Growth of functions

CSE 3318 – Algorithms and Data Structures
Alexandra Stefan

Based on presentations by
Vassilis Athitsos and Bob Weems

University of Texas at Arlington

Math Background & book reference

L'Hopital rule:

If $\lim_{n\to\infty} f(n)$ and $\lim_{n\to\infty} g(n)$ are both 0 or $\pm\infty$

and if $\lim_{n\to\infty} \frac{f'(n)}{g'(n)}$ is a constant or $\pm\infty$

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

- Limits
  - From the Limits cheat sheet see:
    - Properties,
    - Basic limit evaluations at $\pm\infty$ (focus on the +),
    - **Polynomials at infinity (in Evaluation techniques)**
  - L'Hopital's rule on wikipedia
- Derivatives
  - needed to Apply L'Hopital's rule
  - From the Derivatives cheat sheet see:
    - "Basic Properties and Formulas" and
    - "Common Derivatives" (especially for: polynomial, logarithmic and exponential functions)
- Logarithm properties
  - See the class cheat sheet
- Cheat sheets and other useful links are on the Slides and Resources webpage.
  - Math cheat sheets and links: Integrals, Derivatives , Limits Algebraic properties of equality, L'Hospital's rule on wikipedia.
- Book: Chapter 3.2 has a useful math review

Book

- Read chapter 3
  - Including 3.2 which has useful math review

Motivation

Understand the formal meaning of Θ,Ω,O, ω, o.

Be able to understand (read and calculate) Big-Oh notation. E.g.

- Alg 1 is $O(N^2)$
- Alg 2 is $\Theta(N\lg N)$
- Alg 3 is $\Omega(N)$

Function plots

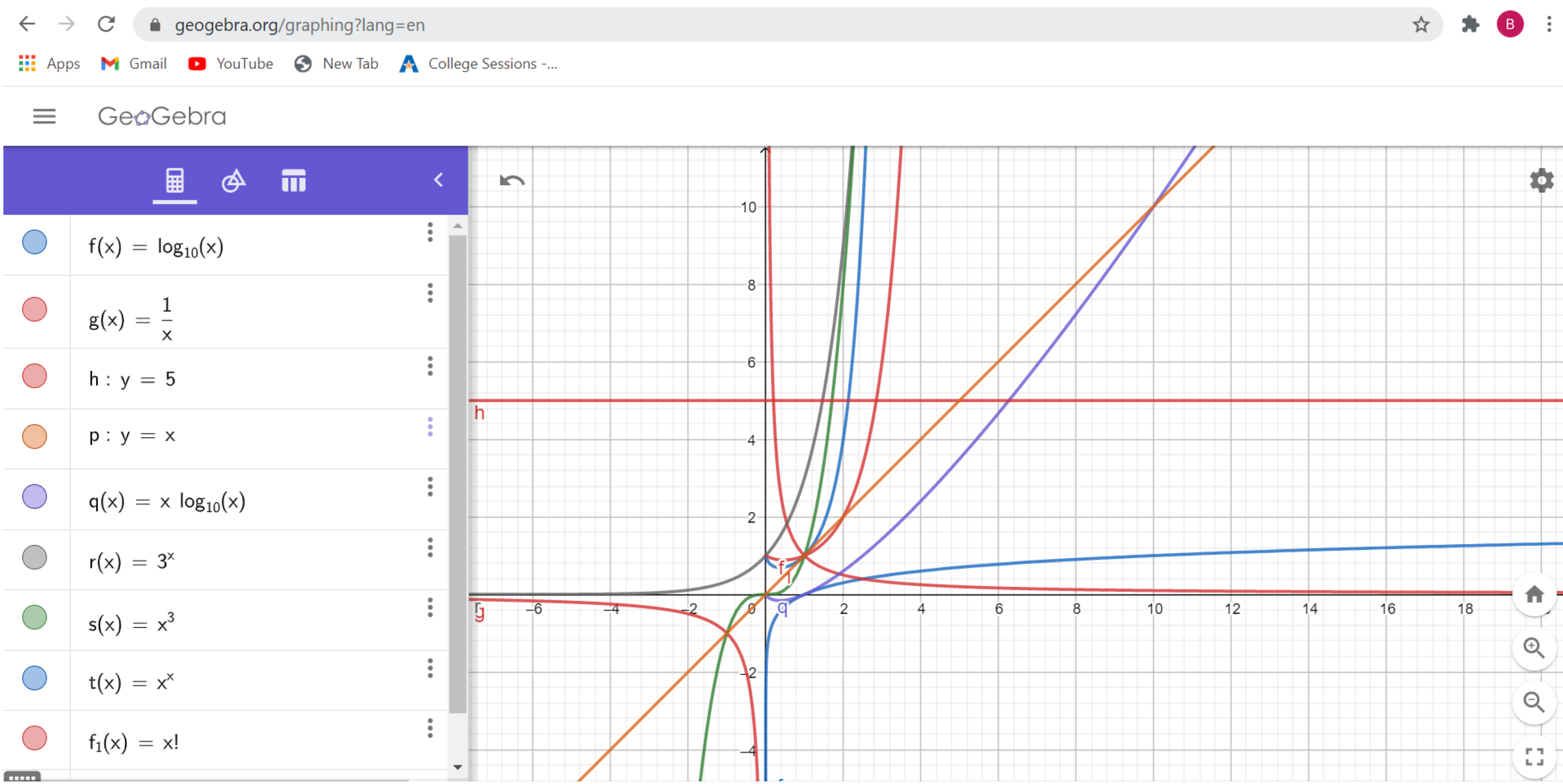


Image from Biruk Kebede

Asymptotic Bounds Notation

- Theta – is called an ***asymptotic tight*** bound
- Goal: we want to express lower and upper bounds as well. E.g.:
  - Selection sort will take time *strictly* proportional to n^2 $\in \Theta(n^2)$
  - Insertion sort will take time *at most* proportional n^2 $\in O(n^2)$
    - Use big-Oh for upper bounding complex functions of n.
    - Note that we can still say that the worst case for insertion sort is $\Theta(n^2)$.
  - Any sorting algorithm will take time *at least* proportional to n. $\in \Omega(n)$
- Math functions that are:
  - $\Theta(n^2)$:
  - $O(n^2)$:
  - $\Omega(n^2)$:

Abuse notation:
$$f(n) = O(g(n))$$
instead of:
$$f(n) \in O(g(n))$$

Informal definition:
f(n) grows 'proportional' to g(n) if:
$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = c \neq 0$$
(c is a non-zero constant)

= .... Θ tight bound
≤ .... O upper bound (big-Oh – bigger)
≥ .... Ω lower bound

Asymptotic Bounds and Notation (CLRS chapter 3)

- $f(n) = \Theta(g(n)) \Leftrightarrow \lim_{n\to\infty} \frac{f(n)}{g(n)} = c \neq 0$ (limit is a non-zero constant)
  - *g(n)* is an asymptotic **tight** bound for *f(n)*.
  - $f(n) = \Theta(g(n)) \Leftrightarrow$ there exist positive constants c_0, c_1 and n_0 such that $c_0\, g(n) \le \mathbf{f(n)} \le c_1\, g(n)\ \ \forall n \ge n_0$.
- $f(n) = O(g(n)) \Leftrightarrow \lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$ or c (limit is a constant)
  - *g(n)* is an asymptotic **upper** bound for *f(n)*.
  - $f(n) = O(g(n)) \Leftrightarrow$ there exist positive constants c_0 and n_0 such that $\mathbf{f(n)} \le c_0\, g(n)$, for all $n \ge n_0$.
- $f(N) = \Omega(g(n)) \Leftrightarrow \lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$ or c
  - *g(n)* is an asymptotic **lower** bound for *f(n)*.
  - $f(N) = \Omega(g(n)) \Leftrightarrow$ there exist positive constants c_0 and n_0 such that $c_0\, g(n) \le \mathbf{f(n)}$ for all $n \ge n_0$.

- Typically, *f(n)* is the running time of an algorithm. This can be a complicated function.
- We try to find a *g(n)* that is **simple** (e.g. n^2), and such that $f(n) = O(g(n))$. *E.g.* $7n^2+10n+26 = O(n^2)$

Asymptotic Bounds and Notation (CLRS chapter 3)

"little-oh": **o**

- **Theorem:** if $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \mathbf{0}$, then $f(n) \in o(g(n))$
- *f(n)* is **o**(g(n)) if for any positive constant c_0, there exists n_0 s.t.: $\boldsymbol{f(n)} < c_0\, g(n)$ for all $n \geq n_0$.
- *g(N)* is an asymptotic upper bound for *f(N)* (but NOT tight).
- *E.g.:* $n = o(n^2)$, $n = o(n\lg n)$, $n^2 = o(n^4)$,…

"little-omega": **ω**

- **Theorem:** if $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$, then $f(n) \in \omega(g(n))$
- *f(N)* is **ω**(*g(n)*) if for any positive constant c_0, there exists n_0 s. t.: $c_0\, g(n) < \boldsymbol{f(n)}$ for all $n \geq n_0$.
- *g(n)* is an asymptotic lower bound for *f(n)* (but NOT tight).
- *E.g.:* $n^2 = \omega(n)$, $n\lg n = \omega(n)$, $n^3 = \omega(n^2)$,…

Theta vs Big-Oh

- The Theta notation is more strict than the Big-Oh notation:
  - $n^2 = O(n^3)$ is true
  - $n^2 = \Theta(n^3)$ is false

Properties of O, Ω and Θ

1. $f(n) = \boldsymbol{O}(g(n)) \Rightarrow g(n) = \boldsymbol{\Omega}(f(n))$

2. $f(n) = \boldsymbol{\Omega}(g(n)) \Rightarrow g(n) = \boldsymbol{O}(f(n))$

3. $f(n) = \boldsymbol{\Theta}(g(n)) \Rightarrow g(n) = \boldsymbol{\Theta}(f(n))$

4. *If* $f(n) = O(g(n))$ ***and*** $f(n) = \Omega(g(n)) \Rightarrow f(n) = \Theta(g(n))$

5. *If* $f(n) = \Theta(g(n)) \Rightarrow f(n) = O(g(n))$ ***and*** $f(n) = \Omega(g(n))$

Transitivity (proved in future slides):

6. *If* $f(n) = O\big(g(n)\big)$ *and* $g(n) = O(h(n))$, *then* $f(n) = O(h(n))$.

7. *If* $f(n) = \Omega\big(g(n)\big)$ *and* $g(n) = \Omega(h(n))$, *then* $f(n) = \Omega(h(n))$.

Simplifying Big-Oh Notation

- Let $f(n) = 35n^2 + 41n + lg(n) + 1532.$
- We say that $f(n) = O(n^2).$
- Also correct, but too detailed (do not use them):
  - $f(n) = O(n^2+n)$
  - $f(n) = O(35n^2 + 41n + lg(n) + 1532).$

Polynomial functions

- If *f(n)* is a polynomial function, then it is ***Θ*** of the dominant term.

- E.g. $f(n) = 15n^3 + 7n^2 + 3n + 20$,

  find g(n) s.t. $f(n) = \Theta(g(n))$:
  - find the dominant term: $15n^3$
  - Ignore the constant, left with: n^3
  - => $g(n) = n^3$
  - => $f(n) = \Theta(n^3)$

You cannot use the dominant term method if f(n) is a summation that has a number of terms that depends on n.
E.g.: $f(n) = n^2 + (n-1)^2 + \ldots + 2^2 + 1$

See Summations for techniques for solving these.

Big-Oh Hierarchy

- $1 = O(\lg(n))$
- $lg(n) = O(n)$
- $n = O(n^2)$
- If $0 \leq c \leq$d, then $n^c = O(n^d)$.
  - Higher-order polynomials always get larger than lower-order polynomials, eventually.
- For any d, if $c > 1$, $n^d = o(c^n)$. (e.g. $n^{100} = o(2^n)$)
  - Exponential functions always get larger than polynomial functions, eventually.
- You can use these facts in your assignments.
- You can apply transitivity to derive other facts, e.g., that $lg(n) = O(n^2)$.

$1/n$, 1, lgn , n^ε, $\sqrt{n}$, n $nlgn$, n^2, n^3, n^d, c^n, $n!$, n^n *(where $0<\varepsilon<0.5$)*

n!

- Compare the following functions (in terms of o,ω)

$$n!\,, \qquad 2^n, \qquad n^n$$

- We can upper and lower bound n!

$n! = o(n^n)$ $\qquad$ *n^n is a strictly upper bound for n!*
$n! = \omega(2^n)$ $\qquad$ *2^n is a strictly lower bound for n!*

- Extra material:
  - We have a Θ bound on lg(n!):

$$\lg(n!) = lg1 + lg2 + lg3 + \ldots + \lg(n-1) + lgn = \Theta(n \lg n)$$

  - But not on n!: $n! \neq \Theta(n^n)$
  - To understand why, see this case, consider:

$$\lg(n^2) = \Theta(\lg(n)) \text{ but}$$
$$n^2 \neq \Theta(n)$$

Useful logarithm properties

- $c^{lg(n)} = n^{lg(c)}$
  - Proof: apply lg on both sides and you get two equal terms:

$$lg(c^{lg(n)}) = lg(n^{lg(c)}) \qquad =>$$

$$lg(n) * lg(c) = lg(n) * lg(c)$$

  - This equality helps identify "false exponentials". E.g. $3^{lg(n)}$ *may look like an exponential growth, but is really polynomial:* $n^{lg(3)}$ *.*

- Can we also say that $c^n = n^c$?
  - NO!

Proofs using the c definition: O

- Let $f(n) = 35n^2 + 41n + lg(n) + 1532.$

  Show (using the definition) that $f(n) = O(n^2)$.

- Proof:

  Want to find n_0 and c_0 s.t., for all $n \geq n_{0:}$ $f(n) \leq c_0 n^2$.

  Upperbound each term in the f(n) expression.

Version 1:

- Pick c_0 large enough to cover the coefficients of all the terms:

  $f(n) = 35n^2 + 41n + lg(n) + 1532 \leq$

  $\leq 35n^2 + 41n^2 + n^2 + 1532n^2 = 1609n^2$

  => *use* $c_0 = 1609$ and $n_0 = 1$

Version 2:

- *Upper bound each term by n^2 for large n (e.g. $n \leq 1532$)*

$f(n) = 35n^2 + 41n + lg(n) + 1532$

$\leq 35n^2 + n^2 + n^2 + n^2 = 38n^2$ *for all* $n \geq 1536$

=> $f(n) = 35n^2 + 41n + lg(n) + 1532 \leq 38n^2$, *for all* $n \geq 1536$ (here $c_0 = 38$, $n_0 = 1536$)

Proofs using the c definition: *Ω, Θ*

- Let $f(n) = 35n^2 + 41n + lg(n) + 1532.$

- Proof that $\boldsymbol{f(n) = \Omega(n^2)}$:

  Want to find n_1 and c_1 s.t., for all $n \geq n_1$: $f(n) \geq c_1 n^2$.
  - Use: $c_1 = 1$, $n_1 = 1$

    $f(n) = 35n^2 + 41n + lg(n) + 1532 \geq n^2$, *for all* $n \geq 1$

- Proof that $\boldsymbol{f(n) = \Theta(n^2)}$ **:**

Version 1: use *property 4, page 10:*

We have proved $f(n) = O(n^2)$ *and* $f(n) = \Omega(n^2)$, therefore $f(n) = \Theta(n^2)$ holds

Version 2: We found $c_0 = 38$, $n_0 = 1536$ and $c_1 = 1$, $n_1 = 1$ s.t.:

$f(n) = 35n^2 + 41n + lg(n) + 1532 \leq 38n^2$, *for all* $n \geq 1536$

$f(n) = 35n^2 + 41n + lg(n) + 1532 \geq n^2$, *for all* $n \geq 1$

$\Rightarrow n^2 \leq f(n) \leq 38n^2$, *for all* $n \geq 1536$ $\Rightarrow f(n) = \Theta(n^2)$

Using Limits: Example 1

- Suppose that we are given this running time:
  $f(n) = 35n^2 + 41n + lg(n) + 1532.$
- Use the limits theorem to show that $f(n) = O(n^2)$.

Summary

- Definitions
- Properties: transitivity, reflexivity, …
- Using limits
- Big-Oh hierarchy
- Example problems
- Asymptotic notation for two parameters
- $a^{log_b(n)} = n^{log_b(a)}$ $(a^n \neq n^a)$ (note $\log_b$ in the exponent)

EXTRA
NOT REQUIRED

Example Problem 1

- Is $n = O(\sin(n)\, n^2)$?
- Answer:

Example Problem 2

- Show that $max(f(n), g(n))$ is $\Theta(f(n) + g(n))$
  - Show O:

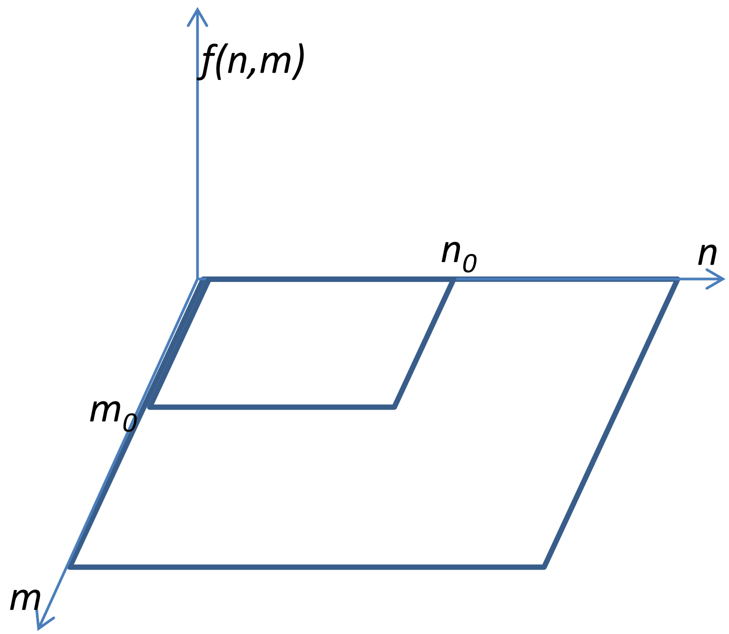
  - Show Ω:

Asymptotic notation for two parameters (CLRS)

$f(n,m)$ is $O(g(n,m))$ if there exist constants c_0, n_0 and m_0 such that:

$f(n,m) \le c_0 g(n,m)$ for all pairs (n,m) s.t.

either $n \ge n_0$ or $m \ge m_0$



Using Limits

- if $\lim_{n\to\infty} \frac{f(n)}{g(n)}$ = c is a **non-zero** constant, then f(n) = ____(g(n)).
  - In this definition, both zero and infinity are excluded.
  - In this case we can also say that g(n) = Θ(f(n)).
    This can easily be proved using the limit or the reflexivity property of Θ.

- if $\lim_{n\to\infty} \frac{g(n)}{f(n)}$ = c is a constant, then f(n) = _____(g(n)).
  - "constant" includes zero, but cannot be infinity.

- if $\lim_{n\to\infty} \frac{g(n)}{f(n)} = \infty$ then f(n) = _____(g(n)).
  - g(n) grows much faster than f(n)

- if $\lim_{n\to\infty} \frac{f(n)}{g(n)}$ = c is a constant, then f(n) = _____(g(n)).
  - "Constant" includes zero, but cannot be infinity.

Using Limits

- if $\lim_{n \to \infty} \frac{f(n)}{g(n)}$ = c is a **non-zero** constant, then f(n) = **Θ**(g(n)).
  - In this definition, both zero and infinity are excluded.
  - In this case we can also say that g(n) = Θ(f(n)).

    This can easily be proved using the limit or the reflexivity property of Θ.

- if $\lim_{n \to \infty} \frac{g(n)}{f(n)}$ = c is a constant, then f(n) = **Ω**(g(n)).
  - "constant" includes zero, but cannot be infinity.

- if $\lim_{n \to \infty} \frac{g(n)}{f(n)} = \infty$ then f(n) = **o**(g(n)).
  - g(n) grows much faster than f(n)

- if $\lim_{n \to \infty} \frac{f(n)}{g(n)}$ = c is a constant, then f(n) = **O**(g(n)).
  - "Constant" includes zero, but cannot be infinity.

Asymptotic Notation in Expressions (if needed)

In the recurrence formulas and proofs, you may see these notations (see CLRS, page 49):

- $f(n) = 2n^2 + \Theta(n)$
  - There is a function $h(n)$ in $\Theta(n)$ s.t. $f(n) = 2n^2 + h(n)$
- $2n^2 + \Theta(n) = \Theta(n^2)$.
  - For any function $h(n)$ in $\Theta(n)$, there is a function $g(n)$ in $\Theta(n^2)$ s.t. $2n^2 + h(n) = g(n)$.
  - For any function $h(n)$ in $\Theta(n)$, $2n^2 + h(n)$ is in $\Theta(n^2)$.

Big-Oh Transitivity - Proof

- If $f(n) = O\big(g(n)\big)$ and $g(n) = O(h(n))$, then $f(n) = O(h(n))$.

Proof:

We want to find c_3 and n_3 s.t. $f(n) \le c_3 h(n)$, for all $n \ge n_3$.

We know:

$f(n) = O\big(g(n)\big)$ => there exist c_1, n_1, s.t. $f(n) \le c_1 g(n)$, for all $n \ge n_1$

$g(n) = O\big(h(n)\big)$ => there exist c_2, n_2, s.t. $g(n) \le c_2 h(n)$, for all $n \ge n_2$

$\Rightarrow f(n) \overset{\boxed{n \ge n_1}}{\le} c_1 g(n) \overset{\boxed{n \ge n_2}}{\le} c_1 c_2 h(n)$, for all $n \ge \max(n_1, n_2)$

$\Rightarrow$ Use: $c = c_1 * c_2$, and $n \ge max(n_1, n_2)$

Extra
Using Substitutions

- If $\lim_{x \to \infty} h(x) = \infty$, and *h(x)* is monotonically increasing then:

$$f(\boldsymbol{x}) = O(g(\boldsymbol{x})) \Rightarrow f(h(x)) = O(g(h(x))).$$

(This can be proved)

- How do we use that?
- For example, prove that:

$$(\lg n)^{10} = O(n)$$

$$(\, for : n^2 (\lg n)^{10} = O(n^3))$$

Proof: Use substitution: $h(n) = \lg(n)$

and: $y^{10} = O(2^y)$

(y = h(n))