Journal of Computational Physics 321 (2016) 151–168

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A comparative study on low-memory iterative solvers for FFT-based homogenization of periodic media

Nachiketa Mishra [a], Jaroslav Vondřejc [b,a], Jan Zeman [a,*]

[a] Department of Mechanics, Faculty of Civil Engineering, Czech Technical University in Prague, Thákurova 7, 166 29 Prague 6, Czech Republic
[b] Institute of Scientific Computing, Technische Universität Braunschweig, Mühlenpfordtstraße 23, D-381 06 Braunschweig, Germany

ARTICLE INFO

Article history:
Received 9 August 2015
Received in revised form 16 May 2016
Accepted 19 May 2016
Available online 27 May 2016

Keywords:
Numerical homogenization
Fourier–Galerkin method
Fast Fourier Transform
Guaranteed bounds
Richardson iteration
Conjugate gradient algorithm
Chebyshev semi-iterative method
Eyre–Milton scheme

ABSTRACT

In this paper, we assess the performance of four iterative algorithms for solving non-symmetric rank-deficient linear systems arising in the FFT-based homogenization of heterogeneous materials defined by digital images. Our framework is based on the Fourier–Galerkin method with exact and approximate integrations that has recently been shown to generalize the Lippmann–Schwinger setting of the original work by Moulinec and Suquet from 1994. It follows from this variational format that the ensuing system of linear equations can be solved by general-purpose iterative algorithms for symmetric positive-definite systems, such as the Richardson, the Conjugate gradient, and the Chebyshev algorithms, that are compared here to the Eyre–Milton scheme — the most efficient specialized method currently available. Our numerical experiments, carried out for two-dimensional elliptic problems, reveal that the Conjugate gradient algorithm is the most efficient option, while the Eyre–Milton method performs comparably to the Chebyshev semi-iteration. The Richardson algorithm, equivalent to the still widely used original Moulinec–Suquet solver, exhibits the slowest convergence. Besides this, we hope that our study highlights the potential of the well-established techniques of numerical linear algebra to further increase the efficiency of FFT-based homogenization methods.

1. Introduction

Various experimental and simulation techniques, such as serial sectioning [1], computed tomography [2], statistical reconstruction [3], or digital models [4] are currently available to characterize microstructures of heterogeneous materials in a degree of realism not possible before. When combined with the tools of homogenization theories, e.g. [5–7], these advances have made it possible to establish the structure-property relations of complex engineering materials across length scales ranging from micrometers to tens of centimeters. The scale transitions rely on the solution of the *corrector problem* — a boundary value problem defined on a representative cell of the material, typically involving periodic boundary conditions. Since the input data are provided in the form of pixel- or voxel-based geometries, the need therefore arises for efficient solvers that employ images as discretization grids. Although several finite element or finite difference solvers have been developed for this purpose (e.g. [8–10]) methods based on the Fast Fourier Transform (FFT) generally offer the best computational efficiency, because of the regular grid, the simple shape of the computational domain, and the periodic boundary conditions.

* Corresponding author.
E-mail addresses: nachiketa.mishra@fsv.cvut.cz (N. Mishra), vondrejc@gmail.com (J. Vondřejc), zemanj@cml.fsv.cvut.cz (J. Zeman).

http://dx.doi.org/10.1016/j.jcp.2016.05.041
0021-9991/

In the field of computational micromechanics of materials, the first FFT-based homogenization solver was proposed by Moulinec and Suquet in 1994 [11] and more than twenty years later, it is still widely used because of its simplicity and computational speed. The crux of the method is to reformulate the corrector problem as an integral equation of the Lippmann–Schwinger type solved by fixed-point iterations, while taking advantage of the fact that the kernel action can be efficiently handled using FFT. Later extensions of the basic algorithm were driven by the need to (i) *accelerate its convergence* for high-contrast problems [12–16]; (ii) increase *accuracy of local fields* by incorporating inclusion shapes [17], modified kernels [18,19], or local smoothing of coefficients [20,21]; and (iii) *prove the convergence of approximate solutions* in the framework of spectral collocation methods [22–24], the Galerkin discretization of the non-classical Hashin–Shtrikman functionals with piecewise-constant approximation spaces [25,26], and standard Fourier–Galerkin methods [23].

Apart from providing theoretical justification to the original scheme, the Fourier–Galerkin setting has also been found convenient from the numerical point of view. For instance, it has clarified the effects of numerical quadrature [27], and led to the development of fully explicit guaranteed error bounds on homogenized properties based on a primal–dual variational approach [23,28], which were later shown to be more restrictive than the corresponding Hashin–Shtrikman bounds [29]. The purpose of this paper is to complement these studies by examining the performance of four low-memory iterative methods for solving linear systems associated with the Fourier–Galerkin discretizations. Our comparison involves general-purpose short-recurrence solvers, namely the Richardson scheme [30], the Conjugate gradient method [31], the Chebyshev semi-iteration [32], together with the Eyre–Milton algorithm [12] — the most efficient of the accelerated schemes developed specifically for FFT-based homogenization problems, according to the recent study [33].

Related work. Previous comparative studies on FFT-based homogenization algorithms fall into two categories. The aim of the first group of works is to compare their results with finite element solvers for material-specific applications, such as particle-reinforced composites with elasto-plastic phases [34], visco-plastic models of polycrystalline materials [35–37], or transport processes and creep in concrete-like materials [38,39]. Results of these studies consistently reveal that FFT-based methods offer at least an order-of-magnitude improvement in the computational time while predicting very similar distributions of local fields. The second group of studies is dedicated to accelerated schemes, namely to benchmarking their computational performance for high-contrast problems [40] and to revealing that they can be derived from a common recurrence relation [33].

Contributions. Although considerable effort has been spent on benchmarking FFT-based algorithms, neither of the studies above addresses conventional iterative solvers for symmetric positive-definite systems, the applicability of which follows naturally from the Fourier–Galerkin setting [41,23]. We aim to fill this gap while utilizing the standard techniques and results of numerical linear algebra. In particular, we discuss in detail the (i) eigenvalue distribution of the system matrix, (ii) effects of numerical integration, and reduction in (iii) algebraic errors and (iv) guaranteed bounds on homogenized properties during iterations. To the best of our knowledge, this is the first study addressing such aspects for FFT-based homogenization solvers.

Limitations. Because our goal is to provide basic insight into the behavior of the different linear solvers for FFT-based homogenization, we restrict our attention to the two-dimensional scalar linear elliptic problems with isotropic phases, moderate contrasts in coefficients, and discretizations not exceeding $\approx 3{,}000{,}000$ unknowns (corresponding to a $1{,}999 \times 1{,}999$ pixel image). We also do not provide details about the overall computational time, since all simulations were performed with an experimental Python-based code FFTHomPy, available at https://github.com/vondrejc/FFTHomPy, that is not optimized for speed. However, because our observations are based on well-established results of numerical linear algebra, they extend directly to more involved applications of FFT-based homogenization solvers reported in the literature, as evidenced by recent contributions [42–44].

Organization of the paper. The remainder of the manuscript is organized as follows. The essentials of the Fourier–Galerkin discretization of the periodic corrector problem are briefly reviewed in Section 2 following our more detailed expositions [23,28,27]. In Section 3, we provide details for the linear iterative solvers considered in this study. Results of the numerical experiments are gathered in Section 4, and the paper is concluded with the summary of the most important findings in Section 5.

Notation. We will denote d-dimensional vectors and matrices by boldface letters, e.g. $\boldsymbol{a} = (a_\alpha)_{\alpha=1,\dots,d} \in \mathbb{R}^d$ or $\boldsymbol{A} = (A_{\alpha\beta})_{\alpha,\beta=1,\dots,d} \in \mathbb{R}^{d\times d}$. The Euclidean inner product will be referred to as $(\bullet,\bullet)_{\mathbb{R}^d}$ and the corresponding norm as $\|\bullet\|_{\mathbb{R}^d}$. By $\mathbb{R}^{d\times d}_{\mathrm{spd}}$, we will refer to the space of symmetric positive-definite $d\times d$ matrices.

Vectors and matrices arising from discretization on regular grids will be denoted by the bold serif font in order to highlight their special structures. In particular, for a parameter $\boldsymbol{N} \in \mathbb{N}^d$ related to the discretization along each coordinate and an index set $\mathbb{Z}^d_{\boldsymbol{N}}$ enumerating the degrees of freedom, see ahead to (7) for the exact specification, we use

$$\mathbf{a}_{\boldsymbol{N}} = \left(a^{\boldsymbol{k}}_\alpha\right)^{\boldsymbol{k}\in\mathbb{Z}^d_{\boldsymbol{N}}}_{\alpha=1,\dots,d} \in \mathbb{R}^{d\times \boldsymbol{N}}, \qquad \mathbf{A}_{\boldsymbol{N}} = \left(A^{\boldsymbol{km}}_{\alpha\beta}\right)^{\boldsymbol{k},\boldsymbol{m}\in\mathbb{Z}^d_{\boldsymbol{N}}}_{\alpha,\beta=1,\dots,d} \in [\mathbb{R}^{d\times \boldsymbol{N}}]^2.$$