Asymptotics and Recurrence Equations

Analysis of Algorithms

Prepared by
John Reif, Ph.D.
Asymptotics and Recurrence Equations

a) Computational Complexity of a Program
b) Worst Case and Expected Bounds
c) Definition of Asymptotic Equations
d) Solution of Recurrence Notation
Readings

- Main Reading Selection:
  - CLR, Chapter 3, 4 and Appendix A
Goal of Asymptotic Notation

- To estimate and compare growth rates of functions
- Ignore constant factors of growth
“f(n) is asymptotically equal to g(n)”

\[ f(n) \sim g(n) \]

if

\[ \lim_{n \to \infty} \frac{f(n)}{g(n)} = 1 \]
"f(n) is little o g(n)"

- f(n) is o(g(n)) if \( \lim_{{n \to \infty}} \frac{f(n)}{g(n)} = 0 \)
"f(n) is \textit{big O} g(n)"

- f(n) is \textit{O}(g(n)) if

\[
\exists c, n_0 > 0 \\
\text{where } f(n) \leq c \cdot g(n) \\
\text{for all } n \geq n_0
\]
Example of $f(n)$ is $O(g(n))$

$$\exists \; c, n_0 > 0$$

where $f(n) \leq c \cdot g(n)$

for all $n \geq n_0$
“f(n) is order at least g(n)”

- f(n) is $\Omega(g(n))$ if

$$\exists n_0, c > 0 \quad \text{s.t.} \quad f(n) \geq c \cdot g(n)$$
for all $n \geq n_0$
Example of $f(n)$ is $\Omega(g(n))$
“f(n) is order tight with g(n)”

- f(n) is $\Theta(g(n))$ if

$$\exists n_0, c, c' \text{ s.t. } c \cdot g(n) \leq f(n) \leq c' \cdot g(n)$$
Suppose my algorithm runs in time $O(n)$

- Don’t say:
  - “his algorithm runs in time $O(n^2)$ so is worse”
- But prove:
  - “his algorithm runs in time $\Omega(n^2)$ so is worse”

- Must find a worst case input of length $n$ for which his algorithm takes time $\geq cn^2$ for all $n \geq n_0$
Use of O Notation

- $n$ is $O(n^2)$
  sometimes written
  \[ n = O(n^2) \]

- But $n^2$ is \textit{not} $O(n)$ so \textit{can’t} use identities!

- The two sides of the equality \textit{do not} play a symmetric role
Use of Asymptotic Notation

• Write "f(n) − g(n) is o(h(n))"

• As $f(n) = g(n) + o(h(n))$

• Example

\[
\frac{n}{n-1} = 1 + \frac{1}{n} + O\left(\frac{1}{n^2}\right)
\]

\[
= 1 + \frac{1}{n} + o\left(\frac{1}{n}\right)
\]

\[
= 1 + o(1) \quad \text{as } n \to \infty
\]
Asymptotic Bounds of Sums and Products
Convergent Power Sum

\[ \sum_{i=0}^{\infty} x^i = \frac{1}{1-x} \leq O(1), \text{ for } 0 < x < 1 \]

- A polynomial is asymptotically equal to its leading term as \( x \to \infty \)

\[ \sum_{i=0}^{d} a_i x^i = \theta \left( x^d \right) \]

\[ \sum_{i=0}^{d} a_i x^i = o \left( x^{d+1} \right) \]

\[ \sum_{i=0}^{d} a_i x^i \sim a_d x^d \]
Sums of Powers

\[ \sum_{i=1}^{n} i = \frac{n(n+1)}{2} \]

- for \( n \to \infty \)

\[ \sum_{i=1}^{n} i^d \sim \frac{1}{d+1} n^{d+1} \]

- Or equivalently

\[ \sum_{i=1}^{n} i^d = \frac{1}{d+1} n^{d+1} + o \left( n^{d+1} \right) \]
Examples

- 2\textsuperscript{nd} order asymptotic expansion

\[
\begin{align*}
\sum_{i=1}^{n} i &\sim \frac{n^2}{2} \\
\sum_{i=1}^{n} i^2 &\sim \frac{n^3}{3}
\end{align*}
\]

\[
\sum_{i=1}^{n} i^d = \frac{1}{d+1} n^{d+1} + \frac{1}{2} n^d + o\left(n^{d-1}\right)
\]
Asymptotic Expansion of $f(n)$ as $n \to n_0$

$$f(n) \sim \sum_{i=1}^{\infty} c_i g_i(n)$$

- If
  $$(1) g_{i+1}(n) = o(g_i(n))$$
  for all $i \geq 1$

- and
  $$(2) f(n) = \sum_{i=1}^{k} c_i g_i(n) + o(g_k(n))$$
  for all $k \geq 1$$


Bounding Sums by Integrals

\[ \sum_{k=1}^{n} f(k) \leq \int_{1}^{n+1} f(x) \, dx \leq \sum_{k=1}^{n} f(k+1) \]
Bounding Sums by Integrals (cont’d)

\[ \sum_{k=1}^{n} f(k) \leq \int_{1}^{n+1} f(x) \, dx \leq \sum_{k=1}^{n} f(k+1) \]

• So

\[ \int_{1}^{n+1} f(x) \, dx - f(n+1) + f(1) \leq \sum_{k=1}^{n} f(k) \leq \int_{1}^{n+1} f(x) \, dx \]

• Example if \( f(x) = \ln(x) \) then

\[ \int \ln(x) \, dx = x\ln(x) - x \]
Bounding Sums by Integrals (cont’d)

• So
  \[ \sum_{k=1}^{n} \ln k = (n + 1) \ln(n+1) - n + \theta(\ln(n)) \]

• Since
  \[ \log(n) = \frac{\ln n}{\ln 2} \]

• So
  \[ \sum_{k=1}^{n} \log k = (n + 1) \log(n + 1) - \frac{n}{\ln 2} + \theta(\log n) \]
Other Approximations Derived from Integrals

\[ \sum_{k=1}^{n} k \log k = \frac{(n+1)^2}{2} \log (n+1) - \frac{(n+1)^2}{4 \ln 2} + \theta (n \log n) \]

- Harmonic Numbers

\[ H_n = \sum_{k=1}^{n} \frac{1}{k} \]

\[ H_n = \ln(n) + \gamma + O \left( \frac{1}{n} \right) \]

Euler's constant \( \gamma = 0.577 \ldots \)
Stirling’s Approximation for Factorial

- Factorial \( n! = 1 \cdot 2 \cdot 3 \cdot \cdots (n-1) \cdot n \)

\[
n! \sim \sqrt{2\pi n} \ n^n \ e^{-n} \quad \text{as} \ n \to \infty
\]

- So

\[
\log(n!) = n \log n - n \log e + \frac{1}{2} \log (2\pi n) + \theta (1)
\]

\[
= n \log n - \theta (n)
\]
Recurrence Equations

Approximate Solution of Recurrence Relations
Recurrence Equations (over integers)

- Homogeneous of degree $d$

\[ n > d \]

\[ x_n = a_1 x_{n-1} + a_2 x_{n-2} + ... + a_d x_{n-d} \]

- Given
  \[
  \begin{cases}
  \text{constant coefficients} & a_1, \ldots, a_d \\
  \text{initial values} & x_1, x_2, \ldots, x_d
  \end{cases}
  \]
Example: Fibonacci Sequence

- $n \geq 2$

$$F_n = F_{n-1} + F_{n-2}$$

$$F_0 = 0, \quad F_1 = 1$$
Solution of Fibonacci Sequence

$r_1 = \frac{1}{2} (1 + \sqrt{5}) = 1.618...$
$r_2 = \frac{1}{2} (1 - \sqrt{5})$

$F_n = c_1 r_1^n + c_2 r_2^n$

where

$F_0 = c_1 + c_2 = 0$
$F_1 = c_1 r_1 + c_2 r_2 = 1$

“golden ratio”
Solution of Fibonacci Sequence (cont’d)

- Hence

\[ F_n = \frac{1}{\sqrt{5}} \left[ \left( \frac{1 + \sqrt{5}}{2} \right)^n - \left( \frac{1 - \sqrt{5}}{2} \right)^n \right] \]

\[ \implies F_n \approx \frac{1}{\sqrt{5}} \left( \frac{1 + \sqrt{5}}{2} \right)^n \]
A Useful Theorem

• $c > 0, \ d > 0$

• If

\[
T(n) = \begin{cases}
c_0 & n=1 \\
aT\left(\frac{n}{b}\right) + cn^d & n>1
\end{cases}
\]

• then

\[
T(n) = \begin{cases}
\theta\left(n^{\log_b a}\right) & a > b^d \\
\theta\left(n^d \log_b n\right) & a = b^d \\
\theta\left(n^d\right) & a < b^d
\end{cases}
\]
Proof

\[ T(n) = cn^d \cdot g(n) + a^{\log_b n} \cdot d \]

• Is solution

\[ g(n) = 1 + \frac{a}{b^d} + \left( \frac{a}{b^d} \right)^2 + ... + \left( \frac{a}{b^d} \right)^{\log_b n-1} \]
Cases

(1) \( a > b^d \Rightarrow g(n) \sim \left( \frac{a}{b^d} \right)^{\log_b n - 1} \)

is last term so

\[
T(n) = \theta \left( a^{\log_b n} d \right) = \theta \left( n^{\log_b a} \right)
\]

(2) \( a = b^d \Rightarrow g(n) = \log_b n \)

so \( T(n) = \theta \left( n^d \log_b n \right) \)

(3) \( a < b^d \Rightarrow g(n) \) upper bound by \( O(1) \)

so \( T(n) = \theta \left( n^d \right) \)
Example: Mergesort

**input** list L of length N

*if* \( N = 1 \) *then* return L

*else do*

- let \( L_1 \) be the first \( \left\lfloor \frac{N}{2} \right\rfloor \) elements of L
- let \( L_2 \) be the last \( \left\lceil \frac{N}{2} \right\rceil \) elements of L

\( M_1 \leftarrow \text{Mergesort} \left( L_1 \right) \)
\( M_2 \leftarrow \text{Mergesort} \left( L_2 \right) \)

*return* Merge \( \left( M_1, M_2 \right) \)
Time Bounds of Mergesort

- Initial Value: $T(1) = c_1$

For $N > 1$,

$$T(N) \leq T\left(\frac{N}{2}\right) + T\left(\frac{\lceil N \rceil}{2}\right) + c_2 N$$

for some constants $c_1, c_2 \geq 1$.
Time Bound (cont’d)

- \( N > 1 \)

\[
T(N) \leq 2T\left(\frac{N}{2}\right) + c_2 N
\]

guess

\[
T(N) \leq a N \log N + b
\]

\[
\leq 2 \left( a \frac{N}{2} \log \left( \frac{N}{2} \right) + b \right) + c_2 N
\]

Holds if \( a = c_1 + c_2, \quad b = c_1 \)

Solution

\[
T(N) \leq (c_1 + c_2) N \log N + c_1
\]
Time Bound (cont’d)

• $N > 1$

$$T(N) \leq 2T\left(\frac{N}{2}\right) + c_2 N, \quad T(1) = c_1$$

• Transform Variables

$$n = \log N, \quad N = 2^n$$

$$n - 1 = \log N - \log 2 = \log \left(\frac{N}{2}\right)$$

• Recurrence equation:

$$X_n = T\left(2^n\right) = 2X_{n-1} + c_2 2^n$$

$$X_0 = T\left(2^0\right) = T(1) = c_1$$
• Solve by usual methods for recurrence equations

\[ X_n = O\left( n2^n \right) \]

so \( T(N) = O\left( N \log N \right) \)
Advanced Material

Exact Solution of Recurrence Relations
Homogenous Recurrence Relations (no constant additive term)

- Solve: \[ x_n = a_1 x_{n-1} + a_2 x_{n-2} + \ldots + a_d x_{n-d} \]

try \[ x_n = r^n \]

Multiply by \[ \frac{r^d}{r^n} \]

- Get characteristic equation:

\[ r^d - a_1 r^{d-1} - a_2 r^{d-2} - \ldots - a_d = 0 \]
Case of Distinct Roots

- Distinct Roots
  \( r_1, r_2, \ldots, r_d \)

\[ \Rightarrow x_n = \sum_{i=1}^{d} c_i r_i^n \]

\[ x_n \sim c_i r_i^n \]

- Where \( r_i \) is dominant root

\[ |r_i| > |r_j| \quad \forall \quad j \neq i \]
Other Case

- Roots are not distinct
  \[ r_1 = r_2 = r_3 \]

- Then solutions not independent, so additional terms:

\[ x_n = c_1 \ r_1^n + c_2 \ n \ r_1^n + c_3 \ n^2 \ r_1^n + \sum_{i=4}^{d} c_i \ r_i^n \]
Inhomogenous Recurrence Equations

- Nonzero constant term $a_0 \neq 0$

- Solution Method

1. Solve homogenous equation

\[
x_n = a_1 x_{n-1} + a_2 x_{n-2} + \ldots + a_d x_{n-d} + a_0
\]

\[
Y_n = a_1 Y_{n-1} + a_2 Y_{n-2} + \ldots + a_n Y_{n-d}
\]
Solution Method

1) Solve homogenous equation

\[ Y_n = a_1 Y_{n-1} + a_2 Y_{n-2} + \ldots + a_n Y_{n-d} \]

2) Case

\[ \sum a_i \neq 1 \], add particular solution
Solution Method (cont’d)

Case

\[ \sum a_i = 1 \], add particular solution

\[ x_n = cn = \left( \frac{a_0}{\sum ia_i} \right)^n \]

3) Add particular and homogeneous solutions, and solve for constants

This is all we usually need!!
Asymptotics and Recurrence Equations

Analysis of Algorithms

Prepared by
John Reif, Ph.D.