CS 101

Numerical Geometric Integration

Patrick Mullen
Mathieu Desbrun

Today

Numerics
- when good math goes wrong

Terminology

- good case: 2+2 = 4
- not so good case: 1+π = 4.1416
  - round-off error
- bad case: sin(0.025) = 0.025
  - truncation error
- worst case
  - both truncation and round-off

Truncation Error (warning: ambiguous term at best)

Basically, we can't do infinite sums
- finite number of terms in Taylor
- finite number of iterations in an algo.

Related to order of accuracy
- big "O"

More to come next time on this
- not in this lecture

Round-Off Error

Difference between the *calculated* and *exact* mathematical value
- finite digits to represent infinite digits
  - a form of quantization error
- commonly: floating-point arithmetic
  - as opposed to fixed-point arithmetic
    - fixed location of radix point in the string

Floating-point Representation

Two parts
- mantissa: fixed point value m (sign incl.)
- exponent: integer value e (+bias)
- represents $m.2^e$ (+subtleties…)

IEEE 754-2008
- single precision: 24 bits in m, 8 bits in e
- double precision: 53 bits in m, 11 bits in e

π = 40490FDB (in hexa)

Numerical Consequences

Simple Addition/Subtraction Wrong!

- sum of a large number and a tiny one?
  - can be the original large number
  - more likely a few digits will be lost
- Example:
  - 123456.7 + 101.7654 = 123558.5(654)
  - 123457.1467 - 123456.659 = 0.4

Numerical Consequences

Typical explicit update:

$$y_{n+1} = y_n + \delta_n$$

- δ_n a bit off, of course
- but update leads to rounding error
  - $|\delta_n| \ll |y_n|$
- if dt tiny, rounding error can be huge
  - how can we deal with this?

Compensated Summation

Crucial Idea:

capture rounding errors and feed them back

For n=0, 1, 2...

- a = y_n
- e = e + δ_n
- y_{n+1} = a + e
- e = e + (a-y_{n+1})

accumulated error so far

round-off error

Implicit Update?

Different problem, same remedy

- error mostly due to non-linear solve
  - solver returns result within threshold
- error easily computed as $\varepsilon = DEL$
- so... feed it back the next time
  - as external forcing! [Kharevych]
    - remember: DEL + forcing = 0