{n}\right)^n = e^x$$

Logarithms

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \frac{x^5}{5} - \ldots \quad \text{if } |x| < 1$$

$$\frac{x}{1+x} \le \ln(1+x) \le x$$

- A function $f(n)$ is *polylogarithmically bounded* if $f(n) = \log^{O(1)} n$.
- Any positive polynomial function grows faster than any polylogarithmic function. $\log^b n = o(n^a)$

Factorials

- Stirling's approximation

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

$$n! = o(n^n)$$

$$n! = \omega(2^n)$$

$$\log(n!) = \Theta(n \log n)$$

$$n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\alpha_n}$$

where

$$\frac{1}{12n+1} < \alpha_n < \frac{1}{12n}$$

Function iteration

$$f^{(i)}(n) = \begin{cases} n & \text{if } i = 0, \\ f(f^{(i-1)}(n)) & \text{if } i > 0. \end{cases}$$

For example, if $f(n) = 2n$, then $f^{(i)}(n) = 2^i n$

The iterated logarithm function

$$\lg^{(i)}(n) = \begin{cases} n & \text{if } i = 0 \\ \lg(\lg^{(i-1)} n) & \text{if } i > 0 \text{ and } \lg^{(i-1)} n > 0 \\ \text{undefined} & \text{if } i > 0 \text{ and } \lg^{(i-1)} n \leq 0 \\ & \text{or } \lg^{(i-1)} n \text{ is undefined.} \end{cases}$$

$$\lg^*(n) = \min\{i \geq 0 \mid \lg^{(i)}(n) \leq 1\}$$

$$\lg^* 2 = 1$$

$$\lg^* 4 = 2$$

$$\lg^* 16 = 3$$

$$\lg^* 65536 = 4$$

$$\lg^* 2^{65536} = 5$$

Check also Ackermann function!

The iterated logarithm function (cont.)

- Since the number of atoms in the observable universe is estimated to be about 10^{80}, which is much less than 2^{65536}, we rarely encounter an input size of n such that $\lg^* n > 5$.

Fibonacci numbers

$$F_0 = 0$$

$$F_1 = 1$$

$$F_i = F_{i-1} + F_{i-2} \quad \text{for } i \geq 2$$

$$F_i = \frac{\phi^i - \widehat{\phi}^i}{\sqrt{5}}$$

$$\phi = \frac{1+\sqrt{5}}{2} = 1.61803...$$

$$\widehat{\phi} = \frac{1-\sqrt{5}}{2} = -0.61803...$$