Open topics in applied mathematics:

Fast Methods in Scientific Computation

MAT 393 C

Gunnar Martinsson

(These notes will be posted on the class webpage.)

Purpose of class:

- The central theme is “fast” methods for solving elliptic PDEs such as:
  - The Laplace and Poisson equations.
  - Helmholtz’ equation.
  - Time-harmonic Maxwell’s equation.
  - The equations of linear elasticity.
  - The Stokes equation.

- We will also cover other computational methods, including:
  - FFT and other expansion based fast solvers.
  - Fast methods for N-body problems such as the Fast Multipole Method (FMM).
  - Techniques for accelerating matrix computations — randomized methods for factorizing matrices, sparse solvers, Krylov methods, “rank-structured” matrix computations, etc.

- Light emphasis on proofs and theory. Stronger emphasis on practical computing.

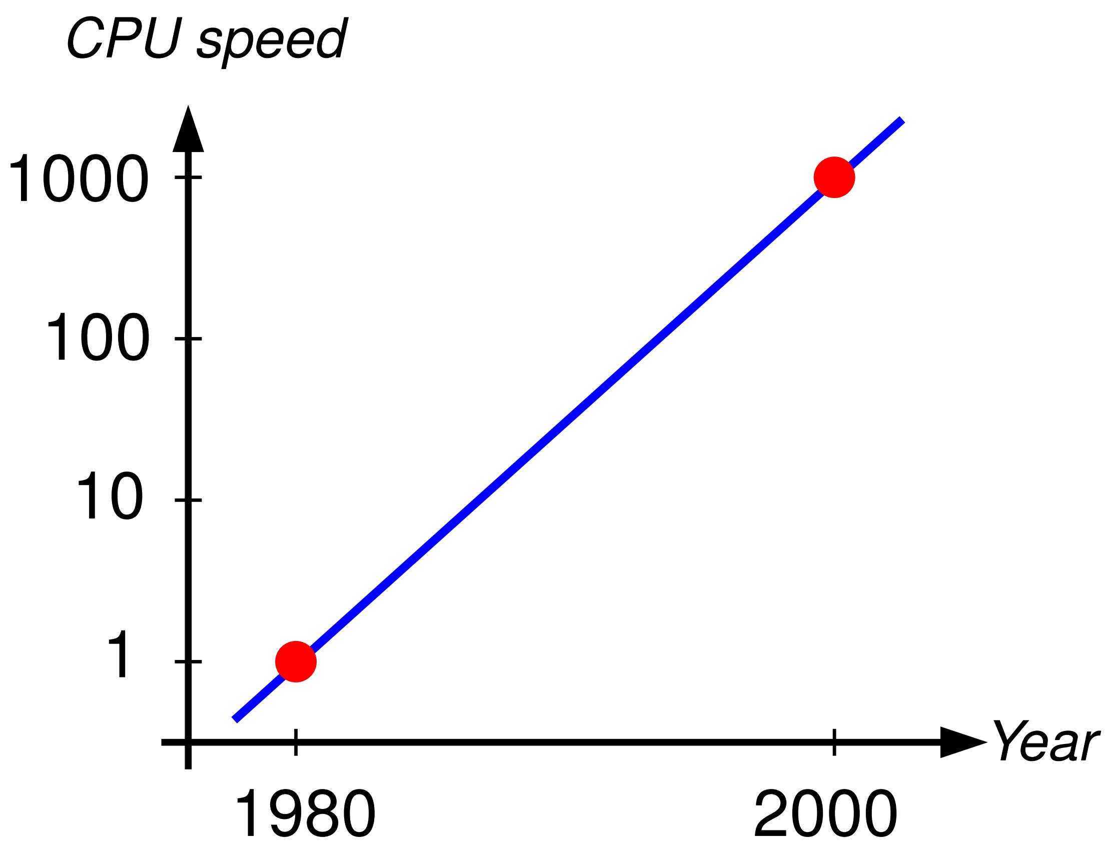
- Focus is on numerical methods and scientific computing, but connections to applications will be discussed as well.

Definition of the term "fast":

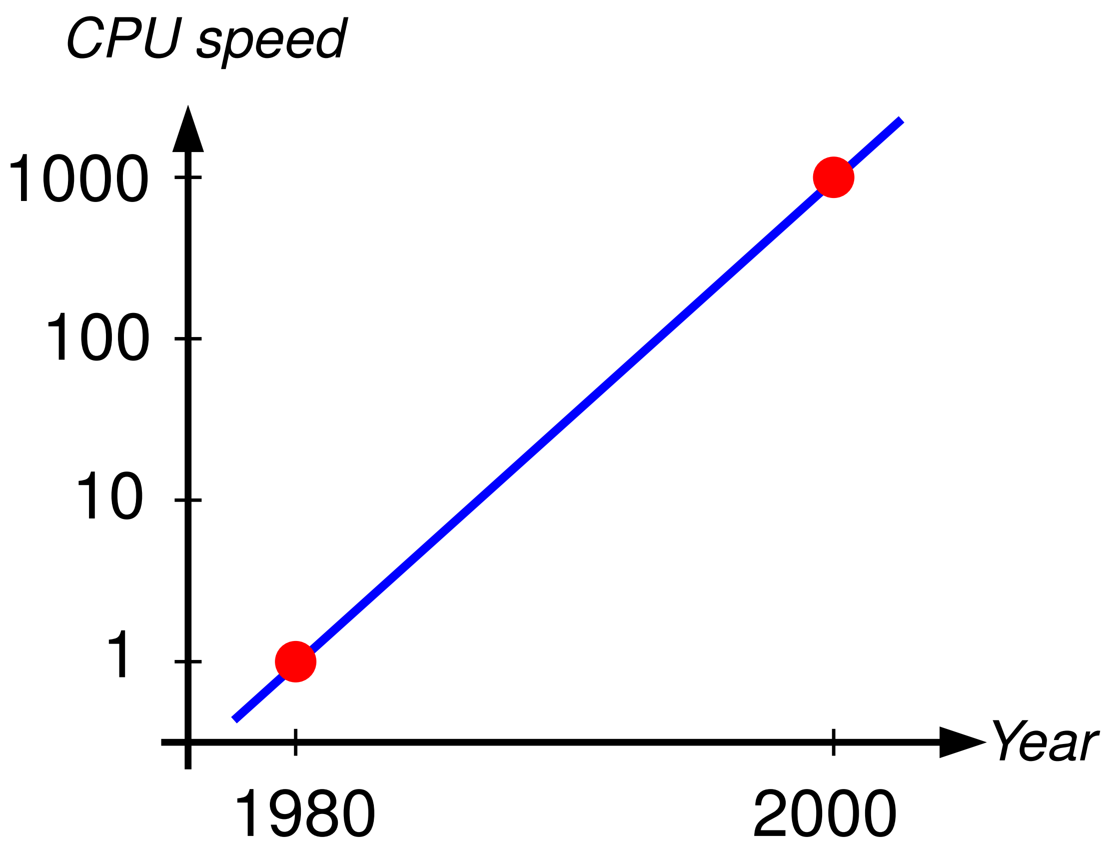
We say that a numerical method is *fast* if its execution time scales as $O(N)$ as the problem size N grows.

Methods whose complexity is $O(N \log N)$ or $O(N \log^2 N)$ are also called "fast".

Growth of computing power and the importance of algorithms

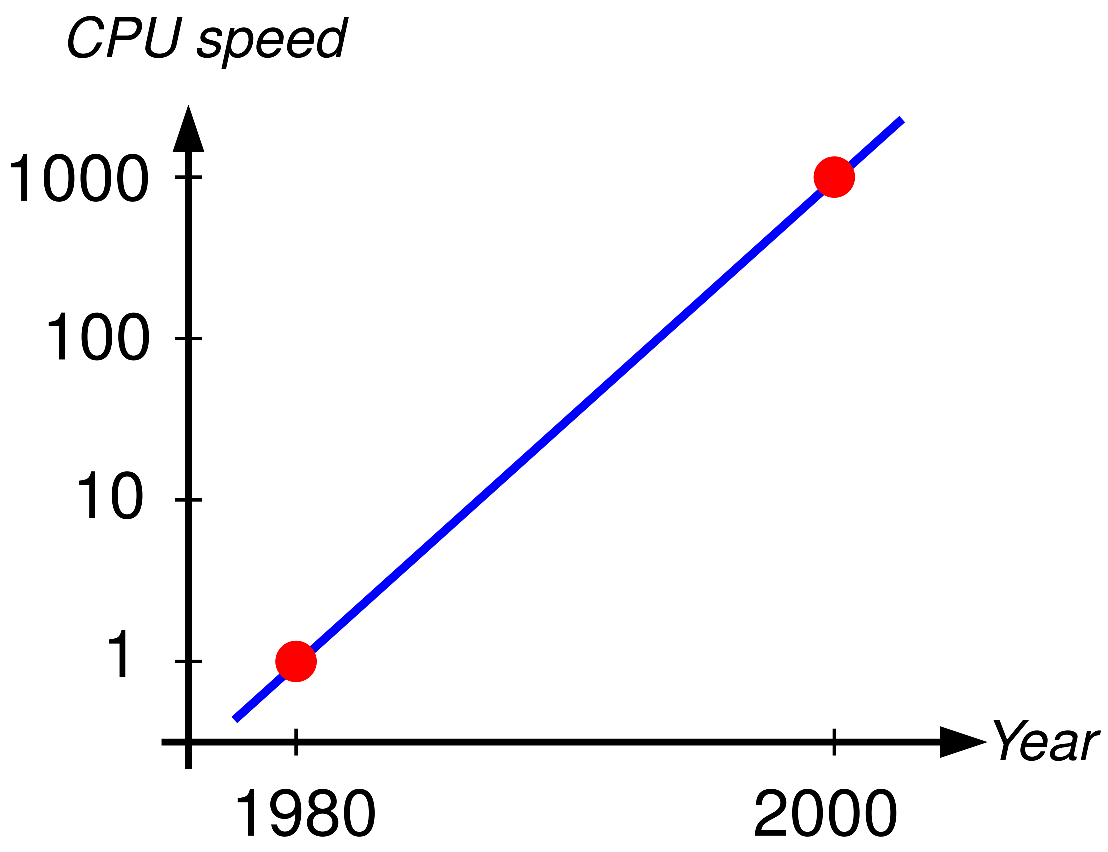


Growth of computing power and the importance of algorithms



Consider the task of solving a linear algebraic system $Ax = b$ of N equations with N unknowns.

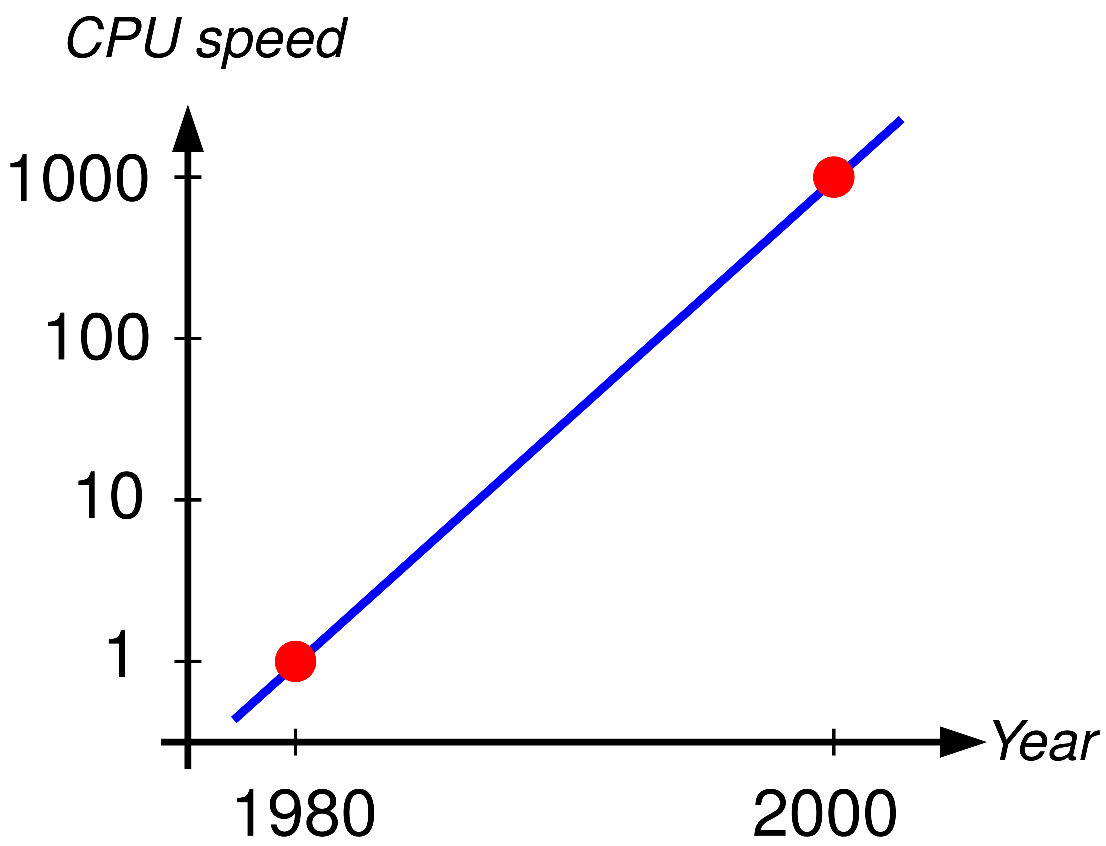
Growth of computing power and the importance of algorithms

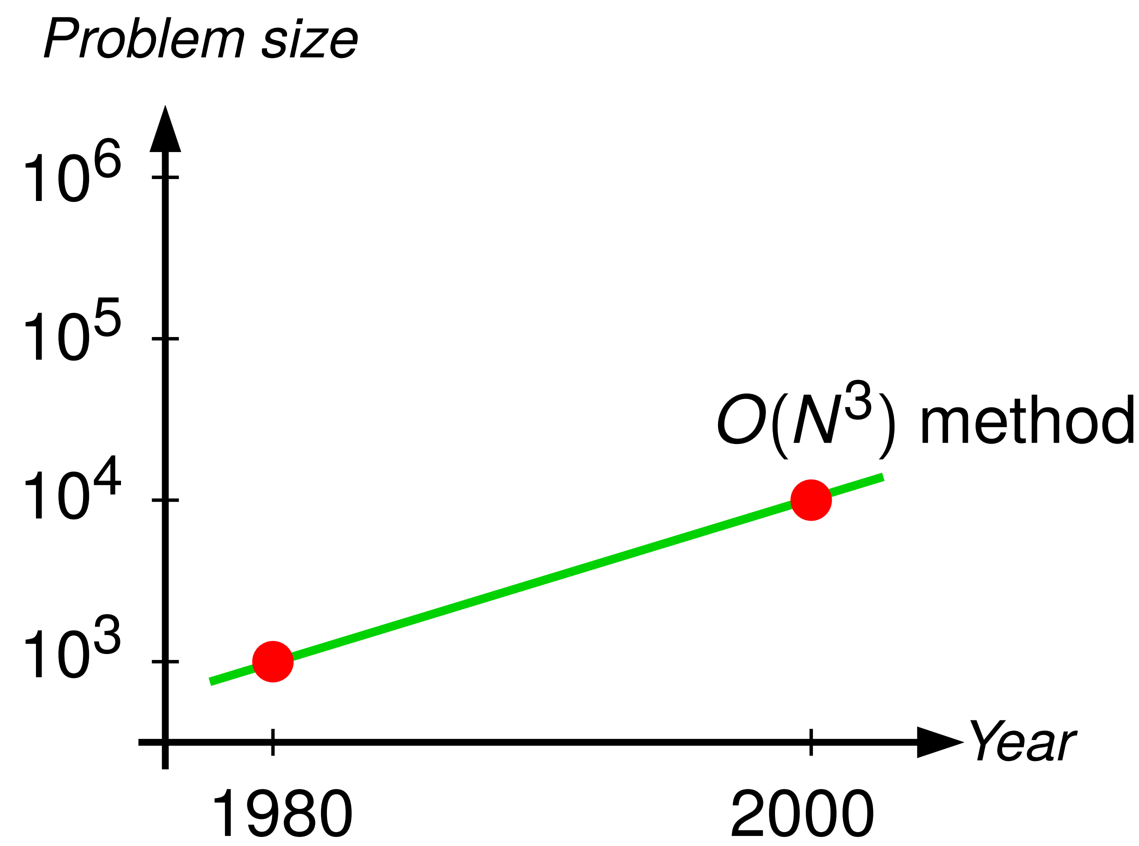


Consider the task of solving a linear algebraic system $Ax = b$ of N equations with N unknowns.

Classical methods such as Gaussian elimination require $O(N^3)$ operations.

Growth of computing power and the importance of algorithms



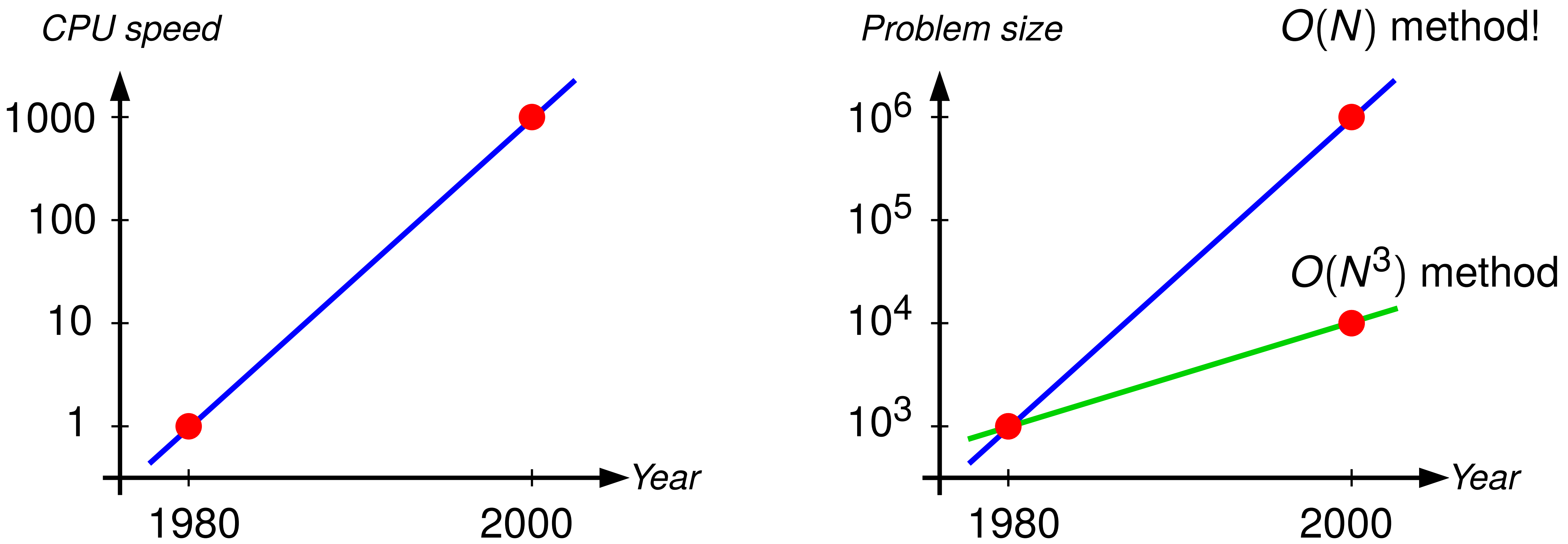


Consider the task of solving a linear algebraic system $Ax = b$ of N equations with N unknowns.

Classical methods such as Gaussian elimination require $O(N^3)$ operations.

Using an $O(N^3)$ method, an increase in computing power by a factor of 1000, enables the solution of problems that are $1000^{1/3} = 10$ times larger.

Growth of computing power and the importance of algorithms



Consider the task of solving a linear algebraic system $Ax = b$ of N equations with N unknowns.

Classical methods such as Gaussian elimination require $O(N^3)$ operations.

Using an $O(N^3)$ method, an increase in computing power by a factor of 1000, enables the solution of problems that are $1000^{1/3} = 10$ times larger.

Using a method that scales as O(N), problems that are 1000 times larger can be solved.

Caveat: It appears that Moore's law is no longer operative.

Processor speed is currently increasing quite slowly.

The principal increase in computing power is coming from *parallelization.*

In consequence, successful algorithms must scale well both with problem size and with the number of processors that a computer has.

To slightly offset the difficulty of parallelization, the *cost of storage is decreasing.* However, the speed of access is increasing only slowly, again reinforcing the need to keep data local in designing algorithms.

Laplace's equation (in two dimensions for simplicity)

Let $u = u(\boldsymbol{x})$ denote a differentiable function of the vector valued variable $\boldsymbol{x} = (x_1, x_2)$. The *Laplace operator* is defined by

$$\Delta u = \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2}.$$

Let Ω denote a *domain* with boundary Γ. Then the *Poisson equation* on Ω is

$$(1) \qquad \begin{cases} -\Delta\, u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Omega, \\ \quad u(\boldsymbol{x}) = g(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma. \end{cases}$$

The function f is a given *body load* and g is a given *boundary data*. If $f = 0$, we call (1) the *Laplace equation.*

The Poisson and Laplace equations are the simplest equations in a large class of so called *elliptic PDEs*. Other examples include Helmholtz, elasticity, Maxwell (for the "time-harmonic case").

The *Laplace and Poisson* equations:

Electrostatics:

$$\begin{cases} -\Delta\, u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = g(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma. \end{cases}$$

u is the electric potential

f is the electric charge density

g is a fixed potential on the boundary (Neumann b.c. $\Rightarrow$ fixed fluxes)

Examples of applications:

- Design of electric engines / turbines / etc.
- Biochemical modeling.
- Design of electronic circuits.

("Magnetostatics" is entirely analogous.)

The *Laplace and Poisson* equations:

Gravity:

$$-\Delta u(\boldsymbol{x}) = f(\boldsymbol{x}), \qquad \boldsymbol{x} \in \mathbb{R}^3,$$

u is the gravitational potential

f is the mass density

Examples of applications:

- Astrophysics

A "hidden" Laplace problem: Consider a situation with N gravitational bodies in $\mathbb{R}^3$. Each body has mass m_i and location $\boldsymbol{x}_i$. Then the force on body i resulting from interactions with the other bodies is

$$\boldsymbol{F}_i = \sum_{j \neq i} G\, m_i\, m_j \frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{|\boldsymbol{x}_i - \boldsymbol{x}_j|^3},$$

where $G \approx 6.67428 \cdot 10^{-11} \mathrm{m}^3/(\mathrm{kg\,s}^2)$ is the gravitational constant.

We now observe that the force $\boldsymbol{F}_i$ can be expressed as

$$\boldsymbol{F}_i = -m_i \sum_{j \neq i} \nabla u_j(\boldsymbol{x}_i),$$

where $u_j = u_j(\boldsymbol{x})$ is the gravitational potential generated by the j'th charge

$$u_j(\boldsymbol{x}) = \sum_{j \neq i} G\, m_j \frac{1}{|\boldsymbol{x} - \boldsymbol{x}_j|}.$$

The potential u_j satisfies

$$-\Delta u_j(\boldsymbol{x}) = m_j\, \delta(\boldsymbol{x} - \boldsymbol{x}_j).$$

The total field $u = \sum_i u_i$ satisfies

$$-\Delta u(\boldsymbol{x}) = f(\boldsymbol{x}) = \sum_i m_j\, \delta(\boldsymbol{x} - \boldsymbol{x}_j).$$

The problem of computing a sum such as

$$\boldsymbol{F}_i = \sum_{j \neq i} G\, m_i\, m_j \,\frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{|\boldsymbol{x}_i - \boldsymbol{x}_j|^3},$$

arises directly in many applications:

- Astrophysics.
- Biochemical simulations (each “particle” is a charged part of a molecule).
- Modeling of semi-conductors (each “particle” is an ion).
- Fluid dynamics (each “particle” is an “vortex”).

It also arises indirectly in many “fast” methods for solving elliptic PDEs.

The naïve computation of $\{\boldsymbol{F}_i\}_{i=1}^{N}$ requires $O(N^2)$ operations since there are $N\,(N-1)/2$ “pair-wise interactions.”

We will study in some detail a method that requires only $O(N)$ operations;

the so called *Fast Multipole Method* or *FMM.*

The *Laplace and Poisson* equations:

Thermostatics:

$$\begin{cases} -\Delta u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = g(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma. \end{cases}$$

u is the temperature

f is the heat source density

g is a fixed temperature on the boundary (Neumann b.c. $\Rightarrow$ fixed flows)

Examples of applications:

- ...

The *Helmholtz* equation:

Recall the *wave equation*:

$$\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} = \frac{\partial^2 u}{\partial t^2}.$$

The wave equation models vibrations in membranes, acoustic waves, certain electro-magnetic waves, and many other phenomena.

Now assume that the time dependence is “time harmonic”:

$$u(\boldsymbol{x}, t) = v(\boldsymbol{x})\ \cos(\omega t).$$

Then $\frac{\partial^2 u}{\partial t^2} = -\omega^2\, u$ and so the wave equation becomes the *Helmholtz equation:*

$$\frac{\partial^2 v}{\partial x_1^2} + \frac{\partial^2 v}{\partial x_2^2} = -\omega^2 v.$$

The *Maxwell equations*

$$\begin{cases} \nabla \cdot \mathbf{E} = \rho & \nabla \times \mathbf{E} = -\dfrac{\partial \mathbf{B}}{\partial t} \\ \nabla \cdot \mathbf{B} = 0 & \nabla \times \mathbf{B} = \mathbf{J} + \dfrac{\partial \mathbf{E}}{\partial t} \end{cases}$$

model electromagnetism. **E** is the electric field, and **B** is the magnetic field.

Consider the stationary case where $\partial \mathbf{E}/\partial t = 0$ and $\partial \mathbf{B}/\partial t = 0$. Since **E** is curl-free, there exists a function $u = u(\boldsymbol{x})$ such that

$$\mathbf{E} = -\nabla u.$$

(The function u is the electric potential.) We now find that

$$\rho = \nabla \cdot \mathbf{E} = \nabla \cdot (-\nabla u) = -\Delta u,$$

and we recover the Poisson equation we saw earlier:

$$-\Delta u = \rho.$$

The *Maxwell equations*

$$\begin{cases} \nabla \cdot \mathbf{E} = \rho & \nabla \times \mathbf{E} = \; -\dfrac{\partial \mathbf{B}}{\partial t} \\ \nabla \cdot \mathbf{B} = 0 & \nabla \times \mathbf{B} = \mathbf{J} + \dfrac{\partial \mathbf{E}}{\partial t} \end{cases}$$

model electromagnetism. **E** is the electric field, and **B** is the magnetic field.

Now consider another simplification: the “time-harmonic” case where

$$\mathbf{E}(\boldsymbol{x}, t) = \mathbf{E}(\boldsymbol{x})\, e^{i\omega t}, \qquad \mathbf{B}(\boldsymbol{x}, t) = \mathbf{B}(\boldsymbol{x})\, e^{i\omega t}.$$

Then

$$\frac{\partial}{\partial t}\mathbf{E} = i\omega\mathbf{E}, \qquad \text{and} \qquad \frac{\partial}{\partial t}\mathbf{B} = i\omega\mathbf{B}.$$

Inserting these relations into the Maxwell equations, we obtain a system of “Helmholtz-like” equations

$$\begin{cases} \nabla \cdot \mathbf{E} = \rho & \nabla \times \mathbf{E} = \; -i\omega\mathbf{B} \\ \nabla \cdot \mathbf{B} = 0 & \nabla \times \mathbf{B} = \mathbf{J} + i\omega\mathbf{E} \end{cases}$$

In special cases, the system simplifies to the plain Helmholtz equation . . .

Recall:

$$\begin{cases} \nabla \cdot \mathbf{E} = \rho & \nabla \times \mathbf{E} = -i\omega \mathbf{B} \\ \nabla \cdot \mathbf{B} = 0 & \nabla \times \mathbf{B} = \mathbf{J} + i\omega \mathbf{E} \end{cases}$$

Suppose $\rho = 0$ and $\mathbf{J} = \mathbf{0}$. Then

$$\nabla \times \nabla \times \mathbf{E} = \nabla \times (-i\omega \mathbf{B}) = -i\omega(\nabla \times \mathbf{B}) = -i\omega(i\omega \mathbf{E}) = \omega^2 \mathbf{E}.$$

Now recall that for any vector field $\mathbf{F}$ we have

$$\nabla \times \nabla \times \mathbf{F} = \nabla(\nabla \cdot \mathbf{F}) - \Delta \mathbf{F}.$$

Consequently:

$$\nabla(\nabla \cdot \mathbf{E}) - \Delta \mathbf{E} = \omega^2 \mathbf{E}.$$

Finally recall that $\nabla \cdot \mathbf{E} = 0$ to obtain the “Helmholtz-like” equation

$$-\Delta \mathbf{E} = \omega^2 \mathbf{E}.$$

The equations of *linear elasticity* in $\mathbb{R}^d$:

$$\sum_{j,k,l=1}^{d} \frac{1}{2} E_{ijkl} \left(\frac{\partial^2 u_k}{\partial x_l \partial x_j} + \frac{\partial^2 u_l}{\partial x_k \partial x_j} \right) = f_i, \qquad i = 1, 2, \dots, d.$$

The function $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}) = (u_1(\boldsymbol{x}), u_2(\boldsymbol{x}), \dots, u_d(\boldsymbol{x}))$ is the displacement of an elastic material subjected to the body load $\boldsymbol{f} = \boldsymbol{f}(\boldsymbol{x})$ at the point $\boldsymbol{x}$.

$(E_{ijkl})_{i,j,k,l=1}^{d}$ is the *stiffness tensor* which describes the material properties.

Many simplifications can be derived from the basic equilibrium equation. For instance, if the material is isotropic, and if $\boldsymbol{f} = 0$, then the displacements satisfy the *biharmonic equation*

$$(-\Delta)^2 \boldsymbol{u} = 0.$$

Another simplification is the displacement of a thin elastic membrane:

$$\begin{cases} (-\Delta)^2 u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = g(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \\ u_n(\boldsymbol{x}) = h(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma. \end{cases}$$

Here f is the body load (e.g. gravity), h is the prescribed deflection at the boundary, and h is the prescribed normal derivative. (Since the equation has order *four*, we need *two* boundary conditions.)

Outline:

Week:	*Material covered:*
1:	Introduction: Objectives of the course. Quick review of basic elliptic PDEs and their connections to physical applications. Analytic solution formulas, and their relationship to numerical methods. Fast algorithms for global operators.
2:	Linear algebra: Review of basic matrix factorizations. Techniques for computing low-rank approximations to matrices. Randomized methods for matrix computations.
4:	Rank-structured matrices: What they are, where they arise in applications, how they enable fast solvers (and fast matrix algebra more generally).
5:	Krylov methods for solving linear systems and computing partial spectral decompositions.
7:	Fast solvers for elliptic PDEs based on the FFT and related techniques.
8:	Direct solvers for elliptic PDEs based on Gaussian elimination combined with nested dissection ordering of the nodes ("multifrontal methods"). Sweeping solvers.
10:	Boundary integral equations. How a PDE can be rewritten as an integral equation. Advantages and disadvantages. Second kind Fredholm equations. Reduction of dimensionality.
12:	The Fast Multipole Method, and fast summation techniques. The kernel evaluation map. Kernel-independent FMMs and $\mathcal{H}$-matrices.
14:	Fast direct solvers for integral equations.
15:	(If time permits…) Johnson-and-Lindenstrauss theory, and connections to analysis of complex high dimensional data sets.

Practicalities:

Text: There is no “official” text. The syllabus is defined by the material covered in class. Extensive latexed notes will be made available on the course website:

`http://users.ices.utexas.edu/~pgm/Teaching/2019_393C`

Comments, errata, suggestions, . . . , are highly appreciated!

Attendance: Strongly encouraged.

Computer programming: Matlab will be used. If you do not have access to a computer with Matlab, please contact the instructor.

Grading: No exam. Final grade is based on homeworks and a project:

- 50%: Five homework problems worth 10% each.
- 10%: Handing in a carefully latexed “reference solution”.
- 40%: Final project.