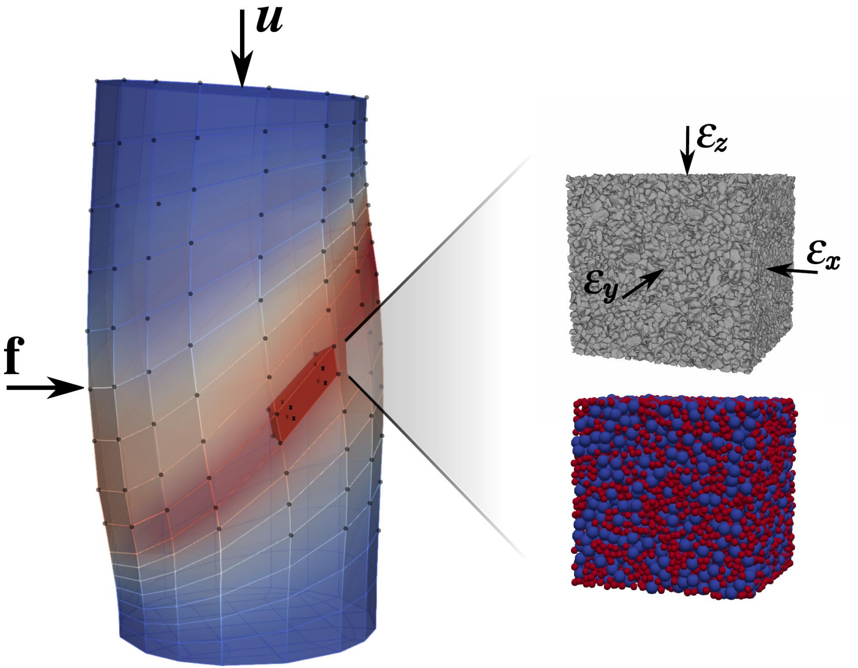


CIVIL-408 Multiscale Modeling in Mechanics

Prof. Kostas Karapiperis

Exercises - Week 11

Quiz

K.Karapiperis

What is the main advantage of the **Data-Driven Computational Mechanics** (DDCM) over **Neural Network** (NN)-based surrogate models?

a. DDCM is easier to train than NN.

b. DDCM can more efficiently interpolate between stresses and strains compared to NN.

c. DDCM indicates exactly what is the prediction error at run-time.

Quiz

What is the main advantage of the **Data-Driven Computational Mechanics** (DDCM) over **Neural Network** (NN)-based surrogate models?

a. DDCM is easier to train than NN.

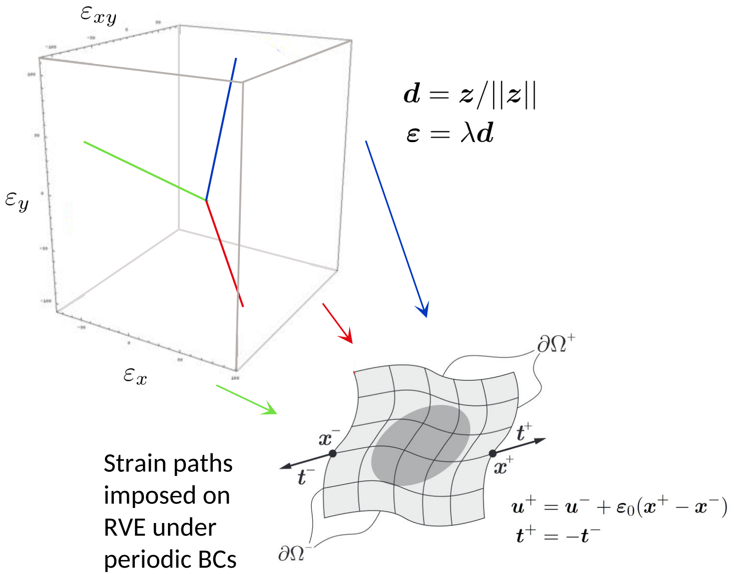
b. DDCM can more efficiently interpolate between stresses and strains compared to NN.

c. DDCM indicates exactly what is the prediction error at run-time.

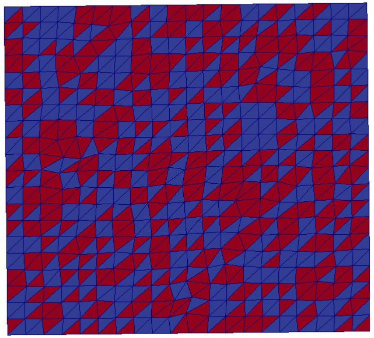
Data-driven multiscale modeling

Application to homogenization of elastic composite: Data

You are given a **material dataset** (same as in Exercise 10) obtained by probing the effective constitutive response of a 2D isotropic composite material along different directions by repeatedly solving a boundary value problem an RVE:



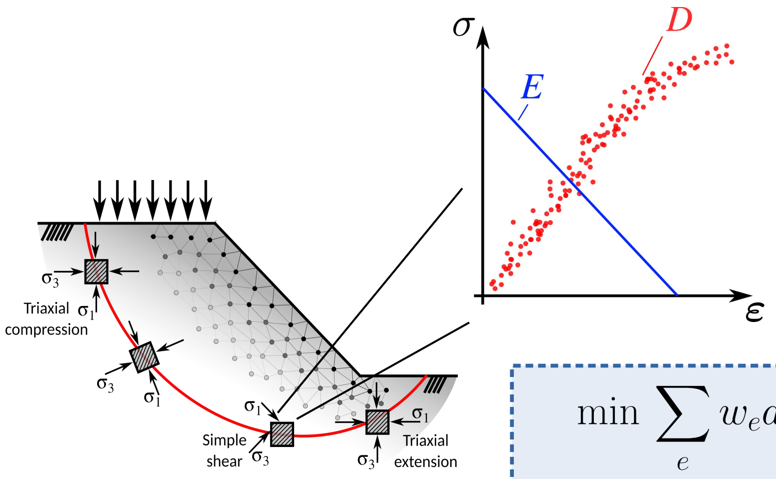
Strain paths imposed on RVE under periodic BCs



Isotropic composite

Data-driven multiscale modeling

Formulation



Perform double minimization in staggered fashion

$$\min_{\boldsymbol{z}\in E} \min_{\boldsymbol{y}\in D} d(\boldsymbol{z}, \boldsymbol{y})$$

$$\min \sum_e w_e d_e(\boldsymbol{\varepsilon}_e, \boldsymbol{\sigma}_e)$$

$$\text{s.t.} \quad \boldsymbol{\varepsilon}_e = \sum_i \boldsymbol{B}_{ei} u_i,$$

$$\sum_e w_e \boldsymbol{B}_{ei}^T \boldsymbol{\sigma}_e = F_i$$

→ Mechanical states

$$\mathbf{z} = \{(\boldsymbol{\varepsilon}^e, \boldsymbol{\sigma}^e)\}_{e=1,\dots,N}$$

$$d_e(\boldsymbol{\varepsilon}_e, \boldsymbol{\sigma}_e) = \min_{\boldsymbol{\varepsilon}_e^*, \boldsymbol{\sigma}_e^* \in D_e} (\boldsymbol{\varepsilon}_e - \boldsymbol{\varepsilon}_e^*)^T \mathbb{C}_e (\boldsymbol{\varepsilon}_e - \boldsymbol{\varepsilon}_e^*) + (\boldsymbol{\sigma}_e - \boldsymbol{\sigma}_e^*)^T \mathbb{C}_e^{-1} (\boldsymbol{\sigma}_e - \boldsymbol{\sigma}_e^*)$$

→ Material states

$$\mathbf{y} = \{(\boldsymbol{\varepsilon}^{e*}, \boldsymbol{\sigma}^{e*})\}_{e=1,\dots,N}$$

The stiffness $\mathbb{C}_e$ is simply a distance-inducing numerical parameter

Data-driven multiscale modeling

Fixed-point algorithm

Iterative scheme, involving:

i) Solution of two modified ‘elasticity’ problems

ii) Database search

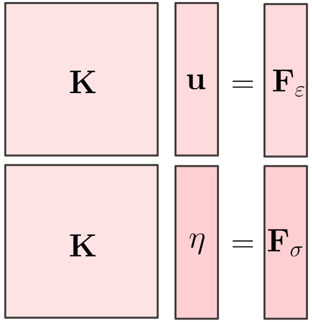
$$\Big(\sum_{e=1}^{M} w_e \mathbf{B}_e^T \mathbb{C}_e \mathbf{B}_e\Big)\mathbf{u}^{(i)} = \sum_{e=1}^{M} w_e \mathbf{B}_e^T \mathbb{C}_e \varepsilon_e^{*(i)} \quad (1)$$

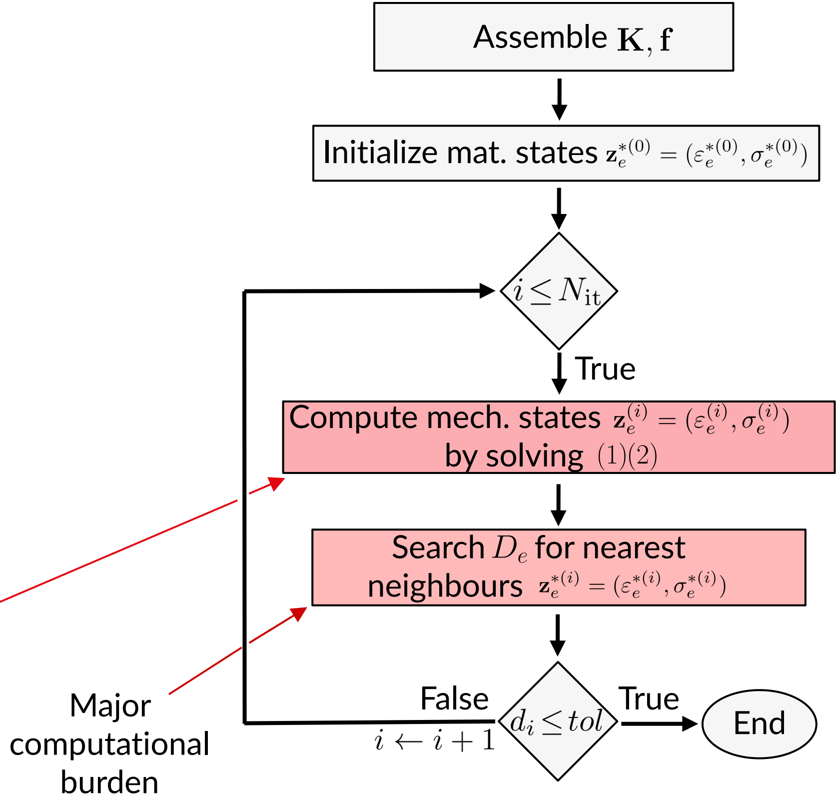
$$\Big(\sum_{e=1}^{M} w_e \mathbf{B}_e^T \mathbb{C}_e \mathbf{B}_e\Big)\boldsymbol{\eta}^{(i)} = \mathbf{f} - \sum_{e=1}^{M} w_e \mathbf{B}_e^T \sigma_e^{*(i)} \quad (2)$$

$$\sigma_e^{(i)} = \sigma_e^{*(i)} + \mathbb{C}_e \sum_{\alpha=1}^{N} \mathbf{B}_{e\alpha} \eta_\alpha^{(i)}$$

Positive-definite distance-inducing tensor

$$|\boldsymbol{z}^e| = \mathbb{C}^e \boldsymbol{\varepsilon}^e \cdot \boldsymbol{\varepsilon}^e + \mathbb{C}^{e^{-1}} \boldsymbol{\sigma}^e \cdot \boldsymbol{\sigma}^e$$



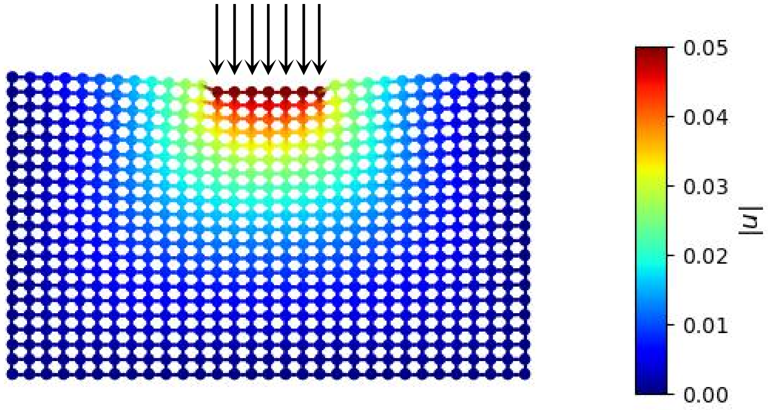


Exercise

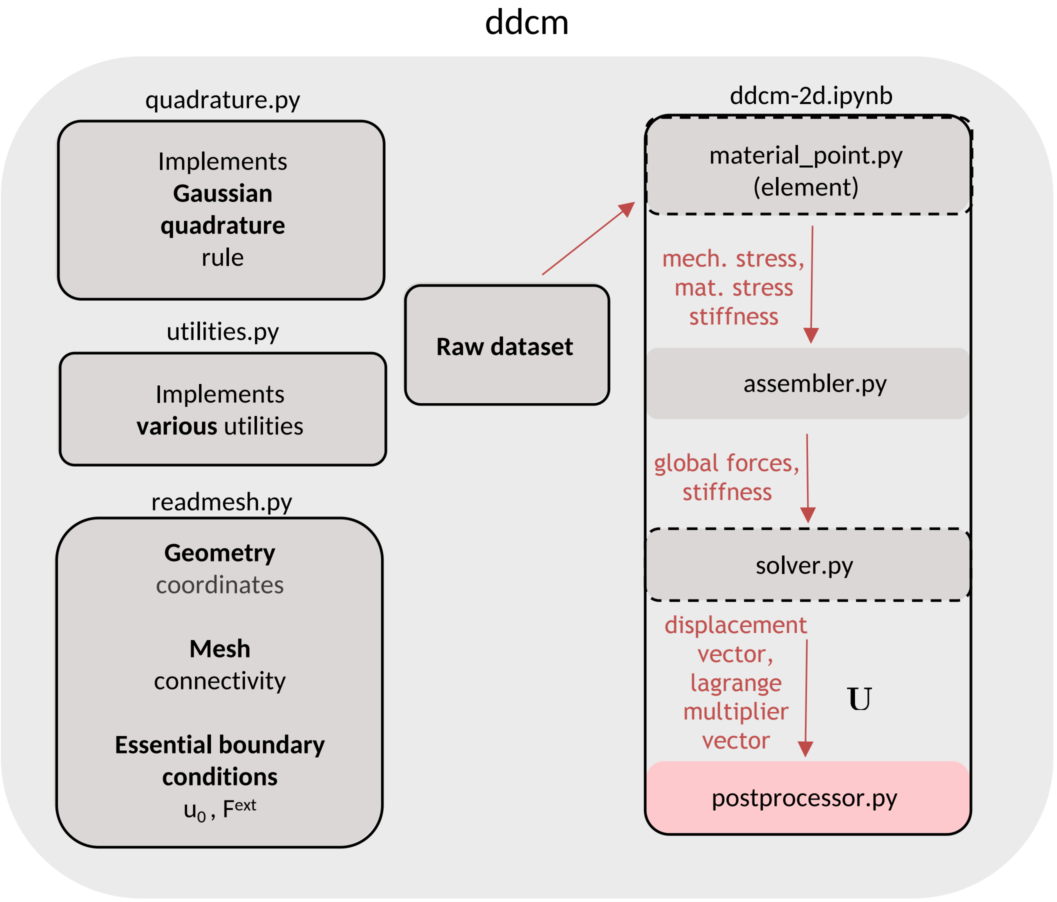
Solve a 2D flat punch problem using DDCM

Our task is to setup the DDCM problem on the provided FEM discretized model.

During the solution, we monitor the decay in the distance until convergence.



Python implementation



(**Task**: Implementation of the assignment of local states - database search)

(**Task**: Implementation of the fixed-point iteration solver)

Legend

To do

Python notebook

Let’s move to the Python notebook