6.006 Introduction to Algorithms
Spring 2008

Lecture 24: Numerics II

Lecture Overview

- Review:
  - high precision arithmetic
  - multiplication
- Division
  - Algorithm
  - Error Analysis
- Termination

Review:

Want millionth digit of $\sqrt{2}$:

$$\lfloor\sqrt{2\cdot 10^{2d}}\rfloor \; d = 10^6$$

Compute $\lfloor\sqrt{a}\rfloor$ via Newton's Method

$$\begin{aligned} \chi_0 &= 1 \quad \text{(initial guess)} \\ \chi_{i+1} &= \frac{\chi_i + a/\chi_i}{2} \quad \leftarrow \text{division!} \end{aligned}$$

Converges quadratically; $\sharp$ correct digits doubles each step.

Multiplication:

1. Naive Divide & Conquer method: $\theta(d^2)$ time
2. Karatsuba: $\theta(d^{\log_2 3}) = \theta(d^{1.584...})$
3. Toom-Cook generalizes Karatsuba (break into $k \geq 2$ parts)

   $$T(d) = 5T(d/3) + \theta(d) = \theta\left(d^{\log_3 5}\right) = \theta\left(d^{1.465...}\right)$$

4. Schonhage-Strassen - almost linear! $\theta(d \lg d \lg\lg d)$ using FFT. All of these are in gmpy package
5. Furer(2007): $\theta\left(n \log n 2^{O(\log^* n)}\right)$ where $\log^* n$ is iterated logarithm. $\sharp$ times log needs to be applied to get a number that is less than or equal to 1.

High Precision Division

We want high precision rep of $\frac{a}{b}$

- Compute high-precision rep of $\frac{1}{b}$ first
- High-precision rep of $\frac{1}{b}$ means $\lfloor\frac{R}{b}\rfloor$ where R is large value s.t. it is easy to divide by R
  Ex: $R = 2^k$ for binary representations

Division

Newton's Method for computing $\frac{R}{b}$

$$\begin{aligned}
f(x) &= \frac{1}{x} - \frac{b}{R} \quad \left(\text{zero at } x = \frac{R}{b}\right) \\
f'(x) &= \frac{-1}{x^2} \\
\chi_{i+1} &= \chi_i - \frac{f(\chi_i)}{f'(\chi_i)} = \chi_i - \frac{\left(\frac{1}{\chi_i} - \frac{b}{R}\right)}{-1/{\chi_i}^2} \\
\chi_{i+1} &= \chi_i + {\chi_i}^2\left(\frac{1}{\chi_i} - \frac{b}{R}\right) = 2\chi_i - \frac{b{\chi_i}^2 \rightarrow \text{multiply}}{R \rightarrow \text{easy div}}
\end{aligned}$$

Example

Want $\frac{R}{b} = \frac{2^{16}}{5} = \frac{65536}{5} = 13107.2$

Try initial guess $\frac{2^{16}}{4} = 2^{14}$

$$\begin{aligned}
\chi_0 &= 2^{14} = 16384 \\
\chi_1 &= 2 \cdot (16384) - 5(16384)^2/65536 = \underline{12}288 \\
\chi_2 &= 2 \cdot (12288) - 5(12288)^2/65536 = \underline{13}056 \\
\chi_3 &= 2 \cdot (13056) - 5(13056)^2/65536 = \underline{13107}
\end{aligned}$$

Error Analysis

$$\begin{aligned}
\chi_{i+1} &= 2\chi_i - \frac{b{\chi_i}^2}{R} \qquad \text{Assume}\, \chi_i = \frac{R}{b}\left(1+\epsilon_i\right) \\
&= 2\frac{R}{b}\left(1+\epsilon_i\right) - \frac{b}{R}\left(\frac{R}{b}\right)^2\left(1+\epsilon_i\right)^2 \\
&= \frac{R}{b}\left(\left(2+2\epsilon_i\right) - \left(1+2\epsilon_i+{\epsilon_i}^2\right)\right) \\
&= \frac{R}{b}\left(1-{\epsilon_i}^2\right) = \frac{R}{b}\left(1+\epsilon_{i+1}\right) \text{ where } \epsilon_{i+1} = -{\epsilon_i}^2
\end{aligned}$$

Quadratic convergence; ♯ digits doubles at each step
Therefore complexity of division = complexity of multiplication

Termination

Iteration: $\chi_{i+1} = \lfloor \frac{\chi_i + \lfloor a/\chi_i \rfloor}{2} \rfloor$
Do floors hurt? Does program terminate?
Iteration is

$$\begin{aligned}
\chi_{i+1} &= \frac{\chi_i + \frac{a}{\chi_i} - \alpha}{2} - \beta \\
&= \frac{\chi_i + \frac{a}{\chi_i}}{2} - \gamma \quad \text{where}\, \gamma = \frac{\alpha}{2} + \beta\, \text{and}\, 0 \leq \gamma < 1
\end{aligned}$$

Since $\frac{a+b}{2} \geq \sqrt{ab}$, $\frac{\chi_i + \frac{a}{\chi_i}}{2} \geq \sqrt{a}$, so subtracting γ always leaves us $\geq \lfloor \sqrt{a} \rfloor$. This won't stay stuck above if $\epsilon_i < 1$ (good initial guess)