INSTITUT DE FRANCE
Académie des sciences

Comptes Rendus

Mécanique

Lionel Gélébart

A modified FFT-based solver for the mechanical simulation of heterogeneous materials with Dirichlet boundary conditions

Volume 348, issue 8-9 (2020), p. 693-704

Published online: 18 November 2020
Issue date: 14 December 2020

https://doi.org/10.5802/crmeca.54

Les Comptes Rendus. Mécanique *sont membres du Centre Mersenne pour l'édition scientifique ouverte*
www.centre-mersenne.org
e-ISSN : 1873-7234

Comptes Rendus
Mécanique
2020, 348, nº 8-9, p. 693-704
https://doi.org/10.5802/crmeca.54

Short paper / *Note*

A modified FFT-based solver for the mechanical simulation of heterogeneous materials with Dirichlet boundary conditions

Lionel Gélébart [a]

[a] CEA Paris-Saclay—Université Paris-Saclay, DES/ISAS/DMN/SRMA, 91191, GIF/YVETTE, France
E-mail: lionel.gelebart@cea.fr

Abstract. Fast Fourier transform based algorithms, relying on the initial idea proposed by Moulinec and Suquet in 1998, are very efficient in the context of periodic homogenization in solid mechanics. The purpose of this short note is to propose a simple modification of these algorithms to extend their application domain from periodic boundary conditions (BC) to Dirichlet BC. The method is validated by a direct comparison with standard finite element simulations with prescribed displacements at the boundary. The convergence properties of the iterative algorithm are then analyzed using a simple example (2D matrix–inclusion) as a function of various parameters (material and algorithm parameters).

Keywords. FFT-based methods, Dirichlet boundary conditions, Heterogeneous materials, Homogenization in solid mechanics.

Manuscript received 18th September 2020, accepted 9th October 2020.

1. Introduction

Fast Fourier transform (FFT) based solvers for mechanics, initially proposed in [1], have attracted increasing interest in the context of simulation of heterogeneous materials. Compared to standard finite element (FE) codes, they are more efficient [2]. Furthermore, they are more efficiently parallelized on distributed memory architectures [3]. However, their use involves periodic boundary conditions (BC), restricting the scope of FFT-based studies on the mechanics of materials. In fact, applying Dirichlet BC (displacement applied on the whole boundary) allows us to address at least two kinds of application. On the one hand, in the context of numerical homogenization of heterogeneous materials, the use of kinematic uniform BC (KUBC) (i.e., a linear displacement field) provides an upper bound for the effective elastic behavior [4–6]. On the other hand, from an experimental point of view, digital imaging techniques combined with digital image correlation provide both the microstructure and the displacement field of a specimen subjected to mechanical loading. This experimental displacement field can be used to apply "realistic" BC at the boundary of an extracted "realistic" subvolume in either 2D [7] or 3D [8].

Taking advantage of efficient FFT-based solvers for solid mechanics to apply Dirichlet BC was addressed in [9]. This was implemented through the use of the so-called explicit jump immersed interface method (EJIIM) based on the application of standard central finite differences, with the simulated domain being embedded in a larger domain subjected to null displacement, and the introduction of "jumps" as additional unknowns. In the present note, a quite different method is proposed. The discretization of equations corresponds to linear (hexahedral) FEs with reduced integration, the embedding part has elastic behavior (with eigenstrain), and the so-called eigendisplacement field is defined at the voxel corners located on the boundary. This eigendisplacement field is unknown and adjusted to fulfill the condition on the displacement prescribed on the boundary (i.e., Dirichlet BC). The linear system is solved through a rather simple modification of the classical fixed-point algorithm [1]. This classical algorithm is combined with the convergence acceleration proposed by Anderson [10], which has been recently applied to efficiently accelerate the same algorithm [3, 11]. This choice, instead of a Krylov linear solver, opens the door toward future extensions to non-linear mechanical behaviors (plasticity, damage, etc.). Indeed, the classical algorithm combined with the convergence acceleration proves to be well suited for both linear and non-linear behaviors [3, 12].

Following a description of the method, a simulation of a simple 2D matrix/inclusion unit cell validates the method through a comparison with a corresponding standard FE simulation. The same unit cell is then used to discuss and analyze the convergence properties of the algorithm.

The notation is as follows. Bold characters correspond to vectors. Parameters σ, τ, and ε are respectively the stress, polarization, and strain symmetric second-order tensors; c, c^B, and c_0 are elastic stiffness fourth-order symmetric tensors (all of them are isotropic in this paper); Γ_0 and G_0 are Green operators acting on polarization fields though a convolution product.

2. Discrete problem, discrete derivations, and FFT-based algorithm with periodic BC

As demonstrated by Schneider [13], the FFT-based method introduced below is strictly equivalent to the FE method with linear hexahedral elements and reduced integration. It is also equivalent to the method proposed by Willot [14], but the presentation is slightly different. This method is quite different from the seminal work of Moulinec [1], which was based on the use of continuous derivatives together with a truncation of the highest Fourier frequencies. Instead, the present method uses discrete derivatives. Hence, this method relies on two grids just like the FE method with nodes for displacements and centers for stresses and strains, whereas the seminal method relies on a single grid of nodes on which all the quantities are defined. As demonstrated in [14], the present FFT-based method improves both the quality of the solution, by reducing spurious oscillations, and the efficiency of the iterative algorithm.

2.1. *Discrete description of unit cell (Figure 1)*

The 3D unit cell Ω is regularly discretized in a 3D image of voxels (rectangular parallelepiped elements equivalent to pixels in 2D). This discretization is associated with a grid of corners and a grid of centers (Figure 1). For a mechanical problem, the material coefficients (e.g., elasticity coefficients) are assumed to be defined at the centers together with the stress σ and the strain ε. These two quantities are related by the constitutive behavior equations (e.g., elastic behavior). On the other hand, the displacement $\boldsymbol{u}$ is defined at the corners together with the stress divergence $\mathbf{div}(\sigma)$.

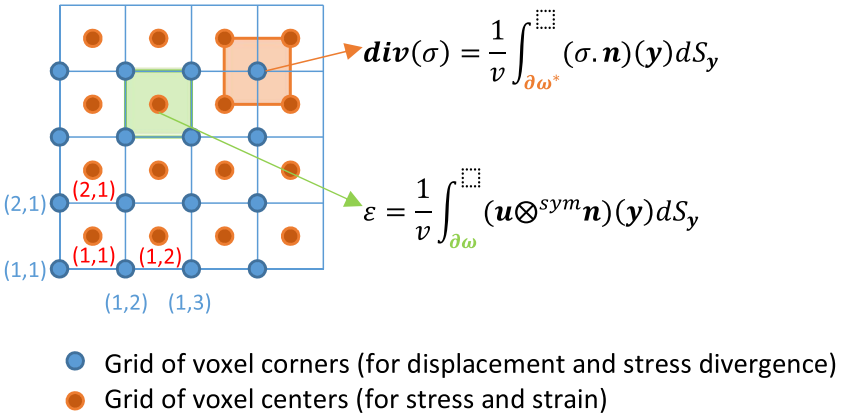

Figure 1. Illustration of unit-cell discretization and schemes used for discrete derivation operators (the two grids of centers and corners have the same size, implicitly assuming their use with periodic BC).

2.2. *FFT evaluation of discrete derivations*

The discrete gradient operator, used to evaluate the strain, results from an average value over the voxel ω (of volume v):

$$\varepsilon = \frac{1}{v}\int_{\omega}\varepsilon(\boldsymbol{y})\,\mathrm{d}V_{\boldsymbol{y}} = \frac{1}{v}\int_{\omega}(\mathrm{grad}(\boldsymbol{u}(\boldsymbol{y})))^{\mathrm{sym}}\,\mathrm{d}V_{\boldsymbol{y}} = \frac{1}{v}\int_{\partial\omega}\left(\boldsymbol{u}\overset{\mathrm{sym}}{\otimes}\boldsymbol{n}\right)(\boldsymbol{y})\,\mathrm{d}S_{\boldsymbol{y}}. \tag{1}$$

In the discrete case, the expression of the strain as a surface integral, defined at the voxel center, is simplified in a discrete sum of the displacement over the eight corners of the voxel (Figure 1). For further details, the reader may consult [15].

The discrete divergence operator is evaluated in a very similar way:

$$\mathbf{div}(\sigma) = \frac{1}{v}\int_{\omega^*}\mathbf{div}(\sigma)(\boldsymbol{y})\,\mathrm{d}V_{\boldsymbol{y}} = \frac{1}{v}\int_{\partial\omega^*}(\sigma.\boldsymbol{n})(\boldsymbol{y})\,\mathrm{d}S_{\boldsymbol{y}}. \tag{2}$$

Here, $\mathbf{div}(\sigma)$ is defined at voxel corners, and ω^* is a translated voxel whose center corresponds to a voxel corner and corners correspond to voxel centers. Hence, the expression of the stress divergence as a surface integral is simplified in a discrete sum of the normal stress ($\sigma\cdot\boldsymbol{n}$) over the eight corners of the translated voxels (i.e., eight voxel centers; see Figure 1).

In the general case, if the size of the grid of centers used to discretize the unit cell is $N_1\times N_2\times N_3$, the size of the grid of corners is $(N_1+1)\times(N_2+1)\times(N_3+1)$. In the case of periodic BC that are implicitly applied when using discrete Fourier transform (DFT) with the FFT algorithm, the last slices of the corners (in each direction) are omitted as they have been already taken into account by the implicit periodic repetitions of the grid. In that case, the two grids (of centers and corners) have the same size and the grid of centers corresponds to the grid of corners translated by the vector of a half-voxel.

The principle of DFT-based derivation is first explained using a 1D grid. The definitions of DFT and inverse DFT are given in (3), and (4) describes the useful translation property of DFT. The parameter N is the grid size, subscripts n and k (with values in $\{0, N-1\}$) refer to variables in real and Fourier spaces, respectively, and $i^2=-1$.

$$\widehat{f_k} = \sum_{n=0}^{N-1} f_n \exp\left(-\frac{2\mathrm{i}\pi kn}{N}\right) \quad f_n = \frac{1}{N}\sum_{k=0}^{N-1}\widehat{f_k}\exp\left(\frac{2\mathrm{i}\pi kn}{N}\right), \tag{3}$$

$$y_n = f_{n\pm1} \quad \widehat{y_k} = \widehat{f_k}\exp\left(\pm\frac{2\mathrm{i}\pi k}{N}\right). \tag{4}$$

Let f_n be the discrete values of a function evaluated on the grid of corners (typically, the displacement). Its discrete derivative Df_n evaluated over the grid of centers (typically, the strain) in real space and then in Fourier space after applying the translation property and some trigonometric manipulations (here, h is the physical size of the grid) reads as

$$Df_n = \frac{1}{h}(f_{n+1} - f_n) \quad \widehat{Df_k} = \mathrm{i}\frac{1}{h}2\sin\left(\frac{\pi k}{N}\right)\widehat{f_k}\exp\left(\frac{\mathrm{i}\pi k}{N}\right). \tag{5}$$

If f_n is evaluated on the grid of centers (typically, the stress) and its discrete derivative Df_n on the grid of corners (typically. the stress divergence), we have

$$Df_n = \frac{1}{h}(f_n - f_{n-1}) \quad \widehat{Df_k} = \mathrm{i}\frac{1}{h}2\sin\left(\frac{\pi k}{N}\right)\widehat{f_k}\exp\left(-\frac{\mathrm{i}\pi k}{N}\right). \tag{6}$$

To summarize, the discrete derivative used in the present paper, depending on where the quantities are defined (corners or centers), reads as

$$\widehat{Df_k} = \mathrm{i}\xi_k^+\widehat{f_k} \quad \text{or} \quad \widehat{Df_k} = \mathrm{i}\xi_k^-\widehat{f_k} \quad \text{with } \xi_k^+\xi_k^- = \widetilde{\xi_k^2}. \tag{7}$$

The extension of these results to 3D grids is quite technical but not difficult. Equation (1) can be written as a sum over the eight corners of the corresponding voxel centers, and as in (5) and (6), the translation property (Equation (4) extended to the 3D case) allows the discrete derivative to be evaluated in Fourier space. Finally, for the displacement evaluated at voxel corners and the gradient evaluated at voxel centers or for the stress evaluated at voxel centers and the stress divergence at voxel corners, we have

$$\widehat{\mathrm{grad}(\boldsymbol{u})}_{\mathbf{k}} = \mathrm{i}\widehat{\boldsymbol{u}}_{\mathbf{k}} \otimes \xi_{\mathbf{k}}^+ \quad \text{and} \quad \widehat{\mathbf{div}(\sigma)}_{\mathbf{k}} = \mathrm{i}\widehat{\sigma}_{\mathbf{k}}.\xi_{\mathbf{k}}^- \quad \text{with } \xi_{\mathbf{k}}^{\pm} = \widetilde{\xi_{\mathbf{k}}}\exp(\pm\mathrm{i}\xi_{\mathbf{k}}.\mathbf{I}/2). \tag{8}$$

Here, $\mathbf{k}$ is a 3D integer vector ($k_i \in \{0, N_i - 1\}$), $\xi_{\mathbf{k}} = \sum_{i=1}^{3}(2\pi k_i/N_i)\boldsymbol{e_i} = \sum_{i=1}^{3}\xi_{k_i}\boldsymbol{e_i}$ ($\boldsymbol{e_i}$ is a vector of the reference orthonormal basis), and $\mathbf{I}$ is the vector $(1,1,1)$. The « modified » frequencies $\widetilde{\xi_{\mathbf{k}}}$ are as follows:

$$\begin{aligned}
\widetilde{\xi_{k_1}} &= \frac{2}{h_1}\cos\left(\frac{\xi_{k_2}}{2}\right)\cos\left(\frac{\xi_{k_3}}{2}\right)\sin\left(\frac{\xi_{k_1}}{2}\right),\\
\widetilde{\xi_{k_2}} &= \frac{2}{h_2}\cos\left(\frac{\xi_{k_1}}{2}\right)\cos\left(\frac{\xi_{k_3}}{2}\right)\sin\left(\frac{\xi_{k_2}}{2}\right),\\
\widetilde{\xi_{k_3}} &= \frac{2}{h_3}\cos\left(\frac{\xi_{k_1}}{2}\right)\cos\left(\frac{\xi_{k_2}}{2}\right)\sin\left(\frac{\xi_{k_2}}{2}\right).
\end{aligned} \tag{9}$$

It must be emphasized that these modified frequencies are equal to the modified frequencies reported in [14]. Instead of using a "trick" that involves rotating the basis to evaluate finite differences, lending the name "rotated scheme" to the method in [14], the present derivation method relies on general expressions, corresponding to surface integrals over a small domain ω, such as proposed equations (1) and (2). These general expressions allow for multiple definitions of first discrete derivatives depending on the choice made for the domain ω (one voxel in the present work); see Figure 11 in [15].

The second-order discrete derivatives are defined here as a succession of two first-order discrete derivatives. After the two opposite transfers from centers to corners and then from corners to centers (or inversely), the translation terms $\exp(\pm\mathrm{i}\xi_{\mathbf{k}}\cdot\mathbf{I}/2)$ vanish.

Finally, as noted in [13], with the choice made for the discrete derivatives (Equations (7) and (8)), the resolution of the problem with the FFT-based algorithm (see below) is equivalent to its resolution with a standard FE code using linear hexahedral elements with reduced integration (one integration point per element).

2.3. *FFT-based algorithm with periodic BC*

The set of equations for a mechanical (elastic) problem to solve, under the small strain assumption, on a unit cell Ω is as follows:

$$\begin{cases} \mathbf{div}(\sigma(\boldsymbol{x})) = 0 \\ \sigma(\boldsymbol{x}) = c(\boldsymbol{x}) : \varepsilon(\boldsymbol{x}) \\ \varepsilon(\boldsymbol{x}) = (\nabla \boldsymbol{u})^{\mathrm{sym}}(\boldsymbol{x}) \\ \boldsymbol{u}(\boldsymbol{x}) = \nabla U \cdot \boldsymbol{x} + \tilde{\boldsymbol{u}}(\boldsymbol{x}) \\ + \text{ boundary conditions on } \partial\Omega. \end{cases} \tag{10}$$

Here, σ, ε, $\boldsymbol{u}$, and c are the stress, strain, displacement, and stiffness tensor fields, respectively. The parameter ∇U denotes the applied average displacement gradient and $\tilde{\boldsymbol{u}}$ is the unknown displacement fluctuation. In the case of periodic BC (on $\partial\Omega$), $\tilde{\boldsymbol{u}}$ is periodic and the traction vector ($\boldsymbol{t} = \sigma \cdot \boldsymbol{n}$) is antiperiodic. In that case, the problem can be solved by using a fixed-point algorithm based on the application of the Green operator Γ_0, which can be advantageously computed in Fourier space. The original fixed-point algorithm proposed in [16] reads as

$$\begin{cases} \tau = (c - c_0) : \varepsilon \\ \tilde{\varepsilon} = -\Gamma_0 * \tau \quad (\rightleftarrows \text{ Fourier}) \\ \varepsilon = \nabla U^{\mathrm{sym}} + \tilde{\varepsilon}. \end{cases} \tag{11}$$

The stiffness tensor c_0 is homogeneous throughout the unit cell and Γ_0 is the periodic Green operator associated with the homogeneous medium of stiffness c_0. To construct Γ_0, the set of equations ($\mathrm{div}(\sigma) = 0$; $\tilde{\varepsilon} = (\mathrm{grad}(\tilde{\boldsymbol{u}}))^{\mathrm{sym}}$; $\sigma = c_0 : \tilde{\varepsilon} + \tau$) must be written in Fourier space and recast to obtain $\widehat{\tilde{\varepsilon}}$ as a function of $\hat{\tau}$ (the reader may refer to the appendix in [1]). The expressions of the discrete divergence and the gradient in Fourier space given in (8) are then used in place of their corresponding continuous expressions from the appendix of [1] for constructing Γ_0.

Following the same idea, a similar Green operator G_0 is defined, relating the displacement fluctuation $\tilde{\boldsymbol{u}}$ to the polarization τ: $\tilde{\boldsymbol{u}} = -\mathrm{G}_0 * \tau$. The parameter G_0 is used in Section 3.2, in algorithm (16), to apply Dirichlet BC.

3. FFT-based algorithm with Dirichlet BC

3.1. *Method*

The strategy used to solve problem (10) with Dirichlet BC ($\tilde{\boldsymbol{u}} = \tilde{\boldsymbol{u}}_{\boldsymbol{d}}$ on $\partial\Omega$) with an FFT-based solver, which assumes periodic BC, consists in defining an elastic buffer zone Ω^{B} around Ω, performing the FFT-based simulation with periodic BC on the larger unit cell Ω^{L} ($= \Omega \cup \Omega^{\mathrm{B}}$), and then prescribing $\tilde{\boldsymbol{u}} = \tilde{\boldsymbol{u}}_{\boldsymbol{d}}$ on $\partial\Omega$, which is now regarded as an internal surface within Ω^{L} (see Figure 2). As the stress/strain fields, solutions of the elastic problem, are independent of any rigid-body translation, the displacement fluctuation can be applied up to a constant, which we fix by setting arbitrarily the average displacement fluctuation field on $\partial\Omega$ to 0:

$$\langle \tilde{\boldsymbol{u}}_{\boldsymbol{d}} \rangle_{\partial\Omega} = 0. \tag{12}$$

The main idea of the method is to (a) define the so-called eigendisplacement fluctuation field $\tilde{\boldsymbol{u}}^*$ on $\partial\Omega$ (defined on the grid of corners), which gives rise to an eigenstrain field ε^* defined at voxel centers in the first layer of voxels in Ω^{B} surrounding $\partial\Omega$ and (b) optimize $\tilde{\boldsymbol{u}}^*$ for fulfilling the Dirichlet BC. In the buffer zone, the elastic behavior reads as

$$\sigma = c^B : (\varepsilon - \varepsilon^*) \quad \text{on } \Omega^{\mathrm{B}}, \tag{13}$$

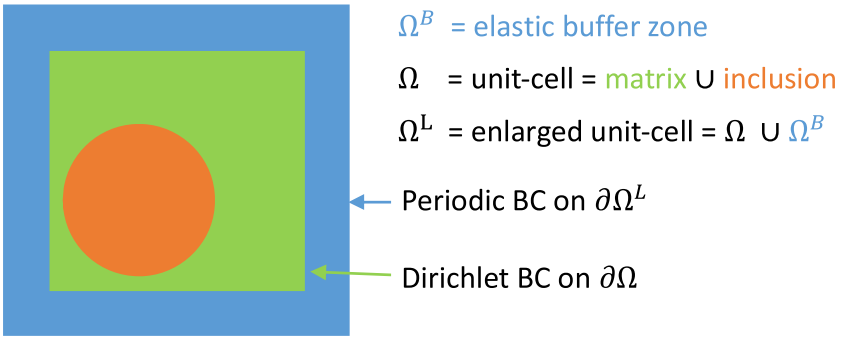

Figure 2. Illustration of the method proposed to apply Dirichlet BC with an FFT-based solver.

where c^B is the homogeneous elastic stiffness tensor of the buffer zone. Note that ε^* is zero except in the first layer of voxels surrounding $\partial\Omega$.

3.2. *Algorithm*

The eigendisplacement fluctuation field $\tilde{\boldsymbol{u}}^*$ is a sum of increments $\delta\tilde{\boldsymbol{u}}^*$ (evaluated from the fixed-point algorithm (16)), which are null everywhere except on $\partial\Omega$, where they are defined as the difference between the applied displacement fluctuation and the current displacement fluctuation (minus their average values to remove rigid-body-translation contributions):

$$\delta\tilde{\boldsymbol{u}}^* = (\tilde{\boldsymbol{u}}_{\boldsymbol{d}} - \langle\tilde{\boldsymbol{u}}_{\boldsymbol{d}}\rangle_{\partial\Omega}) - (\tilde{\boldsymbol{u}} - \langle\tilde{\boldsymbol{u}}\rangle_{\partial\Omega}) \quad \text{on } \partial\Omega. \tag{14}$$

Note that $\langle\tilde{\boldsymbol{u}}_{\boldsymbol{d}}\rangle_{\partial\Omega}$ was arbitrarily set to zero in (12).

The eigenstrain field is then defined on Ω^{B} at voxel centers as

$$\varepsilon^* = (\mathrm{grad}(\tilde{\boldsymbol{u}}^*))^{\mathrm{sym}} \quad \text{on } \Omega^{\mathrm{B}}. \tag{15}$$

Applying the discrete gradient defined in Section 2.2 yields the result that ε^* is null on Ω^{B} except on the first layer of voxels surrounding $\partial\Omega$.

With these definitions in hand, the algorithm detailed in (17) is first introduced in (16). It consists of two iteration steps: step (A) solves the problem on the enlarged unit cell with an eigenstrain field defined on the layer of voxels surrounding $\partial\Omega$ and step (B) defines a new eigenstrain field from the current field and the gap between the displacement fluctuations applied on $\partial\Omega$ and the current displacement fluctuation on $\partial\Omega$ obtained from step (A). The convergence of this simple algorithm is not guaranteed, but it can be improved by using a convergence acceleration technique.

$$\begin{aligned}
&\text{(A)} \begin{cases} \tau = (c - c_0) : (\varepsilon - \varepsilon^*) \\ \tilde{\boldsymbol{u}} = -\mathrm{G}_0 * \tau \quad (\rightleftarrows \text{Fourier}) \\ \varepsilon = \nabla U^{\mathrm{sym}} + (\mathrm{grad}(\tilde{\boldsymbol{u}}))^{\mathrm{sym}}. \end{cases} \\
&\text{(B)} \begin{cases} \delta\tilde{\boldsymbol{u}}^* = (\tilde{\boldsymbol{u}}_{\boldsymbol{d}} - \langle\tilde{\boldsymbol{u}}_{\boldsymbol{d}}\rangle_{\partial\Omega}) - (\tilde{\boldsymbol{u}} - \langle\tilde{\boldsymbol{u}}\rangle_{\partial\Omega}) \\ \varepsilon^* = \varepsilon^* + (\mathrm{grad}(\delta\tilde{\boldsymbol{u}}^*))^{\mathrm{sym}}. \end{cases}
\end{aligned} \tag{16}$$

Finally, a modification of the classical fixed-point algorithm [16] is proposed in algorithm (17) to solve the periodic problem on Ω^{L} with prescribed Dirichlet BC on $\partial\Omega$. The classical algorithm [16] consists of three steps from algorithm (17): (1) evaluation of behavior, (2) application of the Green operator, and (5) evaluation of the equilibrium criterion. Hence, the algorithm modified for applying Dirichlet BC consists of slight modifications of the classical algorithm by the addition

of the following steps: (3) computation of eigendisplacement fluctuation, (4) computation of eigenstrain, and (5) evaluation of the error on Dirichlet BC.

Initialize: $\varepsilon = \nabla U^{\text{sym}}$ on Ω^{L}, $\varepsilon^* = 0$ on Ω^{L}, $\tilde{\boldsymbol{u}}^* = 0$ on $\partial\Omega$
Do while $\text{Err}_\sigma > \text{Err}_{\sigma 0}$ and $\text{Err}_{\text{BC}} > \text{Err}_{\text{BC0}}$

1/ Behavior on Ω^{L}:

$$(\varepsilon, \varepsilon^*) \rightarrow \sigma \rightarrow \tau = \sigma - c_0 : \varepsilon. \tag{17}$$

2/ Green ($\rightleftarrows$ Fourier) on Ω^{L}:

$$\begin{aligned} \tilde{\boldsymbol{u}} &= -G_0 * \tau, \quad \tilde{\varepsilon} = (\text{grad}(\tilde{\boldsymbol{u}}))^{\text{sym}} \\ \varepsilon &= \nabla U^{\text{sym}} + \tilde{\varepsilon} \\ \boldsymbol{u} &= \nabla U \cdot \boldsymbol{x} + \tilde{\boldsymbol{u}}. \end{aligned}$$

3/ Eigendisplacement fluctuation on $\partial\Omega$:

$$\begin{aligned} \delta\tilde{\boldsymbol{u}}^* &= (\tilde{\boldsymbol{u}}_{\boldsymbol{d}} - \langle\tilde{\boldsymbol{u}}_{\boldsymbol{d}}\rangle_{\partial\Omega}) - (\tilde{\boldsymbol{u}} - \langle\tilde{\boldsymbol{u}}\rangle_{\partial\Omega}) \\ \tilde{\boldsymbol{u}}^* &= \tilde{\boldsymbol{u}}^* + \delta\tilde{\boldsymbol{u}}^*. \end{aligned}$$

4/ Eigenstrain on Ω^{B}:

$$\varepsilon^* = (\text{grad}(\tilde{\boldsymbol{u}}^*))^{\text{sym}}.$$

5/ Convergence criteria:

$$\begin{aligned} \text{on } \Omega^{\text{L}}, \quad \text{Err}_\sigma &= \|\mathbf{div}(\sigma)\| / \|\sigma\| \quad \text{(in Fourier space)} \\ \text{on } \partial\Omega, \quad \text{Err}_{\text{BC}} &= \|\delta\tilde{\boldsymbol{u}}\| / \|\boldsymbol{u}\|. \end{aligned}$$

As implemented in the FFT-based code AMITEX_FFTP to accelerate the original fixed-point algorithm [11], the Anderson convergence acceleration procedure is adapted to the present context. To summarize, the Anderson acceleration procedure [10] stores the last N_{ACV} couples (unknown vector U, residual vector R) and, every M_{ACV} iterations, proposes a new unknown vector (instead of the vector proposed by the fixed-point algorithm).

In the present case, the strain components evaluated on the N_{tot} voxel centers of Ω^{L} are arranged in the strain vector U_ε of size $6N_{\text{tot}}$. The eigendisplacement fluctuation components evaluated on the n_{tot} voxel corners defining the contour $\partial\Omega$ are arranged in the eigendisplacement vector $U_{\tilde{\boldsymbol{u}}^*}$ of size $3n_{\text{tot}}$. Residual vectors R_ε and $R_{\tilde{\boldsymbol{u}}^*}$ are constructed in the same manner from $\delta\varepsilon$ and $\delta\tilde{\boldsymbol{u}}^*$ (here, $\delta\varepsilon$ denotes the difference between ε values after and before the application of the Green operator). Vectors $U_{\tilde{\boldsymbol{u}}^*}$ and U_ε on one side and R_ε and $R_{\tilde{\boldsymbol{u}}^*}$ on the other side are then arranged to form the unknown and residual vectors U and R, respectively. Note that as displacements and strains are different quantities, before applying a convergence acceleration procedure to the last N_{ACV} couples of unknown and residual vectors (U, R), the N_{ACV} subvectors $U_{\tilde{\boldsymbol{u}}^*}$ are normalized by the norm of the last stored $U_{\tilde{\boldsymbol{u}}^*}$. The same procedure is adopted for the N_{ACV} subvectors U_ε, R_ε, and $R_{\tilde{\boldsymbol{u}}^*}$.

The "algorithm parameters" influencing the convergence are the stiffness of the reference material c_0, the elastic coefficients and thickness of the buffer zone, and the parameters N_{ACV} and M_{ACV}. The tolerances on the equilibrium and BC criteria are $\text{Err}_{\sigma 0}$ and Err_{BC0}, respectively.

4. Numerical validation and analysis

The choice of c_0, following the rule proposed in [1] ($x_0 = (1/2)(\min(x) + \max(x))$, where x denotes the Lamé coefficients λ and μ), proves to be also relevant to the present algorithm modified for Dirichlet BC.

In our implementation of the convergence acceleration, $N_{\text{ACV}} = 4$. For the simulations of elastic matrix/inclusion considered below, the value $M_{\text{ACV}} = 5$ and the buffer zone behavior

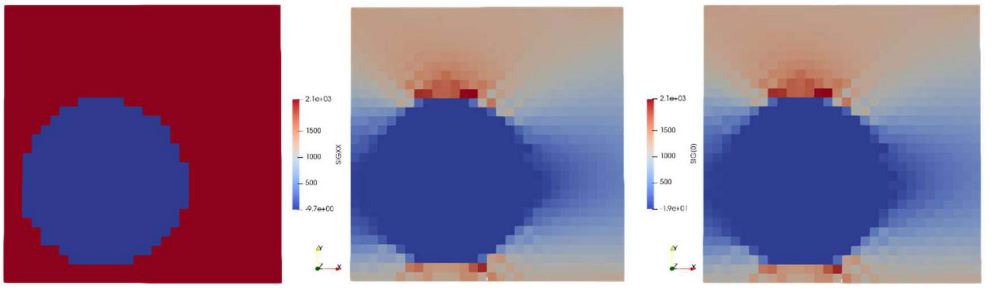

Figure 3. Soft inclusion within a matrix (elastic contrast 10^5) subjected to uniaxial strain (horizontal) with null displacement fluctuations (the so-called KUBC in [4]): discretization (left), axial stress field from the FE (middle), and the FFT-based (right) methods.

given by the stiffest behavior between the matrix and the inclusion are optimized from numerical experiments with respect to convergence. For the sake of conciseness, the results are not reported below. We focus our interest on the mechanical contrast, the applied displacement, the buffer zone thickness, and the spatial resolution. However, note that the buffer zone behavior has an important effect on the convergence and not respecting this "rule" may significantly deteriorate the convergence.

Finally, the tolerances on the criteria of both the equilibrium and the BC are set to 10^{-4}.

4.1. *Validation*

As mentioned in Section 2.2, the FFT-based method proposed in the present paper is equivalent to an FE method using linear elements with reduced integration [13]. Therefore, a direct comparison between them will serve as a validation of the implementation. Since 3D linear elements with reduced integration are unavailable in our FE code (CAST3M [17]), the comparison is performed on a 2D plane-strain simulation with purely uniaxial average strain and null displacement fluctuations ($\boldsymbol{u}(\boldsymbol{x}) = \nabla U \cdot \boldsymbol{x}$ on $\partial\Omega$, where $\nabla U = \varepsilon^0_{11} \boldsymbol{e_1} \otimes \boldsymbol{e_1}$ and $\varepsilon^0_{11} = 0.01$), which corresponds to the so-called KUBC [4]. The 3D FFT-based implementation accounts for a single layer of voxels in the out-of-plane direction. The square unit cell Ω ($D_0 \times D_0$) consists of a disk inclusion of radius $0.3D_0$ with its center being located at $(0.36D_0, 0.36D_0)$ to avoid any symmetry. The discretization is $30 \times 30 \times 1$ voxels for Ω. This configuration is equivalent to a 2D plane-strain simulation. It must be emphasized that the inclusion is quite close to the boundary $\partial\Omega$, where the displacement is prescribed. In addition, the inclusion is very soft with an elastic contrast of 10^5 ($E_m = 10^5$ GPa, $E_i = 1$ GPa, and $\nu_m = \nu_i = 0.3$).

For soft inclusion, the elastic coefficients of the buffer zone are the same as those of the matrix. The thickness of the buffer zone is 5 voxels. Therefore, the size of the enlarged unit cell Ω^{L} is $(30 + 2 \times 5) \times (30 + 2 \times 5) \times 1$.

The comparison of the two axial stress fields in Figure 3 reveals almost perfect agreement between standard FE and FFT-based results up to the precision of the iterative FFT-based method. The range of the color bar, adjusted to the range of the fields, demonstrates that even the extremum values are nicely reproduced. This nice comparison validates our method proposed for applying Dirichlet BC with a FFT-based solver. Note that the spurious oscillations, observed in both simulations, cannot be attributed to any FFT-based artifact (such as the Gibbs phenomenon or aliasing) but rather to an artifact related to the use of linear FE with reduced integration, the *hourglass* [18].

Table 1. Iterations for different elastic contrasts and different boundary conditions

α	∞	10^5	10^2	10^1	10^0	10^{-1}	10^{-3}	10^{-5}	10^{-10}
Dirichlet	46	46	50	50	0	70	110	111	111
Periodic	27	27	25	27	0	16	32	32	32

In spite of quite challenging conditions for the modified FFT-based algorithm (high contrast and proximity of the inclusion to the boundary), convergence is reached after 46 iterations. Section 4.2 provides a deeper convergence analysis.

4.2. *Analysis*

The following analysis relies on the reference simulation described above (same geometry, discretization, elastic stiffness and thickness of the buffer zone, BC, reference material behavior, convergence acceleration parameters, and convergence criteria). The conclusions drawn in the following are limited to similar simulations.

4.2.1. *Mechanical contrast*

The number of iterations for the simulations performed with different elastic contrasts, $\alpha = E_{\text{matrix}}/E_{\text{inclusion}}$, are reported in Table 1 together with those for the simulations with periodic BC (i.e., using the same algorithm without the additional steps for Dirichlet BC and without the buffer zone).

As expected, the convergence of the algorithm is worse with additional conditions to be satisfied (i.e., the Dirichlet BC) than without. This remark holds for different contrasts but with a factor of ∼2 for soft inclusion and a factor of ∼3.5 for stiff inclusion. However, the number of iterations remains reasonable even for very high elastic contrasts.

4.2.2. *Heterogeneous Dirichlet BC*

In the previous simulations, Dirichlet BC are applied with null displacement fluctuations. Here, the axial displacement fluctuations are either random or sinusoidal (other components are null) according to

$$\begin{aligned} \tilde{u}_{d1} &= 0.2\varepsilon^0_{11}\mathrm{D}_0\frac{1}{2}\sin\left(\frac{(x_1+x_2)4\pi}{D_0}\right) - \langle\tilde{u}_{d1}\rangle_{\partial\Omega}, \\ \tilde{u}_{d1} &= 0.2\varepsilon^0_{11}\mathrm{D}_0\,\mathrm{rand} - \langle\tilde{u}_{d1}\rangle_{\partial\Omega}, \end{aligned} \tag{18}$$

where rand is a random generator with values uniformly distributed in $[0,1]$. In both cases, the range of displacement fluctuations covers 20% of the maximum "average" displacement ($\varepsilon^0_{11}\mathrm{D}_0/2$). The axial stress fields simulated with these two types of Dirichlet BC, displayed in Figure 4 for pore inclusion, allow us to visualize their effect. The number of iterations of the simulations with different types of Dirichlet BC are reported in Table 2 for the two extreme cases (pore and very stiff inclusion) and for the homogeneous case. It is striking to observe that the type and the range of the prescribed displacement fluctuations have a very limited effect on the convergence. For the homogeneous case, the algorithm proves to be very efficient with both random and sinusoidal Dirichlet BC. As regards KUBC, the initial guess of the algorithm provides directly the solution (homogeneous strain).

4.2.3. *Convergence evolution*

Up until now, only the number of iterations at convergence was reported to analyze the algorithm performance. The main observations regarding the evolution of the convergence criteria are summarized below.

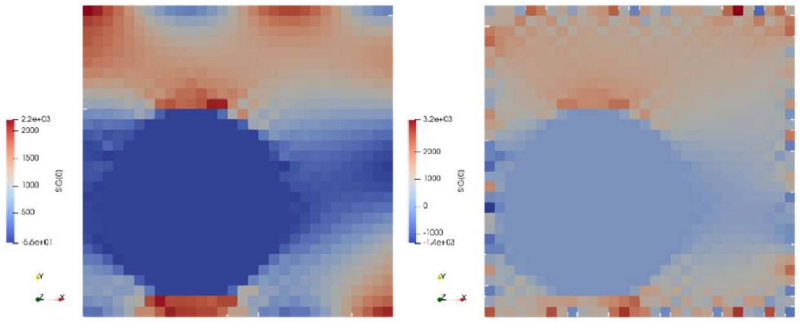

Figure 4. Axial stress fields obtained for pore inclusion with sinusoidal (left) and random (right) Dirichlet BC.

Table 2. Iterations for different Dirichlet BC (null, random, and sinusoidal displacement fluctuations) in two extreme cases (porous and stiff inclusions) and the homogeneous case

	Null (=KUBC)	Random	Sinusoidal
Pore ($\alpha = \infty$)	46	46	52
Homogeneous ($\alpha = 1$)	0	25	20
Stiff inclusion ($\alpha = 10^{-10}$)	111	140	130

For a homogeneous unit cell and non-null Dirichlet BC (sinusoidal for example), the fixed-point algorithm without any convergence acceleration exhibits a monotonic decrease in both the criteria, and convergence is reached after 47 iterations. When using convergence acceleration, convergence decreases monotonically between each acceleration (every set of M_{ACV} iterations, where $M_{\text{ACV}} = 5$). Moreover, each acceleration drastically reduces the criteria up to convergence after 20 iterations.

For soft inclusion, the fixed-point algorithm without acceleration converges monotonically up to an elastic contrast of 2 (with 74 iterations). For higher contrasts, the convergence rate worsens so that acceleration is required to reach convergence. During the $M_{\text{ACV}}(= 5)$ iterations of the fixed point, the two criteria increase or oscillate almost randomly, and each acceleration drastically reduces their values. Considering the criteria evaluated only after accelerations (i.e., every $M_{\text{ACV}}(= 5)$ iterations), the convergence is monotonic.

For stiff inclusion, the same behavior is observed. The non-accelerated fixed-point algorithm converges monotonically up to a contrast of 0.5 (with 46 iterations). For higher contrasts ($\alpha < 0.5$), acceleration is required to reach convergence. However, in this case, even the criteria evaluated only after accelerations (i.e., every $M_{\text{ACV}}(= 5)$ iterations) do not decrease monotonically, which explains the higher number of iterations reported for stiff inclusions compared to soft inclusions in Table 1.

4.2.4. *Buffer zone and spatial resolution*

The thickness of the buffer zone (N_b in voxels) together with the effect of the spatial resolution (N_0 in voxels) is presented in Tables 3 and 4 for the two extreme cases. The sizes of Ω and Ω^L are respectively N_0^2 and $(N_0 + 2N_b)^2$ in the present 2D case.

In spite of the non-monotonic convergence of our algorithm (see Section 4.2.3), some trends can be observed and the following conclusions can be drawn: (a) for a given spatial resolution, the thicker the buffer zone, the better the convergence; (b) for a given thickness of the buffer zone (in voxels), the higher the resolution, the worse the convergence. Ultimately, the ratio N_b/N_0 appears

Table 3. Iterations for various spatial resolution (N_0) and contour thickness (N_b) values: pore inclusion

$N_0 \backslash N_b$	5	10	20	30
30	46	70	45	35
60	195	50	36	40
120	1560	155	50	45

Table 4. Iterations for various spatial resolution (N_0) and contour thickness (N_b) values: stiff inclusion (contrast 10^{-10})

$N_0 \backslash N_b$	5	10	20	30
30	111	70	60	80
60	2465	180	60	50
120	Stop 3000	Stop 3000	205	95

to be a key parameter. Above a given value, the convergence is quite stable. From Tables 3 and 4, this optimal ratio is estimated approximately as 0.2 (5/30) for pore inclusions and 0.33 (10/30) for stiff inclusions.

In practice, the optimization of the ratio N_b/N_0 should take into account not only the number of iterations but also the problem size, which increases as $(N_0 + 2N_b)^2$ in 2D and $(N_0 + 2N_b)^3$ in 3D.

5. Conclusions, discussion, and future prospects

A novel approach has been proposed for applying Dirichlet BC with an FFT-based solver for solid mechanics. The approach relies on (a) an embedding buffer zone surrounding the unit cell, (b) discretization of the equations corresponding to linear FE with reduced integration, (c) a simple modification of the classical fixed-point algorithm [1] with the introduction of the so-called eigendisplacement field at the boundary as an additional unknown, and (d) a convergence acceleration procedure required to attain convergence when the elastic contrast becomes significant.

The approach has been validated by a direct comparison with a mathematically equivalent FE simulation performed using a linear FE with reduced integration. As expected, the solution fields are almost identical up to the precision of the iterative method. Then, the convergence of the method has been analyzed for the case of the 2D matrix/inclusion unit cell used for validation. The method converges with a quite reasonable number of iterations ($\lesssim$100) for both pore and very stiff inclusions with soft inclusions exhibiting better performance. Neither the type (null, random, and sinusoidal) nor the range of the displacement fluctuations applied at the boundary significantly affects the efficiency of the method. Finally, an optimal ratio (between the buffer zone thickness and the unit-cell size) is obtained above which the convergence is quite stable.

The price to pay for applying Dirichlet BC with the modified FFT-based solver when compared to periodic BC with the non-modified solver arises from different points. The number of iterations is higher and the size of the problem with the buffer zone is larger (in addition, each iteration involves additional steps, but this increased cost is probably not predominant). However, the method remains efficient. The method can also be efficiently parallelized, which is a major advantage when compared to standard FE-based codes. In addition, in spite of its reduced performance, this is a convenient method for users to test various BC with the same code and the same inputs.

As a future prospect, a more comprehensive comparison should be conducted with regard to the performance of standard FE-based solvers, especially in 3D. Despite the reduced performance, is the FFT-based method modified for Dirichlet BC still more efficient? Moreover, the applicability of the method will have to be verified and analyzed for non-linear behaviors, for the finite strain framework, and for non-parallelepiped unit cells. The last approach could also benefit from an adaptation of the immersed interface method proposed in [9]. The adaption of the method to mixed BC should also be addressed in the future. Finally, the method will be integrated into the massively parallel FFT-based code AMITEX_FFTP [3].

Acknowledgments

The author would like to acknowledge Yang Chen and Aldo Marano for their kind help with proofreading and fruitful remarks and Noé-Brice Nkoumbou-Kaptchouang for fruitful discussions.

The author also acknowledges the financial support of the Cross-Disciplinary Program on Numerical Simulation by the French Alternative Energies and Atomic Energy Commission (CEA).

References

[1] H. Moulinec, P. Suquet, "A numerical method for computing the overall response of nonlinear composites with complex microstructure", *Comput. Methods Appl. Mech. Eng.* **157** (1998), no. 1–2, p. 69-94.

[2] P. Eisenlohr, M. Diehl, R. A. Lebensohn, F. Roters, "A spectral method solution to crystal elasto-viscoplasticity at finite strains", *Int. J. Plast.* **46** (2013), p. 37-53.

[3] AMITEX_FFTP, http://www.maisondelasimulation.fr/projects/amitex/general/_build/html/index.html.

[4] C. Huet, "Application of variational concepts to size effects in elastic heterogeneous bodies", *J. Mech. Phys. Solids* **38** (1990), no. 6, p. 813-841.

[5] T. Kanit, S. Forest, I. Galliet, V. Mounoury, D. Jeulin, "Determination of the size of the representative volume element for random composites: statistical and numerical approach", *Int. J. Solids Struct.* **40** (2003), no. 13–14, p. 3647-3679.

[6] C. Chateau, L. Gélébart, M. Bornert, J. Crépin, "Micromechanical modeling of the elastic behavior of unidirectional cvi sic/sic composites", *Int. J. Solids Struct.* **58** (2015), p. 322-334.

[7] E. Heripre, M. Dexet, J. Crepin, L. Gélébart, A. Roos, M. Bornert, D. Caldemaison, "Coupling between experimental measurements and polycrystal finite element calculations for micromechanical study of metallic materials", *Int. J. Plast.* **23** (2007), no. 9, p. 1512-1539.

[8] T. T. Nguyen, J. Yvonnet, M. Bornert, C. Chateau, "Modeling of complex microcracking in quasi-brittle materials: Numerical methods and experimental validations", in *Advances in Multi-Physics and Multi-Scale Couplings in Geo-Environmental Mechanics* (F. Nicot, O. Millet, eds.), Elsevier, 2018, p. 171-204.

[9] V. Rutka, A. Wiegmann, H. Andrä, Ejiim for calculation of effective elastic moduli in 3d linear elasticity. Technical Report Berichte des Fraunhofer ITWM, Nr. 93 (2006), Fraunhofer ITWM (2006).

[10] D. G. Anderson, "Iterative procedures for nonlinear integral equations", *J. ACM (JACM)* **12** (1965), no. 4, p. 547-560.

[11] Y. Chen, L. Gélébart, C. Chateau, M. Bornert, C. Sauder, A. King, "Analysis of the damage initiation in a SiC/SiC composite tube from a direct comparison between large-scale numerical simulation and synchrotron X-ray micro-computed tomography", *Int. J. Solids Struct.* **161** (2019), p. 111-126.

[12] A. Marano, L. Gélébart, S. Forest, "Intragranular localization induced by softening crystal plasticity: Analysis of slip and kink bands localization modes from high resolution FFT-simulations results", *Acta Mater.* **175** (2019), p. 262-275.

[13] M. Schneider, D. Merkert, M. Kabel, "FFT-based homogenization for microstructures discretized by linear hexahedral elements", *Int. J. Numer. Methods Eng.* **109** (2017), no. 10, p. 1461-1489.

[14] F. Willot, "Fourier-based schemes for computing the mechanical response of composites with accurate local fields", *C. R. Méc.* **343** (2015), no. 3, p. 232-245.

[15] L. Allais, M. Bornert, T. Bretheau, D. Caldemaison, "Experimental characterization of the local strain field in a heterogeneous elastoplastic material", *Acta Metall.* **42** (1994), no. 11, p. 3865-3880.

[16] H. Moulinec, P. Suquet, "A fast numerical-method for computing the linear and nonlinear mechanical-properties of composites", *C. R. Acad. Sci. Ser. II* **318** (1994), no. 11, p. 1417-1423.

[17] Cast3m, www-cast3m.cea.fr.

[18] T. Belytschko, J. S.-J. Ong, W. K. Liu, J. M. Kennedy, "Hourglass control in linear and nonlinear problems", *Comput. Methods Appl. Mech. Eng.* **43** (1984), no. 3, p. 251-276.