Introduction to FFT-based numerical methods for the homogenization of random materials (14–18 march 2022)

Consistent discretization of the LS equation

Sébastien Brisard

Laboratoire Navier, École des Ponts, Univ. Gustave Eiffel, CNRS, Marne-la-Vallée, France

Outline of Lecture 2

- Weak form of the LS equation
- Galerkin discretization of the LS equation
- The discretized operators
- Applying the discrete Green operator
- Towards linear LS solvers
- The last piece of the jigsaw

Bibliographic notes

- The contents of this lecture is largely based on refs. [1, 2]
- I used the book by Ern and Guermond [3] for the proofs

[1] S. Brisard, L. Dormieux, *Computational Materials Science* **2010**, *49*, 663–671.

[2] S. Brisard, L. Dormieux, *Computer Methods in Applied Mechanics and Engineering* **2012**, *217–220*, 197–212.

[3] A. Ern, J.-L. Guermond, *Theory and Practice of Finite Elements*, Springer-Verlag, New York, **2004**.

Outline of Lecture 2

- Weak form of the LS equation
- Galerkin discretization of the LS equation
- The discretized operators
- Applying the discrete Green operator
- Towards linear LS solvers
- The last piece of the jigsaw

The stress-polarization based LS equation

Remember: equivalent formulation of the corrector problem

Find $\boldsymbol{\varepsilon}, \boldsymbol{\tau} \in \mathcal{T}(\Omega)$ such that $\begin{cases} \boldsymbol{\varepsilon} = \overline{\boldsymbol{\varepsilon}} - \boldsymbol{\Gamma}_0(\boldsymbol{\tau}) \\ \boldsymbol{\tau} = \boldsymbol{\sigma} - \mathbf{C}_0 : \boldsymbol{\varepsilon} = (\mathbf{C} - \mathbf{C}_0) : \boldsymbol{\varepsilon} + \boldsymbol{\varpi} \end{cases}$

Polarization-based form of LS equation [1]

Find $\boldsymbol{\tau} \in \mathcal{T}(\Omega)$ such that $(\mathbf{C} - \mathbf{C}_0)^{-1} : \boldsymbol{\tau} + \boldsymbol{\Gamma}_0(\boldsymbol{\tau}) = \overline{\boldsymbol{\varepsilon}} + (\mathbf{C} - \mathbf{C}_0)^{-1} : \boldsymbol{\varpi}$

Getting rid of eigenstresses

Find $\boldsymbol{\tau} \in \mathcal{T}(\Omega)$ such that $(\mathbf{C} - \mathbf{C}_0)^{-1} : \boldsymbol{\tau} + \boldsymbol{\Gamma}_0(\boldsymbol{\tau}) = \overline{\boldsymbol{\varepsilon}}$

$\overline{\boldsymbol{\varepsilon}} \in \mathcal{T}(\Omega)$ possibly heterogeneous!

[1] J. Willis, *Journal of the Mechanics and Physics of Solids* **1977**, *25*, 185–202.

Weak form of the LS equation

1. Start from strong form

Find $\boldsymbol{\tau} \in \mathcal{T}(\Omega)$ such that, for all $\mathbf{x} \in \Omega$:

$$\left[\mathbf{C}(\mathbf{x}) - \mathbf{C}_0(\mathbf{x})\right]^{-1} : \boldsymbol{\tau}(\mathbf{x}) + \boldsymbol{\Gamma}_0(\boldsymbol{\tau})(\mathbf{x}) = \bar{\boldsymbol{\varepsilon}}(\mathbf{x})$$

2. Multiply by arbitrary test function

Find $\boldsymbol{\tau} \in \mathcal{T}(\Omega)$ such that, for all $\mathbf{x} \in \Omega$ and $\boldsymbol{\theta} \in \mathcal{T}(\Omega)$:

$$\boldsymbol{\theta}(\mathbf{x}) : \left[\mathbf{C}(\mathbf{x}) - \mathbf{C}_0(\mathbf{x})\right]^{-1} : \boldsymbol{\tau}(\mathbf{x}) + \boldsymbol{\theta}(\mathbf{x}) : \boldsymbol{\Gamma}_0(\boldsymbol{\tau})(\mathbf{x}) = \bar{\boldsymbol{\varepsilon}}(\mathbf{x}) : \boldsymbol{\theta}(\mathbf{x})$$

3. Take volume average over Ω

Find $\boldsymbol{\tau} \in \mathcal{T}(\Omega)$ such that, for all $\boldsymbol{\theta} \in \mathcal{T}(\Omega)$

$$\langle \boldsymbol{\theta} : \left(\mathbf{C} - \mathbf{C}_0\right)^{-1} : \boldsymbol{\tau} \rangle + \langle \boldsymbol{\theta} : \boldsymbol{\Gamma}_0(\boldsymbol{\tau}) \rangle = \langle \bar{\boldsymbol{\varepsilon}} : \boldsymbol{\theta} \rangle$$

Outline of Lecture 2

- Weak form of the LS equation
- Galerkin discretization of the LS equation
- The discretized operators
- Applying the discrete Green operator
- Towards linear LS solvers
- The last piece of the jigsaw

Galerkin discretization of the LS equation

The initial variational problem

Find $\boldsymbol{\tau} \in \mathcal{T}(\Omega)$ such that, for all $\boldsymbol{\theta} \in \mathcal{T}(\Omega)$

$$\underbrace{\langle \boldsymbol{\theta} : (\mathbf{C} - \mathbf{C}_0)^{-1} : \boldsymbol{\tau} \rangle + \langle \boldsymbol{\theta} : \boldsymbol{\Gamma}_0(\boldsymbol{\tau}) \rangle}_{a(\boldsymbol{\tau}, \boldsymbol{\theta})} = \underbrace{\langle \bar{\boldsymbol{\varepsilon}} : \boldsymbol{\theta} \rangle}_{\ell(\boldsymbol{\theta})}$$

The approximation space

- $\mathcal{T}^N(\Omega) \subset \mathcal{T}(\Omega)$: finite dimension subspace
- N: discretization parameter (to be defined)

The discretized variational problem

Find $\boldsymbol{\tau}^N \in \mathcal{T}^N(\Omega)$ such that, for all $\boldsymbol{\theta}^N \in \mathcal{T}^N(\Omega)$

$$\langle \boldsymbol{\theta}^N : (\mathbf{C} - \mathbf{C}_0)^{-1} : \boldsymbol{\tau}^N \rangle + \langle \boldsymbol{\theta}^N : \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle = \langle \bar{\boldsymbol{\varepsilon}} : \boldsymbol{\theta}^N \rangle$$

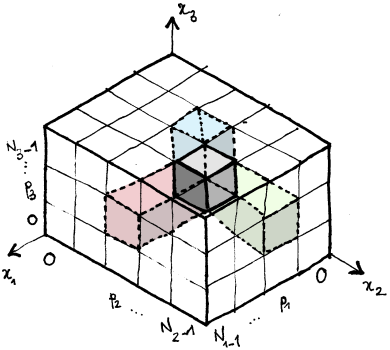
The approximation subspace (1/2)

Discretization over a grid

- **Regular grid** of size $N = (N_1, \dots, N_d)$ over unit-cell Ω
- Grid spacing: $h_i = L_i / N_i$, total number of cells: $\mathcal{N} = N_1 \dots N_d$

Numbering of cells

$$\mathcal{P} = \{0, \dots, N_1 - 1\} \times \dots \times \{0, \dots, N_d - 1\}$$



Cell average

$$\langle \mathcal{Q} \rangle_p \overset{\text{def}}{=} \frac{1}{h_1 \dots h_d} \int_{\mathbf{x} \in \Omega_p} \mathcal{Q}(\mathbf{x}) \, \mathrm{d}x_1 \dots \mathrm{d}x_d$$

Average over whole unit-cell

$$\langle \mathcal{Q} \rangle = \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \langle \mathcal{Q} \rangle_p$$

The approximation subspace (2/2)

Definition of $\mathcal{T}^N(\Omega)$

Space of **cell-wise constant**, 2nd-order, symmetric tensors

$$\text{number of dofs} = \dim \mathcal{T}^N = \tfrac{1}{2}\mathcal{N} d\bigl(d+1\bigr)$$

Trial and test functions defined by their cell-values

$$\boldsymbol{\tau}^N(\mathbf{x}) = \boldsymbol{\tau}_p^N \quad \text{and} \quad \boldsymbol{\theta}^N(\mathbf{x}) = \boldsymbol{\theta}_p^N \qquad (\mathbf{x} \in \Omega_p)$$

Cell-averages of trial and test functions

$$\langle\boldsymbol{\tau}^N\rangle_p = \boldsymbol{\tau}_p^N \quad \text{and} \quad \langle\boldsymbol{\theta}^N\rangle_p = \boldsymbol{\theta}_p^N$$

Outline of Lecture 2

- Weak form of the LS equation
- Galerkin discretization of the LS equation
- The discretized operators
- Applying the discrete Green operator
- Towards linear LS solvers
- The last piece of the jigsaw

Evaluating ℓ over $\mathcal{T}^N(\Omega)$

$$\ell(\boldsymbol{\theta}) = \langle \bar{\boldsymbol{\varepsilon}} : \boldsymbol{\theta} \rangle$$

$$\ell(\boldsymbol{\theta}^N) = \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \langle \boldsymbol{\theta}^N : \bar{\boldsymbol{\varepsilon}} \rangle_p = \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \boldsymbol{\theta}_p^N : \langle \bar{\boldsymbol{\varepsilon}} \rangle_p = \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \boldsymbol{\theta}_p^N : \bar{\boldsymbol{\varepsilon}}_p^N$$

$$\bar{\boldsymbol{\varepsilon}}_p^N \stackrel{\text{def}}{=} \langle \bar{\boldsymbol{\varepsilon}} \rangle_p = \bar{\boldsymbol{\varepsilon}} + \langle \left(\mathbf{C} - \mathbf{C}_0\right)^{-1} : \boldsymbol{\varpi} \rangle_p$$

$\bar{\boldsymbol{\varepsilon}}^N$ can be seen as a cell-wise constant tensor field!

$$\ell(\boldsymbol{\theta}^N) = \langle \bar{\boldsymbol{\varepsilon}}^N : \boldsymbol{\theta}^N \rangle$$

Evaluating a over $\mathcal{T}^N(\Omega)$ (1/3)

$$a(\boldsymbol{\tau}, \boldsymbol{\theta}) = \langle \boldsymbol{\theta} : (\mathbf{C} - \mathbf{C}_0)^{-1} : \boldsymbol{\tau} \rangle + \langle \boldsymbol{\theta} : \boldsymbol{\Gamma}_0(\boldsymbol{\tau}) \rangle$$

$$\langle \boldsymbol{\theta}^N : (\mathbf{C} - \mathbf{C}_0)^{-1} : \boldsymbol{\tau}^N \rangle = \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \langle \boldsymbol{\theta}^N : (\mathbf{C} - \mathbf{C}_0)^{-1} : \boldsymbol{\tau}^N \rangle_p$$

$$= \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \boldsymbol{\theta}_p^N : \langle (\mathbf{C} - \mathbf{C}_0)^{-1} \rangle_p : \boldsymbol{\tau}_p^N$$

$$= \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \boldsymbol{\theta}_p^N : (\mathbf{C}_p^N - \mathbf{C}_0)^{-1} : \boldsymbol{\tau}_p^N$$

$$\mathbf{C}_p^N \stackrel{\text{def}}{=} \mathbf{C}_0 + \left[\langle (\mathbf{C} - \mathbf{C}_0)^{-1} \rangle_p \right]^{-1}$$

$$\langle \boldsymbol{\theta}^N : (\mathbf{C} - \mathbf{C}_0)^{-1} : \boldsymbol{\tau}^N \rangle = \langle \boldsymbol{\theta}^N : (\mathbf{C}^N - \mathbf{C}_0)^{-1} : \boldsymbol{\tau}^N \rangle$$

Evaluating a over $\mathcal{T}^N(\Omega)$ (2/3)

$$a(\boldsymbol{\tau}, \boldsymbol{\theta}) = \langle \boldsymbol{\theta} : \left(\mathbf{C} - \mathbf{C}_0\right)^{-1} : \boldsymbol{\tau} \rangle + \langle \boldsymbol{\theta} : \boldsymbol{\Gamma}_0(\boldsymbol{\tau}) \rangle$$

$$\langle \boldsymbol{\theta}^N : \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle = \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \langle \boldsymbol{\theta}^N : \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p = \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \boldsymbol{\theta}_p^N : \langle \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p$$

Introducing the discrete Green operator

- Let $\boldsymbol{\eta}_p^N = \langle \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p$: cell-values of $\boldsymbol{\eta}^N \in \mathcal{T}^N(\Omega)$
- The mapping $\boldsymbol{\tau}^N \mapsto \boldsymbol{\eta}^N$ is an **endomorphism** over $\mathcal{T}^N(\Omega)$
- This endomorphism is the **discrete Green operator** $\boldsymbol{\Gamma}_0^N$

Evaluating a over $\mathcal{T}^N(\Omega)$ (3/3)

Formal definition of the discrete Green operator

$$\boldsymbol{\Gamma}_0^N : \begin{cases} \mathcal{T}^N(\Omega) \longrightarrow \mathcal{T}^N(\Omega) \\ \boldsymbol{\tau}^N \mapsto \boldsymbol{\eta}^N \end{cases} \quad \text{such that} \quad \boldsymbol{\eta}_p^N = \langle \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p$$

Cell-average of (the opposite of) the strain induced by a cell-wise constant eigenstress

Going back to the bilinear form

$$\langle \boldsymbol{\theta}^N : \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle = \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \boldsymbol{\theta}_p^N : \langle \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p = \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \boldsymbol{\theta}_p^N : \boldsymbol{\eta}_p^N$$

$$= \frac{1}{\mathcal{N}} \sum_{p \in \mathcal{P}} \langle \boldsymbol{\theta}^N : \boldsymbol{\eta}^N \rangle_p = \langle \boldsymbol{\theta}^N : \boldsymbol{\eta}^N \rangle = \langle \boldsymbol{\theta}^N : \boldsymbol{\Gamma}_0^N(\boldsymbol{\eta}^N) \rangle$$

The discrete LS equation

Exact evaluation of the linear and bilinear forms

$$\ell(\boldsymbol{\theta}^N) = \langle \overline{\boldsymbol{\varepsilon}}^N : \boldsymbol{\theta}^N \rangle$$

$$a(\boldsymbol{\tau}^N, \boldsymbol{\theta}^N) = \langle \boldsymbol{\theta}^N : \left(\mathbf{C}^N - \mathbf{C}_0\right)^{-1} : \boldsymbol{\tau}^N \rangle + \langle \boldsymbol{\theta}^N : \boldsymbol{\Gamma}_0^N(\boldsymbol{\eta}^N) \rangle$$

Discrete variational problem

Find $\boldsymbol{\tau}^N \in \mathcal{T}^N(\Omega)$ such that, for all $\boldsymbol{\theta}^N \in \mathcal{T}^N(\Omega)$

$$\langle \boldsymbol{\theta}^N : \left(\mathbf{C}^N - \mathbf{C}_0\right)^{-1} : \boldsymbol{\tau}^N \rangle + \langle \boldsymbol{\theta}^N : \boldsymbol{\Gamma}_0^N(\boldsymbol{\eta}^N) \rangle = \langle \overline{\boldsymbol{\varepsilon}}^N : \boldsymbol{\theta}^N \rangle$$

The associated linear system

$$\left(\mathbf{C}^N - \mathbf{C}_0\right)^{-1} : \boldsymbol{\tau}^N + \boldsymbol{\Gamma}_0^N(\boldsymbol{\tau}^N) = \overline{\boldsymbol{\varepsilon}}^N$$

Outline of Lecture 2

- Weak form of the LS equation
- Galerkin discretization of the LS equation
- The discretized operators
- Applying the discrete Green operator
- Towards linear LS solvers
- The last piece of the jigsaw

Announcing the Fast Fourier Transform

Formal definition of the discrete Green operator

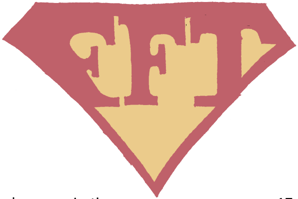
$$\boldsymbol{\Gamma}_0^N : \begin{cases} \mathcal{T}^N(\Omega) \longrightarrow \mathcal{T}^N(\Omega) \\ \boldsymbol{\tau}^N \mapsto \boldsymbol{\eta}^N \end{cases} \quad \text{such that} \quad \boldsymbol{\eta}_p^N = \langle \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p$$

Translation-invariance

$$\langle \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p = \sum_{q \in \mathcal{P}} \boldsymbol{\Gamma}_{0,p,q}^N : \boldsymbol{\tau}_q^N = \sum_{q \in \mathcal{P}} \boldsymbol{\Gamma}_{0,p-q}^N : \boldsymbol{\tau}_q^N$$

$\boldsymbol{\Gamma}_{0,p} : \boldsymbol{\tau}_0$ is the (opposite of the) average strain in cell p induced by the eigenstress $\boldsymbol{\tau}_0$ in the $(0, \dots, 0)$ cell.

This looks like a job for the Fast Fourier Transform!



On the discrete Fourier transform (1/2)

Input and output data are finite sets of numbers

$$(x_p)_p \quad \text{and} \quad (\hat{x}_n)_n \quad \text{with} \quad 0 \leq p_i, n_i < N_i \quad \text{and} \quad i = 1, \dots, d$$

Definition

$$\hat{x}_n \overset{\text{def}}{=} \sum_{p \in \mathcal{P}} x_p U_{np}^{N,*} \quad \text{with} \quad U_n^N \overset{\text{def}}{=} \exp\left[2i\pi\left(\frac{n_1}{N_1} + \cdots + \frac{n_d}{N_d}\right)\right]$$

Output is a discrete, **periodic** series: $\hat{x}_{n+mN} = \hat{x}_n$

Implementation: fast Fourier transform (FFT)
$\mathcal{O}(N \log N)$ **rather than** $\mathcal{O}(N^2)$

Note

$$\begin{cases} n + p \\ np \end{cases} \quad \text{should be understood as the tuple} \quad \begin{cases} (n_1 + p_1, \dots, n_d + p_d) \\ (n_1 p_1, \dots, n_d p_d) \end{cases}$$

On the discrete Fourier transform (2/2)

Inversion

$$x_p = \frac{1}{\mathcal{N}} \sum_{n \in \mathcal{P}} \hat{x}_n U_{np}^N$$

Input can also be seen as a discrete,
periodic series ($x_{p+qN} \overset{\text{def}}{=} x_p$).

Plancherel theorem

$$\sum_{p \in \mathcal{P}} x_p^* \, y_p = \frac{1}{\mathcal{N}} \sum_{n \in \mathcal{P}} \hat{x}_n^* \, \hat{y}_n$$

Circular convolution theorem

$$\widehat{x \star y}_n = \hat{x}_n \, \hat{y}_n \quad \text{with} \quad (x \star y)_p \overset{\text{def}}{=} \sum_{q \in \mathcal{P}} x_{p-q} \, y_q$$

DFTs and the discrete Green operator

Definition of the discrete Green operator

$$\boldsymbol{\eta}^N = \boldsymbol{\Gamma}_0^N(\boldsymbol{\tau}^N) \quad \text{such that} \quad \boldsymbol{\eta}_p^N = \langle \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p$$

Translation-invariant expression

$$\langle \boldsymbol{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p = \sum_{q \in \mathcal{P}} \boldsymbol{\Gamma}_{0,p-q}^N : \boldsymbol{\tau}_q^N$$

Introduce DFT

$$\hat{\boldsymbol{\Gamma}}_{0,n}^N = \mathrm{DFT}_n\left(\boldsymbol{\Gamma}_{0,\bullet}^N\right)$$

Use circular convolution theorem

$$\boldsymbol{\eta}^N = \mathrm{DFT}^{-1}\left(\hat{\boldsymbol{\Gamma}}_0^N : \hat{\boldsymbol{\tau}}^N\right) = \mathrm{DFT}^{-1}\left[\hat{\boldsymbol{\Gamma}}_0^N : \mathrm{DFT}(\boldsymbol{\tau}^N)\right]$$

Note that we still don't know the $\hat{\boldsymbol{\Gamma}}_{0,\bullet}^N$!!!

Pseudo-implementation (1/2)

```
class DiscreteGreenOperator:
    def __init__(self, mu0, nu0, grid_shape):
        self.mu0 = mu0 # Elastic constants of
        self.nu0 = nu0 # reference material
        self.C0 = ...  # Stiffness as a matrix
        self.grid_shape = grid_shape
        self.dim = len(grid_shape)
        self.spatial_axes = tuple(range(self.dim))

    def cell_indices(self):
        ranges = map(range, self.spatial_axes)
        return itertools.product(*ranges)

    # ... To be continued...
```

Pseudo-implementation (2/2)

```python
class DiscreteGreenOperator:
    # ... Continued...

    def fourier_mode(self, n):
        # Return n-th Fourier mode as a matrix
        #
        # !!! WE STILL NEED TO DERIVE THESE GUYS !!!
        #

    def apply(self, tau):
        tau_hat = np.fftn(tau, axes=self.spatial_axes)
        eta_hat = np.empty_like(tau_hat)
        for n in self.cell_indices():
            eta_hat[n] = self.fourier_mode(n) @ tau_hat[n]
        return np.ifftn(eta_hat, axes=self.spatial_axes)
```

Outline of Lecture 2

- Weak form of the LS equation
- Galerkin discretization of the LS equation
- The discretized operators
- Applying the discrete Green operator
- Towards linear LS solvers
- The last piece of the jigsaw

What we have shown so far

The discretized LS equation

$$\left(\mathbf{C}^N - \mathbf{C}_0\right)^{-1} : \boldsymbol{\tau}^N + \boldsymbol{\Gamma}_0^N(\boldsymbol{\tau}^N) = \overline{\boldsymbol{\varepsilon}}^N$$

$$\overline{\boldsymbol{\varepsilon}}_p^N = \overline{\boldsymbol{\varepsilon}} + \langle\left(\mathbf{C} - \mathbf{C}_0\right)^{-1} : \boldsymbol{\varpi}\rangle_p$$

$$\mathbf{C}_p^N = \mathbf{C}_0 + \left[\langle\left(\mathbf{C} - \mathbf{C}_0\right)^{-1}\rangle_p\right]^{-1}$$

$$\boldsymbol{\Gamma}_0^N(\boldsymbol{\tau}^N) = \mathrm{DFT}^{-1}\left[\hat{\boldsymbol{\Gamma}}_0^N : \mathrm{DFT}(\boldsymbol{\tau}^N)\right]$$

Notes

- Convergence wrt grid-size can be proved (using rudimentary FE tools!)
- This linear system must be implemented!

The Kelvin–Mandel representation

2nd order, symmetric, tensors

$$[\mathbf{s}] = \left[s_{11}, s_{22}, s_{33}, \sqrt{2}s_{23}, \sqrt{2}s_{31}, \sqrt{2}s_{12}\right]^{\mathsf{T}}$$

4th order tensors with minor symmetries

$$[\mathbf{T}] = \begin{bmatrix} T_{1111} & T_{1122} & T_{1133} & \sqrt{2}T_{1123} & \sqrt{2}T_{1131} & \sqrt{2}T_{1112} \\ T_{2211} & T_{2222} & T_{2233} & \sqrt{2}T_{2223} & \sqrt{2}T_{2231} & \sqrt{2}T_{2212} \\ T_{3311} & T_{3322} & T_{3333} & \sqrt{2}T_{3323} & \sqrt{2}T_{3331} & \sqrt{2}T_{3312} \\ \sqrt{2}T_{2311} & \sqrt{2}T_{2322} & \sqrt{2}T_{2333} & 2T_{2323} & 2T_{2331} & 2T_{2312} \\ \sqrt{2}T_{3111} & \sqrt{2}T_{3122} & \sqrt{2}T_{3133} & 2T_{3123} & 2T_{3131} & 2T_{3112} \\ \sqrt{2}T_{1211} & \sqrt{2}T_{1222} & \sqrt{2}T_{1233} & 2T_{1223} & 2T_{1231} & 2T_{1212} \end{bmatrix}$$

Some properties

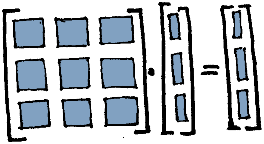
$$\mathbf{s}_1 : \mathbf{s}_2 = [\mathbf{s}_1]^{\mathsf{T}} \cdot [\mathbf{s}_2] \qquad [\mathbf{T} : \mathbf{s}] = [\mathbf{T}] \cdot [\mathbf{s}] \qquad [\mathbf{T}^{-1}] = [\mathbf{T}]^{-1}$$

The structure of the linear system

$$\left(\mathbf{C}^N - \mathbf{C}_0\right)^{-1} : \boldsymbol{\tau}^N + \boldsymbol{\Gamma}_0^N(\boldsymbol{\tau}^N) = \overline{\boldsymbol{\varepsilon}}^N \qquad \Leftrightarrow \qquad A \cdot x = b$$

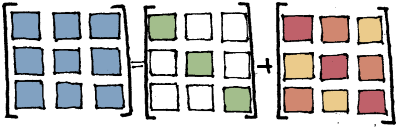
- $s = d(d+1)/2$
- x and b are defined by $s \times 1$ blocks

$$x_p = [\boldsymbol{\tau}_p] \quad \text{and} \quad b_p = [\overline{\boldsymbol{\varepsilon}}_p^N]$$



- A is defined by $s \times s$ blocks

$$A_{pq} = \underbrace{\delta_{pq}[\mathbf{C}^N - \mathbf{C}_0]^{-1}}_{\text{block-diagonal}} + \underbrace{[\boldsymbol{\Gamma}_{0,p-q}^N]}_{\text{block-circulant}}$$



Storage would in principle be possible!
(but we don't do that)

On iterative (matrix-free) linear solvers

$$A \cdot x = b$$

```
import scipy.sparse.linalg

class MyOperator(scipy.sparse.linalg.LinearOperator):
    def __init__(self, ...):
        pass

    def _matvec(self, x):
        # Compute y = A.x
        return y

A = MyOperator()
b = ...
x, info = scipy.sparse.linalg.cg(A, b)
```

Pseudo-implementation (1/2)

```
class LippmannSchwingerOperator(LinearOperator):
    def __init__(self, C, Gamma0):
        self.C = np.copy(C)
        self.Gamma0 = Gamma0
        dim = Gamma0.dim
        sym = (dim * (dim + 1)) // 2
        self.tau_shape = Gamma0.grid_shape + (sym,)

    # ... To be continued...
```

Pseudo-implementation (2/2)

```python
class LippmannSchwingerOperator(LinearOperator):
    # ... Continued...

    def polarization_to_strain(self, tau):
        eta = np.empty_like(tau)
        C0 = self.Gamma0.C0
        for p in self.Gamma0.cell_indices():
            eta[p] = np.linalg.solve(self.C[p]-C0, tau[p])
        return eta

    def _matvec(self, x):
        tau = x.reshape(self.tau_shape)
        eta1 = self.polarization_to_strain(tau)
        eta2 = self.Gamma0.apply(tau)
        return  eta1 + eta2
```

Outline of Lecture 2

- Weak form of the LS equation
- Galerkin discretization of the LS equation
- The discretized operators
- Applying the discrete Green operator
- Towards linear LS solvers
- The last piece of the jigsaw

Fourier coefficients of cell-wise constant functions

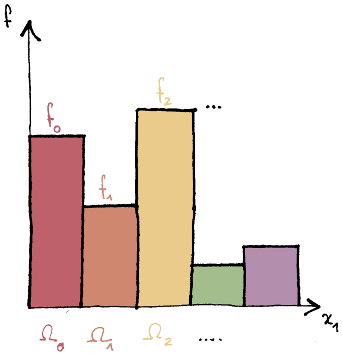
Cell-wise constant functions

$$\mathbf{x} \in \Omega_p : \quad f(\mathbf{x}) = f_p$$

Fourier coefficients

$$\tilde{f}_n = \frac{W_n^N U_{n/2}^{N,*}}{\mathcal{N}} \hat{f}_n$$

$$W_n^N = \operatorname{sinc} \frac{\pi n_1}{N_1} \cdots \operatorname{sinc} \frac{\pi n_d}{N_d}$$



- $\tilde{f}_n$: Fourier coefficients of the **periodic function** f (**infinite** series)
- $\hat{f}_n$: discrete Fourier transform of the **cell values** f_p (**finite** series)

Cell-averages in Fourier space

Let f be *any* Ω-periodic function.

$$\langle f \rangle_p = \sum_{n\in\mathcal{P}} \left[\sum_{m\in\mathbb{Z}^d} (-1)^m W^N_{n+mN} \tilde{f}_{n+mN} \right] U^N_{n(p+1/2)}$$

Let g be a cell-wise constant function. From Plancherel's theorem

$$\langle f\, g^* \rangle = \sum_{n\in\mathbb{Z}^d} \tilde{f}_n\, \tilde{g}^*_n = \frac{1}{\mathcal{N}} \sum_{n\in\mathbb{Z}^d} W^N_n U^N_{n/2} \tilde{f}_n\, \hat{g}^*_n = \frac{1}{\mathcal{N}} \sum_{n\in\mathcal{P}} \sum_{m\in\mathbb{Z}^d} W^N_{n+mN}\, \underbrace{U^N_{(n+mN)/2}}_{=(-1)^m U^N_{n/2}} \tilde{f}_{n+mN}\, \underbrace{\hat{g}^*_{n+mN}}_{=\hat{g}^*_n}$$

Therefore

$$\frac{1}{h_1 \dots h_d} \int_\Omega f\, g^* = \sum_{n\in\mathcal{P}} \left(\sum_{m\in\mathbb{Z}^d} (-1)^m W_{n+mN} \tilde{f}_{n+mN} \right) \hat{g}^*_n U^N_{n/2}$$

Let g be the indicator function of Ω_p ($p \in \mathcal{P}$): $g_q = \delta_{pq}$. Then

$$\hat{g}_n = \sum_{q\in\mathcal{P}} \delta_{pq} U^{N*}_{nq} = U^{N*}_{np}$$

Expression of the discrete Green op. (1/2)

Remember expression of continuous Green operator

$$\mathbf{\Gamma}_0(\boldsymbol{\tau})(\mathbf{x}) = \sum_{n\in\mathbb{Z}^d} \tilde{\Gamma}_0(\mathbf{k}_n) : \tilde{\boldsymbol{\tau}}_n e^{i\mathbf{k}_n\cdot\mathbf{x}}$$

Remember definition of discrete Green operator

$$\boldsymbol{\tau}^N \in \mathcal{T}^N(\Omega) \mapsto \boldsymbol{\eta}^N = \mathbf{\Gamma}_0^N(\boldsymbol{\tau}^N) \in \mathcal{T}^N(\Omega) \quad \text{such that} \quad \boldsymbol{\eta}_p^N = \langle \mathbf{\Gamma}_0(\boldsymbol{\tau}^N)\rangle_p$$

Average strain induced by cell-wise constant eigenstress

$$\tilde{\boldsymbol{\eta}}_n = \tilde{\mathbf{\Gamma}}_0(\mathbf{k}_n) : \tilde{\boldsymbol{\tau}}_n^N = \frac{W_n^N U_{n/2}^{N,*}}{\mathcal{N}} \tilde{\mathbf{\Gamma}}_0(\mathbf{k}_n) : \hat{\boldsymbol{\tau}}_n^N \quad \text{with} \quad \boldsymbol{\eta} = \mathbf{\Gamma}_0(\boldsymbol{\tau}^N)$$

$$\langle\boldsymbol{\eta}\rangle_p = \frac{1}{\mathcal{N}} \sum_{n\in\mathcal{P}} \left[\sum_{m\in\mathbb{Z}^d} \left(W_{n+mN}^N\right)^2 \tilde{\mathbf{\Gamma}}_0(\mathbf{k}_{n+mN}) : \underbrace{\hat{\boldsymbol{\tau}}_{n+mN}^N}_{=\hat{\boldsymbol{\tau}}_n^N} \right] U_{np}^N$$

Expression of the discrete Green op. (2/2)

What we have shown so far

$$\langle \mathbf{\Gamma}_0(\boldsymbol{\tau}^N) \rangle_p = \frac{1}{\mathcal{N}} \sum_{n \in \mathcal{P}} \left[\sum_{m \in \mathbb{Z}^d} \left(W_{n+mN}^N \right)^2 \hat{\mathbf{\Gamma}}_0^\infty(\mathbf{k}_{n+mN}) \right] : \hat{\boldsymbol{\tau}}_n^N \, U_{np}^N$$

Introduce the following quantity

$$\hat{\mathbf{\Gamma}}_{0,n}^N = \sum_{m \in \mathbb{Z}^d} \left(W_{n+mN}^N \right)^2 \hat{\mathbf{\Gamma}}_0^\infty(\mathbf{k}_{n+mN})$$

Then

$$\langle \mathbf{\Gamma}_0(\boldsymbol{\tau}) \rangle_p = \frac{1}{\mathcal{N}} \sum_{n \in \mathcal{P}} \hat{\mathbf{\Gamma}}_{0,n}^N : \hat{\boldsymbol{\tau}}_n^N \, U_{np}^N$$

Recognize an inverse DFT

$$\mathbf{\Gamma}_0^N(\boldsymbol{\tau}^N) = \mathrm{DFT}^{-1}\left(\hat{\mathbf{\Gamma}}_0^N : \hat{\boldsymbol{\tau}}^N \right)$$

Conclusion

Summary of Lecture 2

- Galerkin discretization of the LS equation
- The **consistent** discretized operators
- Using the FFT to apply the discrete Green operator
- Using matrix-free solvers

$$\hat{\boldsymbol{\Gamma}}_{0,n}^{N} = \sum_{m \in \mathbb{Z}^d} \left(W_{n+mN}^{N}\right)^2 \hat{\boldsymbol{\Gamma}}_0^{\infty}(\mathbf{k}_{n+mN})$$

Conclusion

- I must confess something…
- In lecture 3: asymptotically consistent discretizations (aka “variational crimes”)

Thank you for your attention!

sebastien.brisard@univ-eiffel.fr

https://navier-lab.fr/en/equipe/brisard-sebastien

https://cv.archives-ouvertes.fr/sbrisard

https://sbrisard.github.io