E4300: Numerical Methods

Prof. Marc Spiegelman: 211 Mudd
Office Hours 4-5pm Tu/Thurs

Course Website: www.ldeo.columbia.edu/~mspieg/e4300

Lecture 01: Introduction to Numerical Methods

- Introduction and Motivation
  - Examples of Numerical problems
- Overview of Course materials
- Course Logistics
- Prerequisites and "Fluency"

What are Numerical Methods?

1) An extremely broad field -- much broader than this class...

2) Generally speaking -- analysis and application of algorithms to allow computers to solve problems in math, science and engineering.

3) Strictly speaking, numerical methods don't require computers....many predate the modern electronic computer (e.g. *Newton's method* ~17th century)

4) But computers make things practical...and we will consider this class an introduction to computational math...

Why Numerical Methods?

1) Some problems have *no analytic solution*

2) Some problems are too big to be done by hand

3) Sometimes you actually want to compute the answer (rather than show it exists or is unique).

4) Numerics *complement* analytic methods but doesn't replace them

5) But you need both to *understand* your problems

Some Examples

Example #1) The Retirement problem (future value of an annuity)

$A(P,r,n)$ $\quad A = \frac{P}{r}\left((1+r)^n - 1\right)$

$r(A,P,n)$

Where

- P: incremental payment
- r: interest rate per payment period
- n: number of payments

- A: total amount after n payments

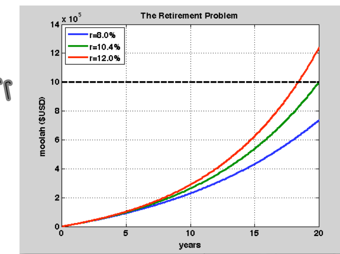
Some Examples

Example #1) The Retirement problem:

The forward problem: find A(P,r,n)

rootfinding problem

$f(r) = 0$

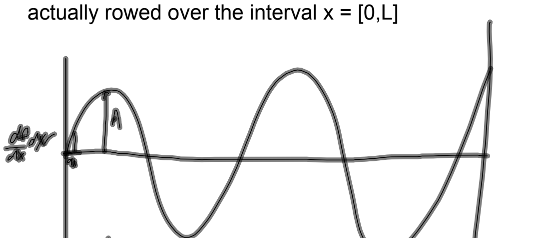


The Inverse problem: find $r(A_{target},P,n)$:
this is a rootfinding problem - no closed solution

Some Examples

Example #2) The BoatRace problem: (numerical quadrature)

Given a sinusoidal river: f(x)=Asin(x), find the total length actually rowed over the interval x = [0,L]



$d\ell = \sqrt{dx^2 + (f' dx)^2}$

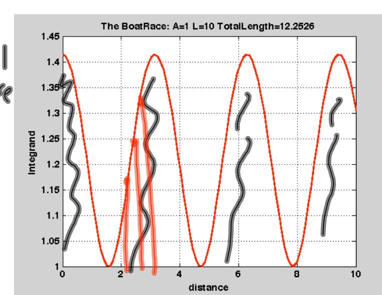
$I = \int \sqrt{1 + (f')^2}\, dx \qquad I = \int_0^L \sqrt{1 + A^2\cos^2 x}\, dx \qquad L = 10\,km$

Some Examples

Example #2) The BoatRace problem: (numerical quadrature)

Given a sinusoidal river: f(x)=Asin(x), find the total length actually rowed over the interval x = [0,L]

Numerical quadrature



Some Examples

Example #3) A simple non-linear population growth model (Lotka-Volterra Predator-Prey model)

$$\frac{dR}{dt} = R(a - bF) \qquad u = \begin{bmatrix} R \\ F \end{bmatrix}$$

$$\frac{dF}{dt} = F(cR + d) \qquad \frac{du}{dt} = f(u,t)$$

Question #1) Are there any Steady states? dR/dt=dF/dt =0 or

$$0 = Ra - bRF$$

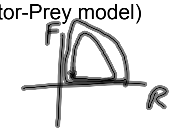
$$0 = cFR + dF$$

Solving: Systems of non-linear equations F(x)=0

Some Examples

Example #3) A simple non-linear population growth model (Lotka-Volterra Predator-Prey model)

dR/dt = R(a-bF)
dF/dt = F(dR-c)



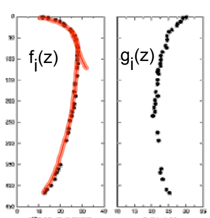
Question #2) How to solve the initial value problem R(t),F(t) given the initial condition R(0),F(0)

Question #3) How to actually understand non-linear dynamical systems!
Question #4) How to evaluate whether this is a good model of Rabbits and Foxes

Some Examples

Example #4) Interpolation/Data fitting/Numerical differentiation

Some real data somebody just sent me



Question: how to evaluate

E(z) $\propto$ d/dz (f dg/dz)/ (f-g)

! differentiation of data is a bad idea: how to fit a smooth function to the data then differentiate

The issues:
Accuracy and Efficiency

Numerical methods, invariably include an enormous range of approximations, each with attendant errors.

Good Numerical methods also return error estimates, and are stable in the presence of floating point error.

The detailed analysis of algorithms and their errors is formally Numerical Analysis

THIS IS NOT A CLASS IN NUMERICAL ANALYSIS

This is principally a Methods class where I will emphasize

- Standard Methods and their errors
- Give insight into how they work (and don't work)
- Give you practice implementing them to solve problems

If you want to design new algorithms...you're in the wrong class

Course Content:

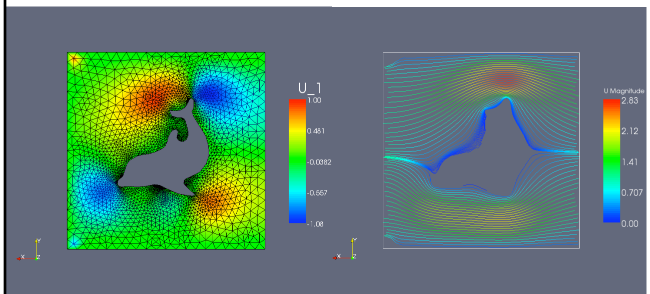
Topics Covered
1) Sources of Error and Error Analysis
2) Rootfinding/Optimization of non-linear functions of one variable, f(x)=0
3) Interpolation
4) Numerical Integration (Quadrature) and Differentiation
5) Solutions of ODE Initial value problems
6) Numerical Linear Algebra
7) Solving systems of non-linear Equations F(x)=0
8) ODE 2-point Boundary value problems (towards numerical PDE's)

Ax=b
x=A\b

Topics not-covered
1) Optimization -- linear programming, constrained optimization
2) Numerical Solution of PDE's (E4301)
3) Mathematical Modeling

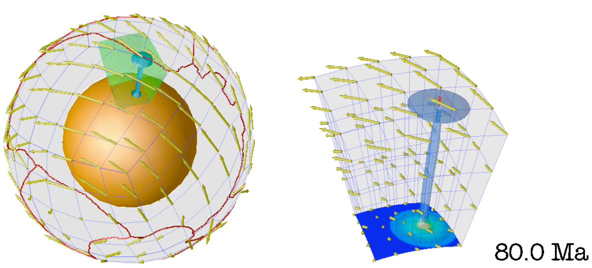
Purpose of this course:

Choose and understand critical methods that prepare you for Numerical PDE's, modeling and scientific computation



Finite element calculation of viscous fluid flow around a "dolfin"
Requires:
Interpolation, numerical quadrature,numerical linear algebra,
+ PDE's, Vector Calculus, functional analysis,computational geometry, C++/Python programming

An example application: Numerical models of the Hawaiin Hotspot (www.geodynamics.org)



Course Logistics:

2 Lectures per week
~1 Homework per unit (60% of grade)
All Matlab based!
1 Midterm (20%)
1 Final (20%)

Text: no required text but
Online resources
Library Reserves
SmartBoard Notes from each class
CVN videos on special request
Matlab: Will need access to matlab. Available through all engineering computers or
Student Version through Columbia Bookstore $99.

All spelled out on web site
www.ldeo.columbia.edu/~mspieg/e4300
(accessible through courseworks)

Final Thoughts:

Numerical methods aren't particularly hard...however...

1) They require significant fluency in
   - Calculus
   - Vector Calculus
   - ODE's
   - Linear Algebra
   - Basic programming and debugging

2) You need to be well prepared, and clear about your objectives

3) Individual pieces can seem dry or disconnected (although interesting math in their own right).

4) However, the ability to
   - put them all together
   - to move between continuous and discrete problems
   - keep track of errors and artifacts
   - stay on top of complex, multi-part problems

...is priceless