MA3227 Numerical Analysis II

Lecture 0: Introduction

Simon Etter

Semester II, AY 2020/2021

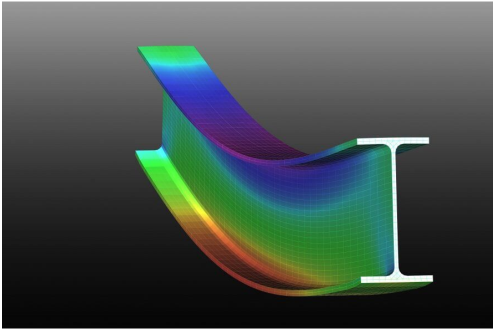
Introduction

Module outline

- Big O notation for talking about runtime and convergence (week 1)
- Bisection and Newton methods for nonlinear equations (week 2)
- Finite difference method for partial differential equations (week 3)
- LU factorisation and Krylov methods for sparse linear systems (weeks 4-6)
- Runge Kutta Methods for ordinary differential equations (weeks 7-9)
- Monte Carlo Methods for computing intractable sums and integrals (weeks 10-12)

Introduction

Partial differential equations and sparse linear systems



Objectives:

- Devise a mathematical model for phenomena like bending beams.
- Translate model into a very large linear system.
- Devise algorithms for solving the resulting linear system efficiently.

https://windowsreport.com/fem-fea-software-pc

Introduction

Ordinary differential equations

Objectives:

- Devise a mathematical model for phenomena like rocket trajectories.
- Translate model into a time-stepping algorithm.
- Do step 2 as efficiently as possible.

`https://pixabay.com/photos/rocket-launch-night-trajectory-693236/`

Introduction

Monte Carlo Methods

The purpose and approach of Monte Carlo methods are best explained by means of an example.

Assume we know the following.

- There is a single COVID patient in Singapore on day 0.
- Every patient infects someone with probability p_1 on any given day.
- Every patient recovers with probability p_2 on any given day.

What is the probability that the disease will infect 100 people?

This question could in principle be answered as follows.

- Go through every possible scenario for the epidemic.
- Sum the probabilities of all those scenarios where the disease died out quickly.

Problem: The number of scenarios is way to large!

Alternative approach: Play out a random sample of scenarios and compute the fraction of scenarios where the disease spreads widely.

See `pandemic()` for a demonstration of this approach.

Introduction

Monte Carlo Methods (continued)

The `pandemic()` function answers a complicated mathematical question essentially by "tossing coins". Algorithms of this type are known as *Monte Carlo methods* in reference to the casino of the same name.

Fun fact: If the Monte Carlo idea had been developed in Singapore, it might have been called the "Marina Bay Sands method" instead.

https://commons.wikimedia.org/wiki/File:Casino_de_Monte-Carlo_(49582351802).jpg