BAG #1

240 hours

Bit; 0 or 1
8 bit = 1 byte

$2^8 - 1$ — 255

$2^{32} - 1$ — 4 byte
— 1 byte

$= 3^{11}0^3 \times 4 - 1$
$= (1024)^3 \times 4 - 1$
$\simeq 4\,000\,000\,000$

0
1
2
3
4

0100 = 4
1011 = 8 + 0 + 2 + 1 = 11

stored no. $= (2^4 - 1) + n$

$11 = 16 - 1 + n$

$\Rightarrow n = -4$

#2

1000 0110 0010

1110

$0.5 = \frac{1}{2}$ $(\ )\frac{1}{2}^{x}$ 111

$0^4 + 0^2 + 0^0$

1010
1111
1001 −6
0110

m

1000
0111
+ 1
1000

rep	(m)
0001	
1110	
1111	

Estimate Round-off errors.

Problem

① smallest # when added to 1 that increases it

② Largest # when added to 1 that leaves it unchanged

−3
−5

$(2^m - 1) - m + 1$

$x = (e, f) = f\, b^{e-q}$

e − 4 bits − exponent

f − 3 bits

Sign − 1 bit

8 byte −

e − 11 bits

−6

0110 0.
1001
1010.

$E = 255\ (2047)$

$f \neq 0$ NaN

& $f = 0$

$E = 255\ (2047)$

$\pm \infty$

↓

Sign

$2^4 (2^{20})$

1.6×10^7

Let $b = 10$

Then $0.5 = 0.5 \times 10^0$

$\Rightarrow f = 0.5$

$e = 1$

$\frac{1}{b} < f < 1$

842 ÷ 5

Bagla #2.1

$x_1 = f_1 b^{e_1 - q}$

$x_2 = f_2 b^{e_2 - q}$

① Equilize exponents

0.32×10^{-1}
0.64×10^{0}

0.032
0.640
0.672
0.67×10^{0}

0.32×10^{-2}
0.64×10^{0}

0.0032
0.64
0.6432
0.64×10^{0}

0.64×10^{-2}
0.64×10^{0}

0.0064
0.6400
0.6464
0.65×10^{0}

(exponents differing by more than 2 have no effect.)

Def°: relative round off error

If $e > e_{max}$ then
overflow — set $e = 255$ for 8 bit
else if $e < 0$ then
underflow — 0
else
:

$x_1 = f_1 b^{e_1 - q}$
$x_2 = f_2 b^{e_2 - q}$
$\rho = x_1 \times x_2 = (f_1 \times f_2) b^{e_1 + e_2 - 2q}$

1) $e = e_1 + e_2 - q$ in $t+2$
1 for sign
1 for overflow

$= f b^{e - q}$

for the sign $(s_1 \times s_2)$ same as addy bits

requires register to be $t+1$ bit if fraction uses t bits for storing.

```
0 . . . . .
0 . . . . .
      5 x 2
    0 1 0 1
  0 1 0 1 0
  0 0 0 0 0
0 0 1 0 1 x
0 0 0 0 0 x x
0 0 0 0 0 x x x
0 0 0 0 1 0 1 0   = 10
```

$2t-1$ bit registers are needed to ~~store~~ multiply.

fast

```
    1 0 0 1
    1 0 0 0
  ---------
    0 0 0 0
  0 0 0 0 x
0 0 0 0 0 x x
1 0 0 1 x x x
---------------
1 0 0 1 0 0 0
```

$t+1$

This can be skipped but will give error

$x = s_1 f_1 b^{e_1 - q}$

$x^{-1} = s_1 f_1^{-1} b^{q - e_1}$

$= s_1 \left(f_1^{-1} \cdot \frac{1}{2}\right) b^{q - e_1 + 1}$

$\frac{1}{b} \le f_1 \le 1$

$1 \le \frac{1}{f_1} \le b$

$b^{-1} \le \frac{1}{f_1} \cdot \frac{1}{b} \le \frac{1}{2} < 1$

Error Analysis

computing error in addition

$S = \sum_{i=1}^{n} x_i$

$= x_1 + x_2 + x_3 + x_4 + \dots$

$S_i = S_{i-1} + x_i$

$S_1 = x_1$

$S_2 = x_1 \oplus x_2$

$= (x_1 + x_2)(1 + \epsilon_2)$

$S_3 = ((x_1 + x_2)(1 + \epsilon_2) + x_3)(1 + \epsilon_3)$

$S_j = x_1 \prod_{i=2}^{j} (1 + \epsilon_i) + \sum_{k=2}^{j} x_k \prod_{i=k}^{j} (1 + \epsilon_j)$

Def° $A \oplus B = (A + B)(1 + \epsilon)$

$|\epsilon_i| = |\epsilon|$ → computer constant

$S_n = \bar{S}_n + E_n$

$|x| \ge |x_i|$

$-\{x_1(n-1)\epsilon + x_2(n-1)\epsilon + \dots + x_{n-1}\epsilon\}(1+\delta) \le E_n \le \{\dots\}(1+\delta)$

$-|x|(1+\delta)\epsilon\{(n-1) + (n-1)\dots + 2 + 1\} \le E_n$

$0 < \delta \ll 1$

$n-1 + \frac{n(n-1)}{2} = \frac{(n-1)(n+2)}{2} \sim n^2$

$|E_n| \leq |x| \frac{n^2}{2} \epsilon(1+\delta)$

$\leq s_n \frac{nt}{2} (1+\delta)$ | claim: realistic $\sqrt{n}$

If you go $x_1 + x_2 + x_3 + x_4 + \cdots + x_{n-1} + x_n$

down: memory
up: accuracy + parallel

error
$\leq \bar{s}_n (\log_2 n) \epsilon(1+\delta)$

Bagla #2.2

Roundoff Errors

$\bar{x} = f(a_1, a_2 \cdots a_n)$

$x = f(a_1(1+\epsilon_1), a_2(1+\epsilon_2), \cdots, a_n(1+\epsilon_n))$

$E = x - \bar{x} \leq \sum_{i=1}^{n} \epsilon_i \left|\frac{\partial f}{\partial a_i}\right| \leq \bar{\epsilon} \sum_{i=1}^{n} \left|\frac{\partial f}{\partial a_i}\right|$

$\frac{|E|}{\bar{x}} \leq \sum_{i=1}^{n} \left|\frac{\epsilon_i}{f}\right| \left|\frac{\partial f}{\partial a_i}\right| = \text{"condition \#"} \ll 1$ (for small error)

Interval Arithmetic
* overestimates at times
* singularities

$a_i : [a_{i\,min}, a_{i\,max}]$

$x : [x_{min}, x_{max}]$

Example: Planetary Motion
(a) Gravity [Newtonian]
(b) Point mass
(c) Two body problem
(d) Radiation Pressure
(e)

Mercury:
- Precession of perihelion
- Quadrupole moment of the sun
- Many body effects
- Relativistic effects

Modelling error: Implicit

Mathematical Model [Modelling error]
↓ [Truncation error]
computation / Model
↓ Numerical Implen [Round off error]

$Area_2 = \frac{(f(x) + f(x+\Delta x))}{2} \Delta x$

$Area_3 = \frac{(f(x) + f(x+\frac{\Delta x}{2}))}{2} \frac{\Delta x}{2} + \frac{[f(x+\frac{\Delta x}{2}) + f(x+\Delta x)]}{2} \frac{\Delta x}{2}$

$= [f(x) + f(x+\Delta x)] \frac{\Delta x}{4} + \frac{\Delta x}{2} f(x+\frac{\Delta x}{2})$

$\Delta A = Area_3 - Area_2 = \frac{\Delta x}{4}\left(2f(x+\frac{\Delta x}{2}) - f(x) - f(x+\Delta x)\right)$

$\simeq -\frac{(\Delta x)^3}{2} f''(x+\frac{\Delta x}{2})$

Target is not to do worse computationally, than the math model you start with.

NB: Truncation error will increase if you decrease the steps.
Round off errors will decrease if you increase steps.

Concl: So there's a trade-off: