Wavelet based solution of flow and diffusion problems in digital materials

A. Arad[1] M. Madadi[2] A. G. Christy[3]
M. Hegland[4]

(Received 31 January 2011; revised 3 August 2011)

Abstract

The computation of physical properties in a digital materials laboratory requires significant computational resources. Due to the complex nature of the media, one of the most difficult problems to solve is the multiphase flow problem, and traditional methods such as Lattice Boltzmann are not attractive as the computational demand for the solution is too high. A wavelet based algorithm reduces the amount of information required for computation. Here we solve the Poisson equation for a large three dimensional data set with a second order finite difference approximation. Constraints and fictitious domains are used to capture the complex geometry. We solve the discrete system using a discrete wavelet transform and thresholding. We show that this method is substantially faster than the original approach and has the same order of accuracy.

http://anziamj.austms.org.au/ojs/index.php/ANZIAMJ/article/view/3960 gives this article, © Austral. Mathematical Soc. 2011. Published August 11, 2011. ISSN 1446-8735.

Contents

1 Introduction

Tomographic imaging can now be routinely performed over three orders of magnitude in length scale with correspondingly high data fidelity. This capability, coupled with the development of advanced computational algorithms for image interpretation, three dimensional visualisation, structural characterisation and computation of physical properties from image data, allows for a new numerical laboratory approach to the study of real complex materials: the Digital Materials Laboratory [3].

The computation of physical properties in a digital materials laboratory requires significant computational resources. For large 3D data sets, the number of data points is of the order of 8 billion; even for a simple binary image, we can benefit from a more efficient numerical approach to solve the problem. Due to the complex nature of the media, one of the most difficult problems to solve is the multiphase flow problem, and traditional methods such as Lattice Boltzmann are not attractive as the computational demand for the solution is too high. A wavelet based algorithm reduces the amount of data needed for the computation. In the next section, we present our method for solving PDEs, illustrated with the Laplace equation, and 3D digital images. Sections 3–6 presents computational results, and Section 7 finishes the paper with a general discussion and outlook.

2 Description of the method

We solve a discrete version of Poisson's equation with Dirichlet boundary conditions. The domain is obtained from a micro-CT scan of a large rectangular (2D) or cuboidal (3D) domain. It is given by a complex system of intermediate spaces (pores) between solid rock. To solve the PDE on these porous media, we use the fictitious domain method [2]. The algorithm is presented in the following subsection.

2.1 Fictitious domain method

The total volume of the 3D images (pores and solid) is $\Omega = \Omega_1 \cup \Omega_2$, where Ω_1 is the pore space domain on which the PDE will be solved and Ω_2 is the solid. From this we get a natural but slightly non-standard formulation of the Dirichlet problem on Ω:

$$-\Delta u(x) = f(x), \quad x \in \Omega_1 , \quad \text{and} \quad u(x) = 0 , \quad x \in \Omega_2 . \tag{1}$$

Here, Δ is the Laplacian. The fictitious domain Ω is open whereas Ω_2, which contains the boundary of Ω_1, is closed. Note that the usual fictitious domain method works with "internal boundary conditions" $u(x) = 0$ on the boundary $\partial\Omega_1$. While it seems at first sight that choosing to satisfy the boundary conditions on the larger set Ω_2 is unnecessarily complicated, it turns out that this does not cause additional work and has the advantage that the complex shaped boundary $\partial\Omega_1$ does not need to be treated explicitly.

The problem has a natural discretisation based on the voxels of the scan, where the solution $u(x)$ of the Poisson problem is characterised by one value per voxel. One obtains the discrete Laplacian from the standard seven point stencil approximation of the Laplacian. The (discrete) fictitious domain is of the form $\mathcal{D} = \{0, \ldots, n_1 - 1\} \times \{0, \ldots, n_2 - 1\} \times \{0, \ldots, n_3 - 1\}$. Then $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2$, where the $\mathcal{D}_i$ are the grid points of $\mathcal{D}$ which are contained in Ω_i. Denote the discrete solution by $\vec{u} \in \mathbb{R}^{\mathcal{D}}$. It will be shown that this $\vec{u}$ is an approximation of $u(x)$ for $x \in \mathcal{D}$. Let $B \in \mathbb{R}^{\mathcal{D}\times\mathcal{D}}$ be the matrix modelling the discrete version of the constraint of $u(x) = 0$ in Ω_2. Then, the discrete solution has to satisfy the condition that it is zero on $\mathcal{D}_2$, that is,

$$B\,\vec{u} = 0\,. \tag{2}$$

Note that B is diagonal and symmetric, and $B^2 = B$. We denote the feasible space by $N(B)$ (the null space of B).

Both theoretically and for the derivation of numerical algorithms, it is most convenient to formulate the discrete Dirichlet problem as a quadratic optimisation problem with linear constraints. Consider the quadratic functional

$$J(\vec{u}) = \frac{1}{2}\vec{u}^T A\vec{u} - \vec{f}^T\vec{u}\,.$$

Let $\vec{u} \in N(B)$ satisfy

$$J(\vec{u}) \leqslant J(\vec{v}) \quad \text{for all } \vec{v} \in N(B). \tag{3}$$

Using Lagrange multipliers $\vec{p} \in \mathbb{R}^{\mathcal{D}}$ one then obtains a saddle point problem

$$\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \vec{u} \\ \vec{p} \end{bmatrix} = \begin{bmatrix} \vec{f} \\ 0 \end{bmatrix}. \tag{4}$$

As $B^T = B$, the vector $B^T\vec{p}$ has only nonzero elements with indices in $\mathcal{D}_2$. It follows that the components in $\mathcal{D}_1$ of the residual $A\vec{u}-\vec{f}$ are zero. Furthermore, the components of $\vec{u}$ in $\mathcal{D}_2$ are also zero.

The algorithm we consider here for the solution of saddle point problems is an *inexact Uzawa algorithm* [2] defined by the iteration

$$\begin{aligned} \vec{u}^{(k+1)} &= \vec{u}^{(k)} + \rho\,(\vec{f} - A\,\vec{u}^{(k)} - B^T\vec{p}^{(k)}), \\ \vec{p}^{(k+1)} &= \vec{p}^{(k)} + \omega\,B\,\vec{u}^{(k+1)}, \end{aligned} \tag{5}$$

where ρ and ω control the convergence of the Uzawa method and have to be chosen such that good convergence is obtained. While, ultimately, a rational choice of these parameters requires extensive error analysis, they can be chosen based on some test problems and by monitoring the convergence.

If we start this iteration with $\vec{u}^{(0)} = \vec{p}^{(0)} = 0$, nonzero values of $\vec{u}^{(k)}$ can occur only for points in $\mathcal{D}_1$ and for points in $\mathcal{D}_2$ which are adjacent to points in $\mathcal{D}_1$. Consequently $B\vec{u}^{(k)}$, and thus $\vec{p}^{(k)}$, can be nonzero only on the points of $\mathcal{D}_2$ which are adjacent to $\mathcal{D}_1$. These points are just the boundary points of $\mathcal{D}_1$ and so one gets a natural representation of the boundary.

An implementation of this version of the Uzawa method requires a sparse storage of the $\vec{u}^{(k)}$ and $\vec{p}^{(k)}$. The operators A and B are stored as sparse matrices acting on sparse vectors. For such an implementation, the complexity of the method is not proportional to the size of the fictitious domain $\mathcal{D}$ but to the size of the actual domain $\mathcal{D}_1$. The next subsection discusses how to further reduce the complexity using wavelets.

2.2 Wavelets

We start with a brief discussion of wavelets. Wavelets are basis functions that are localised in both physical space and wavenumber space. Localisation in physical space occurs due to the compact support, localisation in frequency space occurs due to vanishing moments and smoothness. Consider the analogous case of the classical Fourier transform which has sine and cosine as basis functions that are well localised in wavenumber space, but do not provide localisation in physical space. While the Fourier transform behaves well for continuous functions, the lack of localisation means that it provides poor approximations for images containing discontinuities. Wavelets have four fundamental properties that are highly desirable:

1. a local basis that is
2. easily refined,
3. they are fast to compute and
4. provide a good approximation using only a few terms.

The number of operations required to compute the wavelet transform is $O(N)$. The initial motivation for using a wavelet basis was due to a simple visual inspection; we see a highly heterogeneous structure and as such choosing a basis structure that can capture this information in a compressed way is extremely desirable. A second reason is that using a typical mesh construction of our finite element or finite difference scheme results in a very large linear system of equations whose solution requires a lot of computational power. This is why we study the use of a multilevel approximation scheme (the wavelet transform) on top of the discrete equations. Multilevel methods are a natural choice for flow in porous media and have been used in connection with stochastic models of the pore domain [1].

We demonstrate that the multilevel wavelet approach considered here allows efficient approximation, has a natural interpretation and admits a fast solution. The only disadvantage of wavelets that we have encountered is that they require domains of very regular shape. We had to use cube shaped domains with linear dimensions equal to a power of two in our analysis, and hence had to split our images into subdomains, the largest of which was 512^3 in size.

3 Wavelets for compression

A quick initial study shows the power of using wavelets in Micro-CT image data. Our micro-CT images contain up to 2000^3 voxels, but many of these represent slowly varying parts of the image with little information content. From such data, we typically derive a binary image that distinguishes grain space and pore space, which is subsequently used to model physical properties [4]. The Haar wavelet transform is applied to the binary images to various subsets of the same data set. The Haar wavelet is also the simplest possible wavelet. The Haar wavelet transform is essentially made up of two components. A low pass filter of $1/2$, $1/2$ that acts as the scaling function and a high pass filter of $1/2$, $-1/2$ that allows for the smooth compression. One component of the Haar wavelet is

$$h(x) = \begin{cases} 1, & x \in (0, \frac{1}{2}), \\ -1, & x \in (\frac{1}{2}, 1), \\ 0, & \text{elsewhere}. \end{cases} \tag{6}$$

Figure 1(a) shows a 2D slice through one of our CT images, showing the geometrical complexity. Figures 1(b)–(c) show for two different subvolumes that the number of Haar transform components falls rapidly with increasing magnitude of wavelet coefficient. Hence, much of the information is contained in only a small number of wavelet components. The Haar compression is performed by setting to zero any wavelet coefficient that is initially less than 0.01. We observed that for larger samples, wavelet compression is even more effective. The Haar transform is efficient at capturing the sharp boundaries in the binarized image using a small number of wavelets. Since small subvolumes have larger surface area-to-volume ratios than larger subvolumes, there is less smoothly varying volume available for compression in the smaller subsets. For the Haar transform of a 64^3 subvolume, about 2.5×10^5 wavelet components have magnitude less than one, as shown in Figure 1(b), while for the transform of the 512^3 cube, 1.3×10^8 components are less than one, as shown in Figure 1(c).

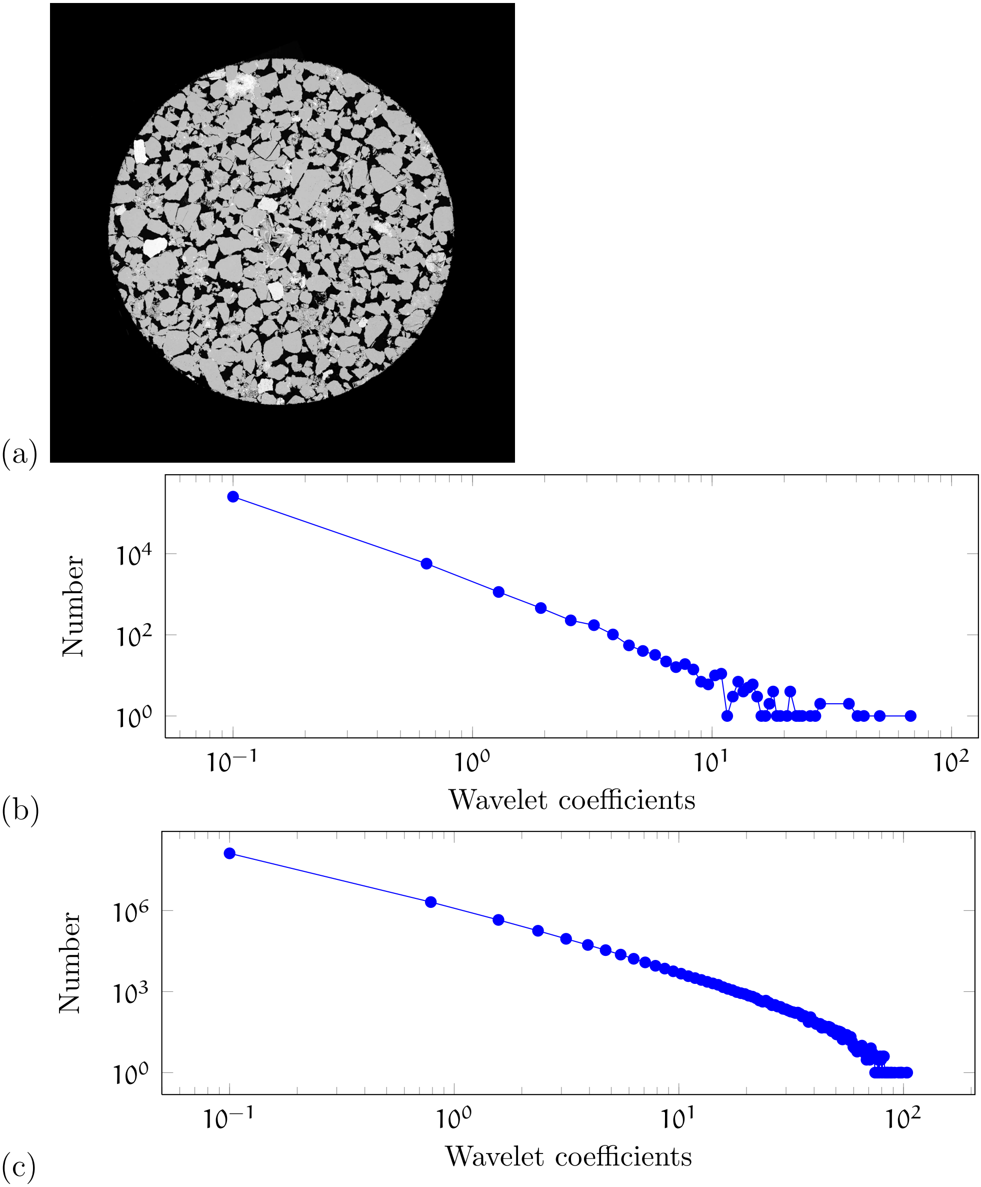


Figure 1: (a) a 2D slice from a micro-CT image of a real sandstone sample. (b) Haar compression for a 64^3 subvolume and (c) a 512^3 subvolume.

4 Operators in wavelet space

What follows is a brief discussion on an efficient construction of the finite difference operator and how we apply the boundary conditions. Here we wish to construct the seven stencil discrete Laplace operator. Let A be the stiffness matrix, then, for the one dimensional case, A has a size of N by N. Similarly, for the 2D and 3D cases, A has dimensions N^2 by N^2 and N^3 by N^3, respectively. One very efficient method for creating the 3D Laplace operator is to use the Kronecker product. We begin with the discrete Laplace operator in 1D,

$$A = A_{1D} = \begin{bmatrix} 2 & -1 & 0 & \cdots \\ -1 & 2 & -1 & \cdots \\ 0 & -1 & 2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{7}$$

whose sparsity pattern is shown in Figure 6(a). To produce, the 2D operator we use the formula $A_{2D} = A \otimes I + I \otimes A$, where I is the identity matrix, and A is the 1D Laplace operator. Similarly for the 3D Laplace operator, we use

$$A_{3D} = A \otimes I \otimes I + I \otimes A \otimes I + I \otimes I \otimes A. \tag{8}$$

We ultimately wish to use the Lagrange multiplier methodology to apply boundary conditions. However, to have a standard to compare against, we first apply the boundary conditions to the operator by removing the row and column and placing a “1” on the diagonal and the known value of “0” on the right hand side of $AU = F$. The elements of A are stored in a sparse format to save memory and increase efficiency. At this point we solve the quadratic minimisation problem with boundary conditions by applying the conjugate gradient method. We compare any subsequent solution to this reference solution. As previously described, the images are segmented into pore and grain space binary images. Ultimately, we wish to apply this methodology to the solution of fluid flow problems, hence we set our boundary conditions in such a way that the grain space has right hand side 0, indicating ‘no flow’,

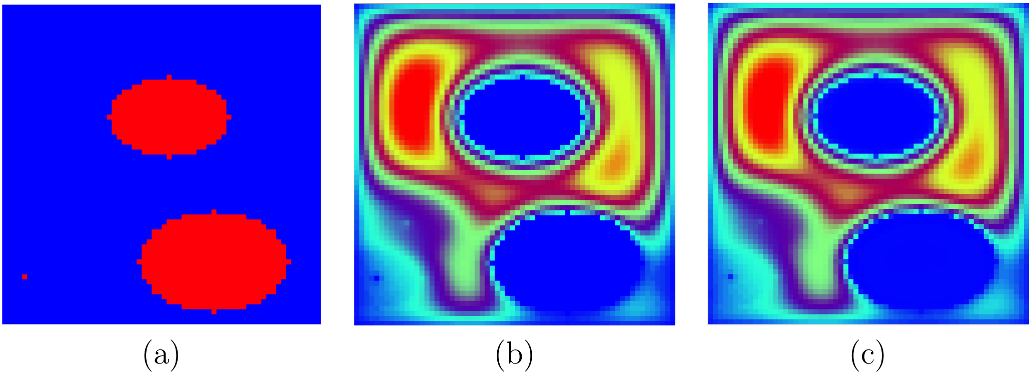


Figure 2: (a) 2D slice of 64^3 sample. (b) Exact Laplace solution for potential. Colour contours are potential which varies from blue (zero) to red (maximum). (c) Solution transformed from wavelet space.

while pore space has right hand side 1, indicating 'normal flow'. In essence, this amounts to a Poisson equation with complicated boundary conditions. A much more efficient way of implementing the boundary conditions is to use the fictitious domain approach, where Lagrange multipliers are used as a sort of penalty term to enforce the boundary conditions on our domain. We note that this fictitious domain implementation is particularly efficient for the Galerkin scheme, where the construction of the stiffness matrix requires no integration. Although it is very computationally efficient to formulate the problem using this approach, there is a drawback. Looking at the block representation shown in Equation (4), it becomes obvious that rather than a quadratic equation, we have a saddle point problem which cannot be solved by the conjugate gradient method.

5 Case study

Now that we have constructed our system and shown that it can be solved, we show how the solution is computed efficiently using Uzawa's algorithm.

Algorithm 1: implement Uzawa's iteration. Here the bar indicates a wavelet transform.

```
1. Assume $\bar{p}_{k=0}$, and $\bar{u}_{k=0}$ equal to zero.
2. Update $\bar{u}_{k+1} \longleftarrow \bar{u}_k + \rho\,(\bar{f} - \bar{A}\bar{u}_k - \bar{B}^T\bar{p}_k)$.
3. Update $\bar{p}_{k+1} \longleftarrow \bar{p}_k + \omega\bar{B}\bar{u}_k$.
4. $k \longleftarrow k + 1$.
5. if $k > m$ then stop, otherwise go to step 2.
```

Let $u(x)$ denote the reference solution and let $\vec{u}(x)$ denote the numerical solution computed using the Uzawa algorithm; thus, the relative error is

$$\text{Error} = \frac{|\vec{u}(x) - u(x)|}{|u(x)|}\,. \tag{9}$$

Algorithm 1 implements Uzawa's iterative scheme in wavelet space as defined by Equation (5).

We start our computational experiments with an artificial granular model of random ellipsoids. A 2D slice of this sample is shown in Figure 2(a). The porosity is about 0.75, and the grain network is not connected. This is clearly a poor example of a rock structure; however, we use it here as a case study for showing the efficacy of the method. The computer simulation ran 100 iterations of the Uzawa algorithm. The inner solver is a preconditioned conjugate gradient which also ran for 100 iterations. Starting with the 64^3 sample, we obtain the reference solution shown in Figure 2(b) and the Uzawa solution shown in Figure 2(c). The error as derived in Equation (9) is calculated at each step of Uzawa's algorithm and plotted for three subsets of the artificial sample. Figure 3(a) shows the number of nonzero values of $\vec{u}$, Figure 3(b) shows the error against the number of iterations, and Figure 3(c) presents the error versus percentage of nonzero values for three different subvolumes of size 32^3, 64^3 and 128^3. For a given error, a larger sample has a smaller percentage of nonzero wavelet coefficients, and hence is compressed further than a smaller subvolume. Conversely, for a given number of iterations, the relative error is greater for a larger subvolume, so a larger number of iterations is required for adequate convergence of a larger data set, although the resulting compression is better.

6 Thresholding

We are now ready to apply the wavelet scheme to our problem. We apply the $W^{\mathsf{T}}AW$ wavelet transform on the operator in one dimension prior to forming the Kronecker product, thus the wavelet transform is carried into the higher dimensions. For this study, we solve the Laplace equation for a 64^3 subvolume of the same artificial sample in Figure 2(a) (but from a slice of different height z) with similarly complex pore structure. However, while employing the Uzawa iteration, we apply a thresholding compression on the solution space at each step of the iteration. The solution at each step is in wavelet space, so to achieve a valid comparison to the reference solution, we apply the inverse wavelet transform. In a normal application of the scheme, this inverse transform will not be performed until convergence in wavelet space is reached. Further, thresholding is only done on $u(x)$ in the solution space. In future work, we will report on the effects of compression on the Lagrange multipliers and the operators. Figure 4 shows the converged solution for compressions of 0%, 50% and 90%. The Lagrange multiplier function p is presented in Figure 4. If the compression becomes too high, there appears to be bleeding into the grain space (Figure 4(f)). Hence, 50% compression appears to be optimal for this Haar wavelet transform. Figure (5) shows the behaviour of the error (9) as a function of the iteration numbers; the sample space is the same one shown in Figure 2(a). The results show that with 50% compression, convergence is achieved after only 20 iterations, approximately five times faster than with no compression. For compression greater than 50%, a systematic error becomes apparent due to the bleeding across the grain boundaries as noted above.

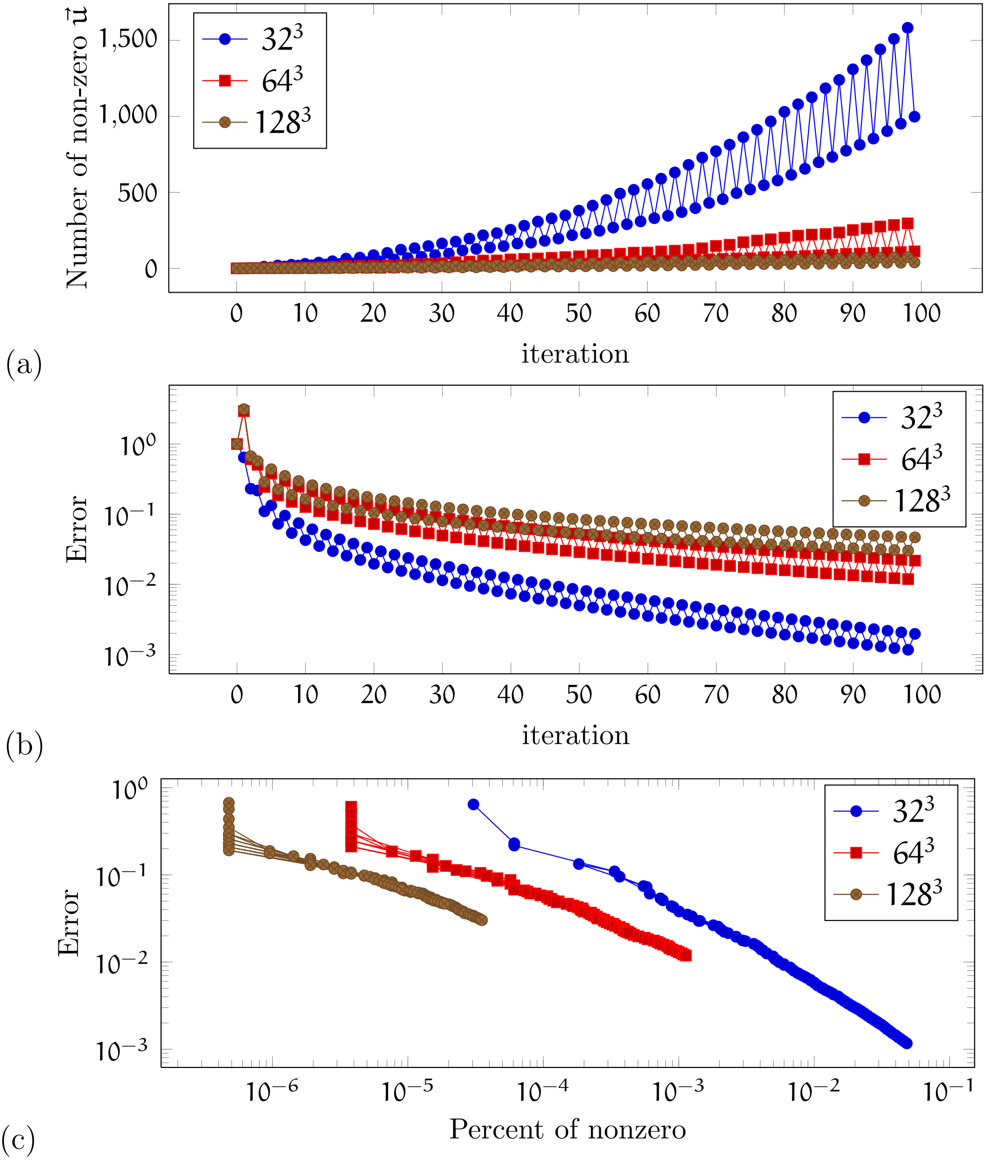
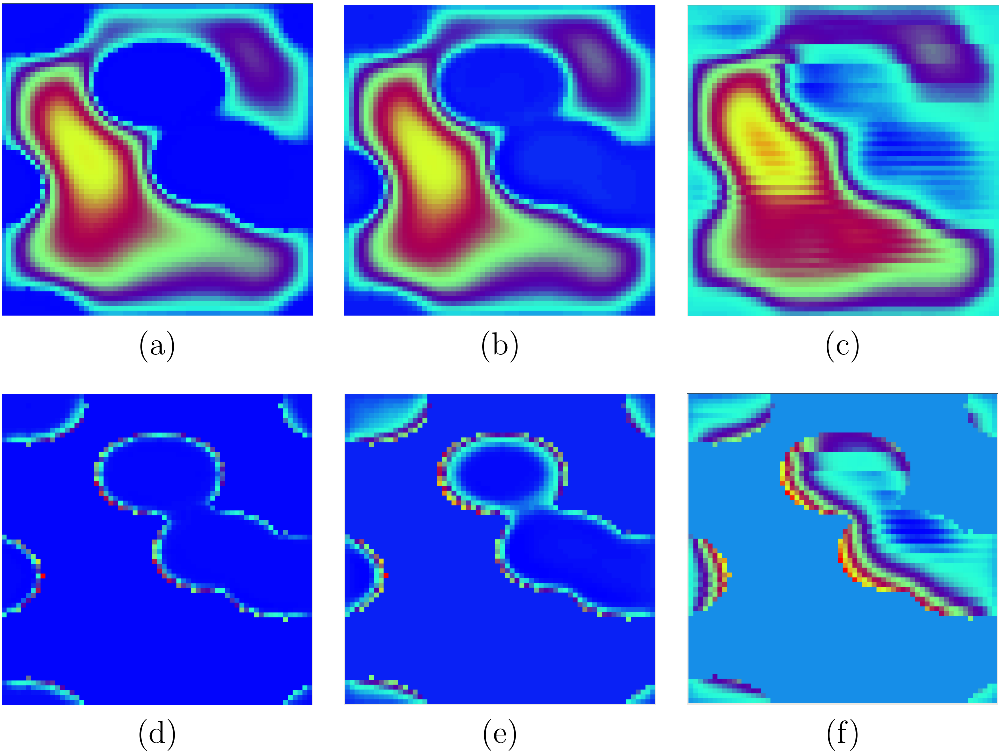


FIGURE 3: (a) Number of nonzero elements of $\vec{u}$. (b) Relative error of Uzawa algorithm. (c) Error versus percentage of nonzero elements.



(a) (b) (c)

(d) (e) (f)

Figure 4: The top plots show the potential for different compression, and bottom graphs show the boundary value parameter, $\vec{\mathsf{p}}$. The compression is 0%, 50% and 90% from left to right.

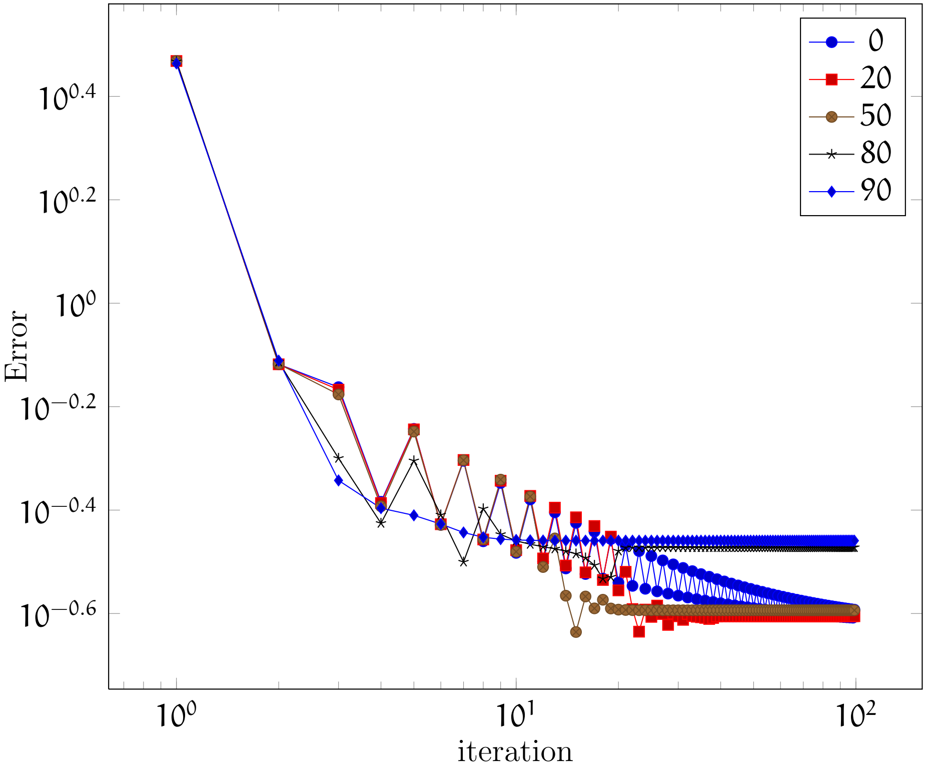
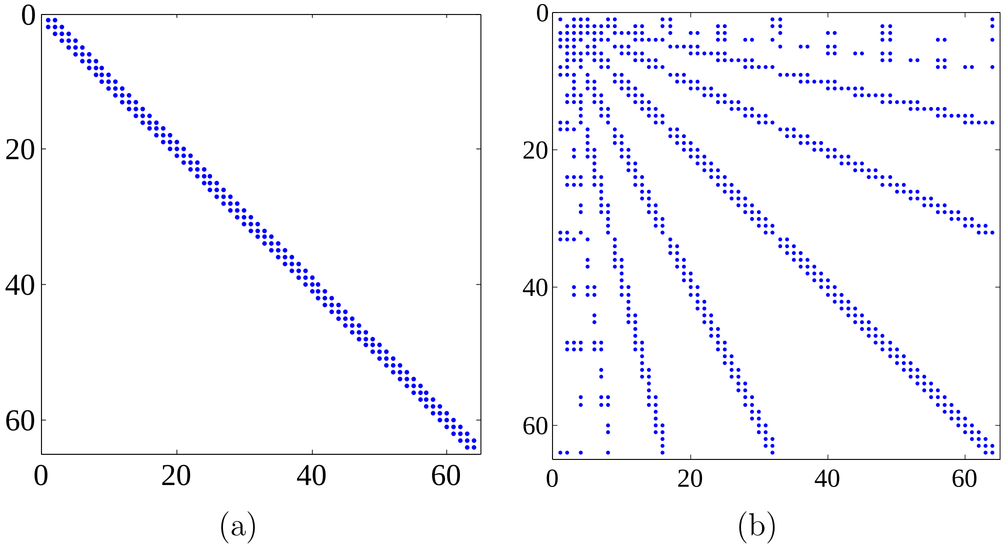


FIGURE 5: Relative error in $\vec{u}$ for different compressions, as a function of iteration number.

7 Discussion and conclusion

It is also possible to perform compression on the operator. A discretisation of the operator A with wavelet bases results in quasi-sparse matrices: the system matrix becomes finger band structured in wavelet coordinates. Figure 6(a) shows a standard discretisation of the Laplace operator as derived from a second order finite difference three point stencil approximation for the one dimensional case, whereas Figure 6(b) shows the clear finger band structure



(a)

(b)

Figure 6: (a) Discrete form of the finite difference Laplace operator; coloured boxes indicate nonzero matrix elements. (b) Operator in wavelet space.

of the same operator after it has undergone a Haar wavelet transform.

We first conclude that the fictitious domain approach works. Secondly, larger systems are more compressible, both geometrically and for PDE solutions. We found that a 50% thresholding allows for a factor of five speed increase in solving the PDE. Third, wavelet bases are such that we get nonphysical results for compressions greater than 50%. We aim to investigate this behaviour further using other wavelet bases.

In future work, we aim to apply compression to the operator by discarding the non-relevant matrix entries. The sparsity pattern will be chosen carefully by a level dependent compression strategy so that we achieve an optimal order of convergence. A diagonal scaling yields a well conditioned linear system of equations.

References

[1] Mark A. Knackstedt, Shane Latham, Mahyar Madadi, Adrian Sheppard, Trond Varslot, and Christoph Arns, Digital rock physics: 3D imaging of core material and correlations to acoustic and flow properties, *The Leading Edge* **28** (2009), no. 1, 28–33. C683

[2] Angela Kunoth, Wavelet techniques for the fictitious domain Lagrange multiplier approach, *Numer Algorithms* **27** (2001), 291–316, doi:10.1023/A:1011891106124 C680, C682

[3] Arthur Sakellariou, Christoph H. Arns, Adrian P. Sheppard, Rob M. Sok, Holger Averdunk, Ajay Limaye, Anthony C. Jones, Tim J. Senden, and Mark A. Knackstedt, Developing a virtual materials laboratory, *Mater Today* **10** (2007), no. 12, 44–51. C679

[4] Adrian P. Sheppard, Rob M. Sok, and Holger Averdunk, Techniques for image enhancement and segmentation of tomographic images of porous materials, *Physica A* **339** (2004), no. 1–2, 145–151. C684

Author addresses

1. **A. Arad**, Research School of Earth Sciences, Australian National University, Canberra, Australia.
   mailto:Alon.Arad@anu.edu.au

2. **M. Madadi**, Department of Applied Mathematics, Research School of Physics and Engineering, Australian National University, Canberra, Australia.
   mailto:Mahyar.Madadi@anu.edu.au

3. **A. G. Christy**, Centre for Advanced Microscopy, Australian National University, Canberra, Australia.
   mailto:Andrew.Christy@anu.edu.au

4. **M. Hegland**, Mathematical Sciences Institute, College of Physical and Mathematical Sciences, Australian National University, Canberra, Australia.
mailto:Markus.Hegland@anu.edu.au