Introduction to Computational Science & Engineering (CSE)

16.0002 / 18.0002 / CSE.01

Lecture 4:
Gaussian elimination + Numpy

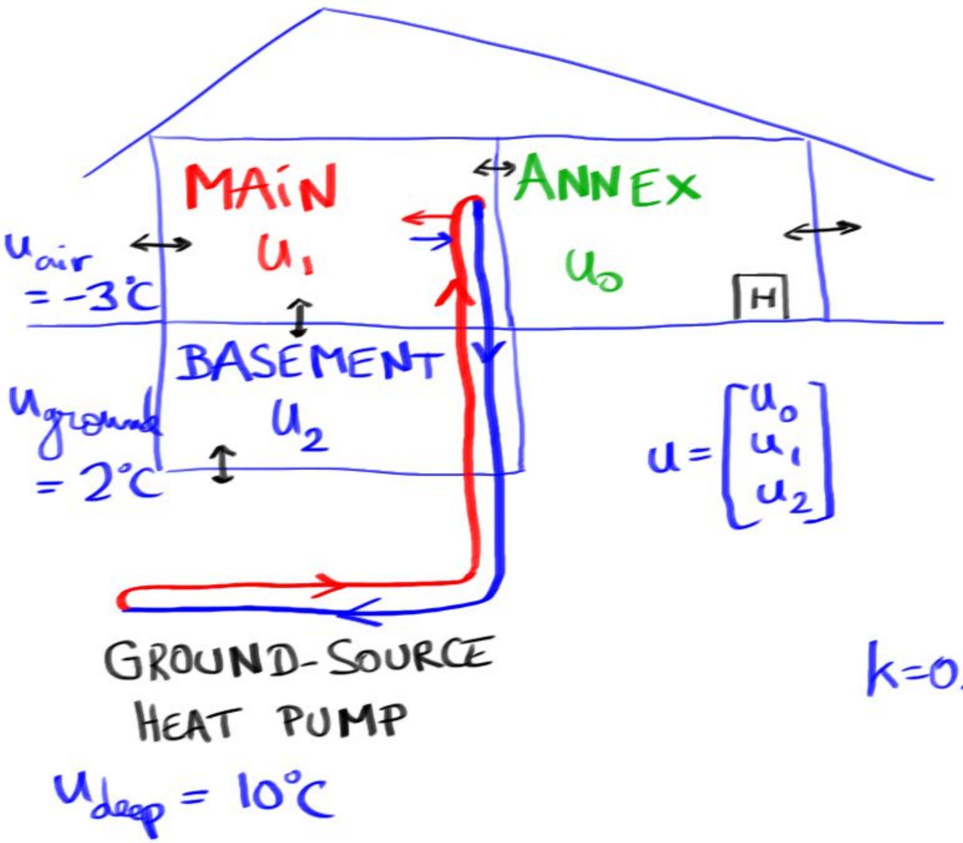
Raul Radovitzky(Aero/Astro),
Laurent Demanet (Math/EAPS)

April 6, 2022

Today: - Numpy arrays
- Gaussian elimination

$$\boxed{\text{IVP}} \begin{cases} \dfrac{du}{dt} = kAu + kb \\ u(0) \text{ given} \end{cases}$$

$$\boxed{\text{Eq.E}} \quad 0 = Au + b$$



$$u = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix}$$

$k = 0.5$

$\boxed{\text{IVP}}$ $\begin{cases} \dfrac{du}{dt} = kAu + kb \\ u(0) \text{ given} \end{cases}$

$\boxed{\text{Eq.E}}$ $0 = Au + b$

$$A = \begin{bmatrix} -2 & 1 & 0 \\ 1 & -3 & 1 \\ 0 & 1 & -2 \end{bmatrix}$$

NUMPY

import numpy as np

WHY? Vector/Matrix operations like +, *

- Array initialization : "ndarray"

v = np.zeros(3) $\rightarrow \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$

A = np.zeros((2,2)) $\rightarrow \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

v = np.array([1.0, 2.0, 3.0])

A = np.array([[1.0, 2.0], (3.0, 4.0]]) $\rightarrow \begin{array}{cc} & j \text{ (col)} \\ i \text{ (row)} & \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \end{array}$ A[i,j]

- Array indexing/slicing

v[0] (access x = v[0] or assign v[0] = x)

A[0,0]

A[0,:]

A[:,1]

A few handy array operations in numpy:

a=np.array([a1, a2, a3]) b=np.array([b1, b2, b3]) c scalar

- a * b is [a1*b1, a2*b2, a3*b3]
- c * b is [c*b1, c*b2, c*b3]
- a ** b is [a1**b1, a2**b2, a3**b3]
- a**c is [a1**c, a2**c, a3**c]
- c**b is [c**b1, c**b2, c**b3]
- K@a or np.matmul(K,a) is matrix-vector mult, not K*a
- use np.array, not np.matrix

$K = \begin{bmatrix} K_{00} & K_{01} \\ K_{10} & K_{11} \end{bmatrix}$ $v = \begin{bmatrix} v_0 \\ v_1 \end{bmatrix}$ K@v is $\begin{bmatrix} K_{00} v_0 + K_{01} v_1 \\ K_{10} v_0 + K_{11} v_1 \end{bmatrix}$

A few more handy operations in numpy:

- $a[r:s] \leftarrow [a[r], a[r+1], \dots, a[s-1]]$
- $a[r:] \leftarrow [a[r], a[r+1], \dots, a[n-1]]$

  ↳ end

- $a[:s] \leftarrow [a[0], a[1], \dots, a[s-1]]$
- $a[:] \leftarrow [a[0], a[1], \dots a[n-1]]$
- $a[r:s:step] \leftarrow [a[r], a[r+step], a[r+2*step], \dots$

  $\dots, a[t]]$

  largest before s

- Assign: $a[r:s] = v$ or Access: $v = a[r:s]$

Yet more handy operations in numpy:

- $a[-1] \leftarrow \underset{\llcorner \text{end}}{a[n-1]}$ (counts in reverse)
- $a[-2:] \leftarrow [a[n-2], a[n-1]]$
- $a[:-2] \leftarrow [a[0], a[1], \ldots, a[n-3]]$
- $a[::-1] \leftarrow [a[n-1], a[n-2], \ldots, a[0]]$ (step -1)
- $a[1::-1] \leftarrow [a[1], a[0]]$
- $a[-3::-1] \leftarrow [a[n-3], a[n-4], \ldots, a[0]]$
- $a[:-3:-1] \leftarrow [a[n-1], a[n-2]]$

Gaussian Elimination for $Ku=f$ (square K)

$$A = \begin{array}{c|c} & \begin{array}{cccc} 0 & 1 & & \end{array} \\ \begin{array}{c} 0 \\ 1 \\ \\ \\ \end{array} & \left[\begin{array}{cccc|c} K_{00} & K_{01} & \cdots & K_{0,n-1} & f_0 \\ \boxed{K_{10}} & K_{11} & \cdots & \vdots & f_1 \\ \boxed{K_{20}} & \vdots & \ddots & & \vdots \\ K_{n-1,0} & \cdots & & K_{n-1,n-1} & f_{n-1} \end{array}\right] \end{array}$$

row ↓ col

$$\text{row } 1 \leftarrow \text{row } 1 - \frac{K_{10}}{K_{00}} \text{row } 0$$

$\vdots$

$$\boxed{\text{row } j \leftarrow \text{row } j - \frac{K_{j0}}{K_{00}} \text{row } 0}$$

$\vdots$

$$\text{row } n\text{-}1 \leftarrow \text{row } n\text{-}1 - \frac{K_{n-1,0}}{K_{00}} \text{row } 0$$

$\boxed{A[j,:]}$

```
for k in range(n+1):
    A[j,k] = A[j,k] - A[j,0]/A[0,0] * A[0,k]
```

$$\text{new } A = \left[\begin{array}{cccc|c} K_{00} & K_{01} & \cdots & K_{0,n-1} & f_0 \\ 0 & & & & \\ 0 & 0 & \text{new stuff} & & \text{new stuff} \\ \vdots & \vdots & 0 & & \\ 0 & 0 & 0 & & \end{array}\right]$$

lower triangular

$$A = \left[\begin{array}{cccccc|c} K_{00} & K_{01} & K_{02} & \cdots & K_{0,n-2} & K_{0,n-1} & f_0 \\ 0 & \tilde{K}_{11} & \tilde{K}_{12} & \cdots & \tilde{K}_{1,n-2} & \tilde{K}_{1,n-1} & \tilde{f}_1 \\ 0 & 0 & \tilde{K}_{22} & & \tilde{K}_{2,n-2} & \tilde{K}_{2,n-1} & \tilde{f}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \tilde{K}_{n-2,n-2} & \tilde{K}_{n-2,n-1} & \tilde{f}_{n-2} \\ 0 & 0 & 0 & \cdots & 0 & \tilde{K}_{n-1,n-1} & \tilde{f}_{n-1} \end{array}\right] = [\tilde{K} \mid \tilde{f}]$$

$$\boxed{\tilde{K} u = \tilde{f}}$$

$$\ldots + 0 + 0 + \tilde{K}_{n-1,n-1}\, u_{n-1} = \tilde{f}_{n-1} \Rightarrow u_{n-1} = \frac{\tilde{f}_{n-1}}{\tilde{K}_{n-1,n-1}}$$

$$0 + 0 + \tilde{K}_{n-2,n-2}\, u_{n-2} + \tilde{K}_{n-2,n-1}\, u_{n-1} = \tilde{f}_{n-2}$$

$$u_{n-2} = \frac{1}{\tilde{K}_{n-2,n-2}} \left(\tilde{f}_{n-2} - \tilde{K}_{n-2,n-1}\, u_{n-1} \right)$$

Back-substitution