A FRAMEWORK FOR ADAPTIVE MULTISCALE METHODS FOR ELLIPTIC PROBLEMS

JAMES NOLEN*, GEORGE PAPANICOLAOU†, AND OLIVIER PIRONNEAU‡

Abstract. We describe a projection framework for developing adaptive multiscale methods for computing approximate solutions to elliptic boundary value problems. The framework is consistent with homogenization when there is scale separation. We introduce an adaptive form of the finite element algorithms for solving problems with no clear scale separation. We present numerical simulations demonstrating the effectiveness and adaptivity of the multiscale method, assess its computational complexity, and discuss the relationship between this framework and other multiscale methods, such as wavelets, multiscale finite element methods, and the use of harmonic coordinates. We prove in detail that the projection based method captures homogenization when there is strong scale separation.

Key words. Multiscale finite elements, numerical homogenization, variational multiscale method

AMS subject classifications. 65N30, 35J20, 35B27

1. Introduction. In this paper we describe a framework for developing adaptive multiscale methods for approximating solutions to the linear elliptic boundary value problem

$$\begin{aligned} -\nabla \cdot (a(x)\nabla u) &= f, \quad x \in D \\ u &\equiv 0, \quad x \in \partial D \end{aligned} \tag{1.1}$$

posed in the bounded domain $D \subset R^d$. We assume that $f \in H^{-1}(D)$ and that the matrix $a(x) = (a_{ij}(x))$ is symmetric, measurable, and positive definite so that $0 < a_*|\xi|^2 \leq \xi^T a(x)\xi \leq a^*|\xi|^2$ holds at almost every $x \in D$ and any vector $\xi \in R^d$ with $|\xi| > 0$. Throughout this paper we use (u, v) to denote the $L^2(D)$ inner product, and we use $\langle f, v\rangle$ to denote the action of $f \in H^{-1}(D)$ on $v \in H_0^1(D)$.

We want to approximate the solution when the coefficients have small-scale features that have a significant effect on the solution. However, the full resolution of these features may require an enormous computational effort. The Galerkin approach to approximating u is to restrict the problem to a finite dimensional subspace $X_C \subset H_0^1$. So, the Galerkin solution $u_{CG} \in X_C$ satisfies

$$(a\nabla u_{CG}, \nabla v) = \langle f, v\rangle, \quad \forall v \in X_C. \tag{1.2}$$

Standard estimates for the Galerkin problem show that

$$\|u - u_{CG}\|_{H_0^1} \leq \frac{a^*}{a_*} \inf_{v \in X_C} \|u - v\|_{H_0^1} = \frac{a^*}{a_*} \|u - \mathcal{P}u\|_{H_0^1}$$

where $\mathcal{P}$ denotes the orthogonal projection of H_0^1 onto the space X_C (orthogonal with respect to the H_0^1 inner product). Therefore, the error between u and the Galerkin solution u_{CG} may be larger than the error between u and $\mathcal{P}u$ by a very large factor (a^*/a_*) that depends on the coefficients.

*Department of Mathematics, Stanford University, Stanford, CA 94305 (jnolen@math.stanford.edu). Supported by an NSF Postdoctoral Fellowship.

†Department of Mathematics, Stanford University, Stanford, CA 94305 (papanico@math.stanford.edu). Supported by grants ONR N00014-02-1-0088 and NSF DMS-0354674-001.

‡Laboratoire Jacques-Louis Lions, Université Pierre et Marie Curie (Paris VI), Boîte courrier 187, 75252 Paris Cedex 05, France (pironneau@ann.jussieu.fr).

Many methods have been proposed to approximate u on a coarse grid while reconstructing relevant information from the small scales. We refer to the work of Brezzi and Hughes, et. al. [9], [26], Hou and Wu [22], Arbogast [2], Engquist and Runborg [17], as well as [8, 23, 15, 24, 14, 1, 31]. When the coefficients are spatially periodic and have the form $a(x/\epsilon)$ these methods may capture the small scale effects very well in the limit $\epsilon \to 0$, as demonstrated numerically and proved analytically in [3, 4, 22, 23, 15, 24, 14, 28, 1, 31]. That is, the approximate solutions converge to the solution of the homogenized equation as long as ϵ is much smaller than the discretization scale.

It is desirable to have an effective multiscale method that does not rely on a periodic structure or on scale separation in the coefficients. The general framework we describe here is based on a decomposition of the space H_0^1 into coarse and fine scales, as is often done with the Lyapunov-Schmidt reduction for nonlinear problems (e.g. [20]). In the context of multiscale methods, the framework we analyze is closely related to the ones described by Hughes [26, 27] and by Engquist and Runborg [17]. The purpose of this paper is to (a) put the projection framework in a form compatible with scale separation and homogenization, (b) introduce an adaptive form of the algorithm for reconstructing the small scale information and assess its computational complexity, (c) implement the algorithm with public domain finite element software [18], and (d) prove the consistency of the projection framework with the homogenization limit. The numerical computations show clearly the spatial localization of the influence of small scales for both periodic and random coefficients. We also see that this localization cannot be known in advance and must be estimated adaptively during the computations.

In Section 2 we introduce the projection framework and in Section 3 we describe various approximation schemes. The projection framework gives us an effective coarse scale equation which is an "upscaled" equation for the coarse-scale features in the solution. Solving the effective equation involves computing an operator that maps the coarse scales to fine scales. This is done by solving a set of locally refined problems that have the form of an elliptic problem with side constraints.

In Section 4 we show the results of numerical computations using FreeFem++ [18]. The projection method and the spatial localization algorithm for the small scale reconstruction operator perform quite well and can dramatically reduce the solution error, measured in both L^2 and H^1 norms. The computational complexity analysis that we provide suggests that the method is scalable and useful for very large problems that are difficult to solve directly.

In Section 5, we compare the projection framework to homogenization theory and to existing multiscale methods. In particular, we prove that in the homogenization limit of strong scale separation, the projection framework with spatial localization has the same form of the homogenized problem in the projection framework. This shows that the projection framework captures correctly the multiscale features of the solution as well as the spatial localization of the small scales that reduces computational complexity.

2. Variational Multiscale Framework. The variational multiscale approach (see [26], [27], [8], [9], [2]) gives a general framework for approximating the course scale function Pu and reconstructing some fine scale features of the solution simultaneously. The framework is based on a direct sum decomposition of the space into coarse scales and fine scales. Let X_C be a finite-dimensional, closed subspace of $H_0^1(D)$ that represents coarse scales that may not fully resolve the oscillations in the solution u. Let

$X_R \subset H_0^1$ be a larger closed space such that $X_C \subset X_R$, and X_R is sufficient to resolve the fine scales in u within a desired tolerance. For now, we assume $X_R = H_0^1$; in a numerical implementation of this framework, X_R may be a finite dimensional space corresponding to full refinement of the details in the coefficients and data.

Let $\mathcal{P}_C : X_R \to X_C$ be a projection of X_R onto X_C (not necessarily orthogonal with respect to the H_0^1 inner product) so that every $u \in X_R$ can be decomposed uniquely as

$$u = \mathcal{P}_C u + \mathcal{Q}_C u = \text{coarse approximation} + \text{details}.$$

where $\mathcal{Q}_C = (\mathcal{I} - \mathcal{P}_C)$. We will use X_F to denote the image of the map $\mathcal{Q}_C$. If $\mathcal{P}_C$ is an orthogonal projection with respect to some inner product, then $X_F = X_C^{\perp}$. In any case, $X_R = X_C \oplus X_F$.

The variational multiscale approach is to replace the quadratic form (1.2) with an effective quadratic form over the coarse space X_C. The solution $u \in X_R$ satisfies

$$(a\nabla u, \nabla v) = \langle f, v\rangle \ \ \forall\, v \in X_R, \tag{2.1}$$

so by choosing $v \in X_C$ or $v \in X_F$, we see that u satisfies both

$$(a\nabla(\mathcal{P}_C u + \mathcal{Q}_C u), \nabla v) = \langle f, v\rangle \ \ \forall\, v \in X_C, \tag{2.2}$$

$$(a\nabla \mathcal{Q}_C u, \nabla v) = \langle f, v\rangle - (a\nabla \mathcal{P}_C u, \nabla v) \ \ \forall\, v \in X_F. \tag{2.3}$$

2.1. The Reconstruction Operator. Here is the simple observation defining the multiscale method: for a given function $u \in X_R$, equation (2.3) defines $\mathcal{Q}_C u$ as an affine functional of $\mathcal{P}_C u$. In fact, $\mathcal{Q}_C u$ is an affine functional of $\nabla \mathcal{P}_C u$. This suggests defining the operator $\mathcal{M} : \nabla X_C \to X_F$ by letting $\mathcal{M}(\nabla \mathcal{P}_C u)$ be the unique function in X_F that satisfies

$$(a\nabla \mathcal{M}(\nabla \mathcal{P}_C u), \nabla v) = \langle f, v\rangle - (a\nabla \mathcal{P}_C u, \nabla v) \ \ \forall\, v \in X_F. \tag{2.4}$$

Now, finding the coarse component $\mathcal{P}_C u$ of the solution to the system (2.2) and (2.3) is equivalent to finding $u_C \in X_C$ that satisfies

$$(a(\mathcal{I} + \nabla\mathcal{M})\nabla u_C, \nabla v) = \langle f, v\rangle \ \ \forall\, v \in X_C. \tag{2.5}$$

By $(\mathcal{I} + \nabla\mathcal{M})\nabla u_C$ we mean $\nabla u + \nabla(\mathcal{M}(\nabla u_C))$. Using the definition of $\mathcal{M}$ and the fact that $\mathcal{M}(\nabla v) \in X_F$ whenever $v \in X_C$, we see that equation (2.5) can also be written in the symmetric form

$$(a(\mathcal{I} + \nabla\mathcal{M})\nabla u_C, (\mathcal{I} + \nabla\mathcal{M})\nabla v) = \langle f, v + \mathcal{M}\nabla v\rangle \ \ \forall\, v \in X_C. \tag{2.6}$$

Because (2.6) may be well-posed even when we approximate $\mathcal{M}$ by $\tilde{\mathcal{M}}$, we will use (2.6) (rather than (2.5)) as the definition of the **effective variational problem at the level** X_C.

Because of the ellipticity assumption, the solution u_C exists and is unique. From (2.2) and (2.3), it is clear that if $u \in X_R$ solves the original problem (2.1), then $\mathcal{M}(\nabla \mathcal{P}_C u) = \mathcal{Q}_C u$. Using the terminology of E and Engquist [13], we will refer to $\mathcal{M}$ as the **reconstruction operator**, since it reconstructs the fine scales of the solution $(\mathcal{Q}_C u)$ from the coarse scale component $(\mathcal{P}_C u)$. This operator is also called the fine scale Green's function in [26]. As we will explain in Section 5.1, it is also related to the corrector function in homogenization theory.

The reconstruction operator $\mathcal{M}$ is an affine operator of the form $\mathcal{M}(\nabla u) = \mu_F + \mathcal{M}^o(\nabla u)$, where the $\mu_F \in X_F$ is independent of u_C and satisfies

$$(a\nabla\mu_F, \nabla v) = \langle f, v\rangle \ \ \forall\, v \in X_F. \tag{2.7}$$

The operator $\mathcal{M}^o : \nabla X_C \to X_F$ is the bounded linear operator defined by (2.4) with $f \equiv 0$. Hence, $\|\mathcal{M}^o\|_{\nabla X_C, X_F} \leq \frac{Ca^*}{a_*}$. If $\langle f, v\rangle = 0$ for all $v \in X_F$, the constant part of the operator $\mathcal{M}$ vanishes: $\mu_F \equiv 0$. In any case, the μ_F terms in the coarse scale equation cancel. That is, solving

$$(a(\mathcal{I} + \nabla\mathcal{M})\nabla u_C, (\mathcal{I} + \nabla\mathcal{M})\nabla v) = \langle f, v + \mathcal{M}\nabla v\rangle \ \ \forall\, v \in X_C, \tag{2.8}$$

is equivalent to solving the equation

$$(a(\mathcal{I} + \nabla\mathcal{M}^o)\nabla u_C, (\mathcal{I} + \nabla\mathcal{M}^o)\nabla v) = \langle f, v + \mathcal{M}^o\nabla v\rangle \ \ \forall\, v \in X_C. \tag{2.9}$$

This follows from the fact that $(a\nabla\mu_F, \nabla\mu_F) = \langle f, \mu_F\rangle$ and for $u_C, v \in X_C$, $(a(\mathcal{I} + \nabla\mathcal{M}^o)\nabla u_C, \nabla\mu_F) = 0 = (a\nabla\mu_F, (\mathcal{I} + \nabla\mathcal{M}^o)\nabla v)$, since $\mu_F \in X_F$ and a is symmetric.

2.2. The Multiscale Equations. In summary, this analysis shows that the solution $u \in X_R = X_C \oplus X_F$ may be written uniquely as $u = u_C + \mathcal{M}(\nabla u_C)$ where $u_C \in X_C$ and $\mathcal{M}(\nabla u_C) \in X_F$ are determined by the relation $\mathcal{M}(\nabla v) = \mu_F + \mathcal{M}^o(\nabla v)$ and the following three equations:

(i) **Effective coarse scale equation:**

$$(a(\mathcal{I} + \nabla\mathcal{M})\nabla u_C, \nabla v) = \langle f, v\rangle \ \ \forall\, v \in X_C. \tag{2.10}$$

or, the equivalent symmetrized and reduced form:

$$(a(\mathcal{I} + \nabla\mathcal{M}^o)\nabla u_C, (\mathcal{I} + \nabla\mathcal{M}^o)\nabla v) = \langle f, v + \mathcal{M}^o\nabla v\rangle \ \ \forall\, v \in X_C. \tag{2.11}$$

(ii) **Fine scale equation defining the linear operator $\mathcal{M}^o$:**

$$(a\nabla(\mathcal{M}^o\nabla w), \nabla v) = -(a\nabla w, \nabla v) \ \ \forall\, v \in X_F,\ w \in X_C. \tag{2.12}$$

or, equivalently:

$$(a(\mathcal{I} + \nabla\mathcal{M}^o)\nabla w), \nabla v) = 0 \ \ \forall\, v \in X_F,\ w \in X_C. \tag{2.13}$$

(iii) **Fine scale equation defining the function $\mu_F \in X_F$:**

$$(a\nabla\mu_F, \nabla v) = \langle f, v\rangle \ \ \forall\, v \in X_F. \tag{2.14}$$

Therefore, if one wants to approximate the coarse scale component u_C, it suffices to approximate the solution to the system formed by the first two equations (2.11) and (2.12). To reconstruct the entire solution $u = u_C + \mathcal{M}^o(\nabla u_C) + \mu_F$, one must also approximate the function μ_F which solves (2.14). The μ_F term is nonzero, in general, and may be quite large. So, f may have a nontrivial effect on both the coarse scale component u_C and the fine scale component μ_F, depending on the projection $\mathcal{P}_C$ and the H^{-1} norm of f. In the coarse scale equation (2.11), the fine scales of f enter the equation through coupling in the term $\langle f, \mathcal{M}^o(\nabla v)\rangle$, since $\mathcal{M}^o\nabla v \in X_F$ has fine scale oscillations. We discuss this issue further in Section 4.3 and Section 5.3.

The coarse scale equation (2.11) is well-posed, as long as there is an $\gamma > 0$ such that

$$\min_{v \in X_F} \|u - v\|_{H^1_0} > \gamma \|u\|_{H^1_0} \tag{2.15}$$

holds for any $u \in X_C$. In this case, the bilinear form (2.11) is coercive on X_C. For example, (2.15) holds if $\mathcal{P}_C$ is the H^1_0 projection, as described in the next section.

2.3. The Projection $\mathcal{P}_C$. The coarse scale solution u_C produced by the multiscale scheme depends on the choice of the projection $\mathcal{P}_C$, which has not yet been specified. If we compute the operator $\mathcal{M}$ exactly, then $u_C = \mathcal{P}_C u$ where u is the solution produced by full resolution in the space X_R. If $\mathcal{P}_C$ is an orthogonal projection with respect to an inner product $(\cdot,\cdot)_{\mathcal{P}_C}$, then u_C will be the best approximation to u in the space X_C with respect to the norm induced by this inner product. That is

$$\|u_C - u\|_{\mathcal{P}_C} = \inf_{v \in X_C} \|v - u\|_{\mathcal{P}_C} \tag{2.16}$$

where $\|v\|^2_{\mathcal{P}_C} = (v,v)_{\mathcal{P}_C}$. In any case, if $\mathcal{M}$ is computed exactly, the fully reconstructed solution is independent of $\mathcal{P}_C$, since u is the standard Galerkin solution in this (highly refined) space and $u = u_C + \mathcal{M}(\nabla u_C) = \mathcal{P}_C u + \mathcal{Q}_C u$. Therefore, in light of these observations, if one wants to minimize the H^1 error in the approximation of u_C, the H^1 projection should be used.

Projections other than the H^1 projection might be used, but they may give poor results. For example, if we choose $\mathcal{P}_C$ to be the orthogonal projection with respect to the inner product $(u,v)_{\mathcal{P}_C} = (a\nabla u, \nabla v)$, then u_C is equal to the Galerkin solution in the coarse space X_C. So, at the level of the coarse scale equation the method will be equivalent to the Galerkin method. In this case, $\mathcal{M}^o(\nabla u) \equiv 0$, and the fine scale reconstruction is contained entirely in the term μ_F (i.e. $\mathcal{M}(\nabla u) = \mu_F$).

For the rest of the paper we will assume $\mathcal{P}_C$ is the H^1_0 projection.

3. Approximating the Fine Scales. So far, there is no approximation of u: if we were able to compute $\mathcal{M}(\nabla u) = \mathcal{M}^o(\nabla u) + \mu_F$ exactly, then $u_C = \mathcal{P}_C u$, where $u \in X_R$ is the solution to the original problem in full detail. Moreover, we could reconstruct the full solution by $u = u_C + \mathcal{M}(\nabla u_C)$. So, we don't gain anything computationally unless we find an approximation of the fine scale reconstruction operator $\mathcal{M}$ that allows us to solve the effective problem (2.5) efficiently. Here we describe various approximation strategies. In Section 5 we will describe their relation to existing methods and to homogenization theory.

Let $\{\phi^C_k\}_{k=1}^{N_c}$ be a basis for the coarse space X_C (finite elements or wavelets). The linear part, $\mathcal{M}^o$, of the reconstruction operator is determined completely by its action on the functions $\{\nabla\phi^C_k\}$. For each k, $\mathcal{M}^o(\nabla\phi^C_k)$ is a function in X_F, defined by the equation

$$(a\nabla\mathcal{M}^o(\nabla\phi^C_k), \nabla v) = -(a\nabla\phi^C_k, \nabla v) \;\; \forall\, v \in X_F. \tag{3.1}$$

The operator $\mathcal{M}^o$ is non-local (this was pointed out in [27], in which $\mathcal{M}^o$ is called the "fine scale Green's function"). This stems from the fact that there are functions $v \in X_R$ such that the supports of $\mathcal{Q}_C v$ and ϕ^C_k intersect, while the support of $\mathcal{Q}_C v$ is not contained in the support of ϕ^C_k.

For each k, we approximate $\mathcal{M}^o(\nabla\phi^C_k)$ by restricting the test space and trial space in problem (3.1). That is, we replace $\mathcal{M}^o(\nabla\phi^C_k)$ by the function $\tilde{\mathcal{M}}^o(\nabla\phi^C_k) \in Y^F_k \subset X_F$ which satisfies

$$(a\nabla\tilde{\mathcal{M}}^o\phi^C_k, \nabla v) = -(a\nabla\phi^C_k, \nabla v) \;\; \forall\, v \in Y^F_k. \tag{3.2}$$

The space Y^F_k is a proper subset of X_F which is somehow localized near the support of ϕ^C_k. If we choose $Y^F_k = X_F$ for all k, then there is no approximation: $\tilde{\mathcal{M}}^o = \mathcal{M}^o$. However, if we choose Y^F_k to be the set of $v \in X_F$ that vanish outside the support of ϕ^C_k, then we have completely localized the problem by ignoring the effect of long-range coupling between fine scale features in the solution.

3.1. The Spaces X_C and X_F. To describe this approach more precisely, we set $d = 2$ and let $\mathcal{T}_C$ denote a conforming triangulation of the set D (we could also use rectangular elements). For each node x_k^C, $k = 1, \ldots, N_c$, let $S_k^C \subset D$ denote the union of triangles that have x_k^C as a common vertex. For each k, let ϕ_k^C denote the continuous, piecewise linear function supported on S_k^C such that $\phi_k^C(x_m^C) = \delta_{k,m}$. Thus, x_k^C is the center vertex of the hat-function ϕ_k^C. Let X_C be the finite element space spanned by the basis $\{\phi_k^C\}_{k=1}^{N_c}$. In the method we describe, we could also choose higher order elements for the X_C basis. For clarity, however, we assume the ϕ_k^C are piecewise linear. Later in Section 3.3.1 we discussion the issue of selecting the space X_C.

For the larger fine scale space X_R we will use a finite element space obtained by refining the coarse grid. We define the refined mesh $\mathcal{T}_R$ which is subordinate to the mesh $\mathcal{T}_C$. By this we mean that $\mathcal{T}_R$ is a refinement of $\mathcal{T}_C$ so that every triangle in $\mathcal{T}_C$ is a union of triangles in $\mathcal{T}_R$. We denote by X_R the corresponding finite element space, which contains X_C.

Let us describe some features of the fine scale space X_F, supposing that we choose $\mathcal{P}_C$ to be the H_0^1 projection. In this case, the condition $v \in X_F$ can be expressed by the constraints $(\nabla v, \nabla \phi_k^C) = 0$ for all ϕ_k^C in the X_C basis. If the coarse scale elements are piecewise linear, then for each coarse scale basis function ϕ_k^C the gradient $\nabla \phi_k^C$ is piecewise constant. Therefore, $v \in X_F$ if and only if for each $k = 1, \ldots N_c$

$$0 = (\nabla v, \nabla \phi_k^C) = \sum_{T_i \in S_k^C} \int_{T_i} \nabla v \cdot \nabla \phi_k^C = \sum_{T_i \in S_k^C} \int_{\partial T_i} v(p_i \cdot \nu_i)\, dx \tag{3.3}$$

where p_i is the value of $\nabla \phi_k^C$ in the triangle T_i, and ν_i is the outward unit normal to T_i. The constants p_i and the triangles T_i depends on k. This implies that X_F must contain all functions $v \in X_R$ that have zero mean across each edge in the coarse mesh, including all elements of X_R whose support is contained within a single coarse scale triangle. Nevertheless, there are elements of X_F which are supported on the entire domain D, rather than locally.

3.2. Spatial Localization and Oversampling. Here we further describe the localization strategy used to approximate $\tilde{\mathcal{M}}^o(\nabla \phi_k^C)$ in (3.2). The function μ_F may be approximated similarly. We define a family of fine-scale spaces $Y_k^F \subset X_F$, indexed by k which corresponds to the coarse scale elements $\{\phi_k^C\}_{k=1}^{N_c}$. For each k, we take Y_k^F to be a subset of functions in X_F that vanish sufficiently far from the support of ϕ_k^C.

For example, the simplest non-trivial approximation of $\tilde{\mathcal{M}}^o(\nabla \phi_k^C)$ is to define the space Y_k^F by $Y_k^F = X_F \cap H_0^1(S_k^C)$ which is the subspace of X_F of functions supported in S_k^C, the support of ϕ_k^C. Since we restrict the support in this way, any $v \in H_0^1(S_k^C)$ might be non-orthogonal to only a few elements in X_C, the immediate neighbor elements of ϕ_k^C. This completely localizes the fine scale problem for each ϕ_k^C. Since the support of $v \in Y_k^F$ coincides with the support of ϕ_k^C, we will refer to this first scenario as the case of no oversampling.

The next level of approximation involves enlarging the support of the functions in Y_k^F. We replace S_k^C by a larger set $\hat{S}_k^C$ which is the union of the supports of ϕ_k^C and one layer of its neighbors. This is in the spirit of "oversampling" proposed for the multiscale methods in [22, 23, 1]. So, for one layer of oversampling, we let the

sampling region be

$$\hat{S}_k^C = \bigcup_{m \in N^C(k)} S_k^C \qquad \text{and} \qquad Y_k^F = X_F \cap H_0^1(\hat{S}_k^C), \tag{3.4}$$

where $N^C(k)$ denotes the set of indices corresponding to coarse scale elements neighboring ϕ_k^C (elements with support that intersects S_k^C, including ϕ_k^C itself). Similarly, we can add multiple layers of oversampling. In the extreme case, $\hat{S}_k^C = D$ and $Y_k^F = X_F$.

Returning to equation (3.2), we may enforce the condition $v \in Y_k^F$ through the use of constraints and Lagrange multipliers. For example, if H_0^1 projection is used with no oversampling, the problem (3.2) defining $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ may be expressed as:

$$\text{minimize } \frac{1}{2}(a\nabla v, \nabla v) + (a\nabla\phi_k^C, \nabla v) \tag{3.5}$$

among all $v \in X_R \cap H_0^1(S_k)$ satisfying satisfying the constraints

$$(\nabla v, \nabla\phi_m^C) = 0, \ \ \forall\, m \in N^C(k). \tag{3.6}$$

So, if the node x_k^C has degree 6, then there will be 7 constraints. The unique minimizer of the constrained problem is taken to be $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$. Notice that because the support of $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ is restricted to S_k^C, the non-oversampling approximation does not alter the structure of the stiffness matrix

$$A_{m,k} = (a(\mathcal{I} + \nabla\tilde{\mathcal{M}}^o)\nabla\phi_k^C, (\mathcal{I} + \nabla\tilde{\mathcal{M}}^o)\nabla\phi_m^C). \tag{3.7}$$

If we use one layer of oversampling, the problem defining $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ may be expressed as:

$$\text{minimize } \frac{1}{2}(a\nabla v, \nabla v) + (a\nabla\phi_k^C, \nabla v) \tag{3.8}$$

among all $v \in X_R \cap H_0^1(\hat{S}_k)$ satisfying satisfying the constraints

$$(\nabla v, \nabla\phi_m^C) = 0, \ \ \forall m \in N^C(r), r \in N^C(k). \tag{3.9}$$

The expression $m \in N^C(r),\ r \in N^C(k)$, means that ϕ_r is a neighbor of ϕ_k^C and ϕ_m^C is a neighbor of ϕ_r^C. Thus, there are more constraints to be satisfied when we use oversampling. The fine scale problem for ϕ_k^C is still localized, but not as localized as in the case of no oversampling. Moreover, the structure of the matrix $A_{m,k}$ defined above will be altered, since the support of $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ extends outside the support of ϕ_k^C.

It is easy to see that this approximation strategy may be continued to define more refined approximations by taking $\hat{S}_k^C$ to include the supports of more distant neighbors of ϕ_k^C. The trade-off, of course, is a higher computational cost associated with a choice of larger sampling regions $\hat{S}_k^C$. More computation is needed to compute the reconstructions $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ and more computation is needed to solve the coarse scale system, since we add more off-diagonal terms to the stiffness matrix when there are multiple oversampling layers. Notice that the number of constraints to be satisfied and the number of off-diagonal terms added to the effective stiffness matrix is independent

of the level of refinement in the space X_R; instead, it depends only on the definition of the coarse subspace X_C and on the number of layers of oversampling used.

One may view the use of oversampling as a way to reduce the numerical effect of boundary layers in the definition of $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$. On the other hand, adding more layers of oversampling incorporates the long-range coupling between fine scales in the solution, which is intrinsic to the problem when there is no strict scale separation. In the future, it would be interesting to study how long range coupling is related to the creation of boundary layers in the definition of $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$.

3.3. Adaptive Oversampling and Refinement. It is not clear a priori how much "oversampling" is needed in the definition of the spaces Y_k^F in order to achieve a certain accuracy; this will depend on the structure of the coefficients. Here we propose an adaptive strategy for determining the optimal level of oversampling, which we intend to implement in future work. For the simulations shown in the following section, however, we use a fixed level of oversampling (either zero, one, or two layers). Nevertheless, the simulations demonstrate the need for an adaptive approach since the wrong choice of the oversampling domain will lead to either increased error in the solution or unnecessary computational cost. For example, we observe that when the coefficients are periodic (Section 4.1), one layer of oversampling is enough to achieve significant error reduction. When the coefficients are generated randomly (Section 4.2), at least two layers are needed. In other simulations, however, these requirements may vary.

Unless the coefficients are periodic and have the form $a(x/\epsilon)$, it also is not clear how refined the space X_R should be in order to fully resolve the fine scales within the region $\hat{S}_k^C$. Indeed there are two sources of error in the above approximation:

(i) **resolution error** due to defining the fine scale spaces Y_k^F too coarsely.

(ii) **localization error** due to the choice of $\hat{S}_k^C$ as a proper subset of D.

To reduce these errors, we propose two separate types of adaptivity which will be analyzed elsewhere. The goal of an adaptive strategy would be to reduce resolution error and localization error with minimal additional computational cost.

The first type of adaptivity aims at reducing the resolution error. For a given coarse scale element ϕ_k^C, the residual

$$R(v) = (a(\mathcal{I} + \nabla\tilde{\mathcal{M}}^o)\nabla\phi_k^C, \nabla v) \tag{3.10}$$

gives a measure of the error associated with approximating (3.1) by (3.2). If the space X_R is sufficiently refined, then $R(v)$ will be small for all $v \in H_0^1 \setminus Y_k^F$ that have support inside $\hat{S}_k^C$. Therefore, we may refine X_R (and thus X_F and Y_k^F) adaptively, until this residual is below some threshold parameter η, for all $v \in H_0^1 \setminus Y_k^F$ obtained by one or two levels of subdivision. To minimize additional computational cost the elements chosen for refinement should be those whose refinement produces the larges decrease in $R(v)$. This may be done using the techniques described in the work of Binev, Cohen, Dahmen, and DeVore [11, 7] and references therein. Other a-posteriori error indicators might be used, as well. Since neighboring sets $\hat{S}_k^C$ and $\hat{S}_j^C$ may overlap, the refinement should be applied at the X_R level; otherwise, it will be more difficult to evaluate the effective stiffness matrix if the meshes on neighboring local spaces do not match.

The second type of adaptivity involves selecting the amount of oversampling adaptively. This can be done once the level of refinement is already chosen. Here is the basic idea. Beginning with the first approximation scheme, $\hat{S}_k^C = S_k^C$, let ∂Y_k^R denote

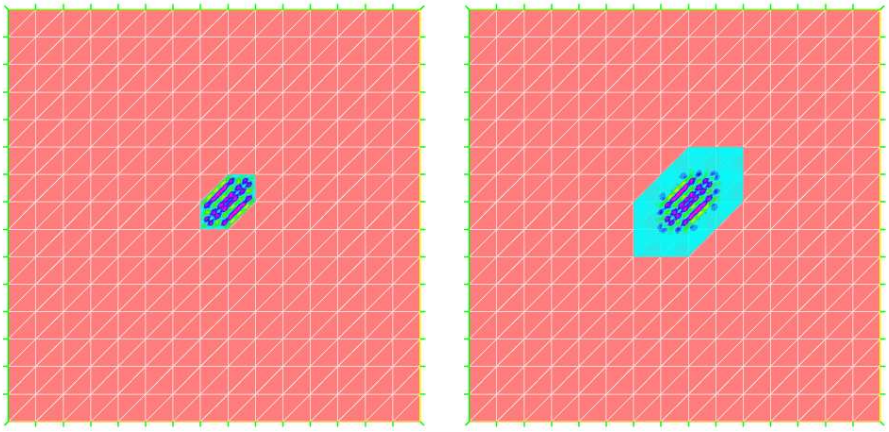

FIG. 3.1. *The function $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ computed with no oversampling (left) and one layer of oversampling (right). The projection is the H^1 projection. The relative H^1 error between these approximations and $\mathcal{M}^o(\nabla\phi_k^C)$ computed with complete oversampling (Figure 3.2b) is* 20% *and* 0.7%*, respectively.*

the set of basis elements in X_R whose central node is supported on the boundary of $\hat{S}_k^C$. These are the fine scale elements that straddle the boundary of $\hat{S}_k^C$. Note that $\partial Y_k^R \cap Y_k^F = \emptyset$, since the elements of Y_k^F are supported within S_k^C. For each of these straddling nodes $v \in \partial Y_k^R$, compute the residual $R(v)$. If $R(v) > \eta$ for some v in this set of elements straddling the boundary of $\hat{S}_k^C$, then we enlarge S_k^C to include the support of v. Also, we enlarge the space Y_k^F accordingly. Then solve for $\mathcal{M}^o(\nabla\phi_k)$ using this new set Y_k^F. Continue the expansion procedure until $R(v) < \eta$ for all of the straddling nodes v or until $\hat{S}_k^C = D$, in which case we are computing $\mathcal{M}^o(\nabla\phi_k^C)$ exactly. There are many variations of this idea. For example, one could test $R(v)$ for just a few functions in ∂Y_k^R and choose to expand Y_k^R in a non-uniform manner. Thus, there might be more oversampling in one direction than another; the values of $R(v)$ suggest which directions in which to expand. To minimize additional computational cost, one should expand in the direction which leads to greatest reduction of $R(v)$ (or another error indicator).

For the numerical simulations described in the next section, we show in Figures 3.1 and 3.2 the functions $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ computed with various levels of oversampling using the H_0^1 projection. The coefficients $a(x)$ are periodic. We observe that the function $\mathcal{M}^o(\nabla\phi_k^C)$ (Figure 3.2b) is localized near the support of ϕ_k^C. The error in the approximation decreases exponentially as a function of the number of layers used in oversampling. With no layers of oversampling (Figure 3.1a), the relative H^1 error between $\mathcal{M}^o(\nabla\phi_k^C)$ and the approximation $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ is 20%. With just one layer of oversampling, this relative error is reduced to 0.7%. With two layers and four layers of oversampling, the error is reduced further to 0.3% and 0.07%, respectively.

3.3.1. Determination of X_C. For some applications, the fine scales of the solution may be of no interest once the effective coarse scale equation (2.11) has been computed. Nevertheless, it is not clear a priori what level of refinement to choose for the space X_C in order that the function $u_C \in X_C$ be close to the true solution u. In other words, it is not clear in advance how to choose the X_C so that $\mathcal{Q}_C u = u - u_C$ be small. This is not an issue of resolving the fine scales; it is an issue of what is

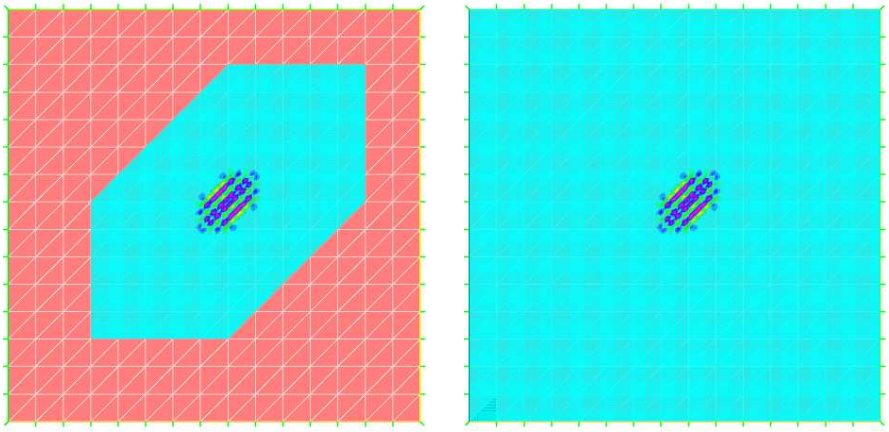

FIG. 3.2. *The function $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ computed with four layers of oversampling (left) and complete oversampling (right). The projection is the H^1 projection. The relative H^1 error between the function shown in the left plot and $\mathcal{M}^o(\nabla\phi_k^C)$ computed with complete oversampling is* 0.07%.

considered coarse and what is considered fine in the decomposition $X_R = X_C \oplus X_F$. For example, the simulations of Section 4.3 show that the reconstructed component μ_F may form a significant portion of the entire solution u. When there is scale separation and the oscillations in the coefficients have an homogenization effect, the space X_C should be resolved enough to give a good approximation of the solution to the effective equation, which may have non-oscillatory coefficients. We discuss this point further in Section 5.1.

The choice of X_C can be made adaptively in the projection framework, as follows. The approximate reconstruction operator $\tilde{\mathcal{M}}^o(\nabla\phi_k)$ gives us a measurement of the error in the coarse scale solution. So, the coarse grid may be refined based on analysis of the functions $\tilde{\mathcal{M}}^o(\nabla\phi_k)$, which are local. We can refine X_C in space (h refinement) or in polynomial order (p refinement), and the refinement choice may be based locally on the interpolation of the function $w_k = \phi_k^C + \tilde{\mathcal{M}}^o(\nabla\phi_k^C)$. Without refinement of X_C, our best approximation of w_k with respect to the projection $\mathcal{P}_C$ is simply $w_k^{old} = \phi_k^C$. Following the idea of automatic hp-adaptivity in the work of Demkowicz [12], we may choose the combination of h and p refinement that significantly reduces the interpolation error: $\|w_k^{new} - w_k\| \leq \lambda\|w_k^{old} - w_k\| = \lambda\|\tilde{\mathcal{M}}^o(\nabla\phi_k^C)\|$, where w_k^{new} denotes the new interpolant of w_k in the (locally) refined coarse basis and $\lambda \in (0,1)$ is a tunable parameter.

4. Numerical Simulations. To test the approximation scheme, we consider the problem (1.1) in the unit square $D = [0,1]^2$ in two dimensions. For the coefficients $a(x)$, we use either a periodic function or a randomly generated function. We implemented the multiscale method using the FreeFem++ software which may be downloaded from [18]. All simulations were performed on a single processor. More implementation details are given in the appendix.

We compare the results with both a highly resolved Galerkin solution (u_{RG}) and a coarse Galerkin solution (u_{CG}). Moreover, we distinguish between u_C, which is the coarse scale component produced by the multiscale algorithm, and $u_{C+} = u_C + \tilde{\mathcal{M}}(\nabla u_C)$ which is the coarse component plus the reconstructed fine scales. The highly resolved Galerkin solution is computed on a mesh that is equivalent to the

resolution given by the multiscale method with reconstruction. Thus, if the operator $\mathcal{M}$ is computed without localization, the multiscale solution and the high resolution Galerkin solution agree. So, if the $\tilde{\mathcal{M}}$ is a good approximation of $\mathcal{M}$, we expect $u_{C+} \approx u_{RG}$, so that $\|u_{RG} - u_{C+}\|_{L^2}$ and $\|u_{RG} - u_{C+}\|_{H^1}$ should be much smaller than $\|u_{RG} - u_{CG}\|_{L^2}$ and $\|u_{RG} - u_{CG}\|_{H^1}$, respectively. Also, we expect that $\|u_{RG} - u_C\|_{L^2}$ will be smaller than $\|u_{RG} - u_{CG}\|_{L^2}$, since u_C is like coarse component of the homogenized solution in homogenization theory. However, we do not expect $\|u_{RG} - u_C\|_{H^1}$ to differ significantly from $\|u_{RG} - u_{CG}\|_{H^1}$, since neither u_C nor u_{CG} contain fine scale structure present in the highly resolved solution u_{RG}.

In the data tables we display the relative change in error. In each table and for each norm (H^1 or L^2), we use $\|u_{CG} - u_{RG}\|$ as the benchmark, since we are interested in whether the multiscale method gives any improvement over the Galerkin method. The various columns show how the error changes with respect to the number of layers of oversampling. However, the reference $\|u_{CG} - u_{RG}\|$ does not depend on the the number of layers used. For example, if we measure the relative change in H^1 error,

$$\text{Relative change} = \frac{\|u_{C+} - u_{RG}\|_{H_0^1} - \|u_{CG} - u_{RG}\|_{H_0^1}}{\|u_{CG} - u_{RG}\|_{H_0^1}} \tag{4.1}$$

Therefore, the entries in the columns for u_{CG} are normalized to 0%, and a value of -50% in the table indicates a 50% reduction in the error relative to the error in the coarse Galerkin solution.

4.1. Simulation 1: Periodic Coefficients. First, we test the method with zero Dirichlet boundary condition and periodic coefficients of the form

$$a(x) = 1.1 + \sin(n2\pi x)\sin(n2\pi y). \tag{4.2}$$

using the H_0^1 projection. The forcing function $f(x)$ is a function in X_C with compact support near the middle of the domain. The highly resolved Galerkin solution was computed on a grid of size 225×225, while the low resolution Galerkin solution was computed on a grid of size 15×15. For the multiscale scheme, the coarse scale components are defined on a grid of size 15×15 (the same grid used for the coarse Galerkin solution). The reconstruction is computed at a scale equivalent to a 225×225 global discretization, so the reconstructed solution has an effective resolution equal to that of the highly resolved Galerkin solution.

Table 4.1 shows the relative change in error when using the H^1 projection and various amounts of oversampling. For this simulation, we used $n = 15$, which corresponds to $\epsilon \sim 1/15$ if we write the coefficients in the form $a(x/\epsilon)$. The relative error in these computations is in very good agreement with the comments we made above (4.1) regarding its behavior in the L^2 and in the H^1 norms. It is also clear that in the case of periodic coefficients one layer of oversampling is sufficient to reduce the error. In this case, taking more layers does not further reduce the error significantly.

4.2. Simulation 2: Random coefficients. Next, we test the method using coefficients constructed randomly. We divide the grid into a 70×70 mesh and set

$$a(x) = \sum_n a_n(\omega)\phi_n(x) \tag{4.3}$$

where $\{\phi_n(x)\}$ is the nodal basis corresponding to the grid, and $\{a_n\}$ are chosen randomly and independently according to the rule $a_n = 0.01$ with probability 0.3 and

TABLE 4.1

Relative change in error (4.1) for Simulation 1 with periodic coefficients (4.2). u_{RG} denotes high-resolution Galerkin solution, u_{CG} denotes the low-resolution Galerkin solution. As the amount of oversampling increases from 0 to 2 layers, the error decreases significantly. When no oversampling is used, the multiscale method produces poor results. See (4.1) and surrounding discussion for a description of the table.

H^1 **projection**	0 Layers	1 Layer	2 Layers	u_{CG}
$\|u_{RG} - u_C\|_{L^2}$	+140%	-69.2%	-69.4%	0%
$\|u_{RG} - u_{C+}\|_{L^2}$	+139%	-86.1%	-89.4 %	-
$\|u_{RG} - u_C\|_{H^1}$	+8.0%	-2.5%	-2.4%	0
$\|u_{RG} - u_{C+}\|_{H^1}$	-9.1%	-77.7%	-84.2%	-

$a_n = 1.0$ with probability 0.7. The function $a(x)$ is shown in Figure 4.1(b). We chose this function as a simple way to model percolating channels through an otherwise dense medium. The forcing function in this simulation is more concentrated than in the periodic case, but has the same structure. We enforce $u = 0$ on the boundary of the domain. For this problem there is no clear scale separation, no "ϵ" as in the periodic case

The relative errors are shown in Table 4.2. As in the periodic case, the relative error in these computations is in very good agreement with the comments we made above (4.1) regarding its behavior in the L^2 and in the H^1 norms. In the case of random coefficients, we note that one layer of oversampling is not sufficient for reducing the error, as was the case with periodic coefficients. In that case, the irregular oscillation of the coefficients requires more oversampling. The amount of oversampling that is needed is not known in advance but must be determined adaptively.

TABLE 4.2

Relative change in error (4.1) for Simulation 2 with random coefficients (4.3). u_{RG} denotes high-resolution Galerkin solution, u_{CG} denotes the low-resolution Galerkin solution. As the amount of oversampling increases from 0 to 2 layers, the error decreases. When no oversampling is used, the multiscale method produces poor results. See (4.1) and surrounding discussion for a description of the table.

H^1 **projection**	0 Layers	1 Layer	2 Layers	u_{CG}
$\|u_{RG} - u_C\|_{L^2}$	+813%	+3.9%	-59.0 %	0%
$\|u_{RG} - u_{C+}\|_{L^2}$	+803%	3.1%	-63.5%	-
$\|u_{RG} - u_C\|_{H^1}$	+98%	-1.1%	-2.9%	0%
$\|u_{RG} - u_{C+}\|_{H^1}$	+435%	-17.6%	-41.7%	-

4.3. Simulation 3: Random coefficients with imposed pressure gradient. Now we test the method using random coefficients with a pressure gradient imposed at the boundary. That is, we impose $u(x_1, x_2) = bx_1, \quad x \in \partial D$ where $b > 0$ is a constant. We impose an external forcing similar to the case above. As before, the function $f \in X_C$ is supported near the middle of the domain. To handle the boundary condition, we compute the function $w = u - bx_1 \in H_0^1$ which satisfies

$$(a(x)\nabla w, \nabla v) = (f, v) - (ba(x), \partial_{x_1} v). \tag{4.4}$$

for all $v \in H_0^1$. The coefficients are defined as before, on a grid of size 70×70. The highly resolved Galerkin solution was computed on a grid of size 225×225, while the low resolution Galerkin solution was computed on a grid of size 15×15. The results are shown in Table 4.3. We see that oscillatory part of the inhomogeneous term in (4.4) makes a significant difference in the behavior of the error in the L^2 norm (lines

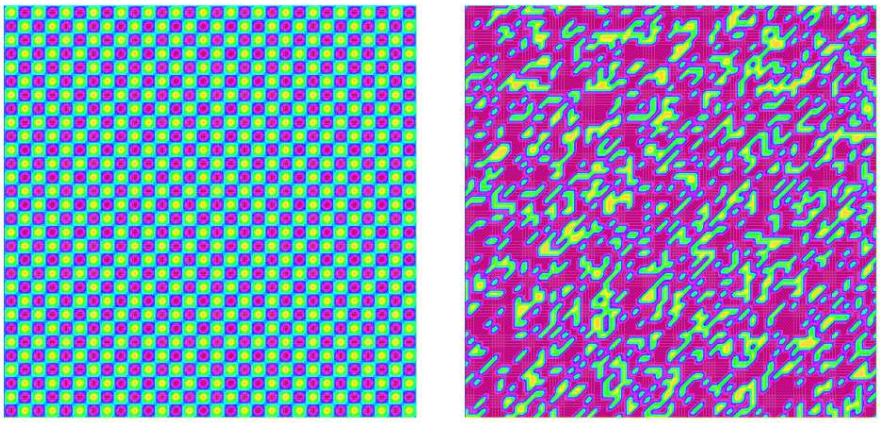

FIG. 4.1. *Left: The periodic function $a(x)$. Here the ratio of domain width to period size is $n = 15$. Right: The randomly generated function $a(x) = 1.0$ (purple) or $a(x) = 0.01$ (yellow).*

one and two in Table 4.3). This is because the μ_F term from (2.14) is not negligible now.

TABLE 4.3

Relative change in error (4.1) for Simulation 3 with random coefficients (4.3). u_{RG} denotes high-resolution Galerkin solution, u_{CG} denotes the low-resolution Galerkin solution. See (4.1) and surrounding discussion for a description of the table.

H^1 **projection**	0 Layers	1 Layer	2 Layers	u_{CG}
$\|u_{RG} - u_C\|_{L^2}$	+72%	-3.1 %	-1.5%	0%
$\|u_{RG} - u_{C+}\|_{L^2}$	+56%	-59.7%	-54.6%	-
$\|u_{RG} - u_C\|_{H^1}$	+0.1%	-0.1%	-0.1%	0%
$\|u_{RG} - u_{C+}\|_{H^1}$	-52.8%	-76.4%	-78.7%	-

4.4. Summary of Numerical Results. In each simulation, we observe that the multiscale method with oversampling significantly reduces the L^2 and H^1 error, when compared with the coarse Galerkin solution. If no oversampling is used, the method produces poor results when the scale of oscillations in $a(x)$ is comparable to the size of the coarse scale elements. This is similar to what the authors of [23, 15] have called "resonance error". We observe that the technique of oversampling reduces this error.

As shown in Figure 4.4, the method reproduces the fine scale features in the solution quite well. The error decreases with more layers of oversampling. We also observe that the computation of the inhomogeneous term μ_F is significant in reproducing the fine scales of the solution. As can be seen in Figure 4.5, the multiscale solution does not resemble the highly resolved solution when this term is ignored.

4.5. Computational Cost. As with other multiscale methods we have mentioned (e.g. [1, 22, 3, 31]), the functions $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ and μ_k^j may be computed independently; so, these computations may be parallelized easily. If the problem must be solved for multiple forcing functions $f(x)$, the functions $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ and the integrals $(a(\mathcal{I} + \nabla\tilde{\mathcal{M}}^o)\nabla\phi_k^C), (\mathcal{I} + \nabla\tilde{\mathcal{M}}^o)\nabla\phi_m^C)$ may be computed once for all instances of f. The function μ_F, however, must be recomputed. Nevertheless, μ_F does not appear

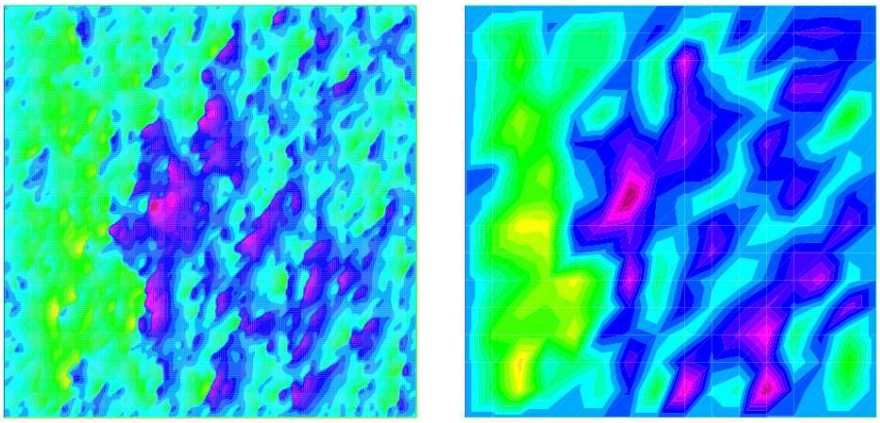

FIG. 4.2. *(Random coefficients with pressure gradient) Left: High resolution Galerkin solution. Right: Low resolution Galerkin solution. The color yellow corresponds to the lowest value; purple corresponds to the highest value.*

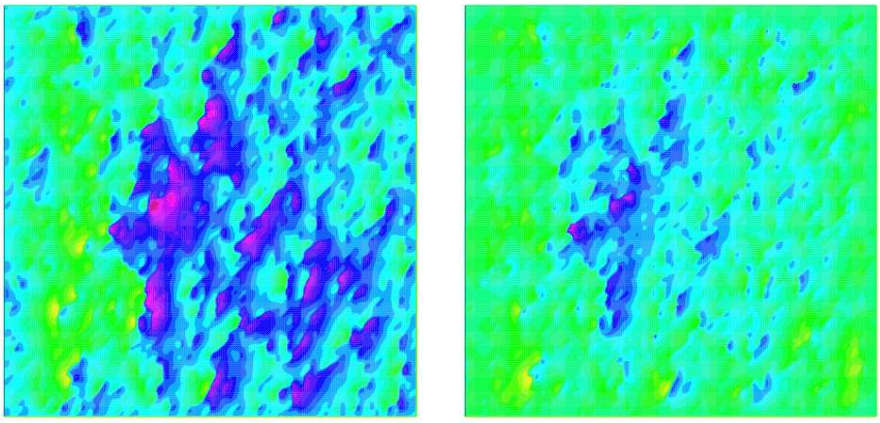

FIG. 4.3. *(Random coefficients with pressure gradient) Left: High resolution Galerkin solution. Right: The multiscale solution* u_C **with reconstruction** *of fine scales using* **no oversampling**. *The fine scale features in the solution are reproduced somewhat, but not as well as in the case when oversampling is used (next figure). Note: we have subtracted off the linear part of the solution.*

in the effective coarse scale equation.

Suppose that there are $O(N^d)$ coarse scale basis functions. Let $L << N$ be the number of layers used in oversampling, $L = 0$ corresponding to no oversampling. To define the fine mesh, we divide each coarse element into $O(M^d)$ fine elements. Thus in the fine mesh, there are $O(N^d M^d)$ basis functions. If we computed the Galerkin solution using the fine scale mesh, the dimension of the stiffness matrix is $O(N^d M^d)$.

With the multiscale method, the local problems are solved on a domain made up of $O((1 + L)^d)$ coarse scale elements, so the basis for each local problem has $O((1+L)^d M^d)$ fine scale elements. For each fine scale problem there are $O((1+L)^d)$ constraints, so that the linear system is of size $O((1+L)^d(M^d+1))$. There are $O(N^d)$ of these problems to solve.

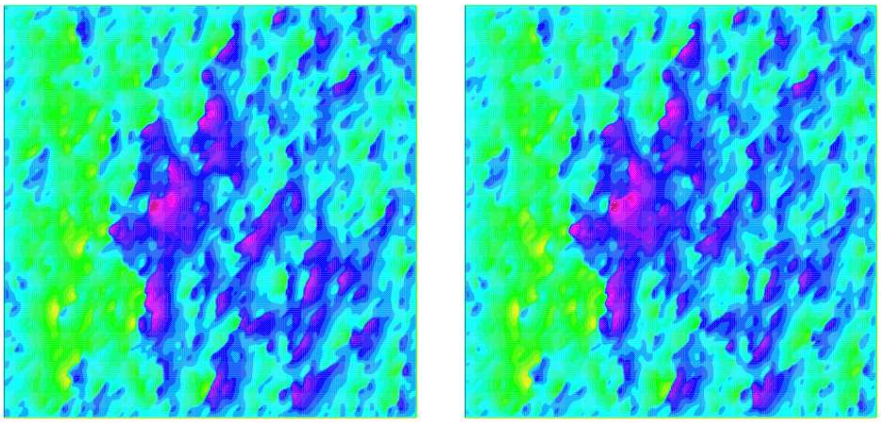

FIG. 4.4. *(Random coefficients with pressure gradient) Left: High resolution Galerkin solution. Right: The multiscale solution* u_C **with reconstruction** *of fine scales using* **one layer of oversampling**. *The fine scale features in the solution are reproduced well. Note: we have subtracted off the linear part of the solution.*

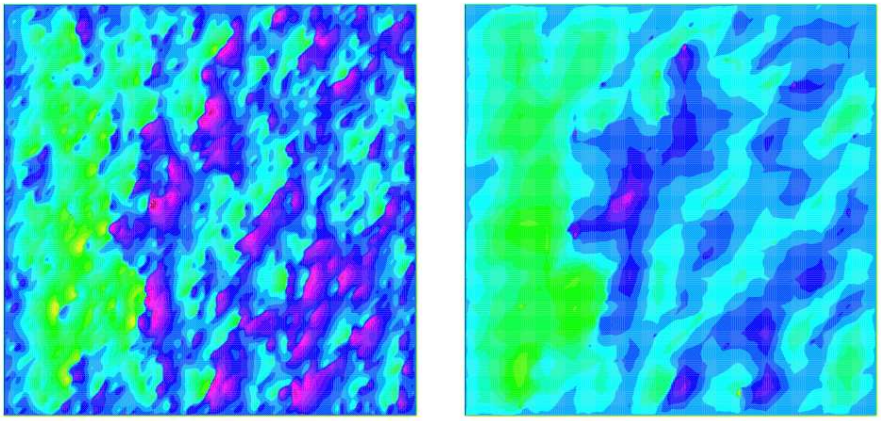

FIG. 4.5. *(Random coefficients with pressure gradient) Left: High resolution Galerkin solution. Right: The multiscale solution reconstructed solution* **without the** μ_F **correction**: $u_C + \tilde{\mathcal{M}}^o(\nabla u_C)$. *Two layers of oversampling are used. This example shows the importance of computing the term* μ_F *in reconstructing the fine scales of the solution.*

Let $S_g(D)$ denote the cost of solving the Galerkin problem with D degrees of freedom, and let $S_{fs}(D, K)$ denote the cost of solving one of the fine scale systems with D degrees of freedom plus K constraints. Then the computational cost of the solving the high resolution Galerkin problem is

$$C_{hres} = S_g(N^d M^d) \tag{4.5}$$

and the cost of computing the multiscale solution is

$$C_{ms} = S_g(N^d) + N^d S_{fs}(O((1+L)^d M^d), O((1+L)^d)), \tag{4.6}$$

which is the cost of the effective coarse scale problem plus the N^d local problems.

If a conjugate gradient algorithm is used to solve the coarse and fine scale systems iteratively, then $S_g(D) = O(D^{3/2})$ if $O(D^{1/2})$ iterations are used, and $S_{fs}(D,K) = O(D^{3/2}K^{3/2})$. In this case,

$$\frac{C_{ms}}{C_{hres}} \leq \frac{O(N^{3d/2} + N^d(1+L)^{9d/4}M^{3d/2})}{O(N^{3d/2}M^{3d/2})} = O(M^{-3d/2}) + O(N^{-d/2}(1+L)^{9d/4}) \tag{4.7}$$

So, as long as $L = o(N^{2/9})$, the cost of computing the multiscale method may be dramatically smaller than the cost of computing the high resolution solution, even when the local problems are not computed in parallel. The width of the oversampling region in physical space is of the size $O(L\Delta x) = O(L/N)$. If L is chosen to satisfy $L = o(N^{2/9})$, then the width of the layer (in spatial units) must vanish as $N \to \infty$.

All of the numerical simulations that we have shown were performed on a single processor, and we observed that the multiscale method was much more computationally expensive than the high-resolution Galerkin method. Nevertheless, the mesh sizes used in the simulations were relatively coarse compared to what might be necessary for a practical problem. As N and M increase, the above bounds show that the multiscale method offers significant savings over the traditional Galerkin approach.

5. Relation to Existing Theory and Methods. In this section, we compare the projection framework presented in Section 2 to homogenization theory when the coefficients have a strong scale separation (for example, see [6], [30]). We also, compare the framework to existing multiscale methods based on wavelets, multiscale finite element bases, and harmonic coordinates.

5.1. Homogenization Theory. Suppose that $a^\epsilon(x)$ has the form $a^\epsilon(x) = a(x/\epsilon)$ for some symmetric matrix $a(y)$ that is periodic in y, and that $f^\epsilon \to f$ strongly in $H^{-1}(D)$. Let $u^\epsilon(x)$ solve

$$-\nabla \cdot (a^\epsilon \nabla u^\epsilon) = f^\epsilon \quad x \in D \tag{5.1}$$

with $u = 0$ on ∂D. It is known that as $\epsilon \to 0$, the solution u^ϵ converges weakly in H^1 to a function $\bar{u} \in H^1(D)$ that solves an effective elliptic equation, an "upscaled" equation. Thus, the difference $(u^\epsilon - \bar{u})$ converges to zero only weakly in $H^1(D)$, and it is this difference that is captured by the fine scale reconstruction $\mathcal{M}(\nabla u_C^\epsilon)$ in the multiscale framework.

The effective equation for $\bar{u} \in H_0^1$ is

$$(\bar{a}\nabla\bar{u}, \nabla v) = (f, v), \quad \forall v \in H_0^1, \tag{5.2}$$

where $\bar{a}$ is the homogenized matrix given by

$$\bar{a}_{ij} = \int_{\mathbb{T}^d} \left(a_{ij}(y) + \sum_{n=1}^{d} a_{in}(y) \frac{\partial \chi^j}{\partial y_n} \right) dy. \tag{5.3}$$

The functions $\chi^j(y)$ are periodic in y and solve $-\nabla_y \cdot a(y)\nabla_y \chi^j = \nabla_y \cdot a(y) e_j$, for $y \in \mathbb{T}^d$. From (5.3) we see that a simple average of $a(x)$, denoted by $\langle a \rangle$, is the wrong homogenized coefficient; the necessary correction will come from the linear part of reconstruction operator.

Suppose that the space X_C is fixed with respect to ϵ. We can decompose the solution u^ϵ as

$$u^\epsilon = u_C^\epsilon + \mathcal{M}^\epsilon(\nabla u_C^\epsilon) = u_C^\epsilon + \mathcal{M}^{o,\epsilon}(\nabla u_C^\epsilon) + \mu_F^\epsilon \tag{5.4}$$

where the operators $\mathcal{M}^\epsilon$ and $\mathcal{M}^{o,\epsilon}$, the coarse component u_C^ϵ, and the function μ_F^ϵ now depend on ϵ. This representation is exact. If we approximate u^ϵ by a function $u^{\epsilon,L}$ using the localization and oversampling strategy described in section 3.2, we decompose $u^{\epsilon,L}$ as

$$u^{\epsilon,L} = u_C^{\epsilon,L} + \tilde{\mathcal{M}}_L^\epsilon(\nabla u_C^{\epsilon,L}) = u_C^{\epsilon,L} + \tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla u_C^{\epsilon,L}) + \mu_F^{\epsilon,L}. \tag{5.5}$$

Here we use $\tilde{\mathcal{M}}_L^{o,\epsilon}$ and $\mu_F^{\epsilon,L}$ to denote approximations to $\mathcal{M}^{o,\epsilon}$ and μ_F^ϵ, respectively, where $L \geq 0$ denotes the number of layers used in oversampling. So, for each coarse basis function $\phi_k \in X_C$, the problem defining $\tilde{\mathcal{M}}_L^{o,\epsilon}$ is:

$$(a^\epsilon \nabla \tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k), \nabla v) = -(a^\epsilon \nabla\phi_k, \nabla v), \quad \forall v \in Y_k^F, \tag{5.6}$$

where Y_k^F denotes the approximation of X_F defined by L layers of oversampling around the support of ϕ_k. The function $\mu_F^{\epsilon,L}$ is defined by

$$\mu_F^{\epsilon,L} = \sum_k \phi_k^C \mu_{F,k}^{\epsilon,L} \tag{5.7}$$

where, for each $k = 1, \ldots, N_c$, $\mu_{F,k}^{\epsilon,L} \in Y_k^F \subset X_F$ satisfies $(a^\epsilon \mu_{F,k}^{\epsilon,L}, \nabla v) = (f, v)$ for all $v \in Y_k^F$. When L is large enough so that the oversampling regions correspond with the entire domain D, then (5.4) and (5.5) are equivalent.

In a similar way, we may decompose the solution to the homogenized problem as

$$\bar{u} = \bar{u}_C + \overline{\mathcal{M}}(\nabla \bar{u}_C) = \bar{u}_C + \overline{\mathcal{M}}^o(\nabla \bar{u}_C) + \bar{\mu}_F \tag{5.8}$$

where the operators $\overline{\mathcal{M}}$ and $\overline{\mathcal{M}}^o$ and the function $\bar{\mu}_F$ are defined as in (2.11 - 2.14) with $a(x)$ replaced by the homogenized coefficients $\bar{a}$. We use $\bar{u}_C$ to denote $\mathcal{P}_C \bar{u}$. The equation for $\bar{u}_C$ is

$$(\bar{a}(\nabla \bar{u}_C + \nabla \overline{\mathcal{M}}^o(\nabla \bar{u}_C)), (\nabla v + \nabla \overline{\mathcal{M}}^o(\nabla v))) = \langle f, v + \overline{\mathcal{M}}^o(\nabla v)\rangle, \tag{5.9}$$

for all $v \in X_C$. This equation is exact for $\bar{u}_C$. However, as with u^ϵ, we could approximate $\bar{u}$ using the localization and oversampling scheme, so that the approximation $\bar{u}^L$ has the decomposition

$$\bar{u}^L = \bar{u}_C^L + \overline{\mathcal{M}}_L(\nabla \bar{u}_C^L) = \bar{u}_C^L + \overline{\mathcal{M}}_L^o(\nabla \bar{u}_C^L) + \bar{\mu}_F^L. \tag{5.10}$$

Here, $\overline{\mathcal{M}}_L$ and $\bar{\mu}_F^L$ are defined as in (5.6) and (5.7) with a^ϵ replaced by $\bar{a}$. The function $\bar{u}_C^L \in X_C$ solves the symmetrized equation (2.11) with $\overline{\mathcal{M}}^o$ and $\bar{\mu}_F$ replaced by the approximations $\overline{\mathcal{M}}_L^o$ and $\bar{\mu}_F^L$, respectively.

The following proposition summarizes how each term in the decomposition of $u^{\epsilon,L}$ behaves in the limit $\epsilon \to 0$. The proof of the proposition is in the appendix.

PROPOSITION 5.1. *Suppose that X_C is defined by piecewise linear elements and that $\mathcal{P}_C$ is the H_0^1 projection. Let $\bar{u}^{\epsilon,L}$ solve the symmetrized equation (2.11) with $\mathcal{M}^{o,\epsilon}$ and μ_F^ϵ replaced by the approximations $\tilde{\mathcal{M}}_L^{o,\epsilon}$ and $\mu_F^{\epsilon,L}$ given by (5.6) and (5.7) for fixed L. As $\epsilon \to 0$,*

(i) $u_C^{\epsilon,L} \to \bar{u}_C^L$ *strongly in* $H_0^1(D)$,

(ii) $\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla u_C^{\epsilon,L}) \to \overline{\mathcal{M}}_L^o(\nabla \bar{u}_C^L)$ *weakly in* $H_0^1(D)$,

(iii) $\mu_F^{\epsilon,L} \to \bar{\mu}_F^L$ *weakly in* $H_0^1(D)$,

(iv) $a^\epsilon \nabla \tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi) \to \bar{a}\nabla \overline{\mathcal{M}}_L^o(\nabla\phi) + (\bar{a} - \langle a\rangle)\nabla\phi$ *weakly in* $(L^2(D))^d$,

(v) $a^\epsilon \nabla \mu_F^{\epsilon,L} \to \bar{a}\nabla\bar{\mu}_F^L$ *weakly in* $(L^2(D))^d$.

Thus, as $\epsilon \to 0$, $u_C^{\epsilon,L}$ *converges to the function* $\bar{u}_C^L$ *which solves*

$$\left(\bar{a}(\mathcal{I} + \nabla\overline{\mathcal{M}}_L^o)\nabla\bar{u}_C^L, (\mathcal{I} + \nabla\overline{\mathcal{M}}_L^o)\nabla v)\right) = (f, v + \overline{\mathcal{M}}_L^o\nabla v), \quad \forall v \in X_C, \tag{5.11}$$

where $\bar{a}$ *is the homogenized matrix given by (5.3).*

The main point of the proposition is that the correct homogenized coefficients ($\bar{a}$) are recovered, regardless of the number of layers used in oversampling. When L is large enough so that the oversampling region is actually the entire domain D, then $\bar{u}_L = \bar{u}$, and (5.11) is the exact equation for $\bar{u}_C = \mathcal{P}_C\bar{u}$. For any $L \geq 0$, however, (5.11) is just the homogenized equation projected onto the N_c-dimensional space spanned by basis functions $\psi_k = \phi_k^C + \overline{\mathcal{M}}_L^o(\nabla\phi_k^C)$.

The operator $\overline{\mathcal{M}}$ and its approximation $\overline{\mathcal{M}}_L^o$ are nonlocal, so the nonlocal nature of the reconstruction operator does not disappear entirely in the limit $\epsilon \to 0$, since $\overline{\mathcal{M}}(\nabla\bar{u}_C) = \mathcal{Q}_C\bar{u} \neq 0$, in general. However, as $\epsilon \to 0$, the effective coefficients are determined locally. So, when we apply the localization and oversampling strategy described in Section 3.2, the method still correctly captures the homogenized coefficients. The nonlocal term that is left in the equation after $\epsilon \to 0$ may be made negligible (in H^1), since $\|\mathcal{Q}_C\bar{u}\|_{H^1}$ may be small, depending on the regularity of $\bar{u}$ and the choice of X_C. Therefore, in the homogenization regime, the space X_C should be chosen so that the solution to the effective equation is sufficiently resolved. This can be achieved adaptively, as described in Section 3.3.1.

The convergence in (ii) and (iii) shows that the distinction between $\tilde{\mathcal{M}}_L^{o,\epsilon}$ and $\mu_F^{\epsilon,L}$ is preserved in the limit. The function $\bar{\mu}_F^L$ does not appear in the coarse scale equation (5.11) for $\bar{u}_C^L$, and (5.11) can be obtained without computing the functions μ_F^ϵ. This suggests that $\mu_F^{\epsilon,L}$ plays no role in determining the homogenized coefficients for the $\bar{u}_C^L$. Indeed, from (iv) we see that the correction comes from the linear part of the reconstruction operator. Nevertheless, the functions $\mu_F^{\epsilon,L}$ will be insignificant in the limit $\epsilon \to 0$ only if the homogenized solution is relatively smooth and f has no fine scale oscillations relative to the space X_C. In that case, $\|\bar{\mu}_F^L\|_{H^1}$ may be negligible, since $\|\bar{\mu}_F^L\|_{H^1} \leq \|\mathcal{Q}_C\bar{u}^L\|_{H^1}$, and the latter may be small, depending on the regularity of $\bar{u}$ and the choice of X_C.

5.2. Wavelet-based methods. The multiscale framework of Section 2 was also used by Engquist and Runborg [16, 17] and Chertock and Levy [10] to derive a compressed form of the differential operator $\mathcal{L} = -\nabla\cdot a\nabla$ acting on a wavelet basis $\{\phi_k\}$. In that context, the authors define the effective variational problem in terms of the Schur complement of a matrix representation of the discretized operator. The spaces X_C and X_F are finite dimensional spaces corresponding to the dyadic level 2^{-j_1} and 2^{-j_2}, respectively, $j_1 < j_2$. If $\mathcal{L}_{k,l} = (a\nabla\phi_l, \nabla\phi_k)$ is the stiffness matrix, then we can decompose $\mathcal{L}_{k,l}$ into four components, according to its operation on the spaces X_C and X_F which represent course-scale elements and fine scale elements, respectively:

$$\mathcal{L}u = \begin{pmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{C} & \mathcal{D} \end{pmatrix} \begin{pmatrix} \mathcal{Q}_C u \\ \mathcal{P}_C u \end{pmatrix} \triangleq \begin{pmatrix} \mathcal{Q}\mathcal{L}\mathcal{Q} & \mathcal{Q}\mathcal{L}\mathcal{P} \\ \mathcal{P}\mathcal{L}\mathcal{Q} & \mathcal{P}\mathcal{L}\mathcal{P} \end{pmatrix} \begin{pmatrix} \mathcal{Q}_C u \\ \mathcal{P}_C u \end{pmatrix} = \begin{pmatrix} \mathcal{Q}_C f \\ \mathcal{P}_C f \end{pmatrix}$$

The coarse scale component $u_C = \mathcal{P}_C u$ satisfies the equation

$$\bar{\mathcal{L}}u_C = (\mathcal{D} - \mathcal{C}\mathcal{A}^{-1}\mathcal{B})u_C = \mathcal{P}f - \mathcal{C}\mathcal{A}^{-1}\mathcal{Q}f. \tag{5.12}$$

The matrix $\bar{\mathcal{L}}$ is known as the Schur complement. If we could compute $\mathcal{A}^{-1}$ and solve (5.12) exactly, then we could obtain the full solution u by back substitution since

$\mathcal{Q}_C u = \mathcal{A}^{-1}\mathcal{B}u_C + \mathcal{A}^{-1}\mathcal{Q}_C f$. This is another way of expressing the relationship (2.4). The equivalence between the two frameworks is seen in the relations

$$\begin{aligned}\mathcal{M}(\nabla v) &= \mathcal{A}^{-1}\mathcal{B}v + \mathcal{A}^{-1}\mathcal{Q}_C f,\\ \mathcal{M}^o(\nabla v) &= \mathcal{A}^{-1}\mathcal{B}v\\ \mu_F &= \mathcal{A}^{-1}\mathcal{Q}_C f\end{aligned}$$

for any $v \in X_C$.

5.3. Finite Element Methods. The multiscale projection framework is also closely related to other finite element methods designed to capture the homogenization effect when there is strong scale separation. The approach of [22, 23, 15, 24, 1, 31] is to transform the typical coarse-scale basis elements into multiscale elements that contain some small-scale information. That is, given a high-dimensional finite element space X_R, we want to find a lower dimensional space X'_R that approximates the solution well, yet the standard basis for X_C will give a poor approximation when there are small scales in the coefficients. So, we define a new basis by mapping $\phi_k^C \mapsto \psi_k \in X_R$:

$$\psi_k = \phi_k^C + \tilde{\mathcal{M}}^o \phi_k^C. \tag{5.13}$$

The dimension of this new basis is the same as the original coarse basis.

This is the perspective of Hou, et.al. [22, 23, 15, 24]. In that approach, the operator $\tilde{\mathcal{M}}^o$ is linear, but may not map into H_0^1. This is because the function $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ is constructed over each triangular component separately and then pieced together. The multiscale basis functions solve $\nabla\cdot(a(x)\nabla\psi_k) = 0$ within the interior of each triangular component of S_k^C (assuming full resolution), yet the functions ψ_k are not continuous across the boundaries of the triangles if oversampling is used. With any level of oversampling used to define ψ_k in this way, the resulting effective coarse-scale equation for u will still be inexact.

The present approach is similar to this approach but not equivalent to it. As we have shown in Section 2 the projection framework, naturally defines the suitable multiscale basis, depending on the projection, and depending on the norm in which one wants to minimize error. In the projection framework, the approximate reconstruction operator $\tilde{\mathcal{M}}^o$ maps into H_0^1.

Within each coarse triangle, the basis functions $\psi_k = \phi_k^C + \tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ satisfy $\nabla\cdot(a\nabla\psi_k) = 0$, as in [22]. However, in the framework we have described the elements are continuous across the boundaries, even with oversampling, and they satisfy (3.3) along the triangle edges. Moreover, if full oversampling is used (oversampling up to the domain boundary), then the resulting coarse scale equation is exact. The coupling of fine scales in the solution is correctly captured, even when there is no scale separation in the coefficients.

Another difference between the two approaches lies in the fact that the multiscale strategy of [22, 23, 15] does not approximate the function μ_F, which is the affine component of the operator $\mathcal{M}$. Of course, if this function is zero, then there is no need to approximate it. However, $\mu_F \equiv 0$ if and only if $\langle f, v\rangle = 0$, for all $v \in X_F$. This condition implies that $f \in H^{-1}$ acts only on the finite dimensional space X_C. In particular, if piecewise linear elements and the H^1 projection are used, and if f has an L^1_{loc} representation, this condition implies that $f \equiv 0$. This follows from the fact that any smooth function η that is compactly supported within a coarse scale

triangle T is in $X_C^\perp$ since

$$(\nabla\eta, \nabla\phi_k) = \int_T \nabla\eta \cdot p_k \, dx = 0 \tag{5.14}$$

where p_k is the constant vector equal to $\nabla_k\phi$ in the triangle T. So, if $f \in L^1_{loc}$ and $f \neq 0$, then the term μ_F will not vanish. As we have demonstrated in the numerical simulations, leaving out the μ_F term may dramatically diminish the quality of the fine scale reconstruction if X_C is not sufficiently large.

The computational cost of the present approach is comparable to the cost of the method in [22]. In [22], the local problem is defined on a small region consisting of one coarse scale triangle and a layer for oversampling. In the present approach, the local problem is solved on the union of triangles sharing a common coarse vertex, plus a layer for over-sampling. So for a fixed level of oversampling, the computational cost of the local problems in the present framework is higher but still proportional to the cost of the local problems in the method of [22].

5.3.1. Harmonic Coordinates. Another approach to constructing multiscale basis functions is to transform the coarse scale elements by a change of variables $F(x) : D \to D$ using harmonic coordinates, which play a fundamental role in the analysis of the homogenization problem for periodic and random media (see Chapter 1 of [29], for example). This is the strategy of Allaire and Brizzi [1] and Owhadi and Zhang [31]. The transformed elements have the form (5.13) since

$$\begin{aligned}\psi_k(x) = \phi_k^C(F(x)) &= \phi_k^C(x) + \left(\phi_k^C(F(x)) - \phi_k^C(x)\right) \\ &= \phi_k^C(x) + \tilde{\mathcal{M}}^o\phi_k^C\end{aligned}$$

Although the operator $\tilde{\mathcal{M}}^o$ is linear and maps into H_0^1, the map $F(x)$ may distort the domain so that ψ_k and ϕ_k^C may have disjoint support. We also note that in this framework, $\tilde{\mathcal{M}}^o$ operates on ϕ_k^C rather than its gradient.

In [1], the coordinate change is defined locally, making the method more feasible computationally for large problems, in the same way that the localization described in Sections 3 and 4.5 improves efficiency. It was shown in [1] that when $\{\phi_k^C\}$ are piecewise linear, the transformed basis functions $\{\psi_k\}$ satisfy the definition used by Hou et.al. [22, 23] described in the preceding section.

The method of [31] is the same, except that the coordinate change is defined globally by $-\nabla \cdot (a\nabla F_i) = 0$, for $x \in D$ with $F_i = x_i$ on ∂D, $i = 1, \ldots, d$. If $a(x)$ and the map $F(x)$ have sufficient regularity (i.e. "stability" of the matrix $\sigma = (\nabla F)^T a \nabla F$, see Definition 1.1 of [31]), the authors show that $u(x) = w(F(x))$ where $w(y)$ satisfies an elliptic equation in non-divergence form. Then using regularity estimates for w they are able to show that (see Theorem 1.6)

$$\|u - u_h\|_{H^1} \leq Ch^\alpha \|f\|_{L^\infty}, \tag{5.15}$$

where u_h solves the Galerkin problem in the multiscale basis $\{\psi_k\}$ and α is some positive number depending on the coefficients. If $a(x)$ is smooth, then the condition necessary to obtain (5.15) is satisfied, but in this case, (5.15) might be obtained without changing variables. For nonsmooth coefficients it is not clear whether the necessary condition will hold.

When the coefficients are periodic and have the structure $a(x/\epsilon)$, then homogenization theory predicts that the map $F(x) : D \to D$ will be close to the identity

map for small ϵ. In general, however, this will not be the case, and the mesh that results from the global change of variables may be very distorted, as shown in [31]. Furthermore, the authors point out (section 1.3 of [31]) that localizing and coarsening the transformed basis may reduce the accuracy of the method. In the variational framework we have described, however, this is not a problem. The solution to the effective coarse scale equation is given in the standard nodal basis. The localization and adaptive refinement we have described is well-suited to a hierarchical decomposition of H^1 and does not require irregular meshes.

5.4. Multigrid methods. If the coefficients are sufficiently smooth, multigrid schemes [21] can approximate the solution with complexity $O(D)$, where D is the number of degrees of freedom. In principle, multigrid schemes could be used to compute the operator $\tilde{\mathcal{M}}^o$ and the function μ_F iteratively, with $O(D)$ complexity. In this case, the estimate (4.6) might be reduced to $O(N^d M^d)$, but the projection framework would present no advantage over multigrid in terms of computational cost, since the cost of computing the high resolution solution directly by multigrid would also be $O(N^d M^d)$. Nevertheless, the projection framework gives a robust method for computing the effective coarse-scale operator that does not require the coefficients to be regular. Moreover, the framework can be implemented easily on unstructured grids.

6. Summary and Conclusions. We have developed a projection method for multiscale finite element computations that captures correctly the homogenization limit of asymptotic scale separation. This method combined with a scheme for localizing the fine scale computations significantly reduces the computational cost for large problems. Numerical simulations using the public domain FreeFem++ [18] show that the projection framework captures well multiscale effects with limited oversampling. We also compare briefly this projection approach to other multiscale methods, such as wavelets, multiscale finite element bases and harmonic coordinates.

The numerical simulations suggest the need for an adaptive strategy to localize the fine scale computations when there is no clear scale separation in the coefficients. We have described such an adaptive strategy, and it would be interesting to compare this approach with other multiscale methods (like [1], [22], [31]), especially when the medium exhibits long-range channels of high permeability, a regime investigated in [19] using a local-global upscaling technique on adaptive grids. Also, it will be interesting to derive estimates of the solution error in terms of the number of layers used in oversampling and in terms of the parameters determining adaptivity. Our numerical experiments demonstrate that the functions $\tilde{\mathcal{M}}^o(\nabla\phi_k)$ decay to zero very quickly away from the support of ϕ_k, which suggests that only a few oversampling layers are needed for good approximation.

Another issue to be explored is the possibility of selective reconstruction of the fine scales. If the oscillations in the coefficients are statistically homogeneous, one might compute $\tilde{\mathcal{M}}^o(\nabla\phi_k)$ for only a few of the ϕ_k. The selection of ϕ_k could also be done adaptively, in a manner similar to the adaptive strategy in [25]. For example, the domain might be decomposed into large blocks, each made up of several coarse scale triangles. A coarse scale element is selected from the block and $\tilde{\mathcal{M}}^o(\nabla\phi_k)$ is computed. The block is then subdivided and the procedure repeated on the smaller blocks. If $(a\nabla\tilde{\mathcal{M}}^o(\nabla\phi_k), \nabla\tilde{\mathcal{M}}^o(\nabla\phi_k))$ changes little after this subdivision procedure, then $\tilde{\mathcal{M}}^o(\nabla\phi_k)$ might be used as the reconstruction for all coarse scale elements within the block. This form of selective reconstruction would be different from other forms of selective sampling of the coefficients which have been shown to be ineffective for random media [5]

Acknowledgment. We thank F. Hecht for assistance with FreeFem++.

7. Appendix A: Implementation Details. We compute u_C in three steps:

1. For each coarse scale basis function ϕ_k^C, $k = 1, \ldots, N_c$, compute the reconstruction $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ by solving (3.2). There are a total of N_c of these uncoupled problems to solve.
2. Compute the inhomogeneous part of the affine reconstruction operator by approximating (2.7).
3. Compute the coarse scale solution u_C by solving (2.6). To do so, we must first add to the original stiffness matrix the terms corresponding to the reconstructed fine scales.

7.1. Computing $\mathcal{M}^o$:. In solving (3.2), we construct a new mesh supported on $\hat{S}_k^C$, depending on the level of oversampling, and define the locally refined space $X_R \cap H_0^1(\hat{S}_k^C)$, of which Y_k^F is a subspace. Suppose that there are s basis elements $\{\phi_{k,p}^F\}_{p=1}^s$ in $X_R \cap H_0^1(\hat{S}_k^C)$. We solve the fine scale problem (3.2) as a constrained optimization problem, as expressed in (3.5) and (3.8). In matrix form the constrained problem (for a given k) has the structure

$$\begin{pmatrix} \mathcal{A} & \mathcal{C} \\ \mathcal{C}^T & 0 \end{pmatrix} \begin{pmatrix} \alpha_k \\ \lambda_k \end{pmatrix} = \begin{pmatrix} F \\ 0 \end{pmatrix} \tag{7.1}$$

Here $\mathcal{A}$ is the $s \times s$ matrix $\mathcal{A}_{q,p} = (a\nabla\phi_{k,p}^F, \nabla\phi_{k,q}^F)$. The variables $\alpha_{k,1}, \ldots, \alpha_{k,s}$ are the coefficients of $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$ in the $X_R \cap H_0^1(\hat{S}_k^C)$ basis. So, if we think of $\tilde{\mathcal{M}}^o$ as a matrix,

$$\tilde{\mathcal{M}}_{k,p}^o = \alpha_{k,p}, \qquad \tilde{\mathcal{M}}^o(\nabla\phi_k^C) = \sum_{p=1}^{s} \alpha_{k,p}(\nabla\phi_{k,p}^F).$$

The matrix $\mathcal{C}$ is the constraint matrix $\mathcal{C}_{p,r} = (\nabla\phi_{k,p}^F, \nabla\phi_r^C)$ where r runs through indices corresponding to neighbors of ϕ_k^C ($1 \le r \le$ #constraints). The vector $\lambda_k = (\lambda_{k,1}, \ldots, \lambda_{k,r})$ is the vector of Lagrange multipliers enforcing the projection constraints (e.g. (3.6) and (3.9)). The vector F is $F_q = -(a\nabla\phi_k^C, \nabla\phi_{k,q}^F)$, $q = 1, \ldots, s$.

7.2. Computing μ_F:. The function μ_F may be approximated in a manner similar to the computation of $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$. To localize the problem for μ_F we let $\mu_F(x) \approx \sum_{k=1}^{N_c} \phi_k^C(x)\mu_{F,k}(x)$ where $\mu_{F,k} \in Y_k^F$ solves the localized problem

$$\text{minimize } \frac{1}{2}(a\nabla v, \nabla v) - \langle f, v\rangle \tag{7.2}$$

among all $v \in X_R \cap H_0^1(\hat{S}_k)$ satisfying the constraints $(\nabla v, \nabla\phi_m^C) = 0$ for all $m \in N^C(r)$ and $r \in N^C(k)$. This corresponds to one layer of oversampling. A similar definition holds for a higher level of oversampling. In the sum, however, the terms $\phi_k^C\mu_{F,k}$ are functions supported in S_k^C, the support of the coarse scale element ϕ_k^C. The sum (7.2) represents an interpolation between the localized problems. In a neighborhood of x_k^C, $\mu_{F,k}(x)$ will be close to the true μ_F. Since the ϕ_k^C sum to one, this gives us an approximation of $\mu_{F,k}(x)$.

7.3. Computing u_C:. After computing each fine scale component $\tilde{\mathcal{M}}^o(\nabla\phi_k^C)$, $k = 1, \ldots, N_c$, we build the stiffness matrix for the coarse scale problem

$$(a(\mathcal{I} + \nabla\tilde{\mathcal{M}}^o)\nabla u_C, (\mathcal{I} + \nabla\tilde{\mathcal{M}}^o)\nabla v) = \langle f, v + \tilde{\mathcal{M}}^o(\nabla v)\rangle \ \ \forall\, v \in X_C, \tag{7.3}$$

which is an approximation of (2.6). This involves computing the integrals $(a\nabla\tilde{\mathcal{M}}^o(\nabla\phi_r^C), \nabla\tilde{\mathcal{M}}^o(\nabla\phi_t^C))$ whenever $\tilde{\mathcal{M}}^o(\nabla\phi_r^C)$ and $\tilde{\mathcal{M}}^o(\nabla\phi_t^C)$ overlap. If no oversampling is used, $\hat{S}_k^C = S_k^C$, and the functions $\tilde{\mathcal{M}}^o(\nabla\phi_r^C)$ and $\tilde{\mathcal{M}}^o(\nabla\phi_t^C)$ overlap if and only if ϕ_r^C and ϕ_t^C are neighbors (the nodes x_r^C and x_k^C are connected by a single edge). So, the stiffness matrix has the same structure as the stiffness matrix for the coarse-scale Galerkin scheme.

With two layers of oversampling, the functions $\tilde{\mathcal{M}}^o(\nabla\phi_r^C)$ and $\tilde{\mathcal{M}}^o(\nabla\phi_t^C)$ overlap if and only if the nodes x_r^C and x_t^C are connected by three edges. This longer range coupling reduces the sparsity in the stiffness matrix. We also must compute the integrals $(a\nabla\tilde{\mathcal{M}}^o(\nabla\phi_r^C), \nabla\phi_t^C)$ whenever $\tilde{\mathcal{M}}^o(\nabla\phi_r^C)$ and ϕ_t^C overlap, and the integrals $(a\nabla\phi_r^C, \tilde{\mathcal{M}}^o(\nabla\phi_t^C))$ whenever ϕ_r^C and $\tilde{\mathcal{M}}^o(\nabla\phi_t^C)$ overlap.

8. Appendix B: Proof of Proposition 5.1. The fact that the convergence in (i) is strong in H^1 is not surprising since $u_C^{\epsilon,L} \in X_C$ lies in a finite dimensional space. Since $u_C^{\epsilon,L}$ is bounded in H^1 independently of ϵ, we can take a subsequence converging strongly to a limit $\bar{\zeta} \in X_C$. After we examine the behavior of $\tilde{\mathcal{M}}_L^{o,\epsilon}$, we will see that this limit is $\bar{\zeta} = \bar{u}_C^L$, the solution to (5.11).

For each $\phi_k \in X_C$, the problem defining $\tilde{\mathcal{M}}_L^{o,\epsilon}$ is:

$$(a^\epsilon\nabla\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k), \nabla v) = -(a^\epsilon\nabla\phi_k, \nabla v), \quad \forall v \in Y_k^F,$$

where Y_k^F denotes the approximation of X_F defined by L layers of oversampling around the support of ϕ_k, as described in Section 3.2. We will now show that

$$\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k) \to \overline{\mathcal{M}}_L^o(\nabla\phi_k) \tag{8.1}$$

weakly in H^1, where $\overline{\mathcal{M}}_L^o(\nabla\phi_k) \in Y_k^F \subset X_F$ is defined by $(\bar{a}\nabla\overline{\mathcal{M}}_L^o(\nabla\phi_k), \nabla v) = -(\bar{a}\nabla\phi_k, \nabla v)$ for all $v \in Y_k^F$. Moreover,

$$a^\epsilon\left(\nabla\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k) + \nabla\phi_k\right) \to \bar{a}\left(\nabla\overline{\mathcal{M}}_L^o(\nabla\phi_k) + \nabla\phi_k\right) \tag{8.2}$$

weakly in $(L^2(D))^d$.

These claims can be proved by the oscillating test function method, as follows (see Section 3.2 of [6]). First, from the boundedness of $\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k)$ in H^1 and the boundedness of $a^\epsilon\nabla\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k)$ in $(L^2(D))^d$, we may extract a subsequence such that $\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k) \to g^k \in Y_k^F$ weakly in H^1 and strongly in L^2, while $a^\epsilon\left(\nabla\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k) + \nabla\phi_k\right) \to \xi^k$ weakly in $(L^2(D))^d$.

Next, we must identify the limits g^k and ξ^k as those given in (8.1) and (8.2). Let $\eta(x)$ have compact support contained within one of the coarse scale triangles. Thus, $\eta \in X_F$. Also, for each k, either $\eta \in Y_k^F$, or the supports of η and $\hat{S}_k^C$ are disjoint. So we may assume $\eta \in Y_k^F$. Let $w^\epsilon(x) = x \cdot e_j + \epsilon\chi^j(x/\epsilon)$, so that $\eta(x)w^\epsilon(x) \in Y_k^F$, as well, and we can use $\eta(x)w^\epsilon(x)$ as a test function:

$$(a^\epsilon\nabla(\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k) + \phi_k), \nabla(\eta w^\epsilon)) = 0. \tag{8.3}$$

By definition of w^ϵ,

$$(a^\epsilon\nabla v, \nabla w^\epsilon) = (\nabla v, a^\epsilon\nabla w^\epsilon) = 0 \tag{8.4}$$

for all $v \in H_0^1(D)$. Therefore, (8.3) and (8.4) imply that

$$(a^\epsilon\nabla(\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k) + \phi_k), (\nabla\eta)w^\epsilon)) = ((\tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla\phi_k) + \phi_k)(\nabla\eta), a^\epsilon\nabla w^\epsilon). \tag{8.5}$$

As $\epsilon \to 0$, w^ϵ converges strongly to $x \cdot e_j$ in $L^2(D)$, while $a^\epsilon \nabla w^\epsilon$ converges weakly in $(L^2(D))^d$ to $\bar{a}_j$, the j^{th} column of the matrix $\bar{a}$. Therefore, as $\epsilon \to 0$ in (8.5),

$$(\xi^k, (\nabla \eta) x \cdot e_j) = ((g^k + \phi_k)(\nabla \eta), \bar{a}_j) = \bar{a}_j \cdot \int_D (g^k(x) + \phi_k(x)) \nabla \eta(x)\, dx. \tag{8.6}$$

Also, we must have $(\xi^k, \nabla(\eta(x) x \cdot e_j)) = 0$, since $\eta(x) x \cdot e_j \in X_F$. Therefore,

$$(\xi^k, (\nabla \eta) x \cdot e_j) = -\int_D \xi_j^k(x) \eta(x)\, dx. \tag{8.7}$$

By combining (8.6) and (8.7) we see that

$$\int_D \xi_j^k(x) \eta(x)\, dx = -\bar{a}_j \cdot \int_D (g^k(x) + \phi_k(x)) \nabla \eta(x)\, dx = \sum_i \bar{a}_{ij} \int_D \frac{\partial (g^k + \phi_k)}{\partial x_i} \eta\, dx.$$

Since η was arbitrarily chosen, this implies that

$$\xi_j^k = \sum_i \bar{a}_{ij} \frac{\partial (g^k + \phi_k)}{\partial x_i} = \sum_i \bar{a}_{ji} \frac{\partial (g^k + \phi_k)}{\partial x_i} \tag{8.8}$$

for almost every x inside each coarse scale triangle. Here we have used the fact that $\bar{a}$ is symmetric, which is implied by the symmetry of a_{ij} (see [6] Chap. 2). Hence, (8.8) holds almost everywhere. This shows that g^k solves

$$(\bar{a}(\nabla g^k + \nabla \phi_k), \nabla v) = 0, \quad \forall v \in Y_k^F.$$

Uniqueness of the solution implies that $g^k = \overline{\mathcal{M}}_L^o(\nabla \phi_k)$, and that the convergence occurs as $\epsilon \to 0$. This proves (8.1) and (8.2), and statement (ii) of the proposition.

Since $u_C^{\epsilon,L} \to \zeta$ strongly in H^1, equation (8.2) implies that

$$a^\epsilon(\nabla u_C^{\epsilon,L} + \nabla \mathcal{M}_L^{o,\epsilon}(\nabla u_C^{\epsilon,L})) \to \bar{a}(\nabla \bar{\zeta} + \nabla \overline{\mathcal{M}}_L^o(\nabla \bar{\zeta})) \tag{8.9}$$

weakly in $(L^2(D))^2$. Therefore, equation (8.2) also implies that

$$a^\epsilon \nabla \mathcal{M}^{o,\epsilon}(\nabla u_C^{\epsilon,L}) \to \bar{a} \nabla \overline{\mathcal{M}}^o(\nabla \bar{\zeta}) + (\bar{a} - \langle a \rangle) \nabla \bar{\zeta} \tag{8.10}$$

weakly in $(L^2(D))^d$, since $a^\epsilon \nabla u_C^{\epsilon,L} \to \langle a \rangle \nabla \bar{\zeta}$ weakly in $(L^2(D))^d$.

The function $a^\epsilon(\nabla u_C^{\epsilon,L} + \nabla \tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla u_C^{\epsilon,L}))$ is divergence free almost everwhere inside each coarse scale triangle, when we consider it as an element of $(L^2(T_i))^d$ for each triangle T_i. In general, this function may not be divergence-free when considered as a function in $(L^2(D))^d$. Nevertheless, we can apply the Div-Curl Lemma [32] within each coarse scale triangle to conclude that

$$\begin{aligned} &\lim_{\epsilon \to 0} (a^\epsilon(\nabla u_C^{\epsilon,L} + \nabla \tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla u_C^{\epsilon,L})), \nabla(v + \tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla v))) \\ &\quad = \sum_{T_i} \lim_{\epsilon \to 0} \int_{T_i} (a^\epsilon(\nabla u_C^{\epsilon,L} + \nabla \tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla u_C^{\epsilon,L}))) \cdot \nabla(v + \tilde{\mathcal{M}}_L^{o,\epsilon}(\nabla v))\, dx \\ &\quad = (\bar{a}(\nabla \bar{\zeta} + \nabla \overline{\mathcal{M}}_L^o(\nabla \bar{\zeta})), \nabla(v + \overline{\mathcal{M}}_L^o(\nabla v))) \end{aligned} \tag{8.11}$$

We have already shown that for fixed $v \in X_C$, $\mathcal{M}_L^{o,\epsilon}(\nabla v) \to \overline{\mathcal{M}}_L^o(\nabla v)$ strongly in $L^2(D)$ as $\epsilon \to 0$. Therefore,

$$\lim_{\epsilon \to 0} (f, v + \mathcal{M}_L^{o,\epsilon}(\nabla v)) = (f, v + \overline{\mathcal{M}}_L^o(\nabla v)) \tag{8.12}$$

Combining (8.11) and (8.12), we see that $\bar{\zeta}$ is the solution to the approximate symmetrized equation (5.11). Hence, $\bar{\zeta} = \bar{u}_C^L$, and convergence holds along the entire sequence. We have now shown that (i), (ii), (iv), and (5.11) hold. The analysis of the $\mu_F^{\epsilon,L}$ term and claims (iii) and (iv) follow from similar arguments applied to the equation for $\mu_F^{\epsilon,L}$. This completes the proof of Proposition 5.1. □

REFERENCES

[1] G. Allaire and R. Brizzi, *A multiscale finite element method for numerical homogenization*, Multiscale Model. Sim. Vol. 4 (2005), No. 3, pp. 790-812.

[2] T. Arbogast, *Analysis of a two-scale, locally conservative subgrid upscaling for elliptic problems.* SIAM J. Numer. Anal. Vol. 42 (2004), No. 2, 576–598.

[3] ———, *Numerical subgrid upscaling of two-phase flow in porous media*, Numerical treatment of multiphase flows in porous media, pp. 35–49, Lect. Notes in Phys., No. 552, Springer, Berlin, 2000.

[4] T. Arbogast, K. Boyd, *Subgrid upscaling and mixed multiscale finite elements*, SIAM J. Numer. Anal. Vol. 44, (2006) No. 3, pp. 1150-1171.

[5] M. Avellaneda, T. Hou, G. Papanicolaou, *Finite difference approximations for partial differential equations with rapidly oscillating coefficients.* RAIRO Modél. Math. Anal. Numér., Vol. 25 (1991), No. 6, pp. 693-710.

[6] A. Bensoussan, J.L. Lions, G. Papanicolaou, *Asymptotic Analysis for Periodic Structures*, North Holland, Amsterdam, 1978.

[7] P. Binev, W. Dahmen, R. DeVore, *Adaptive finite element methods with convergence rates*, Numer. Math. Vol. 97 (2004), pp. 219-268.

[8] F. Brezzi, *Interacting with the subgrid world*, Chapman & Hall/CRC Research Notes in Mathematics, Vol. 420, Chapman & Hall/CRC, Boca Raton, 2000.

[9] F. Brezzi, L.P. Franca, T.J.R. Hughes, and A. Russo, $b = \int g$, Comput. Methods Appl. Mech. Engrg. Vol. 145 (1997), pp. 329 - 339.

[10] A. Chertock and D. Levy, *On wavelet-based numerical homogenization*, SIAM MMS, 2004.

[11] A. Cohen, W. Dahmen, and R. DeVore, *Adaptive wavelet methods for elliptic operator equations: convergence rates.* Mathematics of Computation, 2000.

[12] L. Demkowicz *Computing with hp-adaptive finite elements* Chapman and Hall, 2006.

[13] W. E and B. Engquist, *The heterogeneous multiscale methods.* Commun. Math. Sci. 1 (2003), no. 1, 87–132.

[14] W. E, P. Ming, and P. Zhang, *Analysis of the heterogeneous multiscale method for elliptic homogenization problems*, J. Am. Math. Soc., Vol. 18 (2005), pp. 121-156.

[15] Y. Efendiev, T. Hou, and X.-H. Wu, *Convergence of a nonconforming multiscale finite element method*, SIAM J. Numer. Anal. Vol 37 (2000), No. 3, pp. 888-910.

[16] B. Engquist and O. Runborg *Waveled-based numerical homogenization with applications*, Lect. Notes Comput. Sci. Eng. 20, Springer 2002.

[17] B. Engquist and O. Runborg, *Projection generated homogenization*, in Multiscale Problems in Science and Technology, pp. 129-150, Springer, Berlin, 2002.

[18] FreeFem++, version 2.13.0, http://www.freefem.org/

[19] M. Gerritsen and J.V. Lambers, *A specialized upscaling method for adaptive grids: tight integration of local-global upscaling and adaptivity leads to accurate solution of flow in heterogeneous formations*, Preprint 2006.

[20] M. Golubitsky and D.G. Schaeffer, *Singularities and groups in bifurcation theory*, Vol. I. Applied Mathematical Sciences, 51. Springer-Verlag, New York, 1985.

[21] W. Hackbusch, *Multigrid methods and applications*, Springer-Verlag, Berlin, 1985.

[22] T. Hou and X.-H. Wu, *A multiscale finite element method for elliptic problems in composite materials and porous media* J. Comp. Phys. Vol. 134 (1997), pp. 169-189.

[23] T. Hou, X.-H. Wu, Z. Cai, *Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients*, Math. Comp. Vol. 68 (1999), No. 227, pp. 913–943.

[24] T. Hou, A. Westhead, and D. Yang, *A framework for modeling subgrid effects for two-phase flows in porous media* Multiscale Model. Simul. Vol. 5 (2006), No. 4, pp. 1087-1127.

[25] T. Hou, D. Yang, and H. Ran, *Multiscale analysis and computation for the 3D incompressible Navier-Stokes equations*, preprint, 2007.

[26] T. J. R. Hughes, *Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formula-*

tion, subgrid scale models, bubbles and the origins of stabilized methods, Comput. Methods Appl. Mech. Engrg. Vol. 127 (1995), no. 1-4, pp. 387–401.

[27] T.J.R. Hughes, G. Feijóo, L. Mazzei, J.-B. Quincy, *The variational multiscale method—a paradigm for computational mechanics* Comput. Methods Appl. Mech. Engrg. Vol. 166 (1998), No. 1-2, pp. 3-24

[28] P. Ming and X. Yue, *Numerical methods for multiscale elliptic problems*, J. Comp. Phys. Vol. 214 (2006), pp. 421-445.

[29] S. Molchanov, *Lectures on random media*, in Lectures on Probability Theory, P. Bernard, Ed. Lecture Notes in Mathematics, No. 1581, Springer-Verlag, Berlin, 1994.

[30] F. Murat and L. Tartar, *H-convergence*. in Topics in the mathematical modeling of composite materials, pp. 21–43, Birkhäuser, Boston, 1997.

[31] H. Owhadi and L. Zhang, *Metric-based upscaling* Comm. Pure Appl. Math. Vol. 59 (2006) pp. 1-49.

[32] L. Tartar, *Compensated compactness and applications to partial differential equations*, in Nonlinear Analysis and Mechanics: Hariot-Watt Symposium, Vol. IV, pp. 136-212. Pitman Publishing Ltd., London, 1979.