Computer Methods in Applied Mechanics and Engineering 433 (2025) 117488

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

A discrete sine–cosine based method for the elasticity of heterogeneous materials with arbitrary boundary conditions

Joseph Paux [a,b], Léo Morin [c,d,*], Lionel Gélébart [e], Abdoul Magid Amadou Sanoko [c,d]

[a] Institut de Thermique, Mécanique, Matériaux (ITheMM, EA 7548), Université de Reims Champagne-Ardenne, Campus Sup Ardenne, 08000 Charleville-Mézières, France
[b] Université Sorbonne Paris Nord, Laboratoire des Sciences des Procédés et des Matériaux, LSPM, CNRS, UPR 3407, F-93430, Villetaneuse, France
[c] Univ. Bordeaux, CNRS, Bordeaux INP, I2M, UMR 5295, F-33400 Talence, France
[d] Arts et Metiers Institute of Technology, CNRS, Bordeaux INP, I2M, UMR 5295, F-33400 Talence, France
[e] Université Paris-Saclay, CEA, SRMA, 91191, Gif/Yvette, France

ARTICLE INFO

Keywords:
FFT-based solvers
Dirichlet/Neumann boundary conditions
Discrete sine–cosine transforms
Elasticity

ABSTRACT

The aim of this article is to extend Moulinec and Suquet (1998)'s FFT-based method for heterogeneous elasticity to non-periodic Dirichlet/Neumann boundary conditions. The method is based on a decomposition of the displacement into a known term verifying the boundary conditions and a fluctuation term, with no contribution on the boundary, and described by appropriate sine–cosine series. A modified auxiliary problem involving a polarization tensor is solved within a Galerkin-based method, using an approximation space spanned by sine–cosine series. The elementary integrals emerging from the weak formulation of the equilibrium are approximated by discrete sine–cosine transforms, which makes the method relying on the numerical complexity of Fourier transforms. The method is finally assessed in several problems including kinematic uniform, static uniform and arbitrary Dirichlet/Neumann boundary conditions.

1. Introduction

This work is concerned with the development of efficient numerical methods for solving problems of micromechanics in heterogeneous materials. Fast Fourier transforms (FFT) based methods, introduced in the seminal paper of Moulinec and Suquet [1], are a class of solvers that are commonly used in micromechanics to compute the local and overall fields of heterogeneous (composite) materials and constitute an alternative to the finite element method (FEM) in problems governed by elliptic equations. The three main advantages of FFT-based solvers are that (i) the numerical complexity of the method is in $O(N \log N)$ (and is therefore lower than the complexity of matrix-assembled methods), (ii) it does not require meshing operations since it is based on voxelized images (that can be directly used from imagery techniques such as scanning electron microscopy (SEM) or tomography), and (iii) it is adapted to parallel implementation as most FFT packages are designed for distributed-memory parallel machines [2]. The method proposed by Moulinec and Suquet [1] concerns periodic microstructures and is based on an iterative solution to an integral Lippmann–Schwinger type equation, which can be solved using a fixed-point method (as done by Moulinec and Suquet [1] and referred as the so-called basic scheme) or accelerated iterative schemes [3–7], which are more suitable for highly-contrasted materials. In the context of periodic homogenization, the method has been successfully applied to a large class of (non-)linear

* Corresponding author at: Univ. Bordeaux, CNRS, Bordeaux INP, I2M, UMR 5295, F-33400 Talence, France.
E-mail address: leo.morin@u-bordeaux.fr (L. Morin).

https://doi.org/10.1016/j.cma.2024.117488
Received 3 July 2024; Received in revised form 19 October 2024; Accepted 19 October 2024
Available online 30 October 2024

problems, including, among others, anisotropic elasticity [1], J2-plasticity [1], crystal viscoplasticity [8], dislocation-mediated plasticity [9,10], electrical conduction [3], piezoelectricity [11], porous ductile solids [12,13] (see the papers of Schneider [14], Lucarini et al. [15] for a comprehensive review of FFT-based method applied to homogenization).

Periodic boundary conditions are intrinsic of the method as it is based on a description of the fields through Fourier series. As a consequence, this method is not adapted in several problems of interest in micromechanics such as (i) materials with non-periodic microstructures (such as fibrous networks and particular situations of polycrystalline materials) which requires to periodicize the microstructures obtained from imagery techniques, (ii) higher-order homogenization[1] with higher-order boundary conditions [18,19], and (iii) transient problems such as elastodynamic with wave propagation [20] for which periodic boundary conditions induce two traveling waves.

The extension of FFT-based solvers to non-periodic boundary conditions has been first proposed in the work of Gélébart [21], devoted to elasticity problems subjected to Dirichlet boundary conditions, through the introduction of a buffer zone surrounding the domain of interest. The complete cell, composed of the domain of interest plus the buffer zone, can be periodic so this technique is based on standard FFT packages with periodic boundary conditions applied on the exterior of the buffer zone. Despite its ability to apply correctly Dirichlet and also Neumann (see [22]) boundary conditions, this method requires (i) a larger unit cell than the microstructure studied because of the (elastic) buffer zone and (ii) additional internal iterations to find the eigendisplacement field at the boundary of the buffer zone. Mention has to be made to the works of Grimm-Strele and Kabel [23] and Monchiet and Bonnet [24] which are based on mirror unit cells and can be used for *uniform* Dirichlet and/or Neumann boundary conditions. This method, initially proposed by Wiegmann [25], takes advantage of the equivalence between particular symmetries of the unit cell and boundary conditions on a portion of the unit cell, and can be related to discrete sine–cosine transforms.

Recently, an alternative method has been proposed for imposing non-periodic boundary conditions using explicitly discrete sine–cosine transforms for uniform [26] or arbitrary boundary conditions [27–29], *for problems of heterogeneous conductivity*. The idea of this approach is to split the solution field (e.g. the temperature field in thermal conductivity problems) of the elliptic problem into a known term verifying the Dirichlet/Neumann boundary conditions and an unknown fluctuation term described by an appropriate trigonometric series with no contribution on the boundary. The work by Paux et al. [29] concerns the conductivity of heterogeneous materials subjected to mixed Dirichlet/Neumann boundary conditions[2] and is based on a Galerkin discretization of the auxiliary problem, using sine–cosine series as trial functions. The work of Gélébart [27], on the other hand, concerns the conductivity of heterogeneous materials subjected to mixed Dirichlet/Neumann (and also periodic) boundary conditions and is based on a finite difference scheme. Both approaches (Galerkin and finite differences based) rely on the use of discrete sine and cosine transforms which can be computed using FFT packages and have thus a computational complexity in $O(N\ \log N)$. It has been shown that both methods lead to very similar results in terms of local and overall fields but they differ in terms of convergence; the convergence rate of the Galerkin-based schemes scales linearly with the contrast, while the number of iterations to convergence is bounded or slowly increases with the contrast using a discrete scheme [29] (see also [30]). Mention has to be made to the recent work of Risthaus and Schneider [31], who extends the approaches for conductivity [26,28] to elasticity problems with kinematic uniform boundary conditions using sine series (as a particular case of Dirichlet boundary conditions). It must be noted that discrete sine–cosine transforms have also been used in the literature to solve efficiently the Poisson equation (discretized by a finite difference scheme) instead of using matrix-inversion (see e.g. [32–34]).

The aim of this work is to develop a numerical method based on discrete sine–cosine transforms for heterogeneous *elasticity* problems subjected to general non-periodic mixed Dirichlet/Neumann boundary conditions (including, as subcases, kinematic uniform boundary conditions (KUBC) and static uniform boundary conditions (SUBC)). Thus this work extends the approach of Morin and Paux [28], Gélébart [27], Paux et al. [29], restricted to conductivity problems, to elasticity. The main difference between elasticity and conductivity equations is that elasticity involves higher-order tensors which will induce extra difficulties in the resolution of the auxiliary problem, mainly due to the presence of cross derivatives in the equilibrium equations (see e.g. [25,26]). The present method thus constitutes a direct extension of Moulinec and Suquet [1]'s method to general non-periodic boundary conditions.

The paper is organized as follows. In Section 2, important results related to sine–cosine series and their associated transforms are recalled. The numerical method for elasticity problems subjected to Dirichlet/Neumann boundary conditions is then presented in Section 3. Finally, the numerical scheme is applied in Section 4 to several problems of heterogeneous elasticity including homogenization under kinematic uniform or static uniform boundary conditions.

2. Sine–cosine series and associated discrete transforms

The resolution of the heterogeneous elasticity problem with arbitrary Dirichlet/Neumann boundary conditions will be based, as in conductivity problems [27–29], on a decomposition of the solution field in a known (given) field verifying the boundary conditions and a fluctuation field having no contribution on the boundary conditions. The approach followed in this work is based on a Galerkin method, which requires the definition of a continuous trial function for the unknown field, described by half-range or quarter-range sine–cosine series depending on the boundary conditions. Moreover, as in the periodic case [1,35], the elementary

[1] Higher-order homogenization can be handled in the periodic setting in several cases, as done by Gélébart [16] by adding strain gradient components for plates of beams or by Tran et al. [17] using the asymptotic expansion method.

[2] The work of Paux et al. [29] extends Morin and Paux [28]'s approach, which was restricted to Dirichlet boundary conditions, to mixed Dirichlet/Neumann boundary conditions.

Table 1
Functions and coefficients defining the four mixed Dirichlet/Neumann boundary conditions on the faces $x = 0$ and $x = L_x$.

	Boundary conditions	g and $\overline{g}$	k_i	α_i	ξ_i
DD	$f(x,y,z) = 0 \quad \forall x = 0$ $f(x,y,z) = 0 \quad \forall x = L_x$	$g(x) = \sin(x)$ $\overline{g}(x) = \cos(x)$	$k_i = \frac{i}{L_x}\pi$	$\alpha_i = 1 \quad \forall i \geq 0$	$\xi_i = k_i$
NN	$\frac{\partial f}{\partial x}(x,y,z) = 0 \quad \forall x = 0$ $\frac{\partial f}{\partial x}(x,y,z) = 0 \quad \forall x = L_x$	$g(x) = \cos(x)$ $\overline{g}(x) = \sin(x)$	$k_i = \frac{i}{L_x}\pi$	$\alpha_0 = 1/2$ $\alpha_i = 1 \quad \forall i \geq 1$	$\xi_i = -k_i$
DN	$f(x,y,z) = 0 \quad \forall x = 0$ $\frac{\partial f}{\partial x}(x,y,z) = 0 \quad \forall x = L_x$	$g(x) = \sin(x)$ $\overline{g}(x) = \cos(x)$	$k_i = \frac{2i+1}{2L_x}\pi$	$\alpha_i = 1 \quad \forall i \geq 0$	$\xi_i = k_i$
ND	$\frac{\partial f}{\partial x}(x,y,z) = 0 \quad \forall x = 0$ $f(x,y,z) = 0 \quad \forall x = L_x$	$g(x) = \cos(x)$ $\overline{g}(x) = \sin(x)$	$k_i = \frac{2i+1}{2L_x}\pi$	$\alpha_i = 1 \quad \forall i \geq 0$	$\xi_i = -k_i$

integrals arising in the weak formulation will be approximated using discrete sine–cosine transforms, which are intimately related to the sine–cosine series [36,37]. It must be noted that discrete sine transforms (DST) and discrete cosine transforms (DCT) are related to standard discrete Fourier transforms (DFT) through symmetry extensions; a given non-periodic signal can be periodized by symmetry extension (that can be periodic or anti-periodic, and located an ending point or a half-element after) and the DFT of this periodized signal corresponds to a DST or DCT depending on the symmetry extension (see [27]). We thus recall, following the presentation of Paux et al. [29], important results related to sine and cosine series and their associated discrete transforms in a generic framework.

2.1. 1-d case

2.1.1. Sine–cosine series in 1-d

We consider a function f defined in the interval $[0, L_x]$ and verifying either null Dirichlet or null Neumann boundary conditions. Such function can be described by half-range or quarter-range sine–cosine series and written under the generic form

$$f(x) = \sum_{i=0}^{+\infty} \alpha_i F_i g(k_i x), \quad \forall x \in [0, L_x], \tag{1}$$

with k_i some "frequency" parameter, g a cosine or a sine function (depending on the type of boundary conditions) and F_i, $i \geq 0$, the sine–cosine series coefficients given by

$$F_i = \frac{2}{L_x} \int_0^{L_x} f(x) g(k_i x) \mathrm{d}x, \quad \forall i \geq 0. \tag{2}$$

The associated *partial* series of order N (for every positive integer N) is then defined as

$$f_N(x) = \sum_{i=0}^{N} F_i g(k_i x), \quad \forall x \in [0, L_x]. \tag{3}$$

A summary of all functions and coefficients is given in Table 1 to cover all types of boundary conditions on $x = 0$ and $x = L_x$ (Dirichlet–Dirichlet (DD), Neumann–Neumann (NN), Dirichlet–Neumann (DN) and Neumann–Dirichlet (ND)).

Derivative rules associated with (1) can be written as

$$\frac{\mathrm{d}f}{\mathrm{d}x}(x) = \sum_{i=0}^{+\infty} \xi_i \alpha_i F_i \overline{g}(k_i x), \quad \forall x \in [0, L_x], \tag{4}$$

$$\frac{\mathrm{d}^2 f}{\mathrm{d}x^2}(x) = \sum_{i=0}^{+\infty} -(\xi_i)^2 \alpha_i F_i g(k_i x), \quad \forall x \in [0, L_x]. \tag{5}$$

In these expressions, $\overline{g}$ is the dual function of g (i.e. a sine function if the initial function is a cosine function and vice versa) and ξ_i is the coefficient arising from the derivation (see Table 1).

2.2. Discrete sine–cosine transforms in 1-d

A function that is described by a sine–cosine series given by Eq. (1) relies on the computation of the associated sine–cosine coefficients given in (2). For an arbitrary function f, there is no analytical formula for calculating the integral in Eq. (2). The discrete sine transform (DST) and the discrete cosine transform (DCT) provide the numerical integration[3] of the coefficients defined by Eq. (2), depending on the type of boundary conditions. An efficient computation of the coefficients of the DFT, DST and DCT is classically done using fast Fourier transforms (FFT) [38].

[3] This numerical integration is similar to the standard Discrete Fourier Transform (DFT) used for the computation of the coefficients of the Fourier series for periodic functions.

Table 2
Computational details of the Discrete sine–cosine Transforms.

BC	Transform	$\widehat{F}_i \ (i = 0, \dots, N+1)$	$f_a \ (a = 0, \dots, N+1)$
DD	DST-I $\widehat{\boldsymbol{F}} = \mathcal{D}_{sI}(\boldsymbol{f})$ $\boldsymbol{f} = \mathcal{D}_{sI}^{-1}(\widehat{\boldsymbol{F}})$	$\widehat{F}_i = \sum_{a=0}^{N+1} \beta_a f_a \ \sin\left(\frac{\pi a i}{N+1}\right)$ $\beta_0 = \beta_{N+1} = 1/2$ $\beta_a = 1$ for $a = 1, \dots, N$	$f_a = \frac{2}{N+1} \sum_{i=0}^{N+1} \gamma_i \widehat{F}_i \ \sin\left(\frac{\pi a i}{N+1}\right)$ $\gamma_0 = \gamma_{N+1} = 1/2$ $\gamma_i = 1$ for $i = 1, \dots, N$
NN	DCT-I $\widehat{\boldsymbol{F}} = \mathcal{D}_{cI}(\boldsymbol{f})$ $\boldsymbol{f} = \mathcal{D}_{cI}^{-1}(\widehat{\boldsymbol{F}})$	$\widehat{F}_i = \sum_{a=0}^{N+1} \beta_a f_a \ \cos\left(\frac{\pi a i}{N+1}\right)$ $\beta_0 = \beta_{N+1} = 1/2$ $\beta_a = 1$ for $a = 1, \dots, N$	$f_a = \frac{2}{N+1} \sum_{i=0}^{N+1} \gamma_i \widehat{F}_i \ \cos\left(\frac{\pi a i}{N+1}\right)$ $\gamma_0 = \gamma_{N+1} = 1/2$ $\gamma_i = 1$ for $i = 1, \dots, N$
DN	DST-III $\widehat{\boldsymbol{F}} = \mathcal{D}_{sIII}(\boldsymbol{f})$ $\boldsymbol{f} = \mathcal{D}_{sIII}^{-1}(\widehat{\boldsymbol{F}})$	$\widehat{F}_i = \sum_{a=0}^{N+1} \beta_a f_a \ \sin\left(\frac{\pi a \left(i+\frac{1}{2}\right)}{N+1}\right)$ $\beta_0 = \beta_{N+1} = 1/2$ $\beta_a = 1$ for $a = 1, \dots, N$	$f_a = \frac{2}{N+1} \sum_{i=0}^{N+1} \gamma_i \widehat{F}_i \ \sin\left(\frac{\pi a (i+\frac{1}{2})}{N+1}\right)$ $\gamma_i = 1$ for $i = 0, \dots, N$ $\gamma_{N+1} = 0$
ND	DCT-III $\widehat{\boldsymbol{F}} = \mathcal{D}_{cIII}(\boldsymbol{f})$ $\boldsymbol{f} = \mathcal{D}_{cIII}^{-1}(\widehat{\boldsymbol{F}})$	$\widehat{F}_i = \sum_{a=0}^{N+1} \beta_a f_a \ \cos\left(\frac{\pi a \left(i+\frac{1}{2}\right)}{N+1}\right)$ $\beta_0 = \beta_{N+1} = 1/2$ $\beta_a = 1$ for $a = 1, \dots, N$	$f_a = \frac{2}{N+1} \sum_{i=0}^{N+1} \gamma_i \widehat{F}_i \ \cos\left(\frac{\pi a (i+\frac{1}{2})}{N+1}\right)$ $\gamma_i = 1$ for $i = 0, \dots, N$ $\gamma_{N+1} = 0$

We consider the 1-d domain of size $[0,\ L_x]$, discretized uniformly with N_x+2 points. The spatial scale associated with the uniform grid is thus $\Delta x = L_x/(N_x+1)$ we denote by $x_a = a\Delta x$ (for $a = 0, \dots, N+1$) the grid points. The grid points values of function f are denoted by $f_a = f(x_a)$ (for $a = 0, \dots, N+1$). The discrete transform associated to each boundary conditions[4] can be written under the generic form as follows

$$\widehat{\boldsymbol{F}} = \mathcal{D}_{\mathrm{x}}(\boldsymbol{f}), \tag{6}$$

where $\boldsymbol{f}$ is the array of size $N+2$ containing the values of function f at the grid points x_a and $\widehat{\boldsymbol{F}}$ is the array of size $N+2$ containing the values of the associated discrete transform coefficients $\widehat{F}_i$. The notation $\mathcal{D}_{\mathrm{x}}(\cdot)$ corresponds to the discrete transform operator (with $\mathrm{x} = sI$ for the DST-I, $\mathrm{x} = cI$ for the DCT-I, $\mathrm{x} = sIII$ for the DST-III and $\mathrm{x} = cIII$ for the DCT-III, depending on the boundary conditions, see footnote 3). The inverse discrete transform can be formally written as

$$\boldsymbol{f} = \mathcal{D}_{\mathrm{x}}^{-1}(\widehat{\boldsymbol{F}}), \tag{7}$$

where the notation $\mathcal{D}_{\mathrm{x}}^{-1}(\cdot)$ corresponds to the associated inverse discrete transform operator. In addition, the inverse transforms of type I (sine and cosine) discrete transforms are type I (sine and cosine) discrete transforms multiplied by a factor $2/(N+1)$, while the inverse transforms of type III (sine and cosine) discrete transforms are type II (sine and cosine) discrete transforms multiplied by a factor $2/(N+1)$.

The expressions of all discrete transforms is given in Table 2. Interestingly, one can notice that the coefficients $\widehat{F}_i$ provide a numerical approximation of the coefficients F_i (of the associated cosine or sine series) through

$$\widehat{F}_i \simeq \frac{N+1}{2} F_i, \quad \forall i \geq 0. \tag{8}$$

Finally, it is interesting to note that the four discrete transforms can be put under the following generic form

$$\widehat{F}_i = \sum_{a=0}^{N+1} \beta_a f_a \ g(k_i x_a), \quad \forall i \geq 0, \tag{9}$$

where β_a, k_i, α_i and $g(k_i x)$ depend on the boundary conditions. In addition, the inverse transforms can be written as

$$f_a = \sum_{i=0}^{N+1} \gamma_i \widehat{F}_i \ g(k_a x_i), \quad \forall a \geq 0. \tag{10}$$

2.3. The 3-d case

2.3.1. 3-d sine–cosine series

We consider a function f defined in the prismatic domain $\Omega = [0, L_x] \times [0, L_y] \times [0, L_z]$ and verifying null Dirichlet boundary condition on faces $\partial\Omega_{\mathrm{D}}$

$$f(x, y, z) = 0, \quad \forall (x, y, z) \in \partial\Omega_{\mathrm{D}}, \tag{11}$$

[4] DST-I encodes Dirichlet–Dirichlet, DCT-I encodes Neumann–Neumann, DST-III encodes Dirichlet–Neumann and DCT-III encodes Neumann–Dirichlet.

and null Neumann boundary condition on faces $\partial\Omega_{\mathrm{N}}$

$$\boldsymbol{\nabla} f(x,y,z)\cdot\mathbf{n}=0,\quad \forall (x,y,z)\in\partial\Omega_N, \tag{12}$$

where $\mathbf{n}$ is the outward normal to the boundary $\partial\Omega_{\mathrm{N}}$ (with $\partial\Omega_{\mathrm{N}}\cup\partial\Omega_{\mathrm{D}}=\partial\Omega$). Function f can be described by a generalized 3-d sine–cosine series as

$$f(x,y,z)=\sum_{i=0}^{+\infty}\sum_{j=0}^{+\infty}\sum_{k=0}^{+\infty}\alpha_i^x\alpha_j^y\alpha_k^z F_{ijk}g^x(k_i^x x)g^y(k_j^y y)g^z(k_k^z z),\quad \forall (x,y,z)\in\Omega, \tag{13}$$

where the functions g^x, g^y, g^z, and the coefficients α_i^x, α_j^y, α_k^z, k_i^x, k_j^y, k_k^z depend on the type of boundary conditions on each couple of opposite faces[5] (see Table 1), and F_{ijk} are the sine–cosine series coefficients, given by

$$F_{ijk}=\frac{8}{L_xL_yL_z}\int_\Omega f(x,y,z)g^x(k_i^x x)g^y(k_j^y y)g^z(k_k^z z)\,\mathrm{d}x\mathrm{d}y\mathrm{d}z,\quad \forall i,j,k\geq 0. \tag{14}$$

The partial derivatives of f read

$$\frac{\partial f}{\partial x}(x,y,z)=\sum_{i=0}^{+\infty}\sum_{j=0}^{+\infty}\sum_{k=0}^{+\infty}\xi_i^x\alpha_i^x\alpha_j^y\alpha_k^z F_{ijk}\overline{g^x}(k_i^x x)g^y(k_j^y y)g^z(k_k^z z),\quad \forall (x,y,z)\in\Omega, \tag{15}$$

$$\frac{\partial^2 f}{\partial x^2}(x,y,z)=\sum_{i=0}^{+\infty}\sum_{j=0}^{+\infty}\sum_{k=0}^{+\infty}-(\xi_i^x)^2\alpha_i^x\alpha_j^y\alpha_k^z F_{ijk}g^x(k_i^x x)g^y(k_j^y y)g^z(k_k^z z),\quad \forall (x,y,z)\in\Omega, \tag{16}$$

where $\overline{g^x}$ is the dual function of g^x (as defined in Table 1).

2.3.2. Discrete sine–cosine transforms in 3-d

The 3-d prismatic domain Ω is discretized uniformly with $(N_x+2)\times(N_y+2)\times(N_z+2)$ points so that the spatial scales are $\Delta x=L_x/(N_x+1)$, $\Delta y=L_y/(N_y+1)$ and $\Delta z=L_z/(N_z+1)$. The coordinates of the grid points are denoted by $x_a=a\Delta x$ (for $a=0,\dots,N_x+1$), $y_b=b\Delta y$ (for $b=0,\dots,N_y+1$) and $z_c=c\Delta z$ (for $c=0,\dots,N_z+1$). The values of the function f at the grid points are denoted by $f_{abc}=f(x_a,y_b,z_c)$ (for $a=0,\dots,N_x+1$, $b=0,\dots,N_y+1$ and $c=0,\dots,N_z+1$).

Discrete sine–cosine transforms in 3-d are simply a composition of 1-d transforms in each spatial directions. The coefficients of the discrete sine–cosine transform ($\widehat{F}_{ijk}$) of function f are then given by (following the conventions of Table 2)

$$\widehat{F}_{ijk}=\sum_{a=0}^{N_x+1}\sum_{b=0}^{N_y+1}\sum_{c=0}^{N_z+1}\beta_a^x\beta_b^y\beta_c^z f_{abc}\ g^x(k_i^x x_a)g^y(k_j^x y_b)g^z(k_k^z z_c), \tag{17}$$

where the functions g^x, g^y, g^z, and the coefficients k_i^x, k_j^y, k_k^z, β_i^x, β_j^y and β_k^z depend on the type of boundary conditions. Let us denote by $\widehat{\boldsymbol{F}}$ the array of size $(N_x+2)\times(N_y+2)\times(N_z+2)$ containing the discrete sine–cosine coefficients and by $\boldsymbol{f}$ the array of size $(N_x+2)\times(N_y+2)\times(N_z+2)$ containing the values of function f at the grid points. The discrete transform (17) is then written as

$$\widehat{\boldsymbol{F}}=\mathcal{D}_{\mathrm{xyz}}(\boldsymbol{f}), \tag{18}$$

where $\mathrm{x}\in\{sI,\ cI,\ sIII,\ cIII\}$, $\mathrm{y}\in\{sI,\ cI,\ sIII,\ cIII\}$ and $\mathrm{z}\in\{sI,\ cI,\ sIII,\ cIII\}$, depending on the boundary conditions applied on the $x-$faces, $y-$faces and $z-$faces, respectively.

The associated inverse transform reads

$$f_{abc}=\frac{8}{(N_x+1)(N_y+1)(N_z+1)}\sum_{i=0}^{N_x+1}\sum_{j=0}^{N_y+1}\sum_{k=0}^{N_z+1}\gamma_i^x\gamma_j^y\gamma_k^z\widehat{F}_{ijk}\ g^x(k_a^x x_i)g^y(k_j^x y_b)g^z(k_k^z z_c), \tag{19}$$

and it can be written as

$$\boldsymbol{f}=\mathcal{D}_{\mathrm{xyz}}^{-1}(\widehat{\boldsymbol{F}}). \tag{20}$$

A useful notation is finally introduced to account for dual mixed sine–cosine transforms. Assuming that the transform $\mathcal{D}_{\mathrm{xyz}}(\cdot)$ corresponds to Eq. (17), then we denote by

$$\widehat{\boldsymbol{F}}=\mathcal{D}_{\bar{\mathrm{x}}\mathrm{yz}}(\boldsymbol{f}), \tag{21}$$

the transform defined by

$$\widehat{F}_{ijk}=\sum_{a=0}^{N_x+1}\sum_{b=0}^{N_y+1}\sum_{c=0}^{N_z+1}\beta_a^x\beta_b^y\beta_c^z f_{abc}\ \overline{g^x}(k_i^x x_a)g^y(k_j^x y_b)g^z(k_k^z z_c). \tag{22}$$

The notation $\mathcal{D}_{\bar{\mathrm{x}}\mathrm{yz}}$ defines the dual transform of $\mathcal{D}_{\bar{x}\mathrm{yz}}$ in the x-direction. For example, if $\mathcal{D}_{\mathrm{xyz}}$ corresponds to the transform $\mathcal{D}_{sIsIsI}$, then $\mathcal{D}_{\bar{\mathrm{x}}\mathrm{yz}}$ corresponds to $\mathcal{D}_{cIsIsI}$.

[5] There is a total of $2^6=64$ different sine–cosine series in the 3-d case (corresponding to the 2 types of boundary conditions on the 6 faces).

3. A fast numerical method for elasticity problems in heterogeneous media subjected to non-periodic boundary conditions

3.1. Equations of elasticity

We consider a problem of linear elasticity in a heterogeneous finite medium. The finite cell is a prismatic domain denoted by $\Omega = [0,\ L_x]\times[0,\ L_y]\times[0,\ L_z]$ in 3-d and a rectangle denoted by $\Omega = [0,\ L_x]\times[0,\ L_y]$ in 2-d. Tensorial components refer to a system of Cartesian coordinates $(\mathbf{e}_x;\mathbf{e}_y;\mathbf{e}_z)$ in 3-d (and $(\mathbf{e}_x;\mathbf{e}_y)$ in 2-d).

The problem of elasticity consists in the computation of the elastic strain $\boldsymbol{\varepsilon}(\mathbf{u}(\mathbf{x}))$, stress $\boldsymbol{\sigma}(\mathbf{x})$ and displacement field $\mathbf{u}(\mathbf{x})$, at each point $\mathbf{x}$ in Ω, for a given elasticity tensor field $\mathbb{C}(\mathbf{x})$. Since the principle of the method is to prescribe the value of the solution field or its normal derivative [29], we first consider mixed Dirichlet/Neumann boundary conditions *on the components of the displacement field* $\mathbf{u}(\mathbf{x})$, i.e.

$$\begin{cases} \forall \mathbf{x}\in\partial\Omega_{\mathrm{D}x}, & u_x(\mathbf{x}) = u_{\partial\Omega_{\mathrm{D}x}}(\mathbf{x}) & \text{(Dirichlet on } u_x\text{)}\\ \forall \mathbf{x}\in\partial\Omega_{\mathrm{N}x}, & \boldsymbol{\nabla} u_x(\mathbf{x})\cdot\mathbf{n} = \mathbf{f}_{\partial\Omega_{\mathrm{N}x}}(\mathbf{x}) & \text{(Neumann on } u_x\text{)}\\ \forall \mathbf{x}\in\partial\Omega_{\mathrm{D}y}, & u_y(\mathbf{x}) = u_{\partial\Omega_{\mathrm{D}y}}(\mathbf{x}) & \text{(Dirichlet on } u_y\text{)}\\ \forall \mathbf{x}\in\partial\Omega_{\mathrm{N}y}, & \boldsymbol{\nabla} u_y(\mathbf{x})\cdot\mathbf{n} = \mathbf{f}_{\partial\Omega_{\mathrm{N}y}}(\mathbf{x}) & \text{(Neumann on } u_y\text{)}\\ \forall \mathbf{x}\in\partial\Omega_{\mathrm{D}z}, & u_z(\mathbf{x}) = u_{\partial\Omega_{\mathrm{D}z}}(\mathbf{x}) & \text{(Dirichlet on } u_z\text{)}\\ \forall \mathbf{x}\in\partial\Omega_{\mathrm{N}z}, & \boldsymbol{\nabla} u_z(\mathbf{x})\cdot\mathbf{n} = \mathbf{f}_{\partial\Omega_{\mathrm{N}z}}(\mathbf{x}) & \text{(Neumann on } u_z\text{)}, \end{cases} \tag{23}$$

where $u_{\partial\Omega_{\mathrm{D}x}}$, $u_{\partial\Omega_{\mathrm{D}y}}$ and $u_{\partial\Omega_{\mathrm{D}z}}$ are the prescribed components of the displacement field (corresponding to the Dirichlet condition), and $\mathbf{f}_{\partial\Omega_{\mathrm{N}x}}$, $\mathbf{f}_{\partial\Omega_{\mathrm{N}y}}$ and $\mathbf{f}_{\partial\Omega_{\mathrm{N}z}}$ are the prescribed values for the normal derivative of components of the displacement field (corresponding to the Neumann condition *on the displacement field*). The boundaries associated to the field u_x ($\partial\Omega_{\mathrm{D}x}$ and $\partial\Omega_{\mathrm{N}x}$) are necessary the union of entire faces with of course $\partial\Omega_{\mathrm{D}x}\cup\partial\Omega_{\mathrm{N}x} = \partial\Omega$ (the same properties also hold for the boundaries associated to the fields u_y and u_z).

It must be noted that standard Neumann boundary conditions in elasticity involve *non-normal derivatives* of the displacement components, as they are applied through the normal stress vector, i.e.

$$\boldsymbol{\sigma}(\mathbf{x})\cdot\mathbf{n} = \mathbb{C} : \boldsymbol{\nabla}\mathbf{u}(\mathbf{x})\cdot\mathbf{n} = \mathbf{F}_{\partial\Omega_{\mathrm{N}}}, \quad \forall \mathbf{x}\in\partial\Omega_{\mathrm{N}}, \tag{24}$$

where $\mathbf{F}_{\partial\Omega_{\mathrm{N}}}$ is the prescribed surface force and we used $\mathbb{C} : \boldsymbol{\varepsilon} = \mathbb{C} : \boldsymbol{\nabla}\mathbf{u}$, due to the minor symmetry of the stiffness tensor $\mathbb{C}$. Therefore, they cannot be treated by the present method *as is*. To overcome this issue, a thin external layer, with appropriate elastic properties that remove non-normal derivative of the displacement field in the expression of the stress-vector, is added to the domain. Expanding (24) on the boundary $x = L_x$ (without loss of generality) leads to

$$\begin{cases} C_{11kl}(L_x,y,z)\dfrac{\partial u_k}{\partial x_l}(L_x,y,z) = F_x(y,z)\\ C_{12kl}(L_x,y,z)\dfrac{\partial u_k}{\partial x_l}(L_x,y,z) = F_y(y,z)\\ C_{13kl}(L_x,y,z)\dfrac{\partial u_k}{\partial x_l}(L_x,y,z) = F_z(y,z). \end{cases} \tag{25}$$

The non-normal derivatives are then artificially removed by considering a modified elasticity law in the external layer, leading to

$$\begin{cases} C_{11k1}(L_x,y,z)\dfrac{\partial u_k}{\partial x}(L_x,y,z) = F_x(y,z)\\ C_{12k1}(L_x,y,z)\dfrac{\partial u_k}{\partial x}(L_x,y,z) = F_y(y,z)\\ C_{13k1}(L_x,y,z)\dfrac{\partial u_k}{\partial x}(L_x,y,z) = F_z(y,z). \end{cases} \tag{26}$$

This modification is made by considering a modified elasticity behavior in the extra-layer, denoted by $\mathbb{C}^{\text{layer}}$ and defined by the following tensor

$$C^{\text{layer}}_{ijkl} = \begin{cases} C_{ijkl} & \text{if } i = l\\ 0 & \text{if } i \neq l. \end{cases} \tag{27}$$

One might remark that, contrary to the classical elasticity tensor, this modified tensor does not verify the minor and major symmetries. Still, it is a positive definite tensor, which ensures that the problem is well-posed under Neumann boundary conditions. For an infinitely thin extra layer, the modified problem is equivalent to the classical elasticity problem under Neumann boundary conditions. In practice, a one voxel layer will be used.

The local problem to be solved is

$$\begin{cases} \forall \mathbf{x}\in\Omega, & \mathbf{div}\ \boldsymbol{\sigma}(\mathbf{x}) = \mathbf{0}\\ \forall \mathbf{x}\in\Omega, & \boldsymbol{\sigma}(\mathbf{x}) = \mathbb{C}(\mathbf{x}) : \boldsymbol{\varepsilon}(\mathbf{u}(\mathbf{x}))\\ \forall \mathbf{x}\in\Omega, & \boldsymbol{\varepsilon}(\mathbf{u}(\mathbf{x})) = \dfrac{1}{2}\left(\boldsymbol{\nabla}\mathbf{u} + \boldsymbol{\nabla}^T\mathbf{u}\right), \end{cases} \tag{28}$$

where the boundary conditions considered are given by (23). In Eq. (28), $\mathbb{C}(\mathbf{x})$ is the fourth-order stiffness tensor. In the particular case of isotropic elasticity, it reads (in the case of a 3-d medium)

$$\mathbb{C}(\mathbf{x}) = 3\kappa(\mathbf{x})\mathbb{J} + 2\mu(\mathbf{x})\mathbb{K}, \tag{29}$$

with $\kappa(\mathbf{x})$ and $\mu(\mathbf{x})$ are respectively the local bulk and shear moduli. $\mathbb{J}$ and $\mathbb{K}$ are linearly independent isotropic tensors defined by

$$\mathbb{J} = \frac{1}{3}\mathbf{I}_3 \otimes \mathbf{I}_3, \quad \mathbb{K} = \mathbb{I} - \mathbb{J}, \tag{30}$$

with $\mathbb{I}$ the fourth order identity tensor and $\mathbf{I}_3$ is the second-order identity tensor.

The principle of resolution (see [29] for conductivity problems) is to

1. Write the local problem (28) as an auxiliary problem involving a reference homogeneous material and a polarization term (following the idea of Moulinec and Suquet [1] in the periodic setting);
2. Split each component of the displacement field $\mathbf{u}$ in two contributions, a known term $\mathbf{u}^0$ verifying the boundary conditions (23) (by construction) and an unknown fluctuation term $\widetilde{\mathbf{u}}$, defined such that it is null on the Dirichlet boundaries and it has normal derivatives null on the Neumann boundaries, to be determined;
3. Write the weak (Galerkin) formulation by (i) describing the fluctuation term by a partial 3−d mixed sine–cosine series (depending on the type of boundary conditions) and (ii) using trial functions based on the same 3−d mixed sine–cosine series;
4. Solve approximately the linear problem arising from the weak formulation (corresponding to the auxiliary problem for a given polarization tensor) using discrete sine and cosine transforms (to calculate, and in some cases approximate, the integrals defined in the weak formulation);
5. Find the polarization term solution of the problem using an iterative scheme (as done by Moulinec and Suquet [1] in the periodic setting with a fixed-point method).

3.2. The classical auxiliary problem and the associated difficulties

Following [1], the local elasticity problem (28), subjected to the boundary conditions (23), is written as the following auxiliary problem

$$\begin{cases} \forall \mathbf{x} \in \Omega, & \mathbf{div}\ \boldsymbol{\sigma}(\mathbf{x}) = \mathbf{0} \\ \forall \mathbf{x} \in \Omega, & \boldsymbol{\sigma}(\mathbf{x}) = \mathbb{C}_0 : \boldsymbol{\varepsilon}(\mathbf{u}(\mathbf{x})) + \boldsymbol{\tau}(\mathbf{x}) \\ \forall \mathbf{x} \in \Omega, & \boldsymbol{\varepsilon}(\mathbf{u}(\mathbf{x})) = \frac{1}{2}\left(\boldsymbol{\nabla}\mathbf{u}(\mathbf{x}) + \boldsymbol{\nabla}^T\mathbf{u}(\mathbf{x})\right), \end{cases} \tag{31}$$

where $\mathbb{C}^0$ is the stiffness tensor of a homogeneous reference material and

$$\boldsymbol{\tau}(\mathbf{x}) = \left(\mathbb{C}(\mathbf{x}) - \mathbb{C}^0\right) : \boldsymbol{\varepsilon}(\mathbf{u}(\mathbf{x})) \tag{32}$$

is the polarization tensor. At this stage, the homogeneous reference material is classically taken isotropic (with Lamé's coefficients λ_0 and μ_0) as in most works related to FFT (see e.g. [1]):

$$C^0_{ijkl} = \mu_0\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{kj}\right) + \lambda_0\delta_{ij}\delta_{kl}, \quad \forall i,j,k,l = 1,2,3, \tag{33}$$

where δ is the Kronecker symbol. Despite the fact that $\mathbb{C}^0$ is isotropic one should remark that the (possible) anisotropy of the phases is still contained in the expression (32) of the polarization tensor.

Assuming (momentarily) that the polarization field $\boldsymbol{\tau}$ is known, the auxiliary problem defines a system of three equations on the unknown field $\mathbf{u}$:

$$\begin{cases} \left(\lambda_0+2\mu_0\right)\dfrac{\partial^2 u_x}{\partial x^2} + \mu_0\left(\dfrac{\partial^2 u_x}{\partial y^2} + \dfrac{\partial^2 u_x}{\partial z^2}\right) + \left(\lambda_0+\mu_0\right)\left(\dfrac{\partial^2 u_y}{\partial x\partial y} + \dfrac{\partial^2 u_z}{\partial x\partial z}\right) + \dfrac{\partial \tau_{xx}}{\partial x} + \dfrac{\partial \tau_{xy}}{\partial y} + \dfrac{\partial \tau_{xz}}{\partial z} = 0 \\ \left(\lambda_0+2\mu_0\right)\dfrac{\partial^2 u_y}{\partial y^2} + \mu_0\left(\dfrac{\partial^2 u_y}{\partial x^2} + \dfrac{\partial^2 u_y}{\partial z^2}\right) + \left(\lambda_0+\mu_0\right)\left(\dfrac{\partial^2 u_x}{\partial y\partial x} + \dfrac{\partial^2 u_z}{\partial y\partial z}\right) + \dfrac{\partial \tau_{xy}}{\partial x} + \dfrac{\partial \tau_{yy}}{\partial y} + \dfrac{\partial \tau_{yz}}{\partial z} = 0 \\ \left(\lambda_0+2\mu_0\right)\dfrac{\partial^2 u_z}{\partial z^2} + \mu_0\left(\dfrac{\partial^2 u_z}{\partial x^2} + \dfrac{\partial^2 u_z}{\partial y^2}\right) + \left(\lambda_0+\mu_0\right)\left(\dfrac{\partial^2 u_x}{\partial z\partial x} + \dfrac{\partial^2 u_y}{\partial z\partial y}\right) + \dfrac{\partial \tau_{xz}}{\partial x} + \dfrac{\partial \tau_{yz}}{\partial y} + \dfrac{\partial \tau_{zz}}{\partial z} = 0. \end{cases} \tag{34}$$

The boundary conditions are taken into account by considering the displacement field under the form

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_0(\mathbf{x}) + \widetilde{\mathbf{u}}(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega, \tag{35}$$

with $\mathbf{u}_0$ a *known* field verifying the boundary conditions (see [29] for examples of reconstruction of the function, depending on the type of boundary conditions) and $\widetilde{\mathbf{u}}$ a fluctuation vector field which does not interfere with the boundary conditions. Each component of $\widetilde{\mathbf{u}}$ is thus described by an appropriate sine–cosine series as explained in Section 2:

$$\begin{cases} \widetilde{u}_x(x,y,z) &= \displaystyle\sum_{i=0}^{N_x+1}\sum_{j=0}^{N_y+1}\sum_{k=0}^{N_z+1} \alpha_i^{xx}\alpha_j^{xy}\alpha_k^{xz}U^x_{ijk}\, g^{xx}(k_i^{xx}x)g^{xy}(k_j^{xy}y)g^{xz}(k_k^{xz}z), \quad \forall (x,y,z) \in \Omega, \\ \widetilde{u}_y(x,y,z) &= \displaystyle\sum_{i=0}^{N_x+1}\sum_{j=0}^{N_y+1}\sum_{k=0}^{N_z+1} \alpha_i^{yx}\alpha_j^{yy}\alpha_k^{yz}U^y_{ijk}\, g^{yx}(k_i^{yx}x)g^{yy}(k_j^{yy}y)g^{yz}(k_k^{yz}z), \quad \forall (x,y,z) \in \Omega, \\ \widetilde{u}_z(x,y,z) &= \displaystyle\sum_{i=0}^{N_x+1}\sum_{j=0}^{N_y+1}\sum_{k=0}^{N_z+1} \alpha_i^{zx}\alpha_j^{zy}\alpha_k^{zz}U^z_{ijk}\, g^{zx}(k_i^{zx}x)g^{zy}(k_j^{zy}y)g^{zz}(k_k^{zz}z), \quad \forall (x,y,z) \in \Omega. \end{cases} \tag{36}$$

In the description of the fluctuation (36), we consider a priori different possible boundary conditions on the components $\widetilde{u}_x$, $\widetilde{u}_y$ and $\widetilde{u}_z$. Therefore, each component is separately described with its own combination of functions, which implies that nine different sine–cosine type functions are required and are denoted by g^{xx}, g^{xy}, g^{xz}, g^{yx}, g^{yy}, g^{yz}, g^{zx}, g^{zy} and g^{zz}; the first superscript corresponds to the component of $\widetilde{\mathbf{u}}$ that is modeled, and the second superscript corresponds to the spatial direction described by the function. This convention will also be adopted for the parameters α, k and ξ. The discrete sine–cosine transforms, respectively associated with this description of $\widetilde{u}_x$, $\widetilde{u}_y$ and $\widetilde{u}_z$, are denoted by $\mathcal{D}^x_{\mathrm{xyz}}$, $\mathcal{D}^y_{\mathrm{xyz}}$ and $\mathcal{D}^z_{\mathrm{xyz}}$.

As explained in [29], the last coefficients U^x_{ijk}, U^y_{ijk} and U^z_{ijk} may be poorly estimated for $i = N_x + 1$ or $j = N_y + 1$ or $k = N_z + 1$ using discrete sine–cosine transforms. Therefore, the "high frequencies" are taken null:

$$U^x_{N_x+1jk} = U^x_{iN_y+1k} = U^x_{ijN_z+1} = U^y_{N_x+1jk} = U^y_{iN_y+1k} = U^y_{ijN_z+1} = U^z_{N_x+1jk} = U^z_{iN_y+1k} = U^z_{ijN_z+1} = 0. \tag{37}$$

The resolution thus consists in finding the "Fourier" modes associated with the three components of the displacement fluctuation, i.e., U^x_{ijk}, U^y_{ijk} and U^z_{ijk} (for $i = 0, \dots, N_x$, $j = 0, \dots, N_y$ and $k = 0, \dots, N_z$). The principle of resolution is to assume that the polarization field $\boldsymbol{\tau}$ is known and to solve, for this given value of the polarization, the equilibrium Eqs. (34) with $\mathbf{u}$ described by Eqs. (35) and (36). The resolution can then be done using a Galerkin-type method. The 3-d prismatic domain is discretized uniformly with $(N_x+2)\times(N_y+2)\times(N_z+2)$ points and the spatial scales associated with the uniform grid are $\Delta x = L_x/(N_x+1)$, $\Delta y = L_y/(N_y+1)$ and $\Delta z = L_z/(N_z+1)$. The coordinates of the grid points are denoted by $x_a = a\Delta x$ (for $a = 0, \dots, N_x + 1$), $y_b = b\Delta y$ (for $b = 0, \dots, N_y + 1$) and $z_c = c\Delta z$ (for $c = 0, \dots, N_z + 1$).

The weak formulation of Eq. (31) reads

$$\int_\Omega (\mathbf{div}\,\boldsymbol{\sigma}(\mathbf{x})) \cdot \mathbf{v}^i_{lmn}(\mathbf{x}) \mathrm{d}\Omega = \mathbf{0}, \tag{38}$$

with $\mathbf{v}^i_{lmn}$, $i = 1, 2, 3$, $l = 0, \dots, N_x$, $m = 0, \dots, N_y$, $n = 0, \dots, N_z$, the trial functions, reading

$$\mathbf{v}^1_{lmn}(\mathbf{x}) = \begin{pmatrix} v^x_{lmn} \\ 0 \\ 0 \end{pmatrix} \quad ; \quad \mathbf{v}^2_{lmn}(\mathbf{x}) = \begin{pmatrix} 0 \\ v^y_{lmn} \\ 0 \end{pmatrix} \quad ; \quad \mathbf{v}^3_{lmn}(\mathbf{x}) = \begin{pmatrix} 0 \\ 0 \\ v^z_{lmn} \end{pmatrix}, \tag{39}$$

with

$$v^x_{lmn} = g^{xx}(k^{xx}_l x) g^{xy}(k^{xy}_m y) g^{xz}(k^{xz}_n z), \tag{40}$$

$$v^y_{lmn} = g^{yx}(k^{yx}_l x) g^{yy}(k^{yy}_m y) g^{yz}(k^{yz}_n z), \tag{41}$$

$$v^z_{lmn} = g^{zx}(k^{zx}_l x) g^{zy}(k^{zy}_m y) g^{zz}(k^{zz}_n z). \tag{42}$$

This leads to a system of three equations which can be written under the generic form

$$\begin{cases} I^x_{lmn} + J^x_{lmn} &= -K^x_{lmn} - I^{x0}_{lmn} - J^{x0}_{lmn} \\ I^y_{lmn} + J^y_{lmn} &= -K^y_{lmn} - I^{y0}_{lmn} - J^{y0}_{lmn} \\ I^z_{lmn} + J^z_{lmn} &= -K^z_{lmn} - I^{z0}_{lmn} - J^{z0}_{lmn}, \end{cases} \tag{43}$$

with

$$\begin{cases} I^x_{lmn} &= \displaystyle\int_\Omega \left((\lambda_0 + 2\mu_0) \frac{\partial^2 \widetilde{u}_x}{\partial x^2}(\mathbf{x}) + \mu_0 \left(\frac{\partial^2 \widetilde{u}_x}{\partial y^2}(\mathbf{x}) + \frac{\partial^2 \widetilde{u}_x}{\partial z^2}(\mathbf{x}) \right) \right) v^x_{lmn}(\mathbf{x})\, \mathrm{d}\Omega, \\ I^{x0}_{lmn} &= \displaystyle\int_\Omega \left((\lambda_0 + 2\mu_0) \frac{\partial^2 u^0_x}{\partial x^2}(\mathbf{x}) + \mu_0 \left(\frac{\partial^2 u^0_x}{\partial y^2}(\mathbf{x}) + \frac{\partial^2 u^0_x}{\partial z^2}(\mathbf{x}) \right) \right) v^x_{lmn}(\mathbf{x})\, \mathrm{d}\Omega, \\ J^x_{lmn} &= (\lambda_0 + \mu_0) \displaystyle\int_\Omega \left(\frac{\partial^2 \widetilde{u}_y}{\partial x \partial y}(\mathbf{x}) + \frac{\partial^2 \widetilde{u}_z}{\partial x \partial z}(\mathbf{x}) \right) v^x_{lmn}(\mathbf{x})\, \mathrm{d}\Omega, \\ J^{x0}_{lmn} &= (\lambda_0 + \mu_0) \displaystyle\int_\Omega \left(\frac{\partial^2 u^0_y}{\partial x \partial y}(\mathbf{x}) + \frac{\partial^2 u^0_z}{\partial x \partial z}(\mathbf{x}) \right) v^x_{lmn}(\mathbf{x})\, \mathrm{d}\Omega, \\ K^x_{lmn} &= \displaystyle\int_\Omega \left(\frac{\partial \tau_{xx}}{\partial x}(\mathbf{x}) + \frac{\partial \tau_{xy}}{\partial y}(\mathbf{x}) + \frac{\partial \tau_{xz}}{\partial z}(\mathbf{x}) \right) v^x_{lmn}(\mathbf{x})\, \mathrm{d}\Omega. \end{cases} \tag{44}$$

Similar expressions for I^y_{lmn}, J^y_{lmn}, K^y_{lmn}, I^{y0}_{lmn}, J^{y0}_{lmn}, I^z_{lmn}, J^z_{lmn}, K^z_{lmn}, I^{z0}_{lmn} and J^{z0}_{lmn} can be written using Eq. (34). One can notice that only I^x_{lmn}, J^x_{lmn}, I^y_{lmn}, J^y_{lmn}, I^z_{lmn} and J^z_{lmn} are unknown terms. The calculation of these integrals will be done analytically or numerically, depending on the integral:

- The integrals of type I^x_{lmn} and J^x_{lmn} are calculated analytically as they are based on the calculation of elementary integrals involving different type of sine and cosine functions (related to the orthogonality of the modes) which are provided in Appendix A.
- The integrals of type I^{x0}_{lmn}, J^{x0}_{lmn} and K^x_{lmn} involve functions that are defined (a priori) numerically ($\boldsymbol{\tau}$ and $\mathbf{u}_0$); therefore these integrals will be calculated approximately using discrete sine–cosine transforms (defined in Section 2); this is similar to the standard numerical integration of the Moulinec and Suquet [1] scheme in the periodic setting using discrete Fourier transforms (see [35]).

Hereafter, we discuss the integrals I^x_{lmn} and J^x_{lmn}, as this will be very similar for the other quantities. The detailed calculation of I^x_{lmn} is given in Appendix B and leads to

$$I^x_{lmn} = -\frac{L_x L_y L_z}{8}\left(\left(\lambda_0 + 2\mu_0\right)(\xi^{xx}_l)^2 + \mu_0\left((\xi^{xy}_m)^2 + (\xi^{xz}_n)^2\right)\right) U^x_{lmn}. \tag{45}$$

The computation of J^x_{lmn} is more troublesome as it implies the functions $\widetilde{u}_y$, $\widetilde{u}_z$ and v^x_{lmn}, which, depending on the boundary conditions, can be of different types of sine–cosine series; this leads to two cases:

(1) If $\partial^2 u_y/\partial x\partial y$, $\partial^2 u_z/\partial x\partial z$ and v^x_{lmn} are of the same type of sine–cosine series, J^x_{lmn} is given by

$$J^x_{lmn} = \frac{L_x L_y L_z}{8}\left(\lambda_0 + \mu_0\right)\left(\xi^{yx}_l \xi^{yy}_m U^y_{lmn} + \xi^{zx}_l \xi^{zz}_n U^z_{lmn}\right), \tag{46}$$

and similar relations are obtained for J^y_{lmn} and J^z_{lmn}. Thus, the three equations defining (43) for a triplet (l, m, n) only involve the three coefficients U^x_{lmn}, U^y_{lmn} and U^z_{lmn}, leading to an easy-to-solve system of three equations with three unknowns, which has to be solved for every values of l, m and n. One might remark that these situations correspond to the periodicity compatible mixed uniform boundary conditions (PMUBC) treated in [23] using mirror unit cells.

(2) Otherwise, the computation of J^x_{lmn} relies on "cross-mode" integrals (see Appendix A), implying that J^x_{lmn} will depend on terms U^x_{opq}, U^y_{opq} and U^z_{opq} with $(o, p, q) \neq (l, m, n)$. As a consequence, the weak formulation (38) will lead to a hard-to-solve (non diagonal, non diagonal by blocks, non sparse) linear system of equations, which will require high memory and computational time to solve. Thus, in the general case, the resolution of the *standard* auxiliary problem (31) leads to a computational cost higher than finite element method (and the main advantage of FFT-based methods is lost).

The reason why this auxiliary problem is not suitable is the presence of mixed derivatives in Eq. (34) which leads to a non-diagonal linear system. A modified auxiliary problem is therefore required to keep the computational complexity of FFT.

3.3. The modified auxiliary problem

To resolve the issue explained in Section 3.2, the mixed derivatives involved in Eq. (34) are "transferred" in the polarization tensor. To do so, the elasticity problem is first expressed as a function of the gradient of the displacement field (unsymmetrized). Eq. (28) is rewritten as

$$\begin{cases} \mathbf{div}\,\boldsymbol{\sigma}(\mathbf{x}) = 0 & \forall \mathbf{x} \in \Omega \\ \boldsymbol{\sigma}(\mathbf{x}) = \mathbb{C}(\mathbf{x}) : \boldsymbol{\nabla}\mathbf{u} & \forall \mathbf{x} \in \Omega, \end{cases} \tag{47}$$

since $\mathbb{C} : \boldsymbol{\varepsilon} = \mathbb{C} : \boldsymbol{\nabla}\mathbf{u}$ due to the minor symmetry of the stiffness tensor $\mathbb{C}$. Then, we introduce a modified polarization tensor as

$$\boldsymbol{\tau}(\mathbf{x}) = \left(\mathbb{C}(\mathbf{x}) - \mathbb{B}^0\right) : \boldsymbol{\nabla}\mathbf{u}(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega, \tag{48}$$

where $\mathbb{B}^0$ is a fourth-order tensor reading

$$B^0_{ijkl} = \mu_0 \delta_{ik}\delta_{jl} + (\lambda_0 + \mu_0)\delta_{ik}\delta_{jl}\delta_{kl}, \quad \forall i, j, k, l = 1, 2, 3, \quad \text{(no summation on the indices)}, \tag{49}$$

which has been obtained by putting to zero the components related to cross-derivatives of $\mathbb{C}^0$ (given by Eq. (33)). One might remark that, contrary to $\mathbb{C}^0$, $\mathbb{B}^0$ does not verify the minor and major symmetries; it is thus only a mathematical tool serving the numerical resolution of the auxiliary problem.

The modified auxiliary problem thus reads

$$\begin{cases} \mathbf{div}\,\boldsymbol{\sigma}(\mathbf{x}) = \mathbf{0} & \forall \mathbf{x} \in \Omega \\ \boldsymbol{\sigma}(\mathbf{x}) = \mathbb{B}^0 : \boldsymbol{\nabla}\mathbf{u}(\mathbf{x}) + \boldsymbol{\tau}(\mathbf{x}) & \forall \mathbf{x} \in \Omega. \end{cases} \tag{50}$$

One must remark that $\mathbb{B}^0$ is a positive definite tensor, since

$$\boldsymbol{\nabla}\mathbf{u} : \mathbb{B}^0 : \boldsymbol{\nabla}\mathbf{u} = (\lambda_0 + 2\mu_0)\sum_{i=1}^{3}\left(\frac{\partial u_i}{\partial x_i}\right)^2 + \mu_0 \sum_{i \neq j}\left(\frac{\partial u_i}{\partial x_j}\right)^2, \tag{51}$$

which ensures the existence and the uniqueness of a solution for the modified auxiliary problem. As previously, assuming (momentarily) that the polarization field $\boldsymbol{\tau}$ is known, the modified auxiliary problem defines a system of three equations on the unknown field $\mathbf{u}$, which is freed from mixed derivatives:

$$\begin{cases} \left(\lambda_0 + 2\mu_0\right)\dfrac{\partial^2 u_x}{\partial x^2} + \mu_0\left(\dfrac{\partial^2 u_x}{\partial y^2} + \dfrac{\partial^2 u_x}{\partial z^2}\right) + \dfrac{\partial \tau_{xx}}{\partial x} + \dfrac{\partial \tau_{xy}}{\partial y} + \dfrac{\partial \tau_{xz}}{\partial z} = 0 \\ \left(\lambda_0 + 2\mu_0\right)\dfrac{\partial^2 u_y}{\partial y^2} + \mu_0\left(\dfrac{\partial^2 u_y}{\partial x^2} + \dfrac{\partial^2 u_y}{\partial z^2}\right) + \dfrac{\partial \tau_{xy}}{\partial x} + \dfrac{\partial \tau_{yy}}{\partial y} + \dfrac{\partial \tau_{yz}}{\partial z} = 0 \\ \left(\lambda_0 + 2\mu_0\right)\dfrac{\partial^2 u_z}{\partial z^2} + \mu_0\left(\dfrac{\partial^2 u_z}{\partial x^2} + \dfrac{\partial^2 u_z}{\partial y^2}\right) + \dfrac{\partial \tau_{xz}}{\partial x} + \dfrac{\partial \tau_{yz}}{\partial y} + \dfrac{\partial \tau_{zz}}{\partial z} = 0. \end{cases} \tag{52}$$

The weak formulation associated with the system of partial differential equations reduces to

$$\begin{cases} I^x_{lmn} &= -K^x_{lmn} - I^{x0}_{lmn} \\ I^y_{lmn} &= -K^y_{lmn} - I^{y0}_{lmn} \\ I^z_{lmn} &= -K^z_{lmn} - I^{z0}_{lmn}, \end{cases} \tag{53}$$

where I^x_{lmn}, K^x_{lmn} and I^{x0}_{lmn} are still given by Eq. (44). Considering Eq. (45) (and similar relations for I^y_{lmn} and I^z_{lmn}), the resolution of the weak form of the auxiliary problem read

$$\left\{\begin{array}{lll} U^x_{lmn} = & \dfrac{8}{L_xL_yL_z}\left(\left(\lambda_0+2\mu_0\right)(\xi^{xx}_l)^2+\mu_0\left((\xi^{xy}_m)^2+(\xi^{xz}_n)^2\right)\right)^{-1}\left(K^x_{lmn}+I^{x0}_{lmn}\right) \\ U^y_{lmn} = & \dfrac{8}{L_xL_yL_z}\left(\left(\lambda_0+2\mu_0\right)(\xi^{yy}_m)^2+\mu_0\left((\xi^{yx}_l)^2+(\xi^{yz}_n)^2\right)\right)^{-1}\left(K^y_{lmn}+I^{y0}_{lmn}\right) \quad , \quad \forall l,m,n. \\ U^z_{lmn} = & \dfrac{8}{L_xL_yL_z}\left(\left(\lambda_0+2\mu_0\right)(\xi^{zz}_n)^2+\mu_0\left((\xi^{zx}_l)^2+(\xi^{zy}_m)^2\right)\right)^{-1}\left(K^z_{lmn}+I^{z0}_{lmn}\right) \end{array}\right. \tag{54}$$

The detailed calculations of K^x_{lmn} and I^{x0}_{lmn} are given in Appendix B and rely on the use of discrete sine–cosine transforms (similar relations can be obtained for K^y_{lmn}, I^{y0}_{lmn}, K^z_{lmn} and I^{z0}_{lmn}).

Let us denote by $\boldsymbol{U}^x$, $\boldsymbol{U}^y$ and $\boldsymbol{U}^z$ the arrays of size $(N_x+2)\times(N_y+2)\times(N_z+2)$ containing the values of the sine–cosine series coefficients U^x_{lmn}, U^y_{lmn} and U^z_{lmn} respectively. Eq. (54) can be written alternatively

$$\left\{\begin{array}{ll} \boldsymbol{U}^x & = \boldsymbol{A}^x \odot \left[D^x_{\mathrm{xyz}}(\boldsymbol{\Delta u^0_x}) + \boldsymbol{\xi}^{xx} \odot D^x_{\bar{\mathrm{x}}\mathrm{yz}}(\widetilde{\boldsymbol{\tau}}_{xx}) + \boldsymbol{\xi}^{xy} \odot D^x_{\mathrm{x}\bar{\mathrm{y}}\mathrm{z}}(\widetilde{\boldsymbol{\tau}}_{xy}) + \boldsymbol{\xi}^{xz} \odot D^x_{\mathrm{xy}\bar{\mathrm{z}}}(\widetilde{\boldsymbol{\tau}}_{xz}) + \boldsymbol{S}^x \right] \\ \boldsymbol{U}^y & = \boldsymbol{A}^y \odot \left[D^y_{\mathrm{xyz}}(\boldsymbol{\Delta u^0_y}) + \boldsymbol{\xi}^{yx} \odot D^y_{\bar{\mathrm{x}}\mathrm{yz}}(\widetilde{\boldsymbol{\tau}}_{yx}) + \boldsymbol{\xi}^{yy} \odot D^y_{\mathrm{x}\bar{\mathrm{y}}\mathrm{z}}(\widetilde{\boldsymbol{\tau}}_{yy}) + \boldsymbol{\xi}^{yz} \odot D^y_{\mathrm{xy}\bar{\mathrm{z}}}(\widetilde{\boldsymbol{\tau}}_{yz}) + \boldsymbol{S}^y \right] \\ \boldsymbol{U}^z & = \boldsymbol{A}^z \odot \left[D^z_{\mathrm{xyz}}(\boldsymbol{\Delta u^0_y}) + \boldsymbol{\xi}^{zx} \odot D^z_{\bar{\mathrm{x}}\mathrm{yz}}(\widetilde{\boldsymbol{\tau}}_{zx}) + \boldsymbol{\xi}^{zy} \odot D^z_{\mathrm{x}\bar{\mathrm{y}}\mathrm{z}}(\widetilde{\boldsymbol{\tau}}_{yy}) + \boldsymbol{\xi}^{zz} \odot D^z_{\mathrm{xy}\bar{\mathrm{z}}}(\widetilde{\boldsymbol{\tau}}_{zz}) + \boldsymbol{S}^z \right], \end{array}\right. \tag{55}$$

where $\boldsymbol{A}^x$, $\boldsymbol{A}^y$ and $\boldsymbol{A}^z$ are the $(N_x+2)\times(N_y+2)\times(N_z+2)$ arrays given by

$$\left\{\begin{array}{ll} (\boldsymbol{A})^x_{ijk} & = \dfrac{1}{\left(\lambda_0+2\mu_0\right)\left(\xi^{xx}_i\right)^2+\mu_0\left(\left(\xi^{xy}_j\right)^2+\left(\xi^{xz}_k\right)^2\right)} \times \dfrac{8}{(N_x+1)(N_y+1)(N_z+1)} \\ (\boldsymbol{A})^y_{ijk} & = \dfrac{1}{\left(\lambda_0+2\mu_0\right)\left(\xi^{yy}_j\right)^2+\mu_0\left(\left(\xi^{yx}_i\right)^2+\left(\xi^{yz}_k\right)^2\right)} \times \dfrac{8}{(N_x+1)(N_y+1)(N_z+1)} \\ (\boldsymbol{A})^z_{ijk} & = \dfrac{1}{\left(\lambda_0+2\mu_0\right)\left(\xi^{zz}_k\right)^2+\mu_0\left(\left(\xi^{zx}_i\right)^2+\left(\xi^{zy}_j\right)^2\right)} \times \dfrac{8}{(N_x+1)(N_y+1)(N_z+1)}, \end{array}\right. \tag{56}$$

and $\boldsymbol{\xi}^{xx}$, $\boldsymbol{\xi}^{xy}$, $\boldsymbol{\xi}^{xz}$, $\boldsymbol{\xi}^{yx}$, $\boldsymbol{\xi}^{yy}$, $\boldsymbol{\xi}^{yz}$, $\boldsymbol{\xi}^{zx}$, $\boldsymbol{\xi}^{zy}$, $\boldsymbol{\xi}^{zz}$ are the $(N_x+2)\times(N_y+2)\times(N_z+2)$ arrays given by

$$\left\{\begin{array}{lll} (\boldsymbol{\xi}^{xx})_{ijk}=\xi^{xx}_i, & (\boldsymbol{\xi}^{xy})_{ijk}=\xi^{xy}_j, & (\boldsymbol{\xi}^{xz})_{ijk}=\xi^{xz}_k, \\ (\boldsymbol{\xi}^{yx})_{ijk}=\xi^{yx}_i, & (\boldsymbol{\xi}^{yy})_{ijk}=\xi^{yy}_j, & (\boldsymbol{\xi}^{yz})_{ijk}=\xi^{yz}_k, \\ (\boldsymbol{\xi}^{zx})_{ijk}=\xi^{zx}_i, & (\boldsymbol{\xi}^{zy})_{ijk}=\xi^{zy}_j, & (\boldsymbol{\xi}^{zz})_{ijk}=\xi^{zz}_k. \end{array}\right. \tag{57}$$

The $(N_x+2)\times(N_y+2)\times(N_z+2)$ arrays $\boldsymbol{S}^x$, $\boldsymbol{S}^y$ and $\boldsymbol{S}^z$ contain the values S^x_{lmn}, S^y_{lmn} and S^z_{lmn} (see Appendix B) and $\odot$ denotes Hadamard product (pointwise product) given by

$$(\boldsymbol{A}\odot\boldsymbol{B})_{ijk} = (\boldsymbol{A})_{ijk}(\boldsymbol{B})_{ijk} \qquad \text{(no summation on the indices)}. \tag{58}$$

It should be noted that Eq. (55) allows the calculation of U^x_{lmn}, U^y_{lmn} and U^z_{lmn} for all frequencies except for $l=N_x+1$, $m=N_y+1$ and $n=N_z+1$ where Eq. (37) holds. Once the arrays $\boldsymbol{U}^x$, $\boldsymbol{U}^y$ and $\boldsymbol{U}^z$ are known, each component of the displacement fluctuation $\widetilde{\mathbf{u}}$ is then known using Eq. (36).

3.4. Iterative scheme

As in the periodic case [1], the determination of the polarization tensor $\boldsymbol{\tau}$ solution of the heterogeneous problem is based on an iterative scheme. We consider a fixed-point iterative scheme based on the modified auxiliary problem. One might remark that, due

to the modification of the auxiliary problem, the convergence of the iterative scheme is not guaranteed a priori and must be proven by a theoretical proof, and the convergence rate should be assessed by numerical tests. The iterative scheme reads:

$$
\begin{cases}
\text{Initialization} & \widetilde{\mathbf{u}}^0(\mathbf{x}) = \mathbf{0} \\
& \boldsymbol{\sigma}^0(\mathbf{x}) = \mathbb{C}(\mathbf{x}) : \boldsymbol{\nabla}\mathbf{u}_0(\mathbf{x}) \\
\text{Iterate n + 1} & \widetilde{\mathbf{u}}^n \text{ and } \boldsymbol{\sigma}^n \text{ being known} \\
\text{(a)} & \boldsymbol{\tau}^n(\mathbf{x}) = \boldsymbol{\sigma}^n(\mathbf{x}) - \mathbb{B}^0 : \left(\boldsymbol{\nabla}\mathbf{u}_0(\mathbf{x}) + \boldsymbol{\nabla}\widetilde{\mathbf{u}}^n(\mathbf{x})\right) \\
\text{(b)} & (\boldsymbol{U}^x)^{n+1} = \boldsymbol{A}^x \odot \left[D^x_{\mathrm{xyz}}(\boldsymbol{\Delta u}^{\mathbf{0}}_{\boldsymbol{x}}) + \boldsymbol{\xi}^{xx} \odot D^x_{\bar{\mathrm{x}}\mathrm{yz}}(\widetilde{\boldsymbol{\tau}}^n_{xx}) + \boldsymbol{\xi}^{xy} \odot D^x_{\mathrm{x}\bar{\mathrm{y}}\mathrm{z}}(\widetilde{\boldsymbol{\tau}}^n_{xy}) + \boldsymbol{\xi}^{xz} \odot D^x_{\mathrm{xy}\bar{\mathrm{z}}}(\widetilde{\boldsymbol{\tau}}^n_{xz}) + \boldsymbol{S}^x \right] \\
& (\boldsymbol{U}^y)^{n+1} = \boldsymbol{A}^y \odot \left[D^y_{\mathrm{xyz}}(\boldsymbol{\Delta u}^{\mathbf{0}}_{\boldsymbol{y}}) + \boldsymbol{\xi}^{yx} \odot D^y_{\bar{\mathrm{x}}\mathrm{yz}}(\widetilde{\boldsymbol{\tau}}^n_{yx}) + \boldsymbol{\xi}^{yy} \odot D^y_{\mathrm{x}\bar{\mathrm{y}}\mathrm{z}}(\widetilde{\boldsymbol{\tau}}^n_{yy}) + \boldsymbol{\xi}^{yz} \odot D^y_{\mathrm{xy}\bar{\mathrm{z}}}(\widetilde{\boldsymbol{\tau}}^n_{yz}) + \boldsymbol{S}^y \right] \\
& (\boldsymbol{U}^z)^{n+1} = \boldsymbol{A}^z \odot \left[D^z_{\mathrm{xyz}}(\boldsymbol{\Delta u}^{\mathbf{0}}_{\boldsymbol{z}}) + \boldsymbol{\xi}^{zx} \odot D^z_{\bar{\mathrm{x}}\mathrm{yz}}(\widetilde{\boldsymbol{\tau}}^n_{zx}) + \boldsymbol{\xi}^{zy} \odot D^z_{\mathrm{x}\bar{\mathrm{y}}\mathrm{z}}(\widetilde{\boldsymbol{\tau}}^n_{zy}) + \boldsymbol{\xi}^{zz} \odot D^z_{\mathrm{xy}\bar{\mathrm{z}}}(\widetilde{\boldsymbol{\tau}}^n_{zz}) + \boldsymbol{S}^z \right] \\
\text{(c)} & \widetilde{u}^{n+1}_x(x,y,z) = \sum_{i=0}^{N_x+1}\sum_{j=0}^{N_y+1}\sum_{k=0}^{N_z+1} \alpha^{xx}_i \alpha^{xy}_j \alpha^{xz}_k (U^x_{ijk})^{n+1} g^{xx}(k^{xx}_i x) g^{xy}(k^{xy}_j y) g^{xz}(k^{xz}_k z) \\
& \widetilde{u}^{n+1}_y(x,y,z) = \sum_{i=0}^{N_x+1}\sum_{j=0}^{N_y+1}\sum_{k=0}^{N_z+1} \alpha^{yx}_i \alpha^{yy}_j \alpha^{yz}_k (U^y_{ijk})^{n+1} g^{yx}(k^{yx}_i x) g^{yy}(k^{yy}_j y) g^{yz}(k^{yz}_k z) \\
& \widetilde{u}^{n+1}_z(x,y,z) = \sum_{i=0}^{N_x+1}\sum_{j=0}^{N_y+1}\sum_{k=0}^{N_z+1} \alpha^{zx}_i \alpha^{zy}_j \alpha^{zz}_k (U^z_{ijk})^{n+1} g^{zx}(k^{zx}_i x) g^{zy}(k^{zy}_j y) g^{zz}(k^{zz}_k z) \\
\text{(d)} & \boldsymbol{\sigma}^{n+1}(\mathbf{x}) = \mathbb{C}(\mathbf{x}) : \left(\boldsymbol{\nabla}\mathbf{u}_0(\mathbf{x}) + \boldsymbol{\nabla}\widetilde{\mathbf{u}}^{n+1}(\mathbf{x})\right) \\
\text{(e)} & \text{Convergence test.}
\end{cases}
\tag{59}
$$

The convergence test consists in verifying the local equilibrium. Then, following Parseval's identity, the L^2 norm of $\mathbf{div}\ \boldsymbol{\sigma}(\mathbf{x})$ reads

$$
\|\mathbf{div}\ \boldsymbol{\sigma}\|^2_{L^2} = \frac{1}{L_x L_y L_z}\int_\Omega (\mathbf{div}\ \boldsymbol{\sigma}(\mathbf{x}))^2\,\mathrm{d}\Omega = \sum_{i=0}^{N_x}\sum_{j=0}^{N_y}\sum_{k=0}^{N_z}\left((L^x_{ijk})^2 + (L^y_{ijk})^2 + (L^z_{ijk})^2\right) \tag{60}
$$

where L^x_{ijk}, L^y_{ijk} and L^z_{ijk} are given by

$$
\begin{cases}
L^x_{ijk} &= \dfrac{8}{L_x L_y L_z}\displaystyle\int_\Omega \left(\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z}\right) v^x_{ijk}(\mathbf{x})\mathrm{d}\Omega \\
L^y_{ijk} &= \dfrac{8}{L_x L_y L_z}\displaystyle\int_\Omega \left(\frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yz}}{\partial z}\right) v^y_{ijk}(\mathbf{x})\mathrm{d}\Omega \\
L^z_{ijk} &= \dfrac{8}{L_x L_y L_z}\displaystyle\int_\Omega \left(\frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z}\right) v^z_{ijk}(\mathbf{x})\mathrm{d}\Omega.
\end{cases}
\tag{61}
$$

The integrals (61) are computed as described in Appendix B for $\mathbf{div}\ \boldsymbol{\tau}$, and the convergence is considered achieved when [27]

$$
\frac{\Delta x\sqrt{\displaystyle\sum_{i=0}^{N_x}\sum_{j=0}^{N_y}\sum_{k=0}^{N_z}\left((L^x_{ijk})^2 + (L^y_{ijk})^2 + (L^z_{ijk})^2\right)}}{\sqrt{\displaystyle\sum_{i=0}^{N_x}\sum_{j=0}^{N_y}\sum_{k=0}^{N_z}(\sigma^2_{xx,ijl} + \sigma^2_{yy,ijl} + \sigma^2_{zz,ijl} + 2\sigma^2_{xy,ijl} + 2\sigma^2_{xz,ijl} + 2\sigma^2_{yz,ijl})}} \leq \mathrm{tol}, \tag{62}
$$

where tol is the tolerance taken at 10^{-8} in the following.

3.5. Proof of convergence associated with the modified auxiliary problem

As the proposed modification of the polarization tensor may alter the convergence of the iterative scheme, a proof of convergence is required to (i) ensure the convergence of the method and (ii) provide guidelines for the determination of an appropriate comparison material. Following the convergence proof for the classical FFT-based method proposed by Michel et al. [39], we look into the eigenvalues of the operator leading from an iteration to another. In the periodic framework proposed by Moulinec and Suquet [1], which operates on the strain field, it is referred as the Green operator and noted $\boldsymbol{\Gamma}^0$. It is slightly different in this work as we operate on the displacement field and with different boundary conditions.

For given boundary conditions (either Dirichlet and/or Neumann), let us consider the linear operator $\boldsymbol{\Lambda}^0$ giving the solution $\mathbf{u}$ of the auxiliary problem (50) under null boundary conditions and for a given polarization tensor $\boldsymbol{\tau}$. Then,

$$
\mathbf{u} = \boldsymbol{\Lambda}^0(\boldsymbol{\tau}) \tag{63}
$$

is the unique solution of (50) under null boundary conditions and therefore verifies

$$\mathbf{div}(\mathbb{B}^0 : \boldsymbol{\nabla}\mathbf{u}(\mathbf{x})) = -\mathbf{div}(\boldsymbol{\tau}(\mathbf{x})), \quad \forall \mathbf{x}. \tag{64}$$

For further use, we remark that, for a polarization field given by

$$\boldsymbol{\tau}(\mathbf{x}) = -\mathbb{B}^0 : \boldsymbol{\nabla}\mathbf{u}(\mathbf{x}), \quad \forall \mathbf{x}, \tag{65}$$

with $\mathbf{u}$ a given displacement field with null boundary conditions, $\mathbf{u}$ is the solution of (64), leading to the identity

$$\mathbf{u} = \boldsymbol{\Lambda}^0(-\mathbb{B}^0 : \boldsymbol{\nabla}\mathbf{u}). \tag{66}$$

Considering the algorithm presented in Section 3.4, the fluctuation field at iteration $n+1$ is solution of

$$\mathbf{div}(\mathbb{B}^0 : \boldsymbol{\nabla}\widetilde{\mathbf{u}}^{n+1}(\mathbf{x})) = -\mathbf{div}(\boldsymbol{\tau}^n(\mathbf{x}) + \mathbb{B}^0 : \boldsymbol{\nabla}\mathbf{u}^0) \quad \forall \mathbf{x}, \tag{67}$$

under null boundary conditions. Then, the iterative scheme can be expressed as

$$\widetilde{\mathbf{u}}^{n+1} = \boldsymbol{\Lambda}^0(\boldsymbol{\tau}^n + \mathbb{B}^0 : \boldsymbol{\nabla}\mathbf{u}^0) = \boldsymbol{\Lambda}^0\left((\mathbb{C} - \mathbb{B}^0) : \boldsymbol{\nabla}(\mathbf{u}^0 + \widetilde{\mathbf{u}}^n) + \mathbb{B}^0 : \boldsymbol{\nabla}\mathbf{u}^0\right). \tag{68}$$

Let us split the fluctuation field $\widetilde{\mathbf{u}}^n$ at iteration n into the fluctuation of the solution field of the actual problem, denoted as $\widetilde{\mathbf{u}}_{\text{sol}}$, and the error field at iteration n, denoted as $\widetilde{\mathbf{u}}^n_{\text{err}}$. It reads

$$\widetilde{\mathbf{u}}^n = \widetilde{\mathbf{u}}_{\text{sol}} + \widetilde{\mathbf{u}}^n_{\text{err}}. \tag{69}$$

Following (68), the solution of the actual problem must verify

$$\widetilde{\mathbf{u}}_{\text{sol}} = \boldsymbol{\Lambda}^0\left((\mathbb{C} - \mathbb{B}^0) : \boldsymbol{\nabla}(\mathbf{u}^0 + \widetilde{\mathbf{u}}_{\text{sol}}) + \mathbb{B}^0 : \boldsymbol{\nabla}\mathbf{u}^0\right). \tag{70}$$

Considering (68), (69) and (70), the evolution of the error field with the iterations can be expressed as

$$\widetilde{\mathbf{u}}^{n+1}_{\text{err}} = \boldsymbol{\Lambda}^0\left((\mathbb{C} - \mathbb{B}^0) : \boldsymbol{\nabla}\widetilde{\mathbf{u}}^n_{\text{err}}\right). \tag{71}$$

Thus, one can ensure the convergence of the iterative scheme by proving that the operator

$$\widetilde{\mathbf{u}}_{\text{err}} \to \boldsymbol{\Lambda}^0\left((\mathbb{C} - \mathbb{B}^0) : \boldsymbol{\nabla}\widetilde{\mathbf{u}}_{\text{err}}\right), \tag{72}$$

is a contracting operator, i.e. that its eigenvalues lie between minus one and one. Let us consider an eigenvector $\widetilde{\mathbf{u}}^\alpha$ of the operator associated with an eigenvalue α, i.e. such that

$$\alpha\widetilde{\mathbf{u}}^\alpha = \boldsymbol{\Lambda}^0\left((\mathbb{C} - \mathbb{B}^0) : \boldsymbol{\nabla}\widetilde{\mathbf{u}}^\alpha\right). \tag{73}$$

Following the reasoning presented in [39], and taking advantage of the identity (66) to obtain $\boldsymbol{\Lambda}^0((1-\alpha)\mathbb{B}^0 : \boldsymbol{\nabla}\widetilde{\mathbf{u}}^\alpha) = -(1-\alpha)\widetilde{\mathbf{u}}^\alpha$, one can rewrite (73) into

$$\widetilde{\mathbf{u}}^\alpha = \boldsymbol{\Lambda}^0\left(\left((\mathbb{C} - (1-\alpha)\mathbb{B}^0\right) - \mathbb{B}^0\right) : \boldsymbol{\nabla}\widetilde{\mathbf{u}}^\alpha\right), \tag{74}$$

which means that $\widetilde{\mathbf{u}}^\alpha$ is solution of

$$\begin{cases} \mathbf{div}\,\boldsymbol{\sigma}(\mathbf{x}) = \mathbf{0} & \forall \mathbf{x} \in \Omega \\ \boldsymbol{\sigma}(\mathbf{x}) = \left(\mathbb{C}(\mathbf{x}) - (1-\alpha)\mathbb{B}^0\right) : \boldsymbol{\nabla}\mathbf{u}(\mathbf{x}) & \forall \mathbf{x} \in \Omega, \end{cases} \tag{75}$$

under null boundary conditions. If $\mathbb{C}(\mathbf{x}) - (1-\alpha)\mathbb{B}^0$ is positive definite (resp. negative definite) $\forall \mathbf{x} \in \Omega$, Eq. (75) has a unique solution, which is $\widetilde{\mathbf{u}}^\alpha = 0$. Thus, for every eigenvalues α, there are $\mathbf{x}$ such that $\mathbb{C}(\mathbf{x}) - (1-\alpha)\mathbb{B}^0$ has negative eigenvalues *and* $\mathbf{x}$ such that $\mathbb{C}(\mathbf{x}) - (1-\alpha)\mathbb{B}^0$ has positive eigenvalues. To study the eigenvalues of $\mathbb{C}(\mathbf{x}) - (1-\alpha)\mathbb{B}^0$, we consider, as done by Michel et al. [39], the particular case of $\mathbb{C}$ isotropic (i.e. that $C_{ijkl}(\mathbf{x}) = \lambda(\mathbf{x})\delta_{ij}\delta_{kl} + \mu(\mathbf{x})(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk})$). Eq. $(75)_2$ is then written under a matrix form, reading

$$\begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{xy} \\ \sigma_{yx} \\ \sigma_{xz} \\ \sigma_{zx} \\ \sigma_{yz} \\ \sigma_{zy} \end{pmatrix} = \begin{pmatrix} \Delta_{\mu\lambda} & \lambda & \lambda & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda & \Delta_{\mu\lambda} & \lambda & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda & \lambda & \Delta_{\mu\lambda} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Delta_\mu & \mu & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & \Delta_\mu & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \Delta_\mu & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu & \Delta_\mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Delta_\mu & \mu \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu & \Delta_\mu \end{pmatrix} \begin{pmatrix} \partial u_x/\partial x \\ \partial u_y/\partial y \\ \partial u_z/\partial z \\ \partial u_y/\partial x \\ \partial u_x/\partial y \\ \partial u_z/\partial x \\ \partial u_x/\partial z \\ \partial u_z/\partial y \\ \partial u_x/\partial z \end{pmatrix} \tag{76}$$

with

$$\Delta_{\mu\lambda} = 2\mu(\mathbf{x}) + \lambda(\mathbf{x}) - (1-\alpha)(2\mu^0 + \lambda^0) \quad ; \quad \Delta_\mu = \mu(\mathbf{x}) - (1-\alpha)\mu^0. \tag{77}$$

The eigenvalues of the matrix in Eq. (76) read

$$\begin{cases} e_1 &= 2\mu(\mathbf{x}) - (1-\alpha)(2\mu_0 + \lambda_0) \\ e_2 &= 2\mu(\mathbf{x}) + 3\lambda(\mathbf{x}) - (1-\alpha)(2\mu_0 + \lambda_0) \\ e_3 &= 2\mu(\mathbf{x}) - (1-\alpha)\mu_0 \\ e_4 &= -(1-\alpha)\mu_0. \end{cases} \tag{78}$$

First, it is easy to remark that $\alpha \geq 1$ leads to $e_i \geq 0$ for $i = 1, 2, 3, 4$, which is excluded. In the case $\alpha < 1$, one has $e_4 < 0$. Therefore, the existence of an eigenvalue α requires $e_1 \geq 0$ or $e_2 \geq 0$ or $e_3 \geq 0$, leading to :

$$\alpha \geq 1 - \frac{2\mu(\mathbf{x})}{2\mu^0 + \lambda^0} \quad \text{or} \quad \alpha \geq 1 - \frac{2\mu(\mathbf{x}) + 3\lambda(\mathbf{x})}{2\mu^0 + \lambda^0} \quad \text{or} \quad \alpha \geq 1 - \frac{2\mu(\mathbf{x})}{\mu^0}. \tag{79}$$

Then, the following conditions,

$$2\mu^0 + \lambda^0 > \mu(\mathbf{x}) \quad \text{and} \quad 2\mu^0 + \lambda^0 > \mu(\mathbf{x}) + \frac{3}{2}\lambda(\mathbf{x}) \quad \text{and} \quad \mu^0 > \mu(\mathbf{x}), \tag{80}$$

enforce $\alpha > -1$, and therefore ensure the convergence of the iterative scheme.

Following [39], one might try to estimate the rate of convergence thanks to the spectral radius of the operator, i.e. the minimum and maximum values of the eigenvalues α. An eigenvalue α can be expressed with respect to e_i, $i = 1, 2, 3, 4$ using (78). It reads

$$\alpha = 1 + \frac{e_1(\mathbf{x}) - 2\mu(\mathbf{x})}{2\mu_0 + \lambda_0} = 1 + \frac{e_2(\mathbf{x}) - (2\mu(\mathbf{x}) + 3\lambda(\mathbf{x}))}{2\mu_0 + \lambda_0} = 1 + \frac{e_3(\mathbf{x}) - 2\mu(\mathbf{x})}{\mu_0} = 1 + \frac{e_4(\mathbf{x})}{\mu_0}, \quad \forall \mathbf{x}. \tag{81}$$

Let us denote as $\alpha_i(\mathbf{x})$ the following expressions, obtained by putting $e_i(\mathbf{x})$ to zero in (81),

$$\begin{cases} \alpha_1(\mathbf{x}) &= 1 - \dfrac{2\mu(\mathbf{x})}{2\mu_0 + \lambda_0}, \\ \alpha_2(\mathbf{x}) &= 1 - \dfrac{2\mu(\mathbf{x}) + 3\lambda(\mathbf{x})}{2\mu_0 + \lambda_0}, \\ \alpha_3(\mathbf{x}) &= 1 - \dfrac{2\mu(\mathbf{x})}{\mu_0}, \\ \alpha_4(\mathbf{x}) &= 1. \end{cases} \tag{82}$$

Considering that, for any eigenvalue α, there are i and $\mathbf{x}$ such that $e_i(\mathbf{x}) \geq 0$ and i and $\mathbf{x}$ such that $e_i(\mathbf{x}) \leq 0$, equation (81) leads to the following bounds

$$\min_{i,\mathbf{x}} \alpha_i(\mathbf{x}) \leq \alpha \leq \max_{i,\mathbf{x}} \alpha_i(\mathbf{x}) = 1. \tag{83}$$

Thus, this approach only proves that the spectral radius is inferior to one, which ensures convergence of the iterative scheme (at least on the finite dimension spaces considered in the practical applications), but does not give any further information about the rate of convergence. Considering the classical scheme and the proof of convergence proposed by Michel et al. [39], in which the spectral radius is bounded by values related to the comparison material and strictly inferior to one, it is interesting to note that the modification of the auxiliary problem leads to different eigenvalues of the local operator (matrix (76)), and particularly to the e_4 eigenvalue, whose eigenvector is related to the antisymmetric part of $\boldsymbol{\nabla}\mathbf{u}$, not considered in the classical framework. This particularity prevents from a better bound of the spectral radius, and therefore, contrary to the periodic case considered in [39], it does not provide an estimate of the rate of convergence. Further numerical assessments will give a better appreciation of the actual rate of convergence of the proposed method.

4. Applications

4.1. Description of the simulations

The discrete sine–cosine based method is now applied to the study of the local and overall response of a composite material made of two isotropic phases,[6] a matrix of elasticity properties (κ_1, μ_1) and an inclusion of elasticity properties (κ_2, μ_2). In the applications considered, we fix Poisson's ratios in each phase to $\nu_1 = \nu_2 = 0.25$, so the contrast between the phases reduce to a single parameter and is defined by $\kappa_2/\kappa_1 = \mu_2/\mu_1 = \lambda_2/\lambda_1 = E_2/E_1$. For the application, we consider a 2-d plane strain case and the 2-d domain $[0, 1]\times[0, 1]$ is discretized with 512×512 grid points, comprising a centered square inclusion of side 0.5. Following the convergence conditions given by (80), we consider the values $\lambda_0 = 1.5 \times \max(\lambda_1, \lambda_2)$ and $\mu_0 = 1.5 \times \max(\mu_1, \mu_2)$ for the homogeneous reference material used to define the auxiliary problem.

Several types of boundary conditions will be considered:

- Kinematic Uniform Boundary Conditions (KUBC) (corresponding to a particular Dirichlet case) associated with the volume average (macroscopic) of the strain field $\langle \boldsymbol{\varepsilon} \rangle$, that is

$$\mathbf{u}(\mathbf{x}) = \langle \boldsymbol{\varepsilon} \rangle \cdot \mathbf{x}, \quad \forall \mathbf{x} \in \partial\Omega. \tag{84}$$

[6] We consider, for illustrative purposes, an isotropic behavior, but the method is naturally written in the anisotropic setting as shown by Eq. (32).

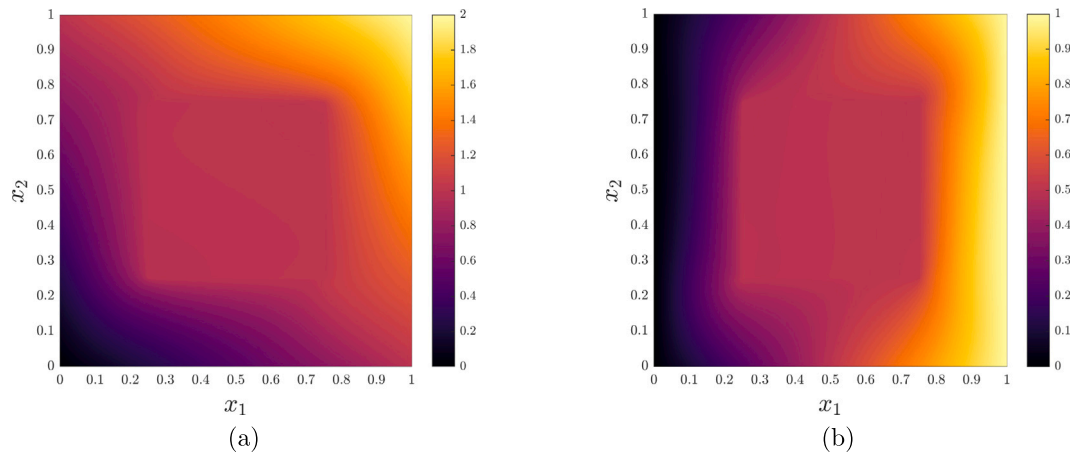

Fig. 1. Distribution of the components of the displacement field in the KUBC case. (a) Component u_1 and (b) Component u_2.

- Static Uniform Boundary Conditions (SUBC) (corresponding to a particular Neumann case) associated with the volume average of the stress field $\langle \boldsymbol{\sigma} \rangle$, that is

$$\boldsymbol{\sigma}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = \langle \boldsymbol{\sigma} \rangle \cdot \mathbf{n}(\mathbf{x}), \quad \forall \mathbf{x} \in \partial\Omega, \tag{85}$$

  where $\mathbf{n}(\mathbf{x})$ is the outward normal.
- Arbitrary Dirichlet–Neumann Boundary Conditions (ADNBC), for which an arbitrary displacement $\mathbf{u}(\mathbf{x})$ is prescribed on the boundary $\partial\Omega_{\mathrm{D}}$ and a surface force $\boldsymbol{\sigma}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x})$ is prescribed on the boundary $\partial\Omega_{\mathrm{N}}$.

The cases of KUBC and SUBC will be used to calculate the macroscopic (effective) elasticity tensor using the relation

$$\langle \boldsymbol{\sigma} \rangle = \overline{\mathbb{C}} : \langle \boldsymbol{\varepsilon} \rangle, \tag{86}$$

where $\langle \boldsymbol{\sigma} \rangle$ is calculated as the volume average of the stress $\boldsymbol{\sigma}$ (in the KUBC case) and $\langle \boldsymbol{\varepsilon} \rangle$ is calculated as the volume average of the strain $\boldsymbol{\varepsilon}$ (in the SUBC case). In the present 2D case, $\overline{\mathbb{C}}$ will be expressed using the following Kelvin notation:

$$\begin{pmatrix} \langle \sigma_{11} \rangle \\ \langle \sigma_{22} \rangle \\ \langle \sigma_{12} \rangle \end{pmatrix} = \begin{pmatrix} \overline{C}_{1111} & \overline{C}_{1122} & \overline{C}_{1112} \\ \overline{C}_{1122} & \overline{C}_{2222} & \overline{C}_{2212} \\ \overline{C}_{1112} & \overline{C}_{2212} & \overline{C}_{1212} \end{pmatrix} \cdot \begin{pmatrix} \langle \varepsilon_{11} \rangle \\ \langle \varepsilon_{22} \rangle \\ 2\langle \varepsilon_{12} \rangle \end{pmatrix}. \tag{87}$$

Given the symmetries of the problem considered, the overall elasticity tensor is expected to be cubic (i.e. $\overline{C}_{1112} = \overline{C}_{2212} = 0$ and $\overline{C}_{1111} = \overline{C}_{2222}$).

In all the applications considered, we take the value $\mathrm{tol} = 10^{-8}$ for the tolerance used in criterion (62).

4.2. Local fields for KUBC and SUBC cases

We consider a contrast $\kappa_2/\kappa_1 = 10^2$.

Kinematic uniform boundary conditions. We perform a KUBC simulation with the following loading

$$\langle \boldsymbol{\varepsilon} \rangle = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}. \tag{88}$$

Convergence of the iterative scheme is reached for about 2000 iterations. The distribution of the components u_1 and u_2 of the displacement field is represented in Fig. 1. One can notice that the boundary conditions (84) are correctly applied, for the loading considered (88). The components ε_{11} and ε_{12} are represented in Fig. 2. These fields show no numerical artifact at the cell boundaries but classical oscillations are observed near the inclusion corner, which is typical when a continuous Green operator is used. The components σ_{11} and σ_{12} are finally represented in Fig. 2.

Considering the contrast $\kappa_2/\kappa_1 = 10^2$, the calculations performed using KUBC lead to the macroscopic elasticity tensor (expressed using Eq. (87)):

$$\overline{\mathbb{C}}_{\mathrm{KUBC}} = \begin{pmatrix} 1.93 & 0.51 & 0 \\ 0.51 & 1.93 & 0 \\ 0 & 0 & 0.63 \end{pmatrix}. \tag{89}$$

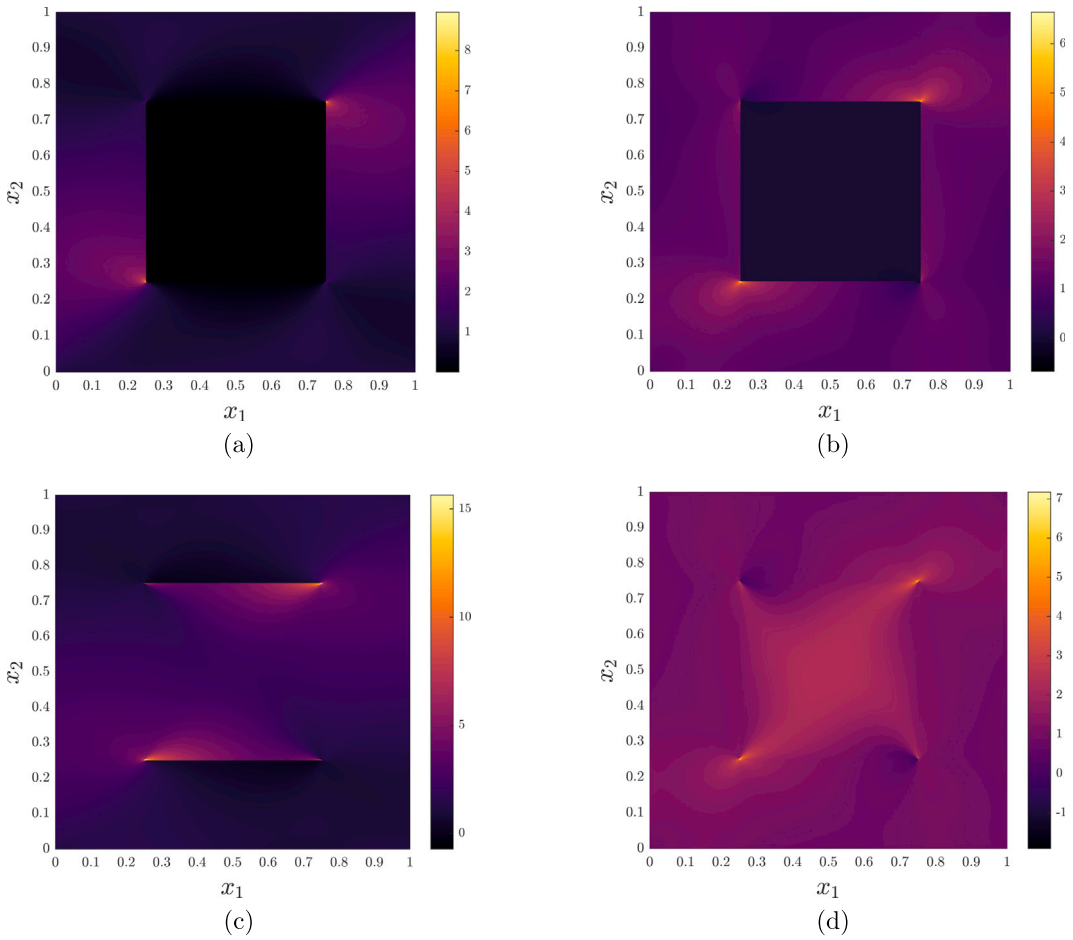

Fig. 2. Distribution of the components of the strain and stress fields in the KUBC case. (a) Strain component ε_{11}, (b) Strain component ε_{12}, (c) Stress component σ_{11} and (d) Stress component σ_{12}.

Static uniform boundary conditions. We continue with a SUBC simulation considering the following loading

$$\langle \boldsymbol{\sigma} \rangle = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}. \tag{90}$$

In that case, convergence of the iterative scheme is reached for about 6500 iterations. The components ε_{11} and ε_{12} are represented in Fig. 3. As previously, no numerical artifact are observed at the cell boundaries but classical oscillations are observed near the inclusion corner. Finally the components σ_{11} and σ_{12} are represented in Fig. 3. One can notice that the boundary conditions (85) are correctly applied, for the loading considered (90).

The calculations performed using SUBC lead to the macroscopic elasticity tensor (expressed using the convention of Eq. (87)):

$$\overline{\mathbb{C}}_{\text{SUBC}} = \begin{pmatrix} 1.81 & 0.59 & 0 \\ 0.59 & 1.81 & 0 \\ 0 & 0 & 0.54 \end{pmatrix}. \tag{91}$$

It is easy to show that $\overline{\mathbb{C}}_{\text{KUBC}} > \overline{\mathbb{C}}_{\text{SUBC}}$ in the sense of quadratic forms.

4.3. Influence of the contrast

We study in this section the influence of the contrast upon (i) the number of iteration to convergence and (ii) the macroscopic elastic properties. We focus on the 2-d plane strain bulk modulus only, which is defined as

$$\overline{\kappa}^{2\text{d}} = \frac{\overline{C}_{1111} + \overline{C}_{1122}}{2}. \tag{92}$$

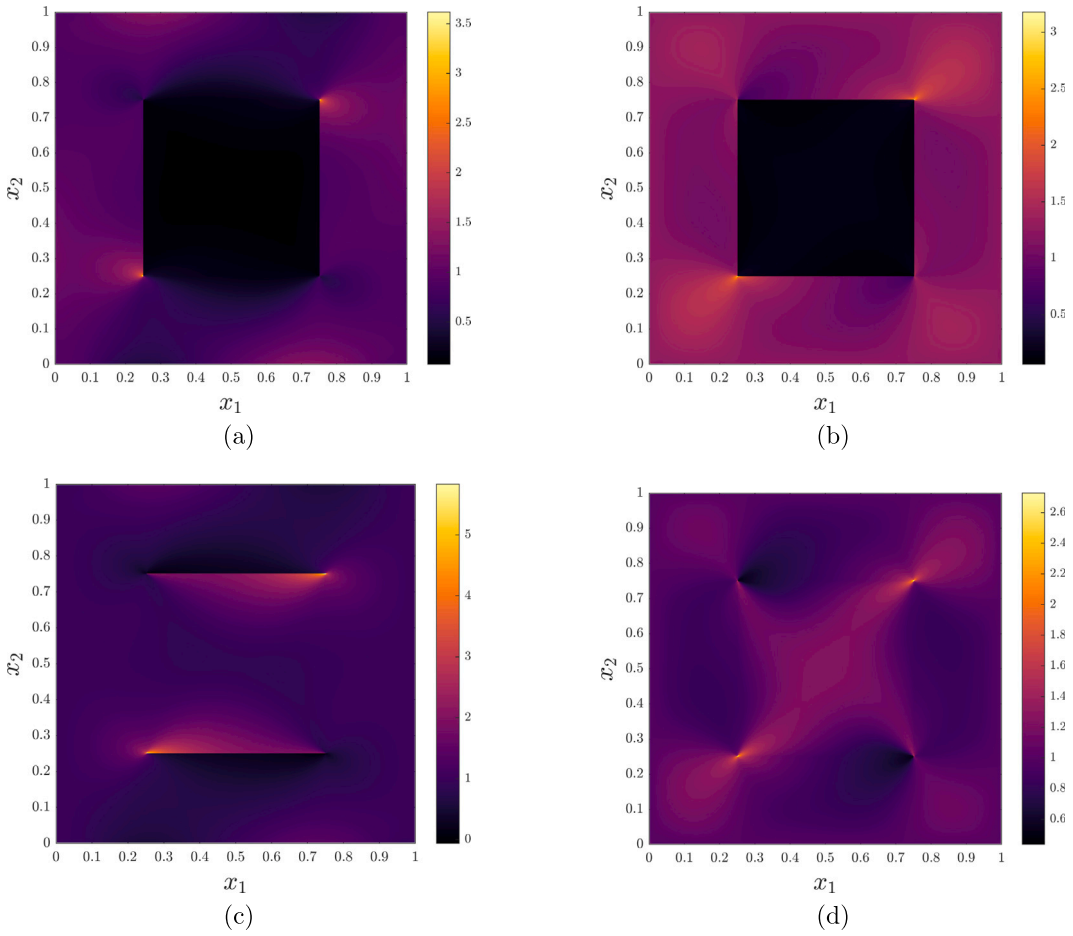

Fig. 3. Distribution of the components of the strain and stress fields in the SUBC case. (a) Strain component ε_{11}, (b) Strain component ε_{12}, (c) Stress component σ_{11} and (d) Stress component σ_{12}.

In addition to the KUBC and SUBC cases, we also consider the periodic case (PBC) in order to provide a numerical comparison of the number of iterations to convergence between the periodic and the non-periodic approaches (and also the macroscopic 2-d bulk modulus).

First, number of iterations up to convergence is represented in Fig. 4 as a function of the contrast. As expected with a continuous Green operator with a fixed-point iterative procedure, the number of iterations increases with the contrast, similarly to the non-periodic conductivity case [29]. One might remark that, despite the poor spectral radius bound derived in Section 3.5 (which is not related to the contrast) one retrieves the classical linear dependency on the contrast, as with the classical scheme (PBC). Accelerated schemes developed in the periodic case are expected to improve the convergence rate (see e.g. [6]). It must be noted that the SUBC case requires extra iterations due to the one-voxel layer that is required for imposing the surface force; this could be possibly improved by adjusting from one iteration to another (i) the material of the layer and (ii) the prescribed displacement field so as to avoid any strong discontinuities across the interface.

Then, the dependency of the (normalized) macroscopic bulk modulus $\overline{\kappa}^{2d}/\kappa_1^{2d}$ with the contrast is represented in Fig. 4b. As expected it increases with the contrast and it is bounded by the asymptotic values $\overline{\kappa}^{2d}/\kappa_1^{2d} = 0.47$ and $\overline{\kappa}^{2d}/\kappa_1^{2d} = 0.4$ when $\kappa_2^{2d}/\kappa_1^{2d} \to 0$, respectively for the KUBC and SUBC cases, and $\overline{\kappa}^{2d}/\kappa_1^{2d} = 1.53$ and $\overline{\kappa}^{2d}/\kappa_1^{2d} = 1.52$ when $\kappa_2^{2d}/\kappa_1^{2d} \to \infty$, respectively for the KUBC and SUBC cases. The value predicted using PBC lies between the bounds obtained using KUBC and SUBC; this is related to the (standard) property $\overline{\mathbb{C}}_{\text{KUBC}} > \overline{\mathbb{C}}_{\text{PBC}} > \overline{\mathbb{C}}_{\text{SUBC}}$ in the sense of quadratic forms.

4.4. Tensile test of square heterogeneous beam

We finally study the case of a square heterogeneous "beam" subjected to a tensile test (see Fig. 5). This case is interesting as it shows the ability of the present approach to be used not only in the context of micromechanics (using KUBC or SUBC) but also for beams (or plates) made of heterogeneous constituents, which can be considered as a first step towards the used of FFT-based for simple structures. The boundary conditions, shown in Fig. 5, can be applied by considering

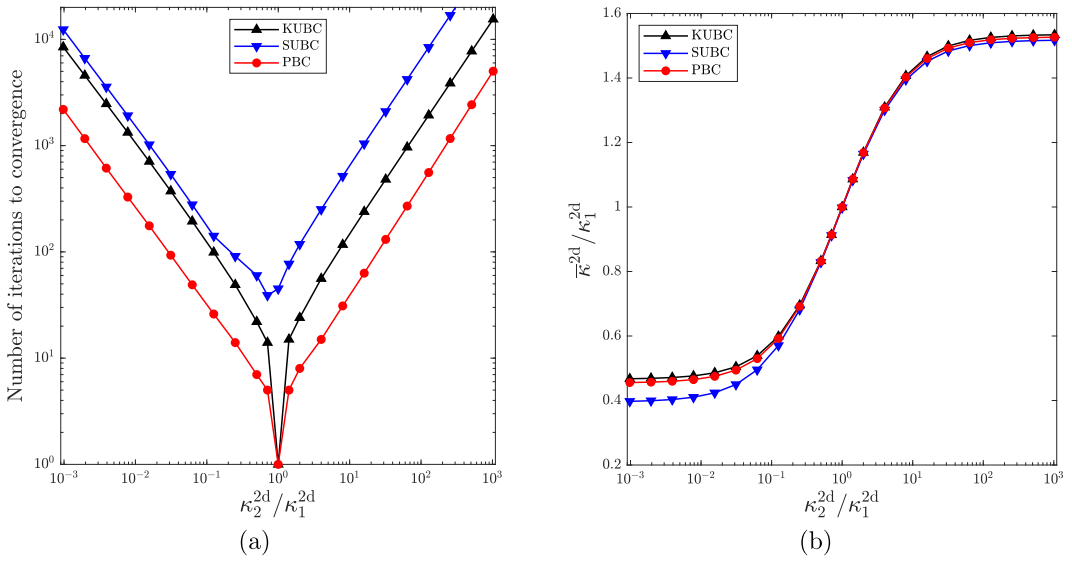

Fig. 4. Influence of the contrast. (a) Number of iterations up to convergence and (b) Normalized macroscopic bulk modulus $\overline{\kappa}^{2d}/\kappa_1^{2d}$.

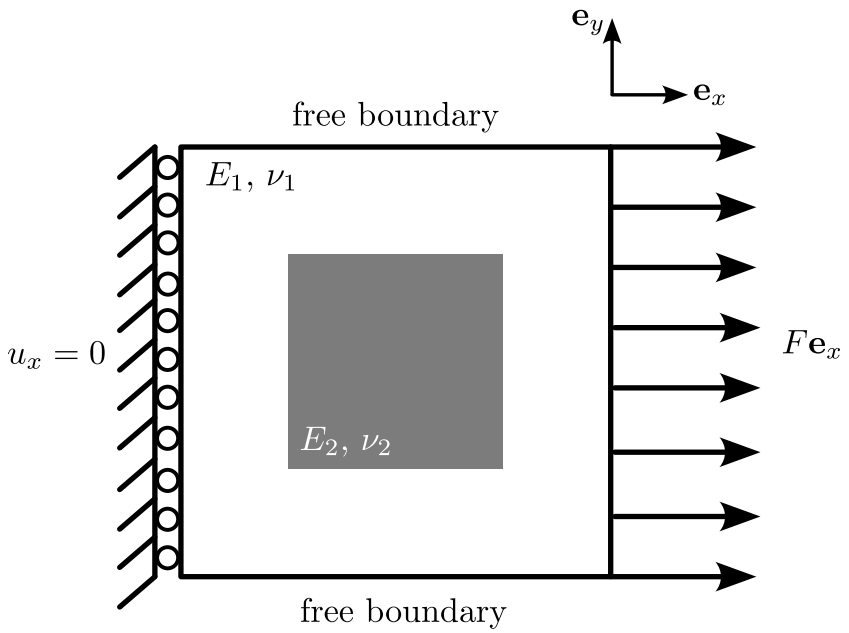

Fig. 5. Tensile test of square heterogeneous "beam".

- For the displacement u_x, Dirichlet–Neumann conditions on the $x-$faces and Neumann–Neumann conditions on the $y-$faces. This leads to a DST-III for the $x-$direction and DCT-I for the $y-$direction;
- For the displacement u_y, Neumann–Neumann conditions on the $x-$faces and Neumann–Neumann conditions on the $y-$faces. This leads to a DCT-I for the $x-$direction and DCT-I for the $y-$direction;

It must be noted that a clamped boundary condition could also be applied by considering, for u_y, DST-III instead of DCT-I. We consider the case $\nu_1 = \nu_2 = 0.25$ and $E_2/E_1 = 1/10$, which corresponds to a soft inclusion.

The distribution of the displacement field is first represented in Fig. 6. One can remark that the boundary conditions are correctly applied as the component u_1 is null on the left boundary. The component u_1 is heterogeneous but it is overall increasing with x which is expected from a tensile test. One can remark that u_1 is heterogeneous on the right boundary ($x = L_x$), as there is no prescribed displacement on this face. The component u_2 is also heterogeneous and with negative and positive values respectively on the top and bottom boundaries, which classically corresponds to a Poisson effect.

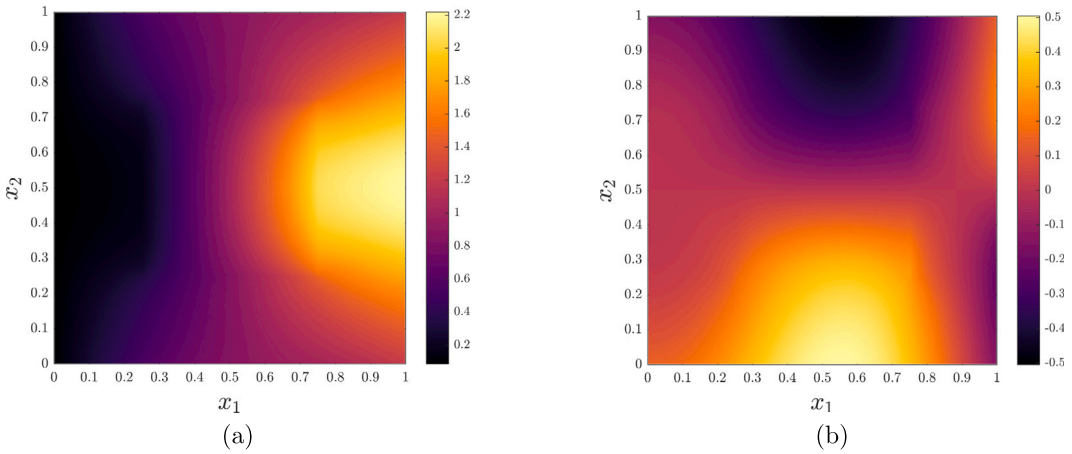

Fig. 6. Distribution of the components of the displacement field in the "tensile test". (a) Component u_1 and (b) Component u_2.

The components of the strain (ε_{11} and ε_{22}) and stress (σ_{11} and σ_{22}) fields are then represented in Fig. 7. The component σ_{11} is uniform on the right boundary since a uniform surface force has been applied. The presence of the inclusion leads to heterogeneous mechanical fields. In particular, the tensile stress is heterogeneous in the left boundary which is due to the inclusion. Overall, those results enhance the importance of the boundary conditions for this kind of situation, as the presence of the inclusion results in significantly different displacement, strain and stress components on opposite faces, which cannot be predicted by the usual periodic boundary conditions.

5. Conclusion

This work was devoted to the development of a new class of spectral FFT-based solvers for materials with heterogeneous mechanical properties and subjected to non-periodic, Dirichlet and/or Neumann, boundary conditions. This works is thus a direct extension of the seminal work of Moulinec and Suquet [1] to non-periodic boundary conditions. The method is based, as in the case of conductivity [27–29], on the decomposition of the displacement field solution of the problem into a known term verifying the boundary conditions and a fluctuation term described by appropriate sine–cosine series inducing no contribution on the boundary. The principle of the method is then very similar to the periodic case, i.e. it is based on an iterative solution of the so-called Lippmann–Schwinger equation [1]. The auxiliary problem is simply modified (through some reference material acting on the displacement gradient and not its symmetrical part) in order not to include cross-derivatives (which were shown to lead to a hard-to-solve linear system). This auxiliary problem is solved using discrete sine–cosine transforms which naturally emerge in the Galerkin method using an approximation space spanned by sine–cosine series. The method therefore relies on the numerical complexity of fast Fourier transforms and allows the consideration of arbitrary boundary conditions on entire faces, including as particular cases kinematic uniform boundary conditions and static uniform boundary conditions. The method has been successfully applied to several cases including the homogenization of a composite (using both KUBC and SUBC) as well as some "structure" (a beam subjected to a tensile loading).

As shown by a study of the influence of the contrast, the present method suffers from the same drawbacks than the initial method of Moulinec and Suquet [1] in the periodic case, i.e. the convergence rate scales linearly with the contrast, making the method not suitable for highly-contrasted materials. This can be improved by the use of a discrete scheme based on finite differences as done by Gélébart [27] in conductivity problems; this will be presented in a future work. This approach based on discrete Green operator is also expected to improve the quality of the solution field, with a notable reduction of spurious oscillations [30]. In addition, accelerated iterative schemes are also expected to improve the convergence of the method [3–7].

This work was restricted to linear elasticity problems for illustrative purposes. The extension of the present method to material non-linearity (e.g. plasticity, damage, etc.) is straightforward [1]: it simply requires to solve an additional *local* problem, as the method itself corresponds to the resolution of the *global step* (i.e. the global equilibrium of the structure).

This work paves the way of using the FFT-based framework of Moulinec and Suquet [1] in applications that are beyond periodic homogenization, including notably inertia effects in elastodynamic problems of heterogeneous materials, higher-order homogenization (of architectured materials) using higher-order boundary type conditions such as quadratic boundary conditions (QBC), and crack propagation in (finite) heterogeneous microstructures (without numerical artifacts in the propagation due to periodicity [40]).

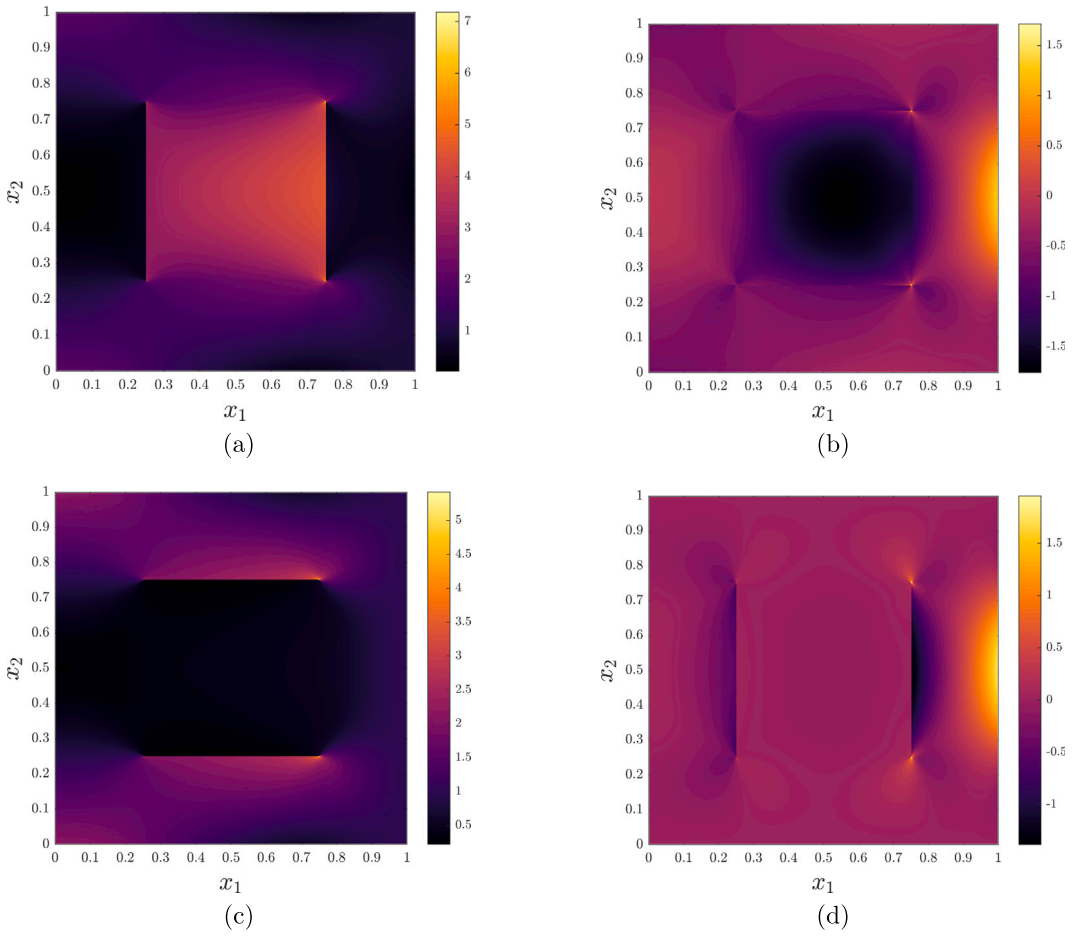

Fig. 7. Distribution of the components of the strain and stress fields in the "tensile test". (a) Strain component ε_{11}, (b) Strain component ε_{22}, (c) Stress component σ_{11} and (d) Stress component σ_{22}.

CRediT authorship contribution statement

Joseph Paux: Writing – review & editing, Writing – original draft, Supervision, Software, Methodology, Investigation, Conceptualization. **Léo Morin:** Writing – review & editing, Writing – original draft, Supervision, Software, Methodology, Investigation, Conceptualization. **Lionel Gélébart:** Writing – review & editing, Writing – original draft, Supervision, Software, Methodology, Investigation, Conceptualization. **Abdoul Magid Amadou Sanoko:** Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Orthogonality of the modes

The weak formulation (38) involves the calculation of elementary integrals which can be of two types, *identical modes* (associated to second order direct partial derivatives) and *cross modes* (associated to cross partial derivatives).

Considering k_1 and k_2 two integers, the elementary integrals associated to identical mode read

$$\begin{cases} \dfrac{2}{L}\displaystyle\int_0^L \sin\left(k_1\frac{x\pi}{L}\right)\sin\left(k_2\frac{x\pi}{L}\right)\mathrm{d}x &= \begin{cases} 0 & \text{if } k_1 \neq k_2 \\ 1 & \text{if } k_1 = k_2 \end{cases} \\ \dfrac{2}{L}\displaystyle\int_0^L \cos\left(k_1\frac{x\pi}{L}\right)\cos\left(k_2\frac{x\pi}{L}\right)\mathrm{d}x &= \begin{cases} 0 & \text{if } k_1 \neq k_2 \\ 1 & \text{if } k_1 = k_2 \neq 0 \\ 2 & \text{if } k_1 = k_2 = 0 \end{cases} \\ \dfrac{2}{L}\displaystyle\int_0^L \sin\left((2k_1+1)\frac{x\pi}{2L}\right)\sin\left((2k_2+1)\frac{x\pi}{2L}\right)\mathrm{d}x &= \begin{cases} 0 & \text{if } k_1 \neq k_2 \\ 1 & \text{if } k_1 = k_2 \end{cases} \\ \dfrac{2}{L}\displaystyle\int_0^L \cos\left((2k_1+1)\frac{x\pi}{2L}\right)\cos\left((2k_2+1)\frac{x\pi}{2L}\right)\mathrm{d}x &= \begin{cases} 0 & \text{if } k_1 \neq k_2 \\ 1 & \text{if } k_1 = k_2 \end{cases} \end{cases} \tag{A.1}$$

Considering again k_1 and k_2 two integers, the elementary integrals associated to cross modes read

$$\begin{cases} \dfrac{2}{L}\displaystyle\int_0^L \cos\left(k_1\frac{x\pi}{L}\right)\sin\left(k_2\frac{x\pi}{L}\right)\mathrm{d}x &= \begin{cases} \dfrac{2k_2}{\pi}\dfrac{(-1)^{k_1+k_2}-1}{k_1^2-k_2^2} & \text{if } k_1 \neq k_2 \\ 0 & \text{if } k_1 = k_2 \end{cases} \\ \dfrac{2}{L}\displaystyle\int_0^L \sin\left((2k_1+1)\frac{x\pi}{2L}\right)\sin\left(k_2\frac{x\pi}{L}\right)\mathrm{d}x &= \dfrac{8}{\pi}\dfrac{(-1)^{k_1+k_2}\,k_2}{(1+2k_1)^2-4k_2^2} \\ \dfrac{2}{L}\displaystyle\int_0^L \cos\left((2k_1+1)\frac{x\pi}{2L}\right)\sin\left(k_2\frac{x\pi}{L}\right)\mathrm{d}x &= -\dfrac{8}{\pi}\dfrac{k_2}{(1+2k_1)^2-4k_2^2} \\ \dfrac{2}{L}\displaystyle\int_0^L \cos\left((2k_1+1)\frac{x\pi}{2L}\right)\cos\left(k_2\frac{x\pi}{L}\right)\mathrm{d}x &= \dfrac{4}{\pi}\dfrac{(-1)^{k_1+k_2}(1+2k_1)}{(1+2k_1)^2-4k_2^2} \\ \dfrac{2}{L}\displaystyle\int_0^L \sin\left((2k_1+1)\frac{x\pi}{2L}\right)\cos\left(k_2\frac{x\pi}{L}\right)\mathrm{d}x &= \dfrac{4}{\pi}\dfrac{1+2k_1}{(1+2k_1)^2-4k_2^2} \\ \dfrac{2}{L}\displaystyle\int_0^L \sin\left((2k_1+1)\frac{x\pi}{2L}\right)\cos\left((2k_2+1)\frac{x\pi}{2L}\right)\mathrm{d}x &= \begin{cases} \dfrac{1+2k_1+(-1)^{1+k_1+k_2}(1+2k_2)}{\pi(k_1-k_2)(1+k_1+k_2)} & \text{if } k_1 \neq k_2 \\ \dfrac{2}{\pi}\dfrac{1}{1+2k_1} & \text{if } k_1 = k_2 \end{cases} \end{cases} \tag{A.2}$$

Appendix B. Detailed calculation of the elementary integrals

The integrals I^x_{lmn}, K^x_{lmn} and I^{x0}_{lmn} involved in the weak formulation of the modified auxiliary problem are calculated as follows.

- The integral I^x_{lmn} reads

$$I^x_{lmn} = \sum_{i=0}^{N_x+1}\sum_{j=0}^{N_y+1}\sum_{k=0}^{N_z+1}\left[-\left(\left(\lambda_0+2\mu_0\right)(\xi_i^{xx})^2+\mu_0\left((\xi_j^{xy})^2+(\xi_k^{xz})^2\right)\right)\alpha_i^{xx}\alpha_j^{xy}\alpha_k^{xz}U^x_{ijk}\times \int_{x=0}^{x=L_x} g^{xx}(k_i^{xx}x)g^{xx}(k_l^{xx}x)\mathrm{d}x\times\int_{y=0}^{y=L_y} g^{xy}(k_j^{xy}y)g^{xy}(k_m^{xy}y)\mathrm{d}y\times\int_{z=0}^{z=L_z} g^{xz}(k_k^{xz}z)g^{xz}(k_n^{xz}z)\mathrm{d}z\right]. \tag{B.1}$$

After calculation of the elementary integrals in Eq. (B.1), it can be shown that

$$I^x_{lmn} = -\frac{L_xL_yL_z}{8}\left(\left(\lambda_0+2\mu_0\right)(\xi_l^{xx})^2+\mu_0\left((\xi_m^{xy})^2+(\xi_n^{xz})^2\right)\right)U^x_{lmn}, \tag{B.2}$$

since

$$\int_{x=0}^{x=L_x} g^{xx}(k_i^{xx}x)g^{xx}(k_l^{xx}x)\mathrm{d}x=\frac{\delta_{il}L_x}{2\alpha_l^{xx}},\quad \int_{y=0}^{y=L_y} g^{xy}(k_j^{xy}y)g^{xy}(k_m^{xy}y)\mathrm{d}y=\frac{\delta_{jm}L_y}{2\alpha_m^{xy}},\quad \int_{z=0}^{z=L_z} g^{xz}(k_k^{xz}z)g^{xz}(k_n^{xz}z)\mathrm{d}z=\frac{\delta_{kn}L_z}{2\alpha_n^{xz}}, \tag{B.3}$$

with δ the Kronecker symbol.

- The integral I^{x0}_{lmn} reads

$$I^{x0}_{lmn} = \int_\Omega\left(\left(\lambda_0+2\mu_0\right)\frac{\partial^2 u_x^0}{\partial x^2}(\mathbf{x})+\mu_0\left(\frac{\partial^2 u_x^0}{\partial y^2}(\mathbf{x})+\frac{\partial^2 u_x^0}{\partial z^2}(\mathbf{x})\right)\right)g^{xx}(k_l^{xx}x)g^{xy}(k_m^{xy}y)g^{xz}(k_n^{xz}z)\mathrm{d}\Omega. \tag{B.4}$$

Assuming that all second derivatives of the function u_x^0 are known analytically, then the integral (B.4) can be simply calculated approximately as

$$I^{x0}_{lmn} = \frac{L_xL_yL_z}{(N_x+1)(N_y+1)(N_z+1)}\left(\mathcal{D}^x_{\mathrm{xyz}}(\boldsymbol{\Delta u_x^0})\right)_{lmn}, \tag{B.5}$$

where $\boldsymbol{\Delta u_x^0}$ is the array of size $(N_x+2)\times(N_y+2)\times(N_z+2)$ containing the values of the function $(\lambda_0+2\mu_0)\frac{\partial^2 u_x^0}{\partial x^2}+\mu_0\left(\frac{\partial^2 u_x^0}{\partial y^2}+\frac{\partial^2 u_x^0}{\partial z^2}\right)$ at the grid points. It must be noted that this only requires the knowledge of the derivatives of u_x^0 at the grid points; consequently, if the analytical expression of the derivatives of the function u_x^0 is not known, the integral I_{lmn}^{x0} can still be determined using an approximation of its derivatives (using finite differences for instance).

- The integral K_{lmn}^x reads

$$K_{lmn}^x = K_{lmn}^{xx} + K_{lmn}^{xy} + K_{lmn}^{xz} \tag{B.6}$$

with

$$K_{lmn}^{xx} = \int_\Omega \frac{\partial \tau_{xx}}{\partial x}(\mathbf{x}) g^{xx}(k_l^{xx}x) g^{xy}(k_m^{xy}y) g^{xz}(k_n^{xz}z)\mathrm{d}\Omega, \tag{B.7}$$

$$K_{lmn}^{xy} = \int_\Omega \frac{\partial \tau_{xy}}{\partial y}(\mathbf{x}) g^{yx}(k_l^{yx}x) g^{yy}(k_m^{yy}y) g^{yz}(k_n^{yz}z)\mathrm{d}\Omega, \tag{B.8}$$

$$K_{lmn}^{xz} = \int_\Omega \frac{\partial \tau_{xz}}{\partial z}(\mathbf{x}) g^{zx}(k_l^{zx}x) g^{zy}(k_m^{zy}y) g^{zz}(k_n^{zz}z)\mathrm{d}\Omega. \tag{B.9}$$

Following [29], the integrals defining K_{lmn}^x are calculated using integration by parts since, in the definition of the auxiliary problem, the known fields are τ_{xx}, τ_{xy} and τ_{xz}, and not their partial derivatives. We only detail hereafter the calculation of K_{lmn}^{xx}:

$$\begin{aligned} K_{lmn}^{xx} = &\int_{y=0}^{y=L_y}\int_{z=0}^{z=L_z} \left(g^{xx}(k_l^{xx}L_x)\tau_{xx}(L_x,y,z) - g^{xx}(0)\tau_{xx}(0,y,z)\right) g^{xy}(k_m^{xy}y) g^{xz}(k_n^{xz}z)\mathrm{d}y\mathrm{d}z \\ &- \xi_l^{xx}\int_\Omega \tau_{xx}(x,y,z)\overline{g^{xx}}(k_l^{xx}x) g^{xy}(k_m^{xy}y) g^{xz}(k_n^{xz}z)\mathrm{d}\Omega. \end{aligned} \tag{B.10}$$

The amount of calculations is reduced by the introduction of the function $\widetilde{\tau}_{xx}$, whose expression depends on the type of boundary conditions associated to u_x on the $x-$faces:

$$\begin{cases} \widetilde{\tau}_{xx}(x,y,z) = \tau_{xx}(x,y,z) & \text{(DD)} \\ \widetilde{\tau}_{xx}(x,y,z) = \tau_{xx}(x,y,z) - \tau_{xx}(0,y,z) - \dfrac{\tau_{xx}(L_x,y,z)-\tau_{xx}(0,y,z)}{L_x}x & \text{(NN)} \\ \widetilde{\tau}_{xx}(x,y,z) = \tau_{xx}(x,y,z) - \tau_{xx}(L_x,y,z) & \text{(DN)} \\ \widetilde{\tau}_{xx}(x,y,z) = \tau_{xx}(x,y,z) - \tau_{xx}(0,y,z) & \text{(ND).} \end{cases} \tag{B.11}$$

The approximate calculation of K_{lmn}^{xx}, using discrete sine–cosine transforms, leads to

$$K_{lmn}^{xx} = \frac{-L_xL_yL_z}{(N_x+1)(N_y+1)(N_z+1)}\left[\xi_l^{xx}\left(D^x_{\bar{\mathrm{x}}\mathrm{yz}}(\widetilde{\boldsymbol{\tau}}_x)\right)_{lmn} + S_{lmn}^{xx}\right], \tag{B.12}$$

where $\widetilde{\boldsymbol{\tau}}_{xx}$ denotes the array of size $(N_x+2)\times(N_y+2)\times(N_z+2)$ containing the values of the function $\widetilde{\tau}_{xx}$ at the grid points. The term S_{lmn}^{xx} is always null, expect in the case $l=0$ of the NN boundary conditions on the x-component of u_x, where it is given by

$$S_{0mn}^{xx} = -\frac{(N_x+1)}{L_x}\left(D^x_{\mathrm{yz}}(\boldsymbol{\tau}_{xx}^0 - \boldsymbol{\tau}_{xx}^{L_x})\right)_{mn}. \tag{B.13}$$

The term K_{lmn}^x finally reads

$$K_{lmn}^x = \frac{-L_xL_yL_z}{(N_x+1)(N_y+1)(N_z+1)}\left[\xi_l^{xx}\left(D^x_{\bar{\mathrm{x}}\mathrm{yz}}(\widetilde{\boldsymbol{\tau}}_{xx})\right)_{lmn} + \xi_m^{xy}\left(D^x_{\mathrm{x}\bar{\mathrm{y}}\mathrm{z}}(\widetilde{\boldsymbol{\tau}}_{xy})\right)_{lmn} + \xi_n^{xz}\left(D^x_{\mathrm{xy}\bar{\mathrm{z}}}(\widetilde{\boldsymbol{\tau}}_{xz})\right)_{lmn} + S_{lmn}^x\right] \tag{B.14}$$

where $\widetilde{\boldsymbol{\tau}}_{xx}$, $\widetilde{\boldsymbol{\tau}}_{xy}$ and $\widetilde{\boldsymbol{\tau}}_{xz}$ respectively denote the arrays of size $(N_x+2)\times(N_y+2)\times(N_z+2)$ containing the values of functions $\widetilde{\tau}_{xx}$, $\widetilde{\tau}_{xy}$ and $\widetilde{\tau}_{xz}$ at the grid points, and S_{lmn}^x is given by

$$S_{lmn}^x = 0 \qquad \text{if } l\neq 0 \text{ and } m\neq 0 \text{ and } n\neq 0 \tag{B.15}$$

$$\begin{aligned} S_{0mn}^x &= -\frac{(N_x+1)}{L_x}\left(D^x_{\mathrm{yz}}(\boldsymbol{\tau}_{xx}^0 - \boldsymbol{\tau}_{xx}^{L_x})\right)_{mn} && \text{if NN on } x-\text{faces} \\ S_{l0n}^x &= -\frac{(N_y+1)}{L_y}\left(D^x_{\mathrm{xz}}(\boldsymbol{\tau}_{xy}^0 - \boldsymbol{\tau}_{xy}^{L_y})\right)_{ln} && \text{if NN on } y-\text{faces} \\ S_{lm0}^x &= -\frac{(N_z+1)}{L_z}\left(D^x_{\mathrm{xy}}(\boldsymbol{\tau}_{xz}^0 - \boldsymbol{\tau}_{xz}^{L_z})\right)_{lm} && \text{if NN on } z-\text{faces,} \end{aligned} \tag{B.16}$$

where $\boldsymbol{\tau}_{xx}^0$, $\boldsymbol{\tau}_{xx}^{L_x}$, $\boldsymbol{\tau}_{xy}^0$, $\boldsymbol{\tau}_{xy}^{L_y}$, $\boldsymbol{\tau}_{xz}^0$ and $\boldsymbol{\tau}_{xz}^{L_z}$ denote the arrays of sizes $(N_y+2)\times(N_z+2)$ for $\boldsymbol{\tau}_{xx}^0$ and $\boldsymbol{\tau}_{xx}^{L_x}$, $(N_x+2)\times(N_z+2)$ for $\boldsymbol{\tau}_{xy}^0$ and $\boldsymbol{\tau}_{xy}^{L_y}$, and $(N_x+2)\times(N_y+2)$ for $\boldsymbol{\tau}_{xz}^0$ and $\boldsymbol{\tau}_{xz}^{L_z}$, containing the values of functions τ_{xx}, τ_{xy} and τ_{xz} on the faces.

Data availability

Data will be made available on request.

References

[1] H. Moulinec, P. Suquet, A numerical method for computing the overall response of nonlinear composites with complex microstructure, Comput. Methods Appl. Mech. Engrg. 157 (1998) 69–94.
[2] L. Gélébart, Amitex, 2022, https://amitexfftp.github.io/AMITEX/index.html.
[3] D.J. Eyre, G.W. Milton, A fast numerical scheme for computing the response of composites using grid refinement, Eur. Phys. J. - Appl. Phys. 6 (1999) 41–47.
[4] J.C. Michel, H. Moulinec, P. Suquet, A computational method based on augmented Lagrangians and fast Fourier transforms for composites with high contrast, Comput. Model. Eng. Sci. 1 (2000) 79–88.
[5] V. Monchiet, G. Bonnet, A polarization-based FFT iterative scheme for computing the effective properties of elastic composites with arbitrary contrast, Internat. J. Numer. Methods Engrg. 89 (2012) 1419–1436.
[6] H. Moulinec, F. Silva, Comparison of three accelerated FFT-based schemes for computing the mechanical response of composite materials, Internat. J. Numer. Methods Engrg. 97 (2014) 960–985.
[7] M. Kabel, T. Böhlke, M. Schneider, Efficient fixed point and Newton–Krylov solvers for FFT-based homogenization of elasticity at large deformations, Comput. Mech. 54 (2014) 1497–1514.
[8] R. Lebensohn, N-site modeling of a 3D viscoplastic polycrystal using Fast Fourier Transform, Acta Mater. 49 (2001) 2723–2737.
[9] R. Brenner, A.J. Beaudoin, P. Suquet, A. Acharya, Numerical implementation of static field dislocation mechanics theory for periodic media, Phil. Mag. 94 (2014) 1764–1787.
[10] N. Bertin, M.V. Upadhyay, C. Pradalier, L. Capolungo, A FFT-based formulation for efficient mechanical fields computation in isotropic and anisotropic periodic discrete dislocation dynamics, Modelling Simul. Mater. Sci. Eng. 23 (2015) 065009.
[11] R. Brenner, Numerical computation of the response of piezoelectric composites using Fourier transform, Phys. Rev. B 79 (2009) 184106, Publisher: American Physical Society.
[12] N. Bilger, F. Auslender, M. Bornert, J.C. Michel, H. Moulinec, P. Suquet, A. Zaoui, Effect of a nonuniform distribution of voids on the plastic response of voided materials: a computational and statistical analysis, Int. J. Solids Struct. 42 (2005) 517–538.
[13] J. Paux, R. Brenner, D. Kondo, Plastic yield criterion and hardening of porous single crystals, Int. J. Solids Struct. 132–133 (2018) 80–95.
[14] M. Schneider, A review of nonlinear FFT-based computational homogenization methods, Acta Mech. 232 (2021) 2051–2100.
[15] S. Lucarini, M.V. Upadhyay, J. Segurado, FFT based approaches in micromechanics: fundamentals, methods and applications, Model. Simul. Mater. Sci. Eng. 30 (2021) 023002, Publisher: IOP Publishing.
[16] L. Gélébart, A simple extension of FFT-based methods to strain gradient loadings-application to the homogenization of beams and plates with linear and non-linear behaviors, J. Theoret. Comput. Appl. Mech. (2022).
[17] T.H. Tran, V. Monchiet, G. Bonnet, A micromechanics-based approach for the derivation of constitutive elastic coefficients of strain-gradient media, Int. J. Solids Struct. 49 (2012) 783–792.
[18] V. Kouznetsova, M.G. Geers, W. Brekelmans, Multi-scale second-order computational homogenization of multi-phase materials: a nested finite element solution strategy, Comput. Methods Appl. Mech. Eng. 193 (2004) 5525–5550.
[19] J. Yvonnet, N. Auffray, V. Monchiet, Computational second-order homogenization of materials with effective anisotropic strain-gradient behavior, Int. J. Solids Struct. 191–192 (2020) 434–448.
[20] R. Sancho, V. Rey-de Pedraza, P. Lafourcade, R.A. Lebensohn, J. Segurado, An implicit FFT-based method for wave propagation in elastic heterogeneous media, Comput. Methods Appl. Mech. Engrg. 404 (2023) 115772.
[21] L. Gélébart, A modified FFT-based solver for the mechanical simulation of heterogeneous materials with Dirichlet boundary conditions, C. R. Mécanique 348 (2020) 693–704.
[22] N.B. Nkoumbou Kaptchouang, L. Gélébart, Multiscale coupling of FFT-based simulations with the LDC approach, Comput. Methods Appl. Mech. Engrg. 394 (2022) 114921.
[23] H. Grimm-Strele, M. Kabel, FFT-based homogenization with mixed uniform boundary conditions, Internat. J. Numer. Methods Engrg. 122 (2021) 7241–7265.
[24] V. Monchiet, G. Bonnet, FFT based iterative schemes for composite conductors with uniform boundary conditions, Eur. J. Mech. A Solids 103 (2024) 105146.
[25] A. Wiegmann, Fast Elliptic Solvers on Rectangular Parallelepipeds, Technical Report, Lawrence Berkeley National Laboratory, 1999.
[26] L. Risthaus, M. Schneider, Imposing different boundary conditions for thermal computational homogenization problems with FFT- and tensor-train-based Green's operator methods, Internat. J. Numer. Methods Engrg. 125 (2024) e7423.
[27] L. Gélébart, FFT-based simulations of heterogeneous conducting materials with combined non-uniform Neumann, periodic and Dirichlet boundary conditions, Eur. J. Mech. - A/Solids (2024) 105248.
[28] L. Morin, J. Paux, A fast numerical method for the conductivity of heterogeneous media with Dirichlet boundary conditions based on discrete sine–cosine transforms, Comput. Methods Appl. Mech. Engrg. 421 (2024) 116772.
[29] J. Paux, L. Morin, L. Gélébart, A discrete sine-cosine transforms galerkin method for the conductivity of heterogeneous materials with mixed Dirichlet/Neumann boundary conditions, Int. J. Numer. Methods Eng. (2024) http://dx.doi.org/10.1002/nme.7615.
[30] F. Willot, B. Abdallah, Y.P. Pellegrini, Fourier-based schemes with modified Green operator for computing the electrical response of heterogeneous media with accurate local fields, Internat. J. Numer. Methods Engrg. 98 (2014) 518–533.
[31] L. Risthaus, M. Schneider, Imposing Dirichlet boundary conditions directly for FFT-based computational micromechanics, Comput. Mech. (2024).
[32] V. Fuka, PoisFFT – A free parallel fast Poisson solver, Appl. Math. Comput. 267 (2015) 356–364.
[33] H. Wang, Y. Zhang, X. Ma, J. Qiu, Y. Liang, An efficient implementation of fourth-order compact finite difference scheme for Poisson equation with Dirichlet boundary conditions, Comput. Math. Appl. 71 (2016) 1843–1860.
[34] F. Caforio, S. Imperiale, A high-order spectral element fast Fourier transform for the Poisson equation, SIAM J. Sci. Comput. 41 (2019) A2747–A2771, Publisher: Society for Industrial and Applied Mathematics.
[35] J. Vondřejc, J. Zeman, I. Marek, An FFT-based Galerkin method for homogenization of periodic media, Comput. Math. Appl. 68 (2014) 156–173.
[36] Z. Wang, Fast discrete sine transform algorithms, Signal Process. 19 (1990) 91–102.
[37] G. Strang, The discrete cosine transform, SIAM Rev. 41 (1999) 135–147.
[38] M. Frigo, S. Johnson, FFTW: an adaptive software architecture for the FFT, in: Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181), vol. 3, 1998, pp. 1381–1384, ISSN: 1520-6149.
[39] J. Michel, H. Moulinec, P. Suquet, A computational scheme for linear and non-linear composites with arbitrary phase contrast, Internat. J. Numer. Methods Engrg. 52 (2001) 139–160.
[40] L. Morin, A. Acharya, Analysis of a model of field crack mechanics for brittle materials, Comput. Methods Appl. Mech. Engrg. 386 (2021) 114061.