ESAIM: M2AN 48 (2014) 449–474
DOI: 10.1051/m2an/2013115

ESAIM: Mathematical Modelling and Numerical Analysis
www.esaim-m2an.org

A MULTISCALE MODEL REDUCTION METHOD FOR PARTIAL DIFFERENTIAL EQUATIONS

Maolin Ci[1], Thomas Y. Hou[1] and Zuoqiang Shi[2]

Abstract. We propose a multiscale model reduction method for partial differential equations. The main purpose of this method is to derive an effective equation for multiscale problems without scale separation. An essential ingredient of our method is to decompose the harmonic coordinates into a smooth part and a highly oscillatory part so that the smooth part is invertible and the highly oscillatory part is small. Such a decomposition plays a key role in our construction of the effective equation. We show that the solution to the effective equation is in H^2, and can be approximated by a regular coarse mesh. When the multiscale problem has scale separation and a periodic structure, our method recovers the traditional homogenized equation. Furthermore, we provide error analysis for our method and show that the solution to the effective equation is close to the original multiscale solution in the H^1 norm. Numerical results are presented to demonstrate the accuracy and robustness of the proposed method for several multiscale problems without scale separation, including a problem with a high contrast coefficient.

Mathematics Subject Classification. 35J15, 65N30.

Received August 3, 2013.
Published online February 20, 2014.

1. Introduction

A broad range of scientific and engineering problems involve partial differential equations (PDE) with multiscale solutions. Due to the large range of scales in these solutions, it is extremely challenging to solve them numerically. Tremendous computational resources are required to solve for the small scales of the solution, which makes it prohibitively expensive. Thus finding an effective equation which governs the large scale solution is very important in many engineering applications. When the problem has scale separation and a periodic structure, the classical homogenization theory provides a powerful tool to derive an effective equation. However, in many engineering applications, the problems usually do not satisfy the scale separation assumption or may not have periodic structures. In this case, it is very difficult to derive an effective equation since the coupling between the small scale solution and the large scale solution is in general nonlinear and nonlocal even if the governing equation is linear.

Keywords and phrases. Model reduction, effective equation, multiscale PDE, harmonic coordinates, decomposition.

[1] Applied and Comput. Math, Caltech, Pasadena, CA 91125, USA. mci@cms.caltech.edu; hou@cms.caltech.edu
[2] Math Science Center, Tsinghua Univ, Beijing 100084, China. zqshi@math.tsinghua.edu.cn

Article published by EDP Sciences

© EDP Sciences, SMAI 2014

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

In this paper, we propose a multiscale model reduction method which aims to derive an effective equation for multiscale problems without scale separation. The tool that we use is the harmonic coordinates, which satisfy the same governing equation with homogeneous source term and linear boundary conditions. An important ingredient of our method is to design an appropriate decomposition of the harmonic coordinates, denoted as F, into a large scale component g plus a small scale component χ. To obtain an accurate effective equation, we require that g be invertible and χ be small. We propose one effective decomposition in which g at the coarse grid nodal points is defined to be the local average of F near those nodal points and we interpolate g from the coarse mesh to the entire domain by using a linear interpolation. Once g is defined, χ is determined uniquely as $\chi = F - g$. One advantage of our approach is that our effective equation can be solved on a regular coarse mesh by any conventional numerical method. This makes the computational implementation very easy.

We have also performed error analysis for our method. We show that the difference between the effective solution and the original solution can be bounded in the H^1 norm by the maximum norm of the oscillatory part of the harmonic coordinates. In general, the oscillatory part of the harmonic coordinates is small. The smallness of the oscillatory part will depend on the regularity of the multiscale coefficient of the problem. In the case when the problem has scale separation and periodic structure, our method recovers the homogenized equation from the classical homogenization theory and the oscillatory part of the harmonic coordinates can be proved to be small in the H^1 norm. One important advantage of our method is that we do not require the problem to have scale separation or periodic structures. This enables us to apply our method to solve more challenging problems arising from various physical applications.

We use the following elliptic equation as an example to illustrate the main idea of our approach

$$\begin{cases} -\nabla \cdot (a(x)\nabla u(x)) = f(x) & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega. \end{cases} \tag{1.1}$$

The multiscale information is described by the multiscale coefficient $a(x)$. We assume that $f(x) \in L^2(\Omega)$ is smooth and $a(x) \in L^\infty(\Omega)$ is a symmetric, positive definite matrix satisfying $\lambda_{min}(x) \geq \gamma > 0$ ($\lambda_{min}(x)$ is the smallest eigenvalue of $a(x)$) for all $x \in \Omega$. For such coefficients, the solutions are only Hölder continuous. If a is highly oscillatory, the solution will become highly oscillatory as well. We would like to design an effective equation in the following form

$$\begin{cases} -\nabla \cdot (a^*(x)\nabla u^*(x)) = f(x) & \text{in } \Omega \\ u^* = 0 & \text{on } \partial\Omega. \end{cases} \tag{1.2}$$

The key is how to construct a smooth effective coefficient a^* so that the solution to the above effective equation approximates the original multiscale solution with some desirable accuracy. As we mentioned above, we will decompose the harmonic coordinates F into a smooth part g plus a highly oscillatory part χ, $F = g + \chi$. Our method does not require F to be invertible. But to motivate the derivation of our method, we assume temporarily that F is invertible. Then we can express u as a function of F. One important property of the harmonic coordinates is that u as a function of F is about one order smoother than u as a function of x (see *e.g.* [29]). Thus, we can write $u(x) = \tilde{u}(F) = \tilde{u}(g + \chi)$, and formally expand $\tilde{u}$ around g by assuming that χ is small. By taking the leading order terms and substituting them into the original equation, we obtain an effective equation of form (1.2) after ignoring the high order terms involving χ. The effective coefficient a^* is defined in terms of a, g, and χ.

One of the main contributions of this paper is that we can show that the effective equation derived by the above formal analysis indeed has the desirable smoothness property. Under some conditions, we will show that the solution to the effective equation is in H^2, which is one order smoother than the original multiscale solution. Thus we can solve the effective equation on a coarse mesh. Moreover, we can show that the error term is small in the H^1 norm under some conditions. From our derivation, we can see that the decomposition of the harmonic coordinates determines the effective coefficient, a^*. An optimal effective coefficient will determine an optimal decomposition. The relationship between these two terms helps us to design a nearly optimal decomposition of F.

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

Our method falls into the category of global upscaling methods. To obtain our effective equation, we need to first solve for the harmonic coordinates, which amounts to solving the original equation n times (n is the space dimension of the problem). If we just solve the elliptic equation once, our method would not save computational cost. But if we need to solve the equation with the same a many times but with different source terms, f, the cost of constructing the effective coefficient is a small overhead in the offline step. The online step of solving the effective equation with multiple right hand sides gives considerable computational saving since the effective equation can be solved on a coarse mesh while the original multiscale problem must be solved on a fine mesh. For time dependent problems such as parabolic, hyperbolic, and convection-diffusion equations, if the coefficients are time-independent, our method gives considerable computational saving even for a single forcing since the overhead of constructing the effective coefficient is negligible compared with the cost of solving the time dependent equations on a fine mesh at the subsequent time steps.

There have been many multiscale methods in the literature, see *e.g.* [2,4,5,10–14,16–24,29,30]. Among these methods, the metric based upscaling method of Owhadi and Zhang [29] shares some common feature with our approach since both methods use harmonic coordinates (see also [1,3] for more discussions on harmonic coordinates). However, there are also some important differences between these two methods. In the metric based upscaling method, they assume that the harmonic map is invertible and use the harmonic map to construct a multiscale basis for the original equation by first transforming the original problem from the physical coordinates to the harmonic coordinates. They then use a standard linear finite element basis in the harmonic coordinates, and transform them from the harmonic coordinates back to the physical coordinates to obtain their multiscale basis. The numerical implementation of their method is more complicated than our method since the coarse mesh in the metric based upscaling method is severely deformed due to the transformation of the harmonic mapping. In our approach, we are interested in deriving a global upscaling equation without the need of constructing local multiscale basis. Moreover, we do not require the harmonic map to be invertible and our effective equation can be solved on a regular coarse mesh using any standard method. This makes our method easier to be implemented and more efficient.

The rest of this paper is organized as follows. In Section 2 we derive our effective equations for several types of equations. Section 3 is devoted to error analysis of our method. We compare our method with homogenization in Section 4. In Section 5, we discuss some numerical implementation issues and present some numerical results to demonstrate the accuracy and effectiveness of our method. Some concluding remarks are given in Section 6.

2. Derivation of effective equations

2.1. Elliptic equations

In this section, we present the derivation of our effective equations. We first consider the elliptic equation (1.1). Let $F(x) = (F_1(x), \dots, F_n(x))$ be the harmonic coordinates associated to (1.1) in dimension n. Then F_k $(k = 1, \dots, n)$ satisfies the following elliptic equation

$$\begin{cases} -\nabla \cdot (a(x)\nabla F_k(x)) = 0 & \text{in } \Omega \\ F_k = x_k & \text{on } \partial\Omega, \end{cases} \tag{2.1}$$

where $x = (x_1, \dots, x_n)$. Write $\tilde{u}_0 = u \circ F^{-1}$. It is known that the solution u is smooth in terms of the harmonic coordinates, *i.e.* $\tilde{u}_0$ is in H^2 (see *e.g.* [29]). If we could make a decomposition $F = g + \chi$ such that g is smooth and χ is small with zero boundary conditions, then we obtain by applying a formal Taylor expansion to $\tilde{u}_0$ and ignoring the high order terms,

$$\tilde{u}_0(F) = \tilde{u}_0(g + \chi) \approx \tilde{u}_0(g) + \chi^T \nabla_g \tilde{u}_0(g). \tag{2.2}$$

Let $u_0(x) = \tilde{u}_0(g(x))$, then we get

$$u(x) = \tilde{u}_0(F) \approx u_0(x) + \chi^T \frac{\partial x}{\partial g} \nabla u_0(x). \tag{2.3}$$

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

Furthermore, we have

$$\nabla u(x) \approx \nabla u_0(x) + \frac{\partial \chi}{\partial x}\frac{\partial x}{\partial g}\nabla u_0(x) + \chi^T \nabla \left(\frac{\partial x}{\partial g}\nabla u_0(x)\right). \tag{2.4}$$

By substituting (2.4) into (1.1) and eliminating the high order terms involving $O(\chi)$, we get a new PDE for u_0

$$\begin{cases} -\nabla \cdot \left(a(x)\left(I + \frac{\partial \chi}{\partial x}\frac{\partial x}{\partial g}\right)\nabla u_0(x)\right) = f(x) & \text{in } \Omega \\ u_0 = 0 & \text{on } \partial\Omega, \end{cases} \tag{2.5}$$

where I is the identity matrix.

We will show in the next section that u_0 is in H^2 so that we can solve (2.5) accurately on a coarse mesh. Moreover, we can show that the H^1 norm of the error, $u-(u_0+\chi^T\frac{\partial x}{\partial g}\nabla u_0)$, is small. Thus we can approximate u by $u_0+\chi^T\frac{\partial x}{\partial g}\nabla u_0$ with a reasonable accuracy. This suggests the following steps of the model reduction method.

1. Solve (2.1) on a fine mesh to get F.
2. Decompose $F = g + \chi$, here g is smooth and χ is small with $\chi = 0$ on $\partial\Omega$.
3. Solve (2.5) on a coarse mesh to get u_0.
4. Approximate u by $u_0 + \chi^T\frac{\partial x}{\partial g}\nabla u_0$.

The first and second steps are 'offline' steps. We can store the information of g and χ so that we could compute u_0 efficiently for different f. The remaining 'online' steps can be solved very efficiently on a coarse mesh. So far we have not defined what we mean by g being 'smooth' and χ being 'small'. We will discuss the guideline in defining g and χ, and give one effective construction later in the paper.

2.2. Parabolic equation

We can apply the similar idea to derive an effective equation for time-dependent equations, such as parabolic and convection-diffusion equations. We first consider a parabolic equation of the form

$$\begin{cases} u_t - \nabla \cdot (a(x)\nabla u(x)) = f(x) & \text{in } (0,T]\times\Omega \\ u = 0 & \text{on } \partial\Omega \\ u = 0 & \text{at } t = 0. \end{cases} \tag{2.6}$$

We can define the harmonic coordinates F in exactly the same way as we did for the elliptic equation. We then decompose $F = g + \chi$ and solve the following effective equation on a coarse mesh

$$\begin{cases} (u_0)_t - \nabla \cdot (a(x)(I + \frac{\partial \chi}{\partial x}\frac{\partial x}{\partial g})\nabla u_0(x,t)) = f(x) & \text{in } (0,T]\times\Omega \\ u_0 = 0 & \text{on } \partial\Omega \\ u_0 = 0 & \text{at } t = 0. \end{cases} \tag{2.7}$$

Again, we approximate u by $u_0 + \chi^T\frac{\partial x}{\partial g}\nabla u_0$.

2.3. Convection-diffusion equation

Next, we consider the convection-diffusion equation with multiscale velocity field (see also [27])

$$\begin{cases} v_t + u \cdot \nabla v = \nabla \cdot (\alpha \nabla v) & \text{in } (0,T]\times\Omega \\ v = 0 & \text{on } \partial\Omega \\ v = \phi(x) & \text{at } t = 0, \end{cases} \tag{2.8}$$

where u is a velocity field which satisfies $\nabla \cdot u = 0$ and α is a positive diffusion coefficient.

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

We define the corresponding harmonic coordinates as follows

$$\begin{cases} u \cdot \nabla F_k = \nabla \cdot (\alpha \nabla F_k) & \text{in } \Omega \\ F_k = x_k & \text{on } \partial\Omega. \end{cases} \tag{2.9}$$

By decomposing $F = g + \chi$ as before, we obtain the following effective equation

$$\begin{cases} (v_0)_t + u \cdot \nabla v_0 = \nabla \cdot \left(\left(\alpha \frac{\partial F}{\partial x} - u\chi^T \right) \frac{\partial x}{\partial g} \nabla v_0 \right) & \text{in } (0,T] \times \Omega \\ \text{or equivalently} \\ (v_0)_t = \sum_{i,j,k=1}^{n} \left(\alpha \frac{\partial F}{\partial x} - u\chi^T \right)_{ij} \frac{\partial g_k}{\partial x_i} \frac{\partial^2 v_0}{\partial g_j \partial g_k} & \text{in } (0,T] \times \Omega \\ v = 0 & \text{on } \partial\Omega \\ v = \phi(x) & \text{at } t = 0. \end{cases} \tag{2.10}$$

Finally, v is approximated by $v_0 + \chi^T \frac{\partial x}{\partial g} \nabla v_0$.

2.4. Generalization to elliptic equations with parameters

We can also generalize our method to solve the multiscale problems with a family of parameters. To illustrate this idea, we consider the following elliptic equation with a parameter $\theta \in \Theta \subset \mathbb{R}^d$

$$\begin{cases} -\nabla \cdot (a(x,\theta) \nabla u(x)) = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega. \end{cases} \tag{2.11}$$

The dimension of the multiscale solution could be very large due to the large parametric space induced by θ. However, in many cases, the effective 'dimension' of the solution space could be small. That is to say, we can find a set of parameters $\{\theta_i\}_{i=1}^M$ so that every solution (with parameter θ) can be approximated by a linear combination of the solutions associated with θ_i to reasonable accuracy. For more discussions on how to choose this sample parameter set in the context of reduced order basis methods, we refer to [9, 18, 25, 31].

We can easily generalize our method to such problems. During the offline stage, we need to find a subset $\{\theta_i\}_{i=1}^M$ such that the solutions $\{F_{(i)}\}_{i=1}^M$ to the following equations

$$\begin{cases} \nabla \cdot (a(x,\theta_i) \nabla F_{(i)}) = 0 & \text{in } \Omega \\ F_{(i)} = x & \text{on } \partial\Omega \end{cases} \tag{2.12}$$

form a basis for all the harmonic coordinates F. This is all we need to do for the offline part and we store all the $F_{(i)}$.

At the online stage, we can approximate $F_{(\theta)}$ for any θ as a linear superposition of $F_{(i)}$

$$F_{(\theta)} = \sum_{i=1}^{M} \beta_i F_{(i)}. \tag{2.13}$$

To find the coefficients β_i, we multiply the equation by some test functions ϕ_j and perform integration by parts to get

$$\sum_{i=1}^{M} \beta_i \int_{\Omega_j} \nabla \phi_j a \nabla F_{(i)} \mathrm{d}x = 0, \tag{2.14}$$

where Ω_j is the support for ϕ_j. Note that if we choose $\phi_j = F_{(j)}$, then $\Omega_j = \Omega$, and the integration will be carried out in the whole domain, which makes the computation very expensive. To reduce the computational

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

cost, we can choose some locally supported test functions ϕ_j, for example the linear hat functions supported in $\{|x - x_j| \leq \delta_j\}$. We only need $M - 1$ such equations and the last equation is determined by the boundary conditions, *i.e.*

$$\sum_{i=1}^{M} \beta_i = 1. \tag{2.15}$$

After getting $F_{(\theta)}$, we decompose it as $F_{(\theta)} = g_{(\theta)} + \chi_{(\theta)}$, and solve

$$\begin{cases} -\nabla \cdot (a(I + \frac{\partial \chi_{(\theta)}}{\partial x} \frac{\partial x}{\partial g_{(\theta)}}) \nabla u_0) = f & \text{in } \Omega \\ u_0 = 0 & \text{on } \partial\Omega. \end{cases} \tag{2.16}$$

Then u is approximated by $u_0 + \chi_{(\theta)}^T \frac{\partial x}{\partial g_{(\theta)}} \nabla u_0$.

3. Analysis

In this section, we will perform error analysis to show that the H^1 norm of the difference between the effective solution and the original solution is small provided that the L^∞ norm of χ is small. We will also prove that the solution to the effective equation is in H^2 under some conditions. We first perform our analysis for the elliptic equation and then the time-dependent equations. Before presenting the general analysis, we will start with a simple 1D elliptic equation to illustrate the main idea.

3.1. A 1D example

Consider a 1D elliptic equation on a unit interval $[0, 1]$:

$$\begin{cases} (a(x)u'(x))' = f(x) & \text{in } (0,1) \\ u(0) = u(1) = 0. \end{cases} \tag{3.1}$$

The corresponding harmonic map F is defined as

$$\begin{cases} (a(x)F'(x))' = 0 & \text{in } (0,1) \\ F(0) = 0,\ F(1) = 1. \end{cases} \tag{3.2}$$

Our effective equation is given by

$$\begin{cases} a(x)F'(x)(u_0'(x)/g'(x))' = f(x) & \text{in } (0,1) \\ u_0(0) = u_0(1) = 0. \end{cases} \tag{3.3}$$

We can solve these equations analytically and get

$$u(x) = C_0 \left(F(x) \int_0^x f(s)\mathrm{d}s - \int_0^x F(s)f(s)\mathrm{d}s + C_1 F(x) \right), \tag{3.4}$$

$$F(x) = \frac{1}{C_0} \int_0^x \frac{\mathrm{d}s}{a(s)}, \tag{3.5}$$

$$u_0(x) = C_0 \left(g(x) \int_0^x f(s)\mathrm{d}s - \int_0^x g(s)f(s)\mathrm{d}s + C_2 g(x) \right), \tag{3.6}$$

where $C_0 = \int_0^1 \frac{\mathrm{d}s}{a(s)}$, $C_1 = -\int_0^1 f(s)\mathrm{d}s + \int_0^1 F(s)f(s)\mathrm{d}s$ and $C_2 = -\int_0^1 f(s)\mathrm{d}s + \int_0^1 g(s)f(s)\mathrm{d}s$. Since $f \in L^2$ and g is smooth, we can see that u_0 is smooth. Let $u_1 = \chi u_0'/g'$, direct computations give

$$(u - u_0 - u_1)' = C_0 \left((C_1 - C_2)F' - \chi f \right). \tag{3.7}$$

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

We also have

$$C_1 - C_2 = \int_0^1 \chi(s) f(s) \mathrm{d}s, \tag{3.8}$$

and

$$F' = \frac{1}{C_0} \frac{1}{a(x)}. \tag{3.9}$$

By the assumption $a(x) \geq \gamma > 0$, we know that F' is bounded. Thus we can bound $\|u - u_0 - u_1\|_{H^1}$ in terms of $\|\chi\|_{L^\infty}$. This implies that as long as we can decompose F so that the oscillatory part χ is small and g is smooth, then $\|u - u_0 - u_1\|_{H^1}$ is small and we can solve u_0 on a coarse grid. In the next section, we will show that this result is true for general multi-dimensional elliptic equations as well.

3.2. An error estimate for the general case

The main result of this section is the following theorem.

Theorem 3.1. *Suppose u, F and u_0 are weak solutions to (1.1), (2.1) and (2.5), respectively. Let $u_1 = \chi^T \frac{\partial x}{\partial g} \nabla u_0$, $F = g + \chi$, and $\chi = 0$ on $\partial\Omega$. Then we have*

$$\|u - u_0 - u_1\|_{H^1(\Omega)} \leq C \|\chi\|_{L^\infty(\Omega)} \left\| \frac{\partial g}{\partial x} \right\|_{L^\infty(\Omega)} \left\| \det\left(\frac{\partial x}{\partial g} \right) \right\|_{L^\infty(\Omega)} |\tilde{u}_0|_{H^2(\Omega)}, \tag{3.10}$$

where C is a constant that depends on n, Ω and a, $\tilde{u}_0 = u_0 \circ g^{-1}$.

Our goal is to estimate the H^1 norm of $z = u - u_0 - u_1$. Zero boundary condition of χ implies zero boundary condition of z, *i.e.* $z \in H_0^1(\Omega)$. Since a is a positive definite matrix whose eigenvalues have a positive lower bound, we conclude that $||z||_{H_0^1(\Omega)}$ is equivalent to the energy norm, $< a\nabla z, \nabla z >$, where $< . >$ is the L^2 inner product. Thus, it is sufficient to perform our estimate using the energy norm.

Proof. Define

$$z = u - u_0 - u_1 \in H_0^1(\Omega),$$

$$p = a\nabla u - a \frac{\partial F}{\partial x} \frac{\partial x}{\partial g} \nabla u_0,$$

$$\eta = -a \frac{\partial}{\partial x} \left(\frac{\partial x}{\partial g} \nabla u_0 \right) \chi.$$

Then we have

$$\nabla \cdot p = \nabla \cdot (a\nabla u) - \nabla \cdot (a \frac{\partial F}{\partial x} \frac{\partial x}{\partial g} \nabla u_0) = f - f = 0, \tag{3.11}$$

and

$$a\nabla z - p = a\nabla u - a\nabla u_0 - a\nabla u_1 - a\nabla u + a \frac{\partial F}{\partial x} \frac{\partial x}{\partial g} \nabla u_0 = a \frac{\partial \chi}{\partial x} \frac{\partial x}{\partial g} \nabla u_0 - a\nabla u_1.$$

Further, we note that

$$\nabla u_1 = \frac{\partial \chi}{\partial x} \frac{\partial x}{\partial g} \nabla u_0 + \frac{\partial}{\partial x} \left(\frac{\partial x}{\partial g} \nabla u_0 \right) \chi.$$

As a result, we get

$$a\nabla z - p = -a \frac{\partial}{\partial x} \left(\frac{\partial x}{\partial g} \nabla u_0 \right) \chi = \eta.$$

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

Thus, we obtain

$$\begin{aligned}\langle a\nabla z, \nabla z\rangle &= \int_\Omega \nabla z \cdot a\nabla z \mathrm{d}x \\ &= \int_\Omega (a\nabla z - p)\cdot \nabla z \mathrm{d}x + \int_\Omega p\cdot \nabla z \mathrm{d}x \\ &= \int_\Omega \eta \cdot \nabla z \mathrm{d}x - \int_\Omega (\nabla \cdot p) z \mathrm{d}x \\ &= \int_\Omega \eta \cdot \nabla z \mathrm{d}x,\end{aligned}$$

where we have used (3.11). By using the ellipticity assumption of a, we get

$$C_1\|\nabla z\|^2_{L^2(\Omega)} \le \langle a\nabla z, \nabla z\rangle \le \|\eta\|_{L^2(\Omega)}\|\nabla z\|_{L^2(\Omega)},$$

which implies

$$C_1\|\nabla z\|_{L^2(\Omega)} \le \|\eta\|_{L^2(\Omega)}.$$

Since z vanishes on $\partial\Omega$, Poincaré's theorem gives

$$\|z\|_{H^1_0(\Omega)} \le C_2\|\nabla z\|_{L^2(\Omega)}.$$

Thus, we get

$$\|z\|_{H^1_0(\Omega)} \le C\|\eta\|_{L^2(\Omega)},$$

where C is a constant that depends on n, Ω and a only.

Let $y = g(x)$ and $\tilde{u}_0(y) = \tilde{u}_0(g(x)) = u_0(x)$, then we have

$$\begin{aligned}\eta &= -a\frac{\partial}{\partial x}\left(\frac{\partial x}{\partial g}\nabla u_0\right)\chi \\ &= -a\frac{\partial g}{\partial x}\frac{\partial x}{\partial g}\frac{\partial}{\partial x}\left(\frac{\partial x}{\partial g}\nabla u_0\right)\chi \\ &= -a\left(\frac{\partial g}{\partial x}\nabla_y^2\tilde{u}_0\right)\chi.\end{aligned}$$

As a result, we obtain

$$\|\eta\|_{L^2} \le C\|a\|_{L^\infty(\Omega)}\|\chi\|_{L^\infty(\Omega)}\left\|\frac{\partial g}{\partial x}\right\|_{L^\infty(\Omega)}\left\|\det\left(\frac{\partial x}{\partial g}\right)\right\|_{L^\infty(\Omega)}|\tilde{u}_0|_{H^2(\Omega)}.$$

The determinant of $\frac{\partial x}{\partial g}$ enters the last step of the above estimate due to a change of variables from x to y. Combining all the results, we get

$$\|z\|_{H^1_0(\Omega)} \le C\|\chi\|_{L^\infty(\Omega)}\left\|\frac{\partial g}{\partial x}\right\|_{L^\infty(\Omega)}\left\|\det\left(\frac{\partial x}{\partial g}\right)\right\|_{L^\infty(\Omega)}|\tilde{u}_0|_{H^2(\Omega)}.$$

This completes the proof of Theorem 3.1. □

The error estimate given by Theorem 3.1 provides us with some insight how to choose our decomposition $F = g + \chi$. If $\|\chi\|_{L^\infty(\Omega)}$ is small and g is smooth in the sense that both $\|\frac{\partial g}{\partial x}\|_{L^\infty(\Omega)}$ and $\|\det(\frac{\partial x}{\partial g})\|_{L^\infty(\Omega)}$ are

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

bounded, then the approximation is accurate provided that $|\tilde{u}_0|_{H^2(\Omega)}$ is bounded. To evaluate $|\tilde{u}_0|_{H^2(\Omega)}$, we make a change of variables from x to $y = g(x)$ in equation (2.5), which has a non-divergence form

$$-\sum_{i,j=1}^{n} B_{ij}(y)\frac{\partial^2 \tilde{u}_0}{\partial y_i \partial y_j} = \tilde{f}(y), \tag{3.12}$$

where

$$B(y) = \left(\left|\det\left(\frac{\partial x}{\partial g}\right)\right| \left(\frac{\partial g}{\partial x}\right)^T a \frac{\partial F}{\partial x}\right) \circ g^{-1}(y) \tag{3.13}$$

and

$$\tilde{f}(y) = \left(\left|\det\left(\frac{\partial x}{\partial g}\right)\right| f\right) \circ g^{-1}(y). \tag{3.14}$$

Note that the determinant term comes from the change of variables in the integral since we consider weak solutions.

If the matrix B satisfies the Cordes condition (see *e.g.* [26]),

$$\frac{\sum_{i,j=1}^{n} B_{ij}^2(y)}{(\sum_{i=1}^{n} B_{ii}(y))^2} \leq \frac{1}{n-1+\epsilon} \text{ in } \Omega, \tag{3.15}$$

for some $\epsilon > 0$ and $M = \sup\left(\frac{\sum_{i=1}^{n} B_{ii}(y)}{\sum_{i,j=1}^{n} B_{ij}^2(y)}\right) < \infty$, we can apply Theorem 1.2.1 in [26] to conclude that

$$|\tilde{u}_0|_{H^2(\Omega)} \leq \frac{M}{1-\sqrt{1-\epsilon}} \|f\|_{L^2(\Omega)}. \tag{3.16}$$

In general, the condition (3.15) is hard to verify. For $n = 2$, we have a more concrete version for ϵ and M. Suppose $\lambda_{\max}(y)$ and $\lambda_{\min}(y)$ are the maximum and minimum eigenvalues of $B(y)$, if $\eta_1 = \sup \frac{\lambda_{\max}(y)}{\lambda_{\min}(y)} < \infty$ and $\eta_2 = \inf \lambda_{\min}(y) > 0$, then we can pick $\epsilon = \frac{1}{\eta_1}$ and $M = \frac{1}{\eta_2}$.

We remark that Theorem 3.1 is an error estimate for the analytical solutions. One should not use it to study the convergence rate of the numerical method. It does not imply that the smaller $\|\chi\|_{L^\infty(\Omega)}$ is, the smaller the numerical error would become. On one hand, if we choose χ to be 0, in which case $g = F$, the error is zero in theory. However, g is no longer smooth and we will have to use a fine mesh to solve the effective equation, which is not what our method is designed for. Similarly, if we let χ decay to zero, g will pick up more small scales and the derivative of g increases. In this case, we will not be able to obtain a small overall error if we use a coarse mesh.

From a view point of numerical implementation, it is usually the case that F may become degenerated in some localized region. The 'smoothness' of g depends on the size of the numerical grid. To avoid degeneracy in constructing g, we can use a finer mesh locally to capture some important information in certain local region, and use a coarser mesh in other regions. By doing this, g is smooth when using a non-uniform mesh, and we can guarantee that χ is small. How to choose an optimal decomposition which would lead to the smallest overall error requires a delicate balance in our decomposition of F. We will discuss more about this issue in Section 5.

3.3. Parabolic and convection-diffusion case

For the parabolic equation, we have the following theorem.

Theorem 3.2. *Suppose u, F and u_0 are weak solutions to (2.6), (2.1) and (2.7), respectively. Let $u_1 = \chi^T \frac{\partial x}{\partial g} \nabla u_0$, $F = g + \chi$, and $\chi = 0$ on $\partial\Omega$. Then we have*

$$\begin{aligned}&\max_{(0,T]} \|u - u_0 - u_1\|_{L^2(\Omega)} + \|u - u_0 - u_1\|_{L^2(0,T;H^1(\Omega))} \\ \leq & C\|\chi\|_{L^\infty(\Omega)} \left\|\frac{\partial g}{\partial x}\right\|_{L^\infty(\Omega)} \left\|\det\left(\frac{\partial x}{\partial g}\right)\right\|_{L^\infty(\Omega)} \left(|\tilde{u}_0|_{H^{2,1}(0,T;\Omega)} + \|(\tilde{u}_0)_t\|_{L^2(0,T;H^1(\Omega))}\right),\end{aligned} \tag{3.17}$$

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

where C is a constant that depends on n, Ω and a, $\tilde{u}_0(y,t) = \tilde{u}_0(g(x),t) = u_0(x,t)$ and $\|u\|^2_{L^2(0,T;H^1(\Omega))} := \int_0^T \int_\Omega (u^2 + |\nabla u|^2) \mathrm{d}x\mathrm{d}t$, $|u|^2_{H^{2,1}(0,T;\Omega)} := \int_0^T \int_\Omega ((u_t)^2 + |\nabla \nabla u|^2) \mathrm{d}x\mathrm{d}t$.

Proof. Let $z = u - u_0 - u_1$, then $z = 0$ at $t = 0$ and on $\partial\Omega$. We define

$$p = a\nabla u - a\frac{\partial F}{\partial x}\frac{\partial x}{\partial g}\nabla u_0,$$

$$\eta_1 = -a\frac{\partial}{\partial x}\left(\frac{\partial x}{\partial g}\nabla u_0\right)\chi,$$

$$\eta_2 = -\chi^T \frac{\partial x}{\partial g}\nabla (u_0)_t.$$

Then we have

$$\begin{aligned} a\nabla z - p =& a\nabla u - a\nabla u_0 - a\nabla u_1 - a\nabla u + a\frac{\partial F}{\partial x}\frac{\partial x}{\partial g}\nabla u_0 \\ =& - a\nabla u_0 - a\nabla u_1 + a\frac{\partial F}{\partial x}\frac{\partial x}{\partial g}\nabla u_0 \\ =& - a\frac{\partial}{\partial x}\left(\frac{\partial x}{\partial g}\nabla u_0\right)\chi = \eta_1, \end{aligned}$$

and

$$\nabla \cdot p = (u_t - f) - ((u_0)_t - f) = u_t - (u_0)_t = z_t + (u_1)_t = z_t - \eta_2.$$

Then for any $\tau \in (0, T]$ we have

$$\begin{aligned} & \int_0^\tau \int_\Omega \nabla z \cdot a\nabla z \mathrm{d}x\mathrm{d}t \\ =& \int_0^\tau \int_\Omega \nabla z \cdot (a\nabla z - p)\mathrm{d}x\mathrm{d}t + \int_0^\tau \int_\Omega \nabla z \cdot p \mathrm{d}x\mathrm{d}t \\ =& \int_0^\tau \int_\Omega \nabla z \cdot \eta_1 \mathrm{d}x\mathrm{d}t - \int_0^\tau \int_\Omega z(\nabla \cdot p)\mathrm{d}x\mathrm{d}t \\ =& \int_0^\tau \int_\Omega \nabla z \cdot \eta_1 \mathrm{d}x\mathrm{d}t - \int_0^\tau \int_\Omega (zz_t - z\eta_2)\mathrm{d}x\mathrm{d}t \\ =& \int_0^\tau \int_\Omega \nabla z \cdot \eta_1 \mathrm{d}x\mathrm{d}t - \int_\Omega \left(\frac{z^2(\tau,x)}{2} - \frac{z^2(0,x)}{2}\right)\mathrm{d}x + \int_0^\tau \int_\Omega z\eta_2 \mathrm{d}x\mathrm{d}t \\ =& \int_0^\tau \int_\Omega \nabla z \cdot \eta_1 \mathrm{d}x\mathrm{d}t - \frac{1}{2}\int_\Omega z^2(\tau,x)\mathrm{d}x + \int_0^\tau \int_\Omega z\eta_2 \mathrm{d}x\mathrm{d}t. \end{aligned}$$

Rearranging the above equation, we get

$$\begin{aligned} \frac{1}{2}\int_\Omega z^2(\tau,x)\mathrm{d}x + \int_0^\tau \int_\Omega \nabla z \cdot a\nabla z \mathrm{d}x\mathrm{d}t &= \int_0^\tau \int_\Omega \nabla z \cdot \eta_1 \mathrm{d}x\mathrm{d}t + \int_0^\tau \int_\Omega z\eta_2 \mathrm{d}x\mathrm{d}t \\ &\le \int_0^\tau \int_\Omega |\nabla z \cdot \eta_1| \mathrm{d}x\mathrm{d}t + \int_0^\tau \int_\Omega |z\eta_2| \mathrm{d}x\mathrm{d}t \\ &\le \int_0^T \int_\Omega |\nabla z \cdot \eta_1| \mathrm{d}x\mathrm{d}t + \int_0^T \int_\Omega |z\eta_2| \mathrm{d}x\mathrm{d}t. \end{aligned}$$

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

Taking the maximum over $\tau \in (0,T]$, we have

$$\begin{aligned}\max_{(0,T]} \frac{1}{2}\int_\Omega z^2(\tau,x)\mathrm{d}x + \int_0^T\int_\Omega \nabla z\cdot a\nabla z \mathrm{d}x\mathrm{d}t \leq& \int_0^T\int_\Omega |\nabla z\cdot\eta_1|\mathrm{d}x\mathrm{d}t + \int_0^T\int_\Omega |z\eta_2|\mathrm{d}x\mathrm{d}t\\ \leq& \|\nabla z\|_{L^2(0,T;\Omega)}\|\eta_1\|_{L^2(0,T;\Omega)} + \|z\|_{L^2(0,T;\Omega)}\|\eta_2\|_{L^2(0,T;\Omega)}.\end{aligned}$$

where the second inequality is due to the Hölder's inequality.

Application of the Poincaré's inequality gives

$$\begin{aligned}\Big(\max_{(0,T]}\|z\|_{L^2(\Omega)} + \|z\|_{L^2(0,T;H^1(\Omega))}\Big)^2 \leq& C\Big(\max_{(0,T]}\frac{1}{2}\int_\Omega z^2(\tau,x)\mathrm{d}x + \int_0^T\int_\Omega \nabla z\cdot a\nabla z\mathrm{d}x\mathrm{d}t\Big)\\ \leq& C\Big(\|\nabla z\|_{L^2(0,T;\Omega)}\|\eta_1\|_{L^2(0,T;\Omega)} + \|z\|_{L^2(0,T;\Omega)}\|\eta_2\|_{L^2(0,T;\Omega)}\Big)\\ \leq& C\Big(\max_{(0,T]}\|z\|_{L^2(\Omega)} + \|z\|_{L^2(0,T;H^1(\Omega))}\Big)\Big(\|\eta_1\|_{L^2(0,T;\Omega)} + \|\eta_2\|_{L^2(0,T;\Omega)}\Big).\end{aligned}$$

Finally we have

$$\begin{aligned}&\max_{(0,T]}\|z\|_{L^2(\Omega)} + \|z\|_{L^2(0,T;H^1(\Omega))}\\ \leq& C\Big(\|\eta_1\|_{L^2(0,T;\Omega)} + \|\eta_2\|_{L^2(0,T;\Omega)}\Big)\\ \leq& C\|\chi\|_{L^\infty(\Omega)}\left\|\frac{\partial g}{\partial x}\right\|_{L^\infty(\Omega)}\left\|\det\left(\frac{\partial x}{\partial g}\right)\right\|_{L^\infty(\Omega)}\left(|\tilde{u}_0|_{H^{2,1}(0,T;\Omega)} + \|(\tilde{u}_0)_t\|_{L^2(0,T;H^1(\Omega))}\right),\end{aligned}$$

where C is a constant that depends on n, Ω and a.

□

For the convection-diffusion equation, we have the following theorem.

Theorem 3.3. *Suppose v, F and v_0 are weak solutions to (2.8), (2.9) and (2.10), respectively. Let $v_1 = \chi^T\frac{\partial x}{\partial g}\nabla v_0$, $F = g + \chi$, and $\chi = 0$ on $\partial\Omega$. Then we have*

$$\begin{aligned}&\max_{(0,T]}\|v - v_0 - v_1\|_{L^2(\Omega)} + \|v - v_0 - v_1\|_{L^2(0,T;H^1(\Omega))}\\ \leq& C\|\chi\|_{L^\infty(\Omega)}\left\|\frac{\partial g}{\partial x}\right\|_{L^\infty(\Omega)}\left\|\det\left(\frac{\partial x}{\partial g}\right)\right\|_{L^\infty(\Omega)}\left(|\tilde{v}_0|_{H^{2,1}(0,T;\Omega)} + \|(\tilde{v}_0)_t\|_{L^2(0,T;H^1(\Omega))}\right),\end{aligned} \tag{3.18}$$

where C is a constant that depends on n, Ω, u, and α, $\tilde{v}_0(y,t) = \tilde{v}_0(g(x),t) = v_0(x,t)$.

The proof of the above theorem is analogous to that of Theorem 3.2. We omit the proof here.

In Theorems 3.2 and 3.3, the quantities $|\tilde{u}_0|_{H^{2,1}(0,T;\Omega)}$ and $|\tilde{v}_0|_{H^{2,1}(0,T;\Omega)}$ could be bounded since the equations in both cases can be written in non-divergence forms in variable $y = g(x)$.

For the parabolic equation, the equation has a form similar to that of the elliptic equation

$$(\tilde{u}_0)_t - \sum_{i,j=1}^n B_{ij}(y)\frac{\partial^2\tilde{u}_0}{\partial y_i\partial y_j} = \tilde{f}(y), \tag{3.19}$$

where B and $\tilde{f}$ are the same as those in (3.13) and (3.14).

For the convection-diffusion equation, the effective equation is

$$(v_0)_t + u\cdot\nabla v_0 = \nabla\cdot\left(\left(\alpha\frac{\partial F}{\partial x} - u\chi^T\right)\frac{\partial x}{\partial g}\nabla v_0\right). \tag{3.20}$$

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

Note that

$$\begin{aligned}
&\nabla\cdot\left(\left(\alpha\frac{\partial F}{\partial x}-u\chi^T\right)\frac{\partial x}{\partial g}\nabla v_0\right)\\
&=\nabla\cdot\left(\left(\alpha\frac{\partial F}{\partial x}-u\chi^T\right)\right)\frac{\partial x}{\partial g}\nabla v_0+\sum_{i,j,k=1}^{n}\left(\alpha\frac{\partial F}{\partial x}-u\chi^T\right)_{ij}\frac{\partial g_k}{\partial x_i}\frac{\partial^2 v_0}{\partial g_j\partial g_k}\\
&=\left(\alpha\Delta F-u\cdot\frac{\partial\chi}{\partial x}\right)\frac{\partial x}{\partial g}\nabla v_0+\sum_{i,j,k=1}^{n}\left(\alpha\frac{\partial F}{\partial x}-u\chi^T\right)_{ij}\frac{\partial g_k}{\partial x_i}\frac{\partial^2 v_0}{\partial g_j\partial g_k}\\
&=\left(u\cdot\frac{\partial F}{\partial x}-u\cdot\frac{\partial\chi}{\partial x}\right)\frac{\partial x}{\partial g}\nabla v_0+\sum_{i,j,k=1}^{n}\left(\alpha\frac{\partial F}{\partial x}-u\chi^T\right)_{ij}\frac{\partial g_k}{\partial x_i}\frac{\partial^2 v_0}{\partial g_j\partial g_k}\\
&=u\cdot\frac{\partial g}{\partial x}\frac{\partial x}{\partial g}\nabla v_0+\sum_{i,j,k=1}^{n}\left(\alpha\frac{\partial F}{\partial x}-u\chi^T\right)_{ij}\frac{\partial g_k}{\partial x_i}\frac{\partial^2 v_0}{\partial g_j\partial g_k}\\
&=u\cdot\nabla v_0+\sum_{i,j,k=1}^{n}\left(\alpha\frac{\partial F}{\partial x}-u\chi^T\right)_{ij}\frac{\partial g_k}{\partial x_i}\frac{\partial^2 v_0}{\partial g_j\partial g_k}.
\end{aligned}$$

So the equation can be written as

$$(\tilde{v}_0)_t-\sum_{i,j=1}^{n}B_{ij}(y)\frac{\partial^2\tilde{v}_0}{\partial y_i\partial y_j}=0, \tag{3.21}$$

where

$$B(y)=\left(\left|\det\left(\frac{\partial x}{\partial g}\right)\right|\left(\alpha\frac{\partial F}{\partial x}-u\chi^T\right)\frac{\partial g}{\partial x}\right)\circ g^{-1}(y). \tag{3.22}$$

In both cases, if the corresponding coefficient matrix B satisfies

$$\frac{\sum_{i,j=1}^{n}B_{ij}^2+1}{(\sum_{i=1}^{n}B_{ii}+1)^2}\leq\frac{1}{n+\epsilon}, \tag{3.23}$$

where n is the dimension and ϵ is a positive number, we can prove that $|\tilde{u}_0|_{H^{2,1}(0,T;\Omega)}$ (or$|\tilde{v}_0|_{H^{2,1}(0,T;\Omega)}$) is bounded (see *e.g.* [26]).

As for the terms $\|(\tilde{u}_0)_t\|_{L^2(0,T;H^1(\Omega))}$ and $\|(\tilde{v}_0)_t\|_{L^2(0,T;H^1(\Omega))}$, we cannot provide an analytical bound based on the assumptions we have so far. Since these terms contain the spacial gradient of $(\tilde{u}_0)_t$ (or $(\tilde{v}_0)_t$), we need stronger requirements on the coefficients as well as the source terms. We will monitor these terms in the numerical experiments. As we will see in Section 5, these terms are bounded in the examples that we consider.

4. Comparison with the homogenization method

In this section, we compare our method with the classical homogenization method. To simplify our presentation, we will restrict our comparison to elliptic equations. The analysis can be also applied to other equations.

First we briefly review the homogenization theory [8]. Consider

$$\begin{cases}-\nabla\cdot(a(\frac{x}{\epsilon})\nabla u(x))=f(x) & \text{in } \Omega\\ u=0 & \text{on } \partial\Omega,\end{cases} \tag{4.1}$$

where $a(y)$ is a symmetric, positive definite matrix, and $f\in L^2$. Furthermore, $a_{ij}(y)$ are periodic smooth functions in y in a unit cube Y.

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

The homogenized coefficients are given by

$$a_{ij}^{*} = \frac{1}{|Y|}\int_{Y} a_{ik}(y)(\delta_{kj} + \frac{\partial}{\partial y_k}\chi_h^j)\mathrm{d}y, \tag{4.2}$$

where χ_h^j (we use the notation χ_h^j to distinguish from χ^j) is the solution to the periodic cell problem

$$\nabla_y \cdot (a(y)\nabla_y \chi_h^j(y)) = -\frac{\partial}{\partial y_i}a_{ij}(y) \tag{4.3}$$

with zero mean, *i.e.* $\int_Y \chi_h^j \mathrm{d}y = 0$.

Let u_0 be the solution to the homogenized equation

$$\begin{cases} -\nabla \cdot (a^*(x)\nabla u_0(x)) = f(x) & \text{in } \Omega \\ u_0 = 0 & \text{on } \partial\Omega. \end{cases} \tag{4.4}$$

Then we have

$$||u - u_0||_{L^2(\Omega)} \leq C\epsilon ||u||_{H^2(\Omega)}. \tag{4.5}$$

Further, we define

$$u_1(x) = \chi_h^j\left(\frac{x}{\epsilon}\right)\frac{\partial u_0}{\partial x_j}(x). \tag{4.6}$$

Note that $u_0 + \epsilon u_1 \neq 0$ on $\partial\Omega$, we introduce a first order correction term θ_ϵ satisfying

$$\begin{cases} -\nabla \cdot (a(\frac{x}{\epsilon})\nabla\theta_\epsilon(x)) = 0 & \text{in } \Omega \\ \theta_\epsilon = u_1 & \text{on } \partial\Omega. \end{cases} \tag{4.7}$$

Then it can be shown that (see *e.g.* [28])

$$||u - u_0 - \epsilon(u_1 - \theta_\epsilon)||_{H^1(\Omega)} \leq C\epsilon ||u_0||_{H^2(\Omega)}. \tag{4.8}$$

The constant C is independent of u_0 and ϵ.

One important advantage of our method is that we could take care of a continuum of scales and do not require periodicity on the microstructure while homogenization theory usually requires scale separation and periodic structures. Moreover, one must include a boundary correction term θ_ϵ to achieve H^1 convergence in the homogenization method. This correction term must be solved on a fine mesh and is expensive to compute. In comparison, there is no need to compute a correction term in our method since we require $\chi = 0$ on $\partial\Omega$.

If only L^2 convergence is needed, both methods do not need correction terms, and the homogenized coefficients are easier to compute (see (4.2) and (4.3)). Our method requires two global solutions on a fine mesh. However, under the conditions for homogenization (periodic smooth $a(y)$), we could modify our method easily so that we can compute the harmonic map with the same cost as the homogenization method. Specifically, the harmonic map F satisfies $\nabla \cdot (a(\frac{x}{\epsilon})\nabla F(x)) = 0$ and $F = g + \chi$. If we set $g = x$, we get the equation for χ as follows

$$\nabla \cdot \left(a\left(\frac{x}{\epsilon}\right)\nabla\chi^j(x)\right) = -\frac{1}{\epsilon}\frac{\partial}{\partial y_i}a_{ij}\left(\frac{x}{\epsilon}\right). \tag{4.9}$$

Now we do not require $\chi = 0$ on the boundary, assume χ^j to be periodic in Y and impose the constraint $\int_Y \chi^j(x)\mathrm{d}x = 0$. Equation (4.9) is still global, but comparing (4.9) with (4.3) gives $\chi(x) = \epsilon\chi_h(\frac{x}{\epsilon})$. So we can solve (4.3) instead of (4.9). Following the proof in Theorem 3.1, we could obtain the same error estimate as (4.5).

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

5. Numerical results

5.1. Decomposition of the harmonic coordinates

In this section, we discuss how to construct the decomposition of the harmonic coordinates. As we know from the previous sections, the decomposition $F = g + \chi$ plays an essential role in our method. Here we discuss some guideline in choosing such decomposition and how to construct it numerically.

The first criterion in choosing our decomposition is to make sure that g is smooth and invertible. We need to define what we mean by g being smooth. The smoothness is relative to the coarse mesh that we will use to solve the effective equation. In our numerical implementation, we use the standard linear finite element method to solve the effective equation. Thus, any linear combination of the nodal basis on the coarse mesh could be considered as a smooth function, and we can choose g in this form.

The second criterion of our decomposition is to make χ small. If we choose the nodal values of g close to those of F, then we expect that the difference between the two would be small. This suggests a natural way to define g, *i.e.* we can choose the nodal values of g at the coarse mesh points to be the same as F at these coarse mesh points. We can then interpolate g from the coarse mesh points to the fine mesh points using the linear interpolation. Once we have defined g globally through linear interpolation, we have also determined $\chi = F - g$. Since F is linear on the boundary, such decomposition guarantees that $g = F$ on the boundary, which implies that $\chi = 0$ on $\partial\Omega$.

The decomposition method described above seems to work very well from our numerical experiments. One advantage of this approach is that it is very easy to implement. However, such decomposition may not work in all cases and may not give the optimal result especially when the harmonic map, F, is not invertible. To overcome this difficulty, we are currently investigating an alternative approach based on optimization. More specifically, we would like to find a smooth g that lives in the linear finite element space generated by the coarse grid mesh and minimizes the difference between g and F in some appropriate norm subject to the constraint that g is invertible. How to formulate this problem as a convex optimization problem would be the key to make this method efficient. We will report the result of this study in a subsequent paper.

We also have another guideline to determine whether the decomposition is effective or not from the view point of numerical implementation. Note that $u_g = g$ solves the following equation exactly:

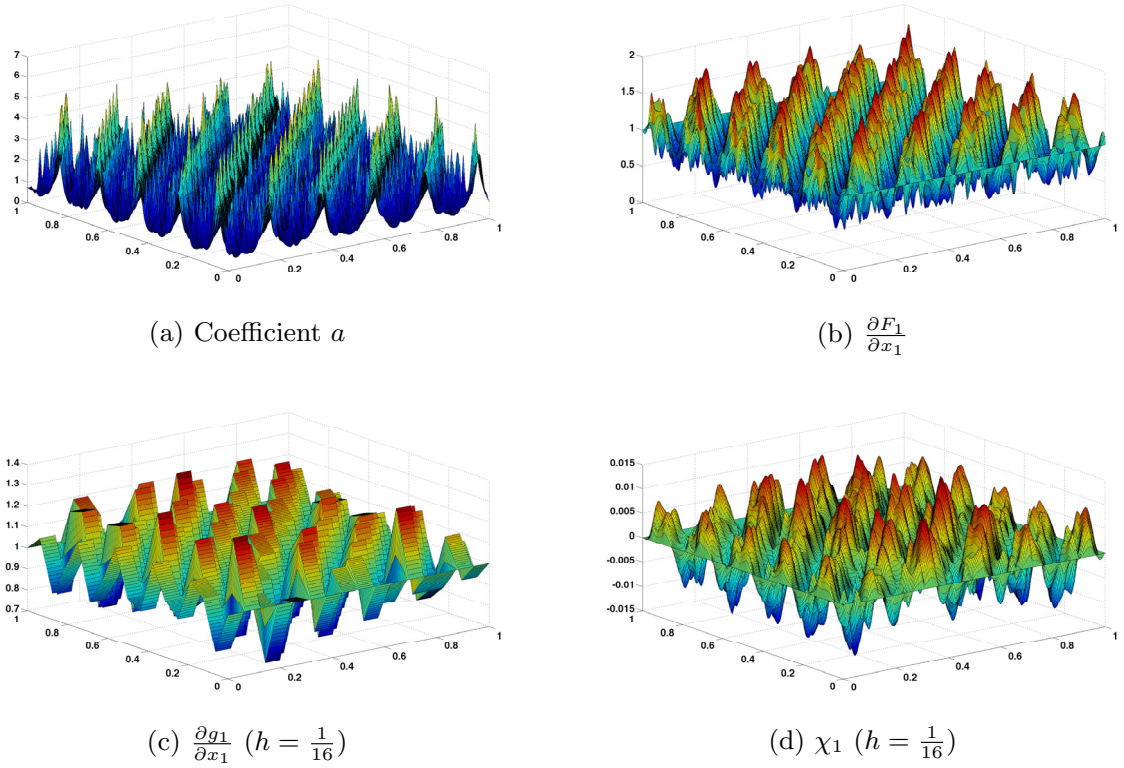
$$\begin{cases} -\nabla \cdot (a(x) \left(I + \frac{\partial \chi}{\partial x} \frac{\partial x}{\partial g}\right) \nabla u_g(x)) = 0 & \text{in } \Omega \\ u_g = x & \text{on } \partial\Omega. \end{cases} \tag{5.1}$$

If g is smooth enough, we should be able to solve (5.1) accurately on a coarse mesh. Thus, we can use the difference between the numerically computed u_g and g to determine whether we obtain a good decomposition for F. The smaller the difference is, the better the method would perform.

5.2. Elliptic equations

The first example that we use to test the performance of our method is an elliptic equation. We perform several numerical experiments to test our method for the elliptic equation (1.1). We take $\Omega = [0,1] \times [0,1]$ in two space dimensions. Since it is difficult to construct a general enough test problem with an analytic solution, we use well resolved numerical solutions in place of exact solutions. In our computations, we use the standard linear finite element method, and choose a 256×256 mesh as the well-resolved solution. To implement our method, the coarse meshes are chosen to be 8×8, 16×16 and 32×32, respectively, and we compare the results on different meshes and calculate the convergence rate.

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press



(a) Coefficient a (b) $\frac{\partial F_1}{\partial x_1}$

(c) $\frac{\partial g_1}{\partial x_1}$ $(h = \frac{1}{16})$ (d) χ_1 $(h = \frac{1}{16})$

FIGURE 1. Example 1.

Example 1. We consider the case when the elliptic coefficient is a scalar defined below

$$a = \frac{1}{2 + 1.6\sin(2\pi(x-y)/\epsilon_1)} + \frac{1}{4 + 1.8(\sin(2\pi x/\epsilon_2) + \sin(2\pi y/\epsilon_2))} + \frac{1}{10(2 + 1.8\sin(2\pi(x-0.5)/\epsilon_3))(2 + 1.8\sin(2\pi(y-0.5)/\epsilon_3))},$$

where $\epsilon_1 = 1/5$, $\epsilon_2 = 1/13$, $\epsilon_3 = 1/17$ and $f = 1,\ x^4 - y^3 + 1,\ \frac{1}{x^2+y^2+1}$.

In Figure 1, we plot the coefficient and the decomposition. Relative errors are shown in Tables 1–3. From these figures, we can see that the coefficient oscillates very rapidly, which generates small scale features in the solution (*e.g.* $\frac{\partial F_1}{\partial x_1}$). The smooth part of F, g, is a summation of some piecewise linear nodal functions, see Figure 1c. The magnitude of χ is indeed small (around 10^{-2}), see Figure 1d. Thus, the conditions of Theorem 3.1 are satisfied. In Tables 1–3, we document the convergence of our method for three different f. We observe convergence in both L^2 and H^1 norms.

We remark that Theorem 3.1 does not give a specific rate of convergence. It is worthwhile to make the following observations on the convergence property of our method. Denote the exact solution as u_e, the solution constructed from the effective equation as u_m and its numerical approximation as u_n. Then the error consists of two parts, *i.e.* $||u_e - u_m|| + ||u_m - u_n||$. The first part is controlled by Theorem 3.1, and it gets smaller as we use a finer size. For fixed u_m, the second part converges at order $O(h^2)$ (L^2 norm) or order $O(h^1)$ (H^1 norm). However, as the mesh size h varies, we have different decompositions, and so u_m is not fixed. Thus, the overall rate of convergence is not necessarily $O(h^2)$ or $O(h^1)$. In all our numerical experiments, we observe

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

TABLE 1. Example 1 – Relative L^2 and H^1 errors, $f = 1$.

	L^2	Convergence order	H^1	Convergence order
$h = \frac{1}{8}$	2.38×10^{-2}	0	1.66×10^{-1}	0
$h = \frac{1}{16}$	7.13×10^{-3}	1.74	8.11×10^{-2}	1.03
$h = \frac{1}{32}$	3.46×10^{-3}	1.04	4.02×10^{-2}	1.01

TABLE 2. Example 1 – Relative L^2 and H^1 errors, $f = x^4 - y^3 + 1$.

	L^2	Convergence order	H^1	Convergence order
$h = \frac{1}{8}$	2.65×10^{-2}	0	1.80×10^{-1}	0
$h = \frac{1}{16}$	7.75×10^{-3}	1.77	8.83×10^{-2}	1.03
$h = \frac{1}{32}$	3.59×10^{-3}	1.11	4.35×10^{-2}	1.02

TABLE 3. Example 1 – Relative L^2 and H^1 errors, $f = \frac{1}{x^2+y^2+1}$.

	L^2	Convergence order	H^1	Convergence order
$h = \frac{1}{8}$	2.39×10^{-2}	0	1.67×10^{-1}	0
$h = \frac{1}{16}$	6.81×10^{-3}	1.81	7.99×10^{-2}	1.06
$h = \frac{1}{32}$	3.51×10^{-3}	0.96	4.03×10^{-2}	0.99

TABLE 4. Example 2 – Relative L^2 and H^1 errors, $f = 1$.

	L^2	Convergence order	H^1	Convergence order
$h = \frac{1}{8}$	2.23×10^{-2}	0	1.46×10^{-1}	0
$h = \frac{1}{16}$	5.78×10^{-3}	1.95	7.12×10^{-2}	1.04
$h = \frac{1}{32}$	2.17×10^{-3}	1.41	3.28×10^{-2}	1.12

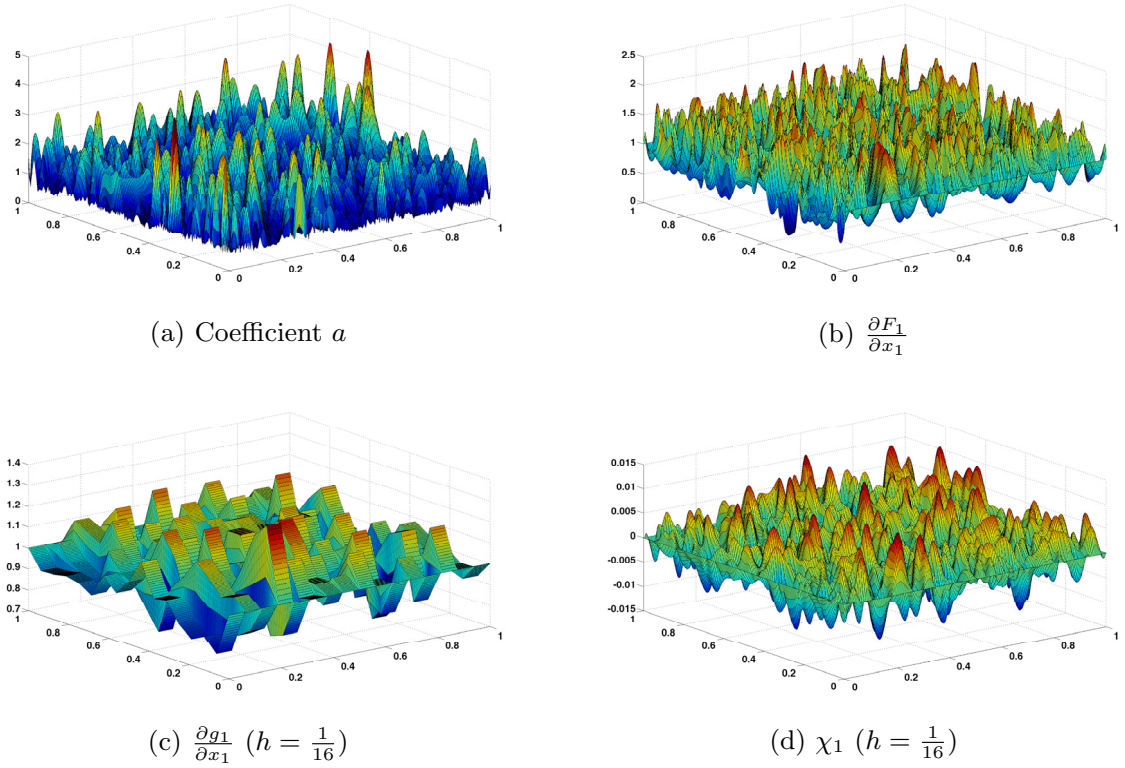
different rates of convergence for different mesh sizes, especially for the L^2 norm. On the finest mesh, the error will eventually converge to zero (if we take the numerical solution on the finest mesh to be the true solution). But we want to perform our method on a coarse mesh instead of a fine one. Our main objective is not to find the optimal convergence rate, but to reduce the error on a given coarse mesh. So we are more interested in the error itself rather than the convergence rate. It is important not to be confused with these two issues.

Example 2. We choose a random and anisotropic field a, and $f = 1$. In this case, $a = \begin{bmatrix} |\tilde{a}| + 0.5 & 0 \\ 0 & 1 \end{bmatrix}$, where $\tilde{a}$ is normally distributed on the mesh (See Fig. 2a). The multiscale coefficient, a, is very rough and does not satisfy scale separation or has any periodic structure. Compared with the first example, both the coefficient and the solution are more singular. This presents a challenging test problem for our method. As we can see from the convergence study presented in Table 4, our method still gives a satisfactory convergence rate and the relative errors are quite small.

Example 3. $a = \begin{cases} 10^4 & \text{in the channel} \\ 1 & \text{otherwise,} \end{cases}$ and $f = x^4 - y^3 + 1$.

Next, we consider an example that has a discontinuous and high contrast coefficient, see Figure 3. The channel is 0.02 wide in both x and y directions, and 0.4 long in x direction and 0.3 long in y direction. There

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press



(a) Coefficient a (b) $\frac{\partial F_1}{\partial x_1}$

(c) $\frac{\partial g_1}{\partial x_1}$ $(h = \frac{1}{16})$ (d) χ_1 $(h = \frac{1}{16})$

FIGURE 2. Example 2.

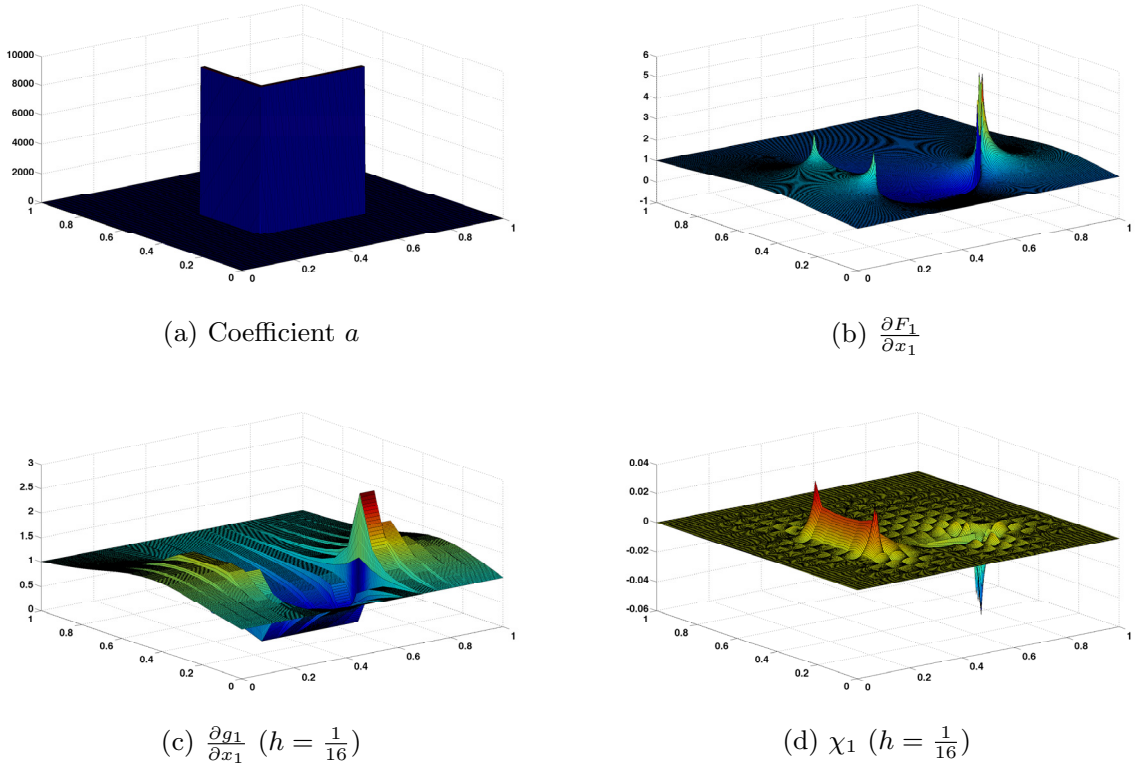
TABLE 5. Example 3 – Relative L^2 and H^1 errors, $f = x^4 - y^3 + 1$.

	L^2	Convergence order	H^1	Convergence order
$h = \frac{1}{8}$	3.24×10^{-2}	0	1.72×10^{-1}	0
$h = \frac{1}{16}$	7.47×10^{-3}	2.12	7.49×10^{-2}	1.20
$h = \frac{1}{32}$	2.92×10^{-3}	1.36	3.53×10^{-2}	1.09

has been a lot of interest in studying multiscale problems with high contrast coefficients in recent years, see *e.g.* [12, 14, 18, 20]. Inside the channel, the coefficient is very large, while the coefficient is small outside the channel. This is known to be a very difficult problem. As we can see, the solution along the channel is quite singular due to the discontinuity and high contrast of the coefficient. Even for such a challenging test problem, our method still captures the important feature of the solution accurately. As we can see from Table 5, the convergence rate remains to be very robust and the relative errors are very small.

Example 4. In this example, we will test our method for an elliptic equation with a parameter. We choose the coefficient to be in the following form (see also [18]) $a = (1 - \mu)a_1 + \mu a_2$, where a_1 and a_2 are both elliptic (plotted in Figs. 4a and 4b) and $0 \leq \mu \leq 1$. We choose $\mu = 0,\ 0.5,\ 1$ as three representative parameters and solve for the corresponding harmonic coordinates F. We express the general solution as a linear combination of such F as described before. Table 6 shows the errors for different μ. As we can see, the method works very well for several choices of the parameter. In Table 7, we show the convergence rate for different h with $\mu = 0.63$, which is consistent with what we observed previously for multiscale coefficients without parameters.

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press



(a) Coefficient a (b) $\frac{\partial F_1}{\partial x_1}$

(c) $\frac{\partial g_1}{\partial x_1}$ $(h = \frac{1}{16})$ (d) χ_1 $(h = \frac{1}{16})$

FIGURE 3. Example 3.

TABLE 6. Example 4 – Relative L^2 and H^1 errors at $h = \frac{1}{16}$, $f = 1$.

	$\mu = 0.11$	$\mu = 0.23$	$\mu = 0.61$	$\mu = 0.86$
L^2	2.90×10^{-2}	2.82×10^{-2}	2.07×10^{-2}	1.37×10^{-2}
H^1	1.33×10^{-1}	1.20×10^{-1}	9.33×10^{-2}	1.12×10^{-1}

TABLE 7. Example 4 – Relative L^2 and H^1 errors, $f = 1$, $\mu = 0.63$.

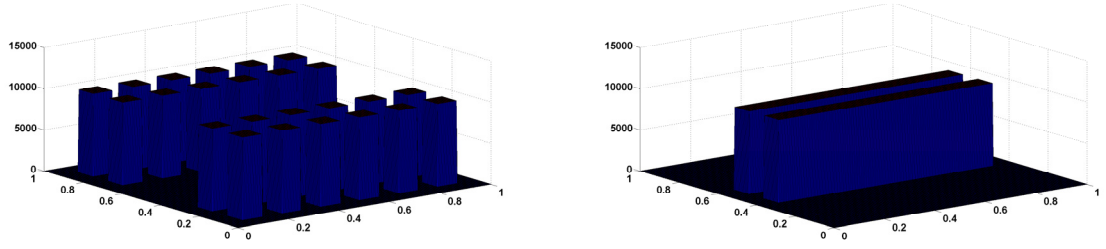
	L^2	Convergence order	H^1	Convergence order
$h = \frac{1}{8}$	6.07×10^{-2}	0	1.94×10^{-1}	0
$h = \frac{1}{16}$	2.01×10^{-2}	1.59	9.32×10^{-2}	1.06
$h = \frac{1}{32}$	1.16×10^{-2}	0.79	4.94×10^{-2}	0.92

5.3. Parabolic and convection-diffusion equations

In this section, we perform numerical experiments to test the convergence of our method for parabolic and convection-diffusion equations in two space dimensions.

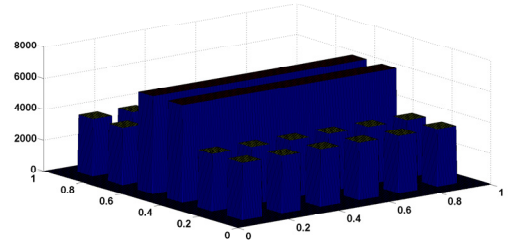
Example 5. A parabolic equation. $a = \frac{2+\sin(2\pi x/\epsilon_1)}{2+\sin(2\pi y/\epsilon_1)} + \frac{2+\sin(2\pi y/\epsilon_2)}{2+\sin(2\pi x/\epsilon_2)} + x^2 + y^2 + 1$ in $\Omega = [0,1] \times [0,1]$, with $\epsilon_1 = \frac{1}{5}$, $\epsilon_2 = \frac{1}{13}$ and $f = 1$. We choose a 128×128 mesh for the well-resolved solution, and compute the solution to $T = 0.1$. Table 8 shows the errors *versus* time with different coarse grid meshes. As we can see from Table 8,

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

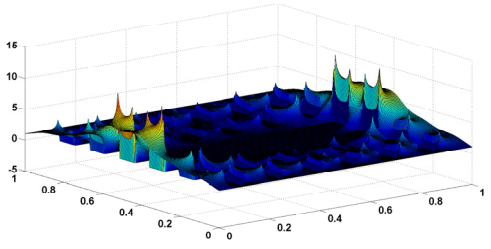


(a) a_1

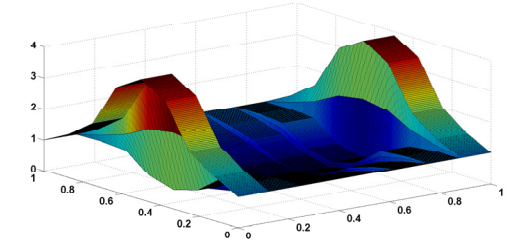
(b) a_2



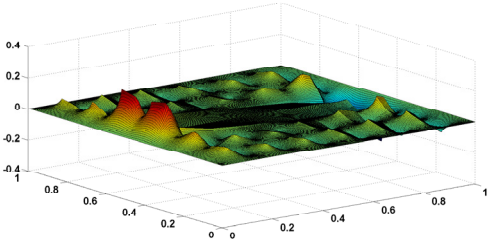
(c) Coefficient a



(d) $\frac{\partial F_1}{\partial x_1}$



(e) $\frac{\partial g_1}{\partial x_1}$ $(h = \frac{1}{16})$



(f) χ_1 $(h = \frac{1}{16})$

FIGURE 4. Example 4 ($\mu = 0.63$).

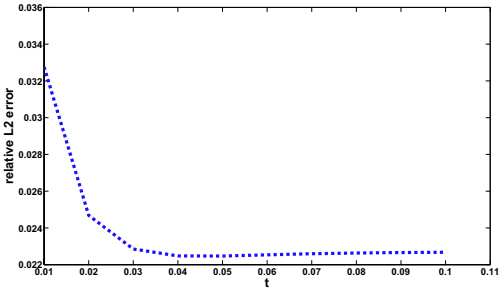
TABLE 8. Example 5 – Relative L^2 and H^1 errors at $T = 0.1$.

	L^2	Convergence order	H^1	Convergence order
$h = \frac{1}{8}$	2.27×10^{-2}	0	1.40×10^{-1}	0
$h = \frac{1}{16}$	5.93×10^{-3}	1.94	6.60×10^{-2}	1.08
$h = \frac{1}{32}$	2.44×10^{-3}	1.28	2.75×10^{-2}	1.26

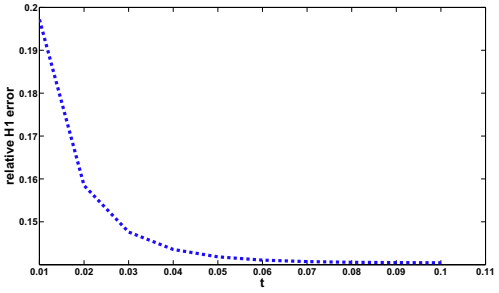
our method gives qualitatively the same performance as for the case of the elliptic equation. In Figure 6, we show the quantity $\int_\Omega |\nabla(\tilde{u}_0)_t|^2 \mathrm{d}x$ as time varies. Although we do not have an analytical bound for it based on the limited assumptions, we can see that it remains bounded.

In order to gain better understanding on the boundedness of the term $\nabla(\tilde{u}_0)_t$, we study several cases for different a and f and monitor the quantity $\int_0^T \int_\Omega |\nabla(\tilde{u}_0)_t|^2 \mathrm{d}x\mathrm{d}t$. In all cases, we choose a 128×128 mesh for

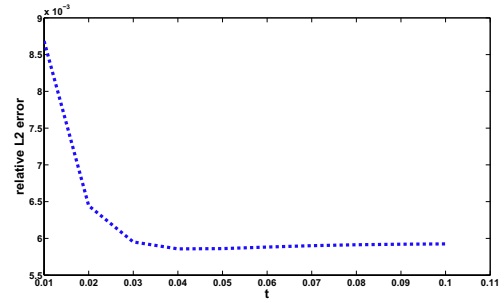
https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press



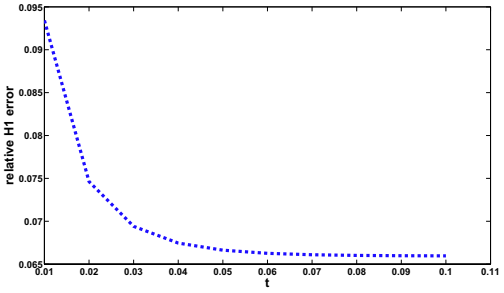
(a) Relative L^2 errors at $h = \frac{1}{8}$



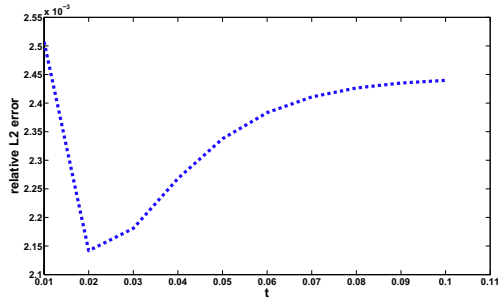
(b) Relative H^1 errors at $h = \frac{1}{8}$



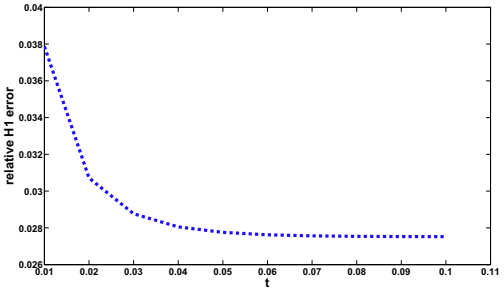
(c) Relative L^2 errors at $h = \frac{1}{16}$



(d) Relative H^1 errors at $h = \frac{1}{16}$



(e) Relative L^2 errors at $h = \frac{1}{32}$



(f) Relative H^1 errors at $h = \frac{1}{32}$

FIGURE 5. Example 5.

the well-resolved solution and a 16×16 mesh for the coarse grid, and $\Omega = [0,1] \times [0,1]$, $T = 0.1$. In the Table 9, $a_1 = \frac{1}{2+1.6\sin(\frac{2\pi(x-y)}{\epsilon_1})} + \frac{1}{4+1.8(\sin(\frac{2\pi x}{\epsilon_2})+\sin(\frac{2\pi y}{\epsilon_2}))}$, $a_2 = |\tilde{a}| + 0.5$, $f_1 = x^4 - y^3 + 1$, $f_2 = \frac{1}{x^2+y^2+1}$, where $\epsilon_1 = \frac{1}{5}$, $\epsilon_2 = \frac{1}{13}$ and $\tilde{a}$ is normally distributed on the mesh. As we can see from the table, the error is small and the quantity $\int_0^T \int_\Omega |\nabla(\tilde{u}_0)_t|^2 \mathrm{d}x\mathrm{d}t$ also remains bounded.

Example 6. A convection-diffusion equation. In this experiment, we choose the stream function $\psi = (1 + \sin(2\pi x/\epsilon))(1 + \cos(2\pi y/\epsilon)) + xy(\epsilon = \frac{1}{13})$ to generate the velocity filed. $u_0 = (\psi_y, -\psi_x)$ and $u = \frac{u_0}{\epsilon||u_0||_{L^\infty}}$. The initial condition is $\phi = xy(1-x)(1-y)$, and $\alpha = 0.05$. The domain is $\Omega = [0,1] \times [0,1]$ as before and the end time is $T = 0.1$. A 128×128 mesh grid is used for the well-resolved solution. Figure 7 shows the errors

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

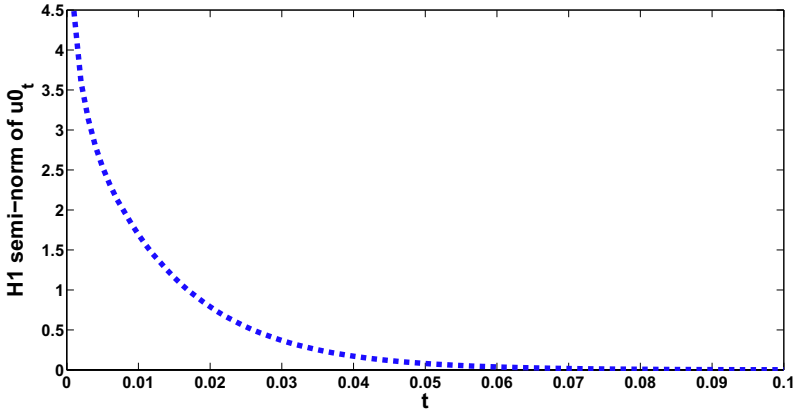


FIGURE 6. $\int_\Omega |\nabla(\tilde{u}_0)_t|^2 \mathrm{d}x$ in Example 5.

TABLE 9. $\int_0^T \int_\Omega |\nabla(\tilde{u}_0)_t|^2 \mathrm{d}x\mathrm{d}t$ for different a and f.

	L^2 error	H^1 error	$\int_0^T \int_\Omega \lvert\nabla(\tilde{u}_0)_t\rvert^2 \mathrm{d}x\mathrm{d}t$
a_1, f_1	8.27×10^{-3}	8.77×10^{-2}	6.52×10^{-1}
a_1, f_2	7.99×10^{-3}	7.97×10^{-2}	4.30×10^{-1}
a_2, f_1	8.03×10^{-3}	7.94×10^{-2}	6.09×10^{-1}
a_2, f_2	6.83×10^{-3}	7.22×10^{-2}	4.06×10^{-1}

TABLE 10. Example 6 – Relative L^2 and H^1 errors at $T = 0.1$.

	L^2	Convergence order	H^1	Convergence order
$h = \frac{1}{8}$	2.60×10^{-2}	0	1.41×10^{-1}	0
$h = \frac{1}{16}$	1.09×10^{-2}	1.25	9.23×10^{-2}	0.61
$h = \frac{1}{32}$	5.43×10^{-3}	1.01	6.91×10^{-2}	0.42

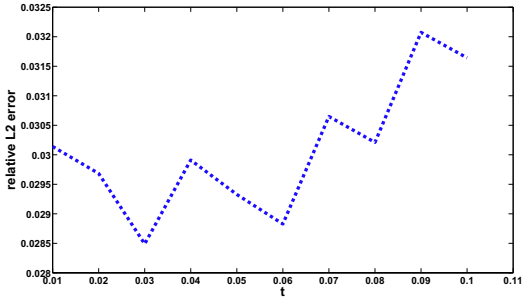
versus time with different coarse grid meshes. Figure 8 shows the boundedness of $\int_\Omega |\nabla(\tilde{v}_0)_t|^2 \mathrm{d}x$. Table 10 shows the error at $T = 0.1$. In this case, we observe that the errors are larger than those presented in the previous examples for the elliptic and the parabolic equations. The reason for this behavior is due to a mild degeneracy of ellipticity since the diffusion coefficient α is relatively small in this convection diffusion problem. From our convergence analysis, the error will be amplified by the smallest eigenvalue of the elliptic coefficient. In this case, the smallest eigenvalue is α, which is smaller than the ones we consider in the previous example. Given that the convection diffusion equation presents a tougher test problem for our method, the performance of our method is still quite encouraging.

5.4. Hyperbolic equations

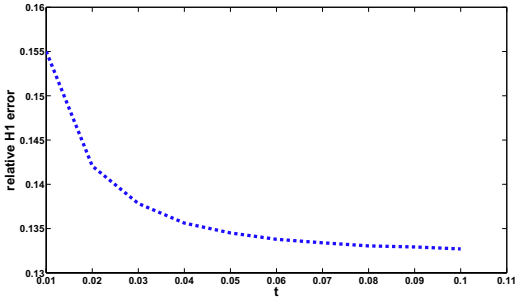
The multiscale model reduction method proposed in this paper can be extended to study hyperbolic partial differential equations with multiscale coefficients. Specifically, we consider the following hyperbolic equation

$$\begin{cases} u_{tt} - \nabla \cdot (a(x)\nabla u(x)) = f(x) & \text{in } (0,T] \times \Omega \\ u = 0 & \text{on } \partial\Omega \\ u = 0 & \text{at } t = 0 \\ u_t = 0 & \text{at } t = 0. \end{cases} \tag{5.2}$$

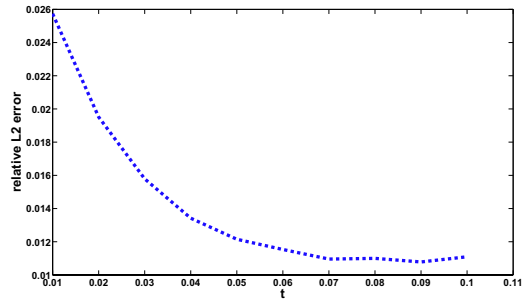
https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press



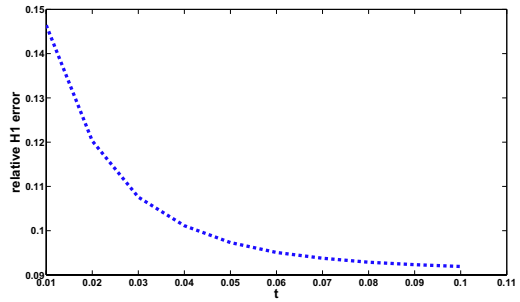
(a) Relative L^2 errors at $h = \frac{1}{8}$



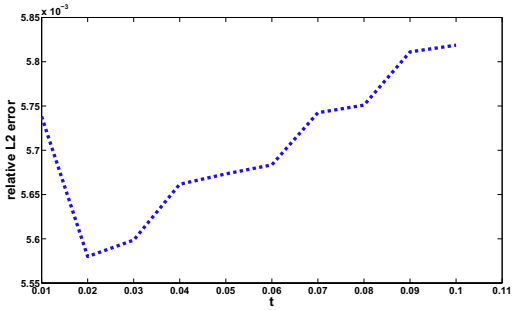
(b) Relative H^1 errors at $h = \frac{1}{8}$



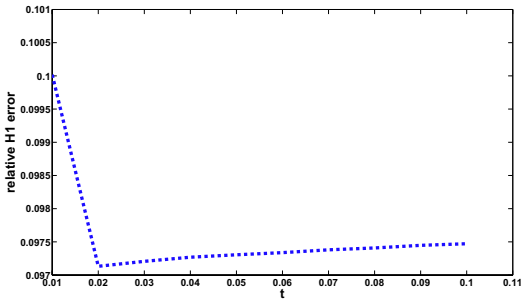
(c) Relative L^2 errors at $h = \frac{1}{16}$



(d) Relative H^1 errors at $h = \frac{1}{16}$



(e) Relative L^2 errors at $h = \frac{1}{32}$



(f) Relative H^1 errors at $h = \frac{1}{32}$

FIGURE 7. Example 6.

It is straightforward to generalize the derivation of our effective equation for an elliptic equation to this hyperbolic equation. The effective equation takes the form

$$\begin{cases} (u_0)_{tt} - \nabla \cdot (a(x)(I + \frac{\partial \chi}{\partial x}\frac{\partial x}{\partial g})\nabla u_0(x,t)) = f(x) & \text{in } (0,T] \times \Omega \\ u_0 = 0 & \text{on } \partial\Omega \\ u_0 = 0 & \text{at } t = 0 \\ (u_0)_t = 0 & \text{at } t = 0, \end{cases} \tag{5.3}$$

where $F = g + \chi$ is defined in the same way as before.

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

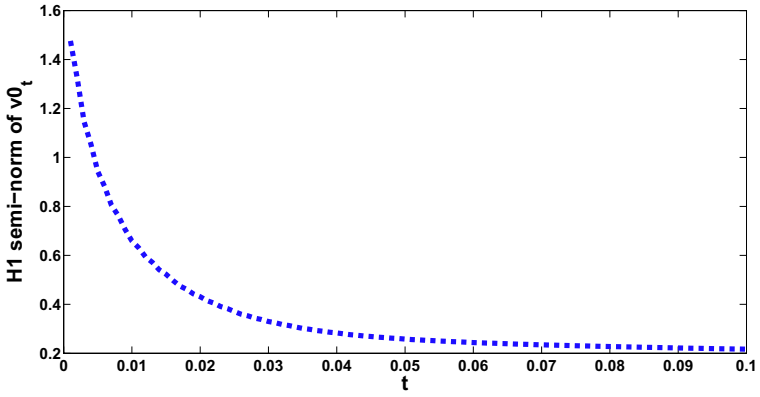


FIGURE 8. $\int_\Omega |\nabla(\tilde{v}_0)_t|^2 \mathrm{d}x$ in Example 6.

TABLE 11. Example 7 – Relative L^2 and H^1 errors at $T = 1$.

	L^2	Convergence order	H^1	Convergence order
$h = \frac{1}{8}$	2.23×10^{-2}	0	1.15×10^{-1}	0
$h = \frac{1}{16}$	7.03×10^{-3}	1.67	5.64×10^{-2}	1.03
$h = \frac{1}{32}$	2.87×10^{-3}	1.29	2.70×10^{-2}	1.06

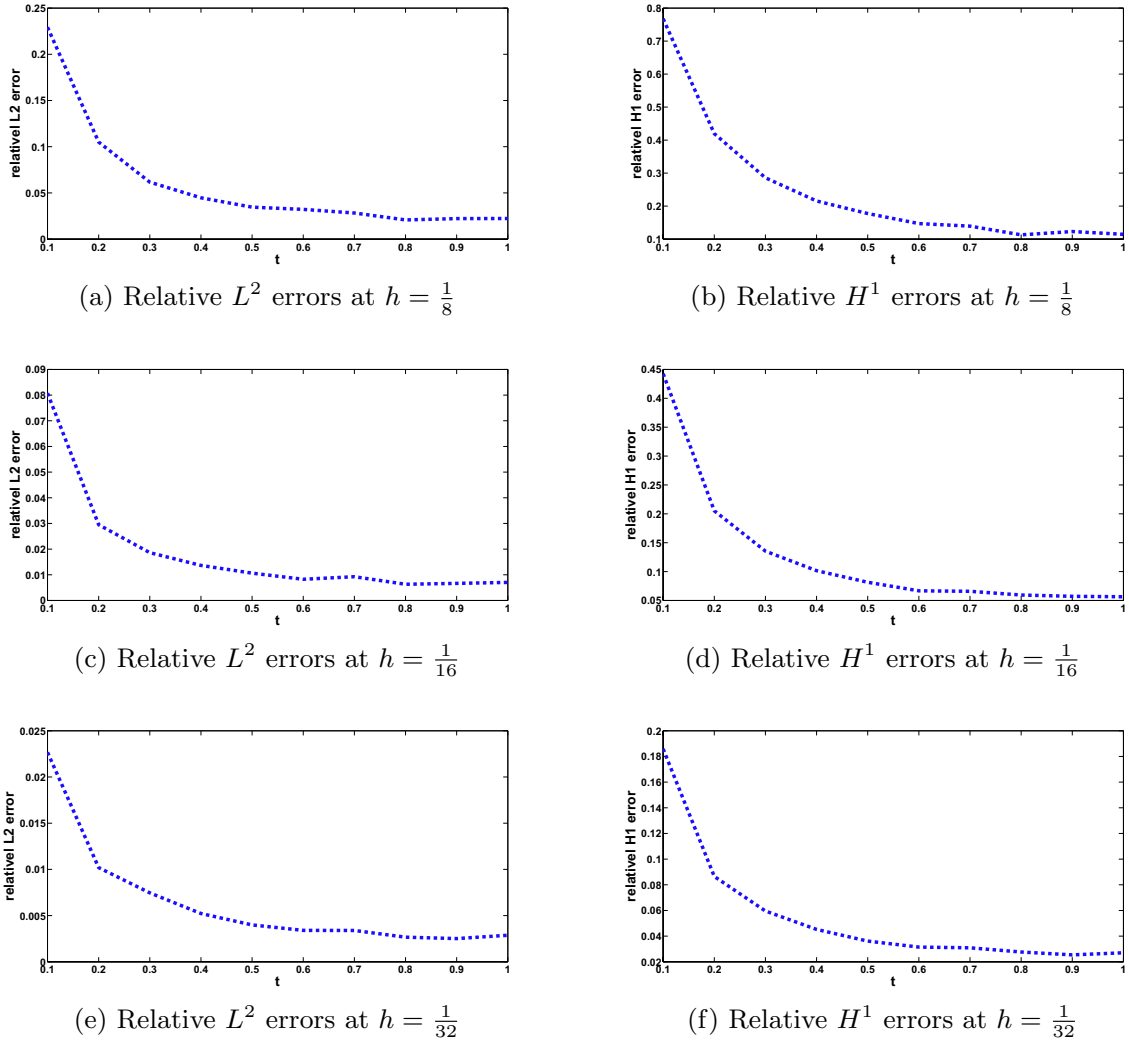
We remark that the convergence analysis of the effective equation for the hyperbolic equation is more complicated than that for the equations that we have considered so far. A straightforward generalization of our previous convergence analysis to the hyperbolic equation would require a stronger regularity assumption on the effective solution. We are currently investigating an alternative approach to perform the convergence analysis of the effective equation. On the other hand, our numerical results seem to indicate that our effective equation still gives a very accurate approximation to the multiscale solution of the hyperbolic PDE in both L^2 and H^1 norms.

Example 7. A hyperbolic equation. In this numerical experiment, we choose $a = \frac{1}{1+|\sin(2\pi x/\epsilon_1)|+|\cos(2\pi y/\epsilon_2)|}$, where $\epsilon_1 = \frac{1}{5}$, $\epsilon_2 = \frac{1}{13}$, and $f = 1$. The domain is still the same as before with $\Omega = [0,1] \times [0,1]$ and the end time is $T = 1$. A 128×128 fine mesh is used for the well-resolved solution. Figure 9 shows the errors *versus* time with different coarse grid meshes, and Table 11 shows the error at $T = 1$. As we can see from Figure 9 and Table 11, our method gives first order convergence in the H^1 norm and better than first order of convergence in the L^2 norm, which is consistent with the convergence rates that we observed earlier for elliptic and parabolic equations.

6. Concluding remarks

We have proposed a multiscale model reduction method for several standard types of elliptic, parabolic, hyperbolic, and convection-diffusion equations. A key ingredient of this method is to decompose the harmonic map into a smooth and invertible part, g, and a small remainder, χ. The effective equation is derived by taking into the account of the interaction between the multiscale coefficient of the governing equation and χ. One advantage of this approach is that we do not require scale separation or periodic structures as in traditional homogenization theory. In the case when the multiscale coefficient has the scale separation property and a periodic structure, our method recovers the homogenized equation. Another advantage is that our effective equation can be solved on a regular coarse mesh and it is easy to implement. An efficient decomposition method

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press



(a) Relative L^2 errors at $h = \frac{1}{8}$

(b) Relative H^1 errors at $h = \frac{1}{8}$

(c) Relative L^2 errors at $h = \frac{1}{16}$

(d) Relative H^1 errors at $h = \frac{1}{16}$

(e) Relative L^2 errors at $h = \frac{1}{32}$

(f) Relative H^1 errors at $h = \frac{1}{32}$

FIGURE 9. Example 7.

has been proposed to decompose the harmonic map into a smooth part plus a small remainder. Under some assumptions on the multiscale coefficient, we analyze the error between the effective solution and the original multiscale solution and show that the error is small in the H^1 norm as long as χ is small and g is smooth. Several numerical examples have been given to demonstrate the robustness and the accuracy of the proposed method. Our numerical results seem to indicate that our method gives an accurate and efficient approximation to the multiscale solution even when the multiscale coefficient does not have the scale separation property or a periodic structure.

There are still some limitations of the current method. One most challenging issue is how to perform the decomposition of the harmonic map when the harmonic map is not invertible due to irregular geometries or three dimensionality. Although our method does not require the harmonic map, F, to be invertible, finding an optimal decomposition, $F = g + \chi$, such that g is smooth and invertible while keeping χ small becomes more challenging when F is not invertible in some local region. One way to overcome this difficulty is to apply our

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

model reduction method locally instead of globally. By using a local mesh refinement, we can capture some nearly singular behavior of F by a locally well-resolved g and still keep the remainder χ small. Another way is to develop an optimization method to generate the optimal g iteratively as we indicated in the numerical section. We can also use a residual error correction to further reduce the error if χ is not small in some localized region due to the degeneracy of F. These issues will be further investigated in our future work.

Acknowledgements. This research was supported in part by a DOE Grant DE-FG02-06ER25727, a AFOSR MURI Grant FA9550-09-1-0613 and an NSF FRG Grant DMS-1159138. The research of Dr. Z. Shi was in part supported by a NSFC Grant 11201257.

References

[1] G. Alessandrini and V. Nesi, Univalent σ-harmonic mappings. *Arch. Rational Mech. Anal.* **158** (2001) 155–171.
[2] G. Allaire and R. Brizzi, A multiscale finite element method for numerical homogenization. *SIAM MMS* **4** (2005) 790–812.
[3] A. Ancona, Some results and examples about the behavior of harmonic functions and Green's funtions with respect to second order elliptic operators. *Nagoya Math. J.* **165** (2002) 123–158.
[4] T. Arbogast, G. Pencheva, M.F. Wheeler and I. Yotov, A multiscale mortar mixed finite element method. *SIAM MMS* **6** (2007) 319–346.
[5] I. Babuška and E. Osborn, Generalized Finite Element Methods: Their Performance and Their Relation to Mixed Methods. *SIAM J. Numer. Anal.* **20** (1983) 510–536.
[6] I. Babuška, G. Caloz and E. Osborn, Special finite element methods for a class of second order elliptic problems with rough coefficients. *SIAM J. Numer. Anal.* **31** (1994) 945–981.
[7] M. Barrault, Y. Maday, N.C. Nguyen and A.T. Patera, An Empirical Interpolation Method: Application to Efficient Reduced-Basis Discretization of Partial Differential Equations. *C.R. Acad. Sci. Paris Series I* **339** (2004) 667–672.
[8] A. Bensoussan, J.-L. Lions and G. Papanicolaou, *Asymptotic Analysis for Periodic Structure*. North-Holland, Amsterdam (1978).
[9] S. Boyaval, C. LeBris, T. Lelièvre, Y. Maday, N. Nguyen and A. Patera, Reduced Basis Techniques for Stochastic Problems. *Arch. Comput. Meth. Eng.* **17** (2012) 435–454.
[10] Y. Chen, L.J. Durlofsky, M. Gerritsen and X.H. Wen, A coupled local-global upscaling approach for simulating flow in highly heterogeneous formations. *Advances in Water Resources* **26** (2003) 1041–1060.
[11] Z. Chen and T.Y. Hou, A mixed multiscale finite element method for elliptic problems with oscillating coefficients. *Math. Comput.* **72** (2002) 541–576.
[12] C.C. Chu, I. Graham and T.Y. Hou, A New multiscale finite element method for high-contrast elliptic interface problems. *Math. Comput.* **79** (2010) 1915–1955.
[13] W. E and B. Engquist, The heterogeneous multi-scale methods. *Commun. Math. Sci.* **1** (2003) 87–133.
[14] Y. Efendiev, J. Galvis and X.H. Wu, Multiscale finite element methods for high-contrast problems using local spectral basis functions. *J. Comput. Phys.* **230** (2011) 937–955.
[15] Y. Efendiev, V. Ginting, T. Hou and R. Ewing, Accurate multiscale finite element methods for two-phase flow simulations. *J. Comput. Phys.* **220** (2006) 155–174.
[16] Y. Efendiev and T. Hou, Multiscale finite element methods. Theory and applications. Springer (2009).
[17] Y. Efendiev, T.Y. Hou and X.H. Wu, Convergence of a nonconforming multiscale finite element method. *SIAM J. Num. Anal.* **37** (2000) 888–910.
[18] Y. Efendiev, J. Galvis and T.Y. Hou, Generalized multiscale finite element methods (GMsFEM). Accepted by *JCP* (2013).
[19] J. Galvis and Y. Efendiev, Domain decomposition preconditioners for multiscale flows in high-contrast media: Reduced dimension coarse spaces. *SIAM MMS* **8** (2009) 1621–1644.
[20] I.G. Graham, P.O. Lechner and R. Scheichl, Domain decomposition for multiscale PDEs. *Numer. Math.* **106** (2007) 589–626.
[21] T.Y. Hou and X.H. Wu, A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.* **134** (1997) 169–189.
[22] T.Y. Hou, X.H. Wu and Z. Cai, Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients. *Math. Comput.* **68** (1999) 913–943.
[23] T. Hughes, G. Feijoo, L. Mazzei and J. Quincy, The variational multiscale method – a paradigm for computational mechanics. *Comput. Methods Appl. Mech. Engrg.* **166** (1998) 3–24.
[24] P. Jenny, S.H. Lee and H. Tchelepi, Multi-scale finite volume method for elliptic problems in subsurface flow simulation. *J. Comput. Phys.* **187** (2003) 47–67.
[25] Y. Maday, *Proceedings of the International Conference Math.*, Madrid. European Mathematical Society, Zurich (2006).
[26] A. Maugeri, D.K. Palagachev and L.G. Softova. Elliptic and parabolic equations with discontinuous coefficients. *Math. Research* **109**, Wiley-VCH (2000).

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press

[27] D.W. Mclaughlin, G.C. Papanicolaou and O.R. Pironneau, Convetion of mircrostructure and related problems. *SIAM J. Appl. Math.* **45** (1985) 780–797.

[28] S. Moskow and M. Vogelius, First-oder corrections to the homogenised eigenvalues of a periodic composite medium. A convergence proof. *Proc. Roy. Soc. Edinburgh.* **127A** (1997) 1263–1299.

[29] H. Owhadi and L. Zhang, Metric based up-scaling. *Commun. Pure Appl. Math.* **LX** (2007) 675–723.

[30] H. Owhadi and L, Zhang, Homogenization of parabolic equations with a continuum of space and time scales. *SIAM J. Numer. Anal.* **46** (2007) 1–36.

[31] G. Rozza, D.B.P. Huynh and A.T. Patera, Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Application to transport and continuum mechanics. *Arch. Comput. Methods Eng.* **15** (2008) 229–275.

https://doi.org/10.1051/m2an/2013115 Published online by Cambridge University Press