Numerical analysis for MX / EL / CH

Michael F. Herbst, Bruno Ploumhans

Mathematics for Materials Modelling (`matmat.org`), EPFL

Spring semester 2025

Leading thought

- Advanced numerical methods are everywhere in science
  - Data collection (Adaptive experimental instruments)
  - Data analysis
  - Visualisation
  - Simulation
  - ...

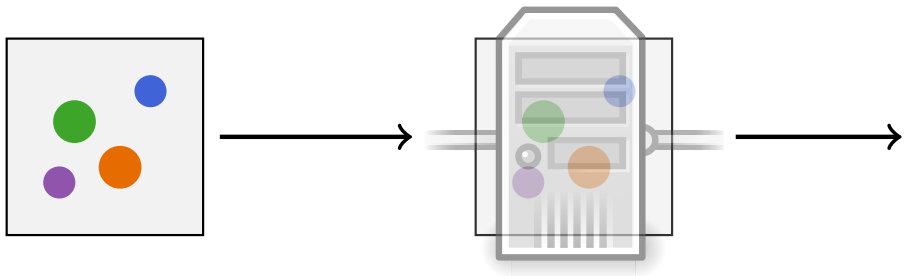
$\Rightarrow$ It's better if you understand how they work and when they fail

- Numerical analysis provides ...
  - **Mathematical formulation** of physical problems
  - **Numerical algorithms** to solve them
  - Mathematical analysis of **accuracy** and **errors**
  - Insight to **efficiency** and **reliability** of computations

We work on: Computational materials discovery

- Nowadays discover materials on a computer

$$\min_{\Psi} \langle \Psi, H\Psi \rangle$$



- $\simeq$ 30% of supercomputer usage $\Rightarrow$ Accurate & efficient numerics

Topics of this course

- In this course we won't go HPC, but we cover the basics:
  - Solving non-linear equations
  - Interpolation
  - Numerical differentiation and integration
  - Solving linear systems
  - Solving eigenvalue problems
  - Solving differential equations

- You're phone / computer / microscope / spectrometer / ... is using all these techniques *right now*

- (... and unfortunately does not always get it right)

Course organisation

- Lecture: **Tuesday, 10:15 - 12:00, CM 1 4**
- Exercise classes:
  - **Wednesday, 10:15 - 12:00, BC07 & 08**
  - Important: You need to **bring your own laptop**
    - You don't own one → Later slide
- Moodle: `https://go.epfl.ch/numerical-analysis`

  ⇒ Single source for all information and material
- `https://teaching.matmat.org/numerical-analysis/`

  ⇒ Online version of the lecture notes

Assistants

- Head teaching assistant
  - **Bruno Ploumhans**, bruno.ploumhans@epfl.ch, MA B1 457
- Teaching assistants
  - **Andrea Pintus** andrea.pintus@epfl.ch
  - **Alessio Siviglia**, alessio.siviglia@epfl.ch
  - **Divya Suman**, divya.suman@epfl.ch
  - **Leo Digonzelli**, leo.digonzelli@epfl.ch
  - **Lorenzo Piersante**, lorenzo.piersante@epfl.ch
  - **Théodore Decaux**, theodore.decaux@epfl.ch

Exercises

- Exercises: **Theoretical** and a **computational** component
  - ("numerical experiments")
- julia programming language: Computational component
  - You **need to do these on your laptop**
  - Installation instructions on moodle
  - Introduction during the first exercise session
  - From week 2 we will assume you have a working setup
    - This will be checked before the exercise session
- Exercises sheets: Available **Tuesday at 12:00** on Moodle
  - Sheets are generally *not graded* (except one → see later slide)
  - Master solution available one week later
  - Beginning of exercise session: 15mins to ask questions on previous sheet
- **The exercises are a preparation for the exam**

A few words about Pluto and Julia

What a Julia notebook looks like

→ `https://teaching.matmat.org/numerical-analysis/02_Julia.html`

Obtaining a laptop

- A laptop is required for the coding exercises
  - This includes the graded sheet (see later)

If you do not have a laptop yet, make sure to order one as soon as possible.

- Financial support:
  - Website on financing your studies:
    `https://www.epfl.ch/education/studies/en/financing-study/`
  - Need-based scholarships:
    `https://www.epfl.ch/education/studies/en/support-scholarships/`
  - Canton of Vaud scholarships:
    `https://www.vd.ch/formation/aides-financieres-aux-etudes-et-a-la-formation-professionnelle-bourses-ou-prets/informations-principales/`

Course materials

- All materials will be published on Moodle:

  `https://go.epfl.ch/numerical-analysis`

- **Pluto notebooks** used during the lecture
- **Weekly** exercise sheets *(no submission)*
- **Graded** exercise sheet *(submission mandatory)*

- Additional materials (see Moodle):
  - Lecture notes of Prof. Fabio Nobile (English & French)
  - Tobin A. Driscoll, Richard J. Braun *Fundamentals of Numerical Computation* (2022).
  - Giray Ökten *First Semester Numerical analysis* (2023).
  - MIT's *Introduction to computational thinking* lecture

Course evaluation

- One graded exercise sheet (20%)
  - One graded sheet around week 11 ($\approx$ **6 May**)
  - Implementation and analysis of algorithms
  - Focuses on *computational* aspects of the course
  - Think of it as a take home exam
  - Individual submissions, interviews and spot checks

- Final exam (80%)
  - Written exam consisting of *theoretical* problems
  - Pen and paper questions

Learning objectives

- We study numerical techniques to solve mathematical problems
- **This is not a programming course**. Most exercises can be done by *understanding* and *using* code, which we provide
- Main objectives of the course are:
  - Obtain a basic idea how an algorithm works
  - Understand its underlying theory
  - Know the advantages and disadvantages of each algorithm
  - Know how to employ julia to solve mathematical problems
  - Learn how to analyse numerical results critically
  - Learn how to choose the best algorithm for each problem
- Fundamental concepts:
  - **Stability** of an algorithm
  - **Accuracy** of an algorithm
  - **Cost** of an algorithm

Questions ?

Your background and prior knowledge

Quick survey

https://etc.ch/K7Ud