Applied Mathematics 205

Unit III: Numerical Calculus

Lecturer: Dr. David Knezevic

Unit III: Numerical Calculus

Chapter $\mathrm{III}.1$: Motivation

Motivation

Since the time of Newton, calculus has been ubiquitous in science

Many (most?) calculus problems that arise in applications do not have closed-form solutions

Numerical approximation is essential!

Epitomizes idea of Scientific Computing as developing and applying numerical algorithms to problems of continuous mathematics

In this Unit we will consider:

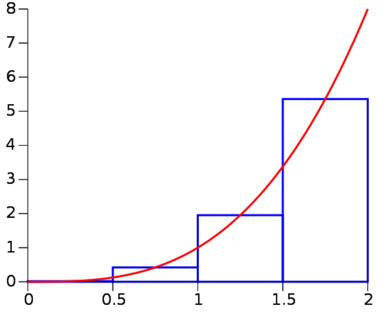
- Numerical integration
- Numerical differentiation
- Numerical methods for ordinary differential equations
- Numerical methods for partial differential equations

Integration

Integration

Approximating a definite integral using a numerical method is called quadrature

The familiar Riemann sum idea suggests how to perform quadrature



We will examine more accurate/efficient quadrature methods

Integration

Question: Why is quadrature important?

We know how to evaluate many integrals analytically, e.g.

$$\int_0^1 e^x \mathrm{d}x \qquad \text{or} \qquad \int_0^\pi \cos x \mathrm{d}x$$

But how about $\int_1^{2000} \exp(\sin(\cos(\sinh(\cosh(\tan^{-1}(\log(x))))))) \mathrm{d}x$?

Integration

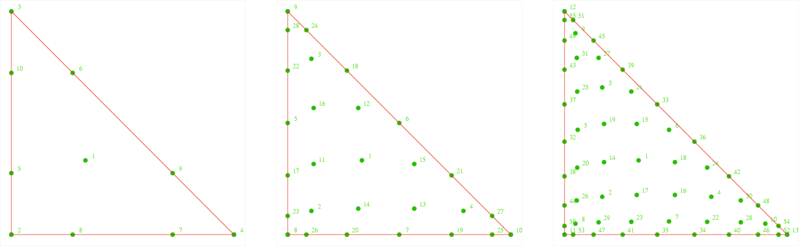
We can numerically approximate this integral in Matlab using quadrature

```
>> tic; quad(@(x)exp(sin(cos(sinh(cosh(atan(log(x))))))),1,2000),
toc

ans =
    1.514780678205574e+03

Elapsed time is 0.005041 seconds.
```

Integration

Quadrature also generalizes naturally to higher dimensions, and allows us to compute integrals on irregular domains

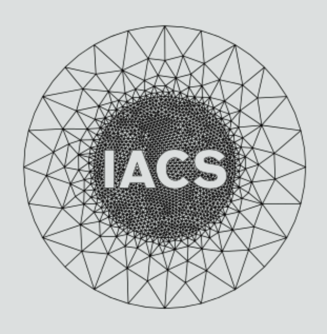
For example, we can approximate an integral on a triangle based on a finite sum of samples at quadrature points



Three different quadrature rules on a triangle

Integration

Can then evaluate integrals on complicated regions by "triangulating" (AKA "meshing")



Differentiation

Differentiation

Numerical differentiation is another fundamental tool in Scientific Computing

We have already discussed the most common, intuitive approach to numerical differentiation: finite differences

$$
\begin{aligned}
f'(x) &= \frac{f(x+h)-f(x)}{h} + O(h) && \text{(forward difference)}\\
f'(x) &= \frac{f(x)-f(x-h)}{h} + O(h) && \text{(backward difference)}\\
f'(x) &= \frac{f(x+h)-f(x-h)}{2h} + O(h^2) && \text{(centered difference)}\\
f''(x) &= \frac{f(x+h)-2f(x)+f(x-h)}{h^2} + O(h^2) && \text{(centered, 2nd deriv.)}\\
&\vdots
\end{aligned}
$$

Differentiation

We will see how to derive these and other finite difference formulas and quantify their accuracy

Wide range of choices, with trade-offs in terms of

- accuracy
- stability
- complexity

Differentiation

We saw in Unit 0 that finite differences can be sensitive to rounding error when h is "too small"

But in most applications we obtain sufficient accuracy with h large enough that rounding error is still negligible[1]

Hence finite differences generally work very well, and provide a very popular approach to solving problems involving derivatives

[1]That is, h is large enough so that rounding error is dominated by discretization error

ODEs

ODEs

The most common situation in which we need to approximate derivatives is in order to solve differential equations

Ordinary Differential Equations (ODEs): Differential equations involving functions of one variable

Some example ODEs:

- $y'(t) = y^2(t) + t^4 - 6t$, $y(0) = y_0$ is a first order Initial Value Problem (IVP) ODE
- $y''(x) + 2xy(x) = 1$, $y(0) = y(1) = 0$ is a second order Boundary Value Problem (BVP) ODE

ODEs: IVP

A familiar IVP ODE is Newton's Second Law of Motion: suppose position of a particle at time $t \geq 0$ is $y(t) \in \mathbb{R}$

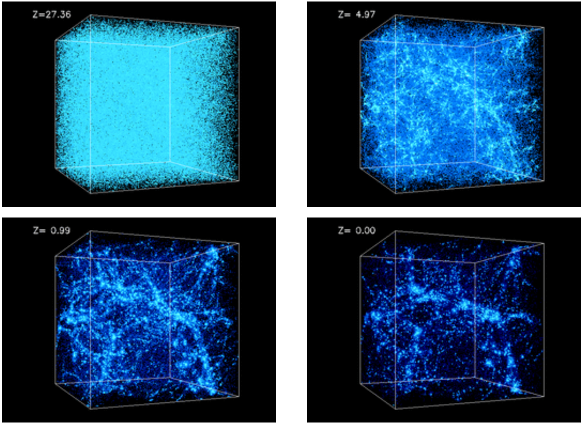
$$y''(t) = \frac{F(t, y, y')}{m}, \qquad y(0) = y_0, y'(0) = v_0$$

This is a scalar ODE ($y(t) \in \mathbb{R}$), but it's common to simulate a system of N interacting particles

e.g. F could be gravitational force due to other particles, then force on particle i depends on positions of the other particles

ODEs: IVP

N-body problems are the basis of many cosmological simulations: Recall galaxy formation simulations from Unit 0



Computationally expensive when N is large!

ODEs: BVP

ODE boundary value problems are also important in many circumstances

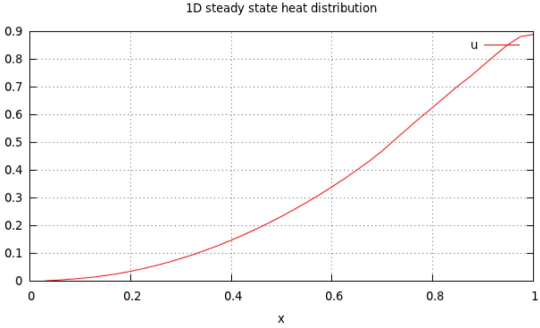
For example, steady state heat distribution in a "1D rod"

Apply heat source $f(x) = x^2$, impose "zero" temperature at $x = 0$, insulate at $x = 1$:

$$-u''(x) = x^2, \quad u(0) = 0, u'(1) = 0$$

ODEs: BVP

We can approximate via finite differences: use F.D. formula for $u''(x)$



PDEs

PDEs

It is also natural to introduce time-dependence for the temperature in the "1D rod" from above

Hence now u is a function of x and t, so derivatives of u are partial derivatives, and we obtain a partial differential equation (PDE)

For example, the time-dependent heat equation for the 1D rod is given by:

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = x^2, \quad u(x,0) = 0, u(0,t) = 0, \frac{\partial u}{\partial x}(1,t) = 0$$

This is an Initial-Boundary Value Problem (IBVP)

PDEs

Also, when we are modeling continua[2] we generally also need to be able to handle 2D and 3D domains

e.g. 3D analogue of time-dependent heat equation on a domain $\Omega \subset \mathbb{R}^3$ is

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} - \frac{\partial^2 u}{\partial z^2} = f(x, y, z), \quad u = 0 \text{ on } \partial\Omega$$

[2] e.g. temperature distribution, fluid velocity, electromagnetic fields,...

PDEs

This equation is typically written as

$$\frac{\partial u}{\partial t} - \Delta u = f(x, y, z), \quad u = 0 \text{ on } \partial\Omega$$

where $\Delta u \equiv \nabla \cdot \nabla u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}$

Here we have:

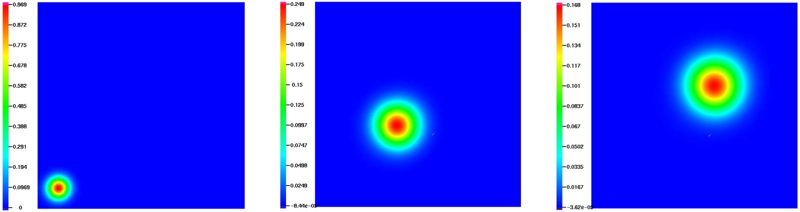
- The Laplacian, $\Delta \equiv \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$
- The gradient, $\nabla \equiv (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$

PDEs

Can add a "transport" term to the heat equation to obtain the convection-diffusion equation, e.g. in 2D we have

$$\frac{\partial u}{\partial t} + (w_1(x,y), w_2(x,y)) \cdot \nabla u - \Delta u = f(x,y), \quad u = 0 \text{ on } \partial\Omega$$

$u(x,t)$ models concentration of some substance, e.g. pollution in a river with current (w_1, w_2)

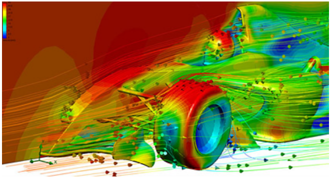


$t = 0$ $\qquad$ $t = 3$ $\qquad$ $t = 5$

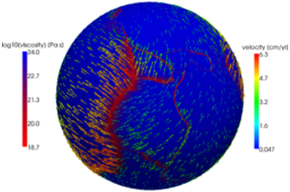
PDEs

Numerical methods for PDEs are a major topic in scientific computing

Recall examples from Unit 0:



CFD



Geophysics

The finite difference method is an effective approach for a wide range of problems, hence we focus on F.D. in AM205[3]

[3]There are many important alternatives, e.g. finite element method, finite volume method, spectral methods, boundary element methods...

Summary

Numerical calculus encompasses a wide range of important topics in scientific computing!

As always, we will pay attention to stability, accuracy and efficiency of the algorithms that we consider