CS370 Notes

Minyang Jiang

January 16, 2017

1 Floating Point Number Systems

1.1 Introduction

$$F(\beta, t, L, u)$$

continas:

$$\begin{gathered}0 \text{ or } \pm 0.\beta_1\beta_2...\beta_t * \beta^d \\ \text{where } \beta_1 \neq 0 \\ 0 \leq \beta_i \leq \beta \\ L \leq d \leq u\end{gathered}$$

Two common systems used today - Base 2

1. Single precision: $F(2, 24, -126, 127)$
2. Double precision: $F(2, 53, -1022, 1023)$

Important concepts
If x is any real number then set $fl(x) =$ floating point representation of x
if we write:

$$x = \pm 0.x_1x_2x_3...x_tx_{t+1}... * \beta^d$$

$$fl(x) = \pm 0.x_1x_2x_3...x_t * \beta^d$$

relative error

$$\delta_x = \frac{fl(x) - x}{x}$$

$$\|\delta_x\| \leq ?$$

$$\begin{aligned}\frac{\|fl(x) - x\|}{\|x\|} &= \frac{0.00\ldots 0x_{t+1}\ldots * \beta^d}{0.x_1x_2\ldots x_{t+1}\ldots * \beta^d} \\ &= \frac{0.x_{t+1}x_{t+2}\ldots * \beta^{-t}}{x_1.x_2\ldots * \beta_{-1}} \\ \delta = \frac{fl(x) - x}{x} & \qquad \text{then } \|\delta\| \leq \epsilon = \begin{cases}\beta^{1-t} \\ \frac{\beta^{1-t}}{2}\end{cases} \\ fl(x) = x(1+\delta) & \qquad |\delta| \leq \epsilon\end{aligned}$$

What about floating point arithmetic?
    x, y real numbers, $x + y$ real
    $x \oplus y =$addition inside floating pt system $= fl(fl(x) + fl(y))$

1.2 Analysis some errors in computation

Example. Addition

$$\begin{aligned}
\left\| \frac{(x+y)-(x\oplus y)}{x+y} \right\| &= \frac{\|(x+y)-fl(fl(x)+fl(y))\|}{x+y} = \frac{\|x+y-x(1+\delta)+y(1+\delta)\|}{x+y} \\
&= \frac{\|x+y-(x+y+\delta_1 x+\delta_2 y+x\delta_3+y\delta_3+\delta_1\delta_3 x+\delta_2\delta_3 y)\|}{x+y} \\
&\leq \frac{\|\delta_1 x\|+\|\delta_2 y\|+\|\delta_3 x\|+\|\delta_3 y\|+\delta_1\delta_3 x+\|\delta 2\delta_3 y\|}{\|x+y\|} \\
&\leq \frac{(\|x\|+\|y\|)(2\epsilon+\epsilon^2)}{\|x+y\|}
\end{aligned}$$

$$\|\delta_1\| \leq \epsilon, \qquad \|\delta_2\| \leq \epsilon, \qquad \|\delta_3\| \leq \epsilon$$

$\Rightarrow$ if x and y have same sign then relative error of addition

$$\left\| \frac{x\oplus y-(x+y)}{x+y} \right\| \leq 2\epsilon+\epsilon^2$$

However if x and y have opposite sign, then you potentially have a problem particularly when $x+y\approx 0$, Situation is called **Catastrophic cancellation**

$$\begin{aligned}
x &= 0.x_1x_2\ldots x_{t-1}x_tx_{t+1}\ldots * \beta^d \\
y &= -0.x_1x_2\ldots x_{t-1}x_tx_{t+1}\ldots * \beta^d \\
x+y &= 0.00\ldots 0??\ldots * \beta^d
\end{aligned}$$

1.3 How about some algorithms?

1.3.1 Example

Given α, compute:

$$I_n = \int_0^1 \frac{x^n}{x+\alpha}dx \qquad n=0,1,\ldots,100\ldots$$

Step 1

$$I_0 = \int_0^1 \frac{1}{x+\alpha}dx = Ln(x+\alpha) = Ln(1+\alpha)-Ln(\alpha) = Ln\left(\frac{1+\alpha}{\alpha}\right)$$

e.g.

$$\begin{aligned}
\alpha = 0.5 &\text{ then } I_0 = 1.098612288668\ldots \\
\alpha = 2.0 &\text{ then } I_0 = 0.405465108108\ldots
\end{aligned}$$

Step 2 Notice:

$$I_{n+1} = \int_0^1 \frac{x^{n+1}}{x+\alpha}dx = \int_0^1 \frac{x^n(x+\alpha-\alpha}{x+\alpha}dx = \int_0^1 x^n dx - \alpha\int_0^1 \frac{x^n}{x+\alpha}dx$$

$$I_{n+1} = \frac{1}{n+1} - \alpha I_n$$

$$
\begin{aligned}
&I_0 \\
&I_1 = 1 - \alpha I_0 \\
&I_2 = \frac{1}{2} - \alpha I_1 \\
&\vdots \\
&I_{100} = \frac{1}{100} - \alpha I_0 0
\end{aligned}
$$

If $\alpha = 0.5$ then $I_{100} = 0.00664$
if $\alpha = 2.0$ then $I_{100} = 2.1 * 10^{22}$

$$\|I_n\| \leq \frac{1}{1+\alpha}$$

Let's analyze what is happening
Math:

$$I_0^{ex}, I_{n+1}^{ex} = \frac{1}{n+1} - \alpha I_n^{ex}$$

CS:

$$I_0^{app}, I_{n+1}^{app} = \frac{1}{n+1} - \alpha I_n^{aop}$$

At every step, therer is some error:

$$
\begin{aligned}
e_0 &= I_0^{ex} - I_0^{app} \\
e_n &= I_0^{ex} - I_n^{app}
\end{aligned}
$$

Notice:

$$
\begin{aligned}
e_{n+1} &= I_{n+1}^{ex} - I_{n+1}^{app} \\
&= \left(\frac{1}{n+1} - \alpha I_n^{ex}\right) - \left(\frac{1}{n+1} - \alpha I_n^{app}\right) \\
&= -\alpha I_n^{ex} + \alpha I_n^{app} \\
&= -\alpha\left(I_n^{ex} - I_n^{app}\right) \\
&= -\alpha e_n \\
&= (-\alpha)^{n+1} e_0
\end{aligned}
$$

If $\|\alpha\| \leq 1$ then $\|e_n\| \rightarrow 0$ as $n \rightarrow \inf$
If $\|\alpha\| \geq 1$ then $\|e_n\| \rightarrow \inf$ as $n \rightarrow \inf$
What to do when $\|\alpha\| \geq 1$:

$$
\begin{aligned}
I_n &= \frac{1}{\alpha(n+1)} - \frac{1}{\alpha} I_{n+1} \\
e_n &= -\frac{1}{\alpha} e_{n+1}
\end{aligned}
$$

If $\|\alpha\| \geq 1$ then work backwards, e.g. I_{100}, Do $I_{200}, I_{199} \ldots$

2 Interpolation

Given n points $(x_1, y_1), \ldots, (x_N, y_N)$ x_i distinct $x_1 < x_2 < \ldots < x_N$
$y = p(x)$, p should be 'nice'
'nice':

- polynomial; piecewise polynomial;

2.1 Polynomial Interpolation

Given n points (x_i, y_i) $i = 1, 2, \ldots, n$
Find a polynomial having degree $< n$ satisfying $p(x_i) = y_i$
Example
$(-1, 3), (1, 1), (2, 2)$

$$p(x) = c_0 + c_1 x + c_2 x^2$$
$$p(-1) = c_0 - c_1 + c_2 = 3$$
$$p(1) = c_0 + c_1 + c_2 = 1$$
$$p(2) = c_0 + 2c_1 + 4c_2 = 2$$
$$\begin{bmatrix} 1 & -1 & 1 & 3 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 2 \end{bmatrix}$$
$$p(x) = \frac{4}{3} - x + \frac{2}{3}x^2$$

Given n points $(x_1, y_1), \ldots, (x_N, y_N)$ x_i distinct $x_1 < x_2 < \ldots < x_N$

1) Does there exist a polynomial $p(x)$ of degree ¡ n which interpolate the n points?
2) If it exists then is it unique?

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$$
$$p(x) = c_1 + c_2 x + \ldots + c_n x^{n-1}$$

n unknowns
n equations

$$c_1 + c_2 x_1 + \ldots + c_n x_1^{n-1} = y_1$$
$$c_1 + c_2 x_2 + \ldots + c_n x_2^{n-1} = y_2$$
$$\vdots$$
$$c_1 + c_2 x_n + \ldots + c_n x_n^{n-1} = y_n$$

Vamdeimonde matrix

$$V \cdot \vec{c} = \vec{y}$$
$$det(V) = \Pi_{i>j}(x_i - x_j) \neq 0 \qquad \text{since all } x_i \text{ distinct}$$

2.2 Lagrange Form of Interpolating Polynomial

$$\begin{aligned} p(x) &= c_1 + c_2 x + c_3 x^2 + \ldots + c_n x^{n-1} \\ &= y_1 L_1(x) + y_2 L_2(x) + \ldots + y_n L_n(x) \end{aligned}$$

where each $L_i(x)$ is a polyominal of degree $< n$ which satisfies $L_i(x_i) = 1, L_i(x_j) = 0$ if $i \neq j$
$L_i(x) \equiv$ Lagrange polynominal

$$L_i(x) = \frac{(x - x_1)\ldots(x - x_{i-1})(x - x_{i+1})\ldots(x - x_n)}{(x_i - x_1)\ldots(x_i - x_{i-1})(x_i - x_{i+1})\ldots(x_i - x_n)}$$

2.3 Hermite Interpolation

Points: $(x_L, y_L), (x_R, y_R)$
Derivative values: S_L, S_R
$S(x)$ degree < 4, $S(X_L) = Y_L, S(X_R) = Y_R, S'(X_L) = S_L, S'(X_R) = S_R$, Find $S(X)$

$$\begin{aligned} S(x) =& c_1 + c_2(x - x_L) + c_3(x - x_L)^2 + c_4(x - x_L)^3 \\ & \begin{bmatrix} 1 & 0 & 0 & 0 & y_L \\ 0 & 1 & 0 & 0 & s_L \\ 1 & \Delta x & \Delta x^2 & \Delta x^3 & y_R \\ 0 & 1 & 2\Delta x & 3\Delta x^2 & s_R \end{bmatrix} \\ =& \begin{bmatrix} 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 & -1 \\ 1 & 2 & 4 & 8 & 4 \\ 0 & 1 & 4 & 12 & -1 \end{bmatrix} \\ y_L' =& \frac{y_R - y_L}{x_R - x_L} = slope \\ c_1 =& y_L \\ c_2 =& s_L \\ c_3 =& \frac{3y_L' - 2s_L - s_R}{\Delta x} \\ c_4 =& \frac{s_L + s_R - 2y_L'}{\delta x^2} \end{aligned}$$

Point Polynomial interpolation is good at the interpolating points but might not be very good elsewhere

We will need to try other methods to fit data, cubic splines

2.4 Cubic Splines

Given N points: (x_i, y_i)
A cubic spline is a function $S(x)$ which satisfies the following conditions

1. in each interval $[xi, x_{i+1}]$, $S(x) = S_i(x)$ is a polynomial of degree at most 3.
2. $S(x)$ interpolates the points $(x_1, y_1), \ldots, (x_N, y_N)$
3. $S'(x)$ exists and in continuous everywhere in $[x_1, x_N]$
4. $S''(x)$ exists and in continuous everywhere in $[x_1, x_N]$
5. 2 Boundary Conditions

As it stands as cubic spline problem has A unknowns and B equations, We want $A = B$
How many unknowns?
4 unknowns per interval, $N-1$ intervals $\Rightarrow 4(N-1) = 4N-4$ unknowns
How many equations?
Condition (2): 2 per interval $\Rightarrow 2(N-1) = 2N-2$
Condition (3): 1 per interier pt $N-2$
Condition (4): 1 per interier pt $N-2$
$2N-2+N-2+N-2 = 4N-6$
There fore need 2 more conditions

Typical Boundary Conditions

1. Natural spline $S''(x_1) = 0, S''(x_N) = 0$
2. Clamped spline $S'(x_1) = s_1, S'(x_N) = s_n$, s_1, s_n given
3. periodic spline $S'(x_1) = S'(x_N), S''(x_1) = S''(x_N)$
4. Not-a-knot (Matlab default) S''' continuous at x_2, x_{N-1}

How to compute a cubic spline?

1. Set a system of $4N-4$ equations in the $4N-4$ unknowns

$$S_i(x) = a_i + b_i(x-x_i) + c_i(x-x_i)^2 + d_i(x-x_i)^3$$

Solving Cost: $O(N^3)$ We want a method which compute a cubic spline using $O(N)$ operations
Set up a linear system for the unknown derivative values $s_1, s_2, \ldots, s_n$ which will be fast to solve and which will give us $S_1(x), \ldots, S_{N-1}(x)$