



# An iteratively adaptive multi-scale finite element method for elliptic PDEs with rough coefficients



Thomas Y. Hou<sup>a</sup>, Feng-Nan Hwang<sup>b,\*</sup>, Pengfei Liu<sup>a</sup>, Chien-Chou Yao<sup>b</sup>

<sup>a</sup> Department of Computing + Mathematical Sciences, California Institute of Technology, Pasadena, CA 91125, USA

<sup>b</sup> Department of Mathematics, National Central University, Jhongli District, Taoyuan City 32001, Taiwan

## ARTICLE INFO

### Article history:

Received 26 June 2015

Received in revised form 28 August 2016

Accepted 2 February 2017

Available online 9 February 2017

### Keywords:

Iteratively adaptive MsFEM

Global-to-local information transfer

Two-level multi-grid method

Elliptic equation with rough coefficients

Convection-dominated diffusion equation

## ABSTRACT

We propose an iteratively adaptive Multi-scale Finite Element Method (MsFEM) for elliptic PDEs with rough coefficients. The choice of the local boundary conditions for the multi-scale basis functions determines the accuracy of the MsFEM numerical solution, and one needs to incorporate the global information of the elliptic equation into the local boundary conditions of the multi-scale basis functions to recover the underlying fine-mesh solution of the equation. In our proposed iteratively adaptive method, we achieve this global-to-local information transfer through the combination of coarse-mesh solving using adaptive multi-scale basis functions and fine-mesh smoothing operations. In each iteration step, we first update the multi-scale basis functions based on the approximate numerical solutions of the previous iteration steps, and obtain the coarse-mesh approximate solution using a Galerkin projection. Then we apply several steps of smoothing operations to the coarse-mesh approximate solution on the underlying fine mesh to get the updated approximate numerical solution. The proposed algorithm can be viewed as a nonlinear two-level multi-grid method with the restriction and prolongation operators adapted to the approximate numerical solutions of the previous iteration steps. Convergence analysis of the proposed algorithm is carried out under the framework of two-level multi-grid method, and the harmonic coordinates are employed to establish the approximation property of the adaptive multi-scale basis functions. We demonstrate the efficiency of our proposed multi-scale methods through several numerical examples including a multi-scale coefficient problem, a high-contrast interface problem, and a convection-dominated diffusion problem.

© 2017 Elsevier Inc. All rights reserved.

## 1. Introduction

Many problems of practical importance in science and engineering have multi-scale features. Porous media simulation and composite materials modeling are typical examples of such kind. The small-scale features of the problem can have significant impact on the large-scale properties of the solutions, and one needs a very fine discretization that resolves the small-scale variation of the problem to obtain faithful numerical results. For these problems, methods that allow people to effectively incorporate the small-scale features of the problem into the large-scale properties of the solutions are desired. In this work, we consider the following second order linear elliptic partial differential equation (PDE) with rough coefficient,

\* Corresponding author. Fax: +886 3 425 7379.

E-mail addresses: hou@cms.caltech.edu (T.Y. Hou), hwangf@math.ncu.edu.tw (F.-N. Hwang), plliu@caltech.edu (P. Liu), ccyao001@gmail.com (C.-C. Yao).

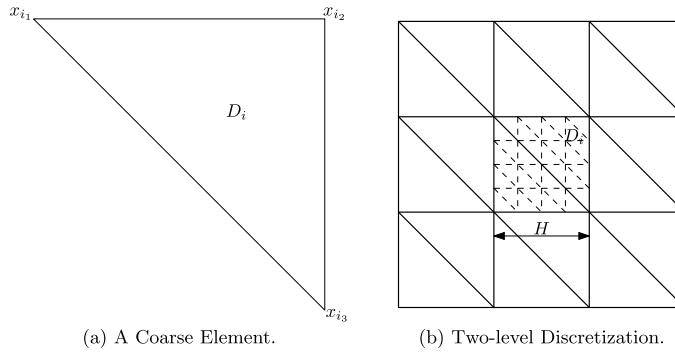


Fig. 1. Discretization of the problem.

$$\begin{cases} -\operatorname{div}(a(x)\nabla u(x)) = f(x), & x \in D, \\ u(x)|_{\partial D} = 0, \end{cases} \tag{1.1}$$

and the proposed method will also be applied to a convection-dominated diffusion problem.

Here we assume that  $D$  is a convex polygon domain in  $R^2$ , and the forcing term  $f(x) \in L^2(D)$ . We also assume that the equation is uniformly elliptic, i.e., there exist  $\lambda_{\min} > 0$  and  $\lambda_{\max} > 0$  such that

$$a(x) \in [\lambda_{\min}, \lambda_{\max}]. \tag{1.2}$$

We do not assume any regularity of the coefficient  $a(x) \in L^\infty(\Omega)$ , which can be arbitrarily rough, thus equation (1.1) can be used to model diffusion process in strongly heterogeneous media. Classical finite element methods use piecewise linear (polynomial) functions to discretize the equation, and their convergence depends on the following approximation property and regularity result

$$\|u(x) - Ju(x)\|_{H_0^1(D)} \leq CH\|u(x)\|_{H^2(D)}, \quad \|u(x)\|_{H^2(D)} \leq C\|f(x)\|_{L^2(D)}, \tag{1.3}$$

where  $Ju(x)$  is the piecewise linear interpolation of  $u(x)$ , and  $H$  is the underlying mesh size. We can see that  $O(H)$  accuracy in the  $H_0^1(D)$  norm can be obtained if a discretization of size  $H$  is employed. However, classical finite element methods may fail if the coefficient  $a(x)$  is rough, because in this case the solution  $u(x)$  loses regularity and  $\|u(x)\|_{H^2(D)}$  cannot be bounded by  $\|f(x)\|_{L^2(D)}$  in (1.3). It is actually shown in [9] that the polynomial finite element methods can perform arbitrarily badly in this setting. In practice, one needs a much finer mesh that resolves the small scale features of  $a(x)$  to achieve the desired  $O(H)$  accuracy.

1.1. The multi-scale finite element method

The Multi-scale Finite Element Methods (MsFEM) [47–49,28,19] were developed to solve these multi-scale PDEs. Instead of piecewise polynomials in the classical finite element methods, MsFEM employs basis functions that incorporate properties of the elliptic operator to discretize the equation (1.1),

$$\phi_1(x), \phi_2(x), \dots, \phi_i(x), \dots, \phi_n(x) \in H_0^1(\Omega). \tag{1.4}$$

Consider a coarse-mesh discretization of the domain with mesh size  $H$ , and an underlying fine mesh of size  $h$  that refines the coarse mesh. This two-level discretization is illustrated in Fig. 1. Let

$$x_1, x_2, \dots, x_i, \dots, x_n \tag{1.5}$$

be the interior node points of the coarse mesh. The MsFEM basis functions  $\phi_i(x)$  in (1.4) are associated with the interior coarse mesh node points  $x_i$  in (1.5), and they satisfy

$$\phi_i(x_j) = \delta_{ij}.$$

Let  $D_i$  be an element of the coarse mesh, see Fig. 1, and let  $x_{i_1}, x_{i_2}, x_{i_3}$  be its node points, where

$$i_1, i_2, i_3 \in \{1, 2, \dots, n\}.$$

Then the multi-scale basis function  $\phi_{i_j}(x)$  on the element  $D_i$  is constructed by solving local elliptic boundary value problem with appropriate local boundary conditions on  $\partial D_i$ ,

$$-\operatorname{div}(a(x)\nabla \phi_{i_j}(x)) = 0, \quad x \in D_i, \quad \phi_{i_j}(x_{i_k}) = \delta_{kj}. \tag{1.6}$$

With the basis functions  $\phi_i(x)$ ,  $i = 1, \dots, n$ , MsFEM finds the numerical solution

$$u_H^{MS}(x) \in V_H = \text{span}\{\phi_1(x), \phi_2(x), \dots, \phi_n(x)\} \subset H_0^1(\Omega), \tag{1.7}$$

using the Galerkin projection. Namely, one finds  $u_H^{MS}(x) \in V_H$ , such that

$$a(u_H^{MS}(x), v(x)) = \langle f(x), v(x) \rangle, \quad \text{for all } v(x) \in V_H, \tag{1.8}$$

where  $a(u(x), v(x)) = \int_{\Omega} \nabla u(x)^t a(x) \nabla v(x) dx$ ,  $\langle f(x), v(x) \rangle = \int_{\Omega} f(x) v(x) dx$ .

The numerical solution defined above satisfies the following optimal property under the energy norm

$$\|u(x) - u_H^{MS}(x)\|_E = \inf_{v(x) \in V_H} \|u(x) - v(x)\|_E, \quad \|u(x)\|_E^2 = a(u(x), u(x)). \tag{1.9}$$

In numerical implementation, the local elliptic problems (1.6) are solved using the underlying fine mesh, thus the resulting multi-scale basis functions (1.4) are represented as piecewise linear functions on the fine mesh. We denote  $V_h$  as the space of piecewise linear basis functions on the underlying fine mesh of size  $h$ , and  $u_h(x)$  as the Galerkin numerical solution to (1.1) using trial space  $V_h$ ,  $u_h(x) \in V_h$ ,

$$a(u_h(x), v(x)) = \langle f(x), v(x) \rangle, \quad \text{for all } v(x) \in V_h(x).$$

Then based on the above optimal property (1.9), the optimal accuracy that one can expect from the MsFEM solution  $u_H^{MS}(x)$  (1.9) is limited by the underlying fine mesh solution  $u_h$ ,

$$\|u_H^{MS}(x) - u(x)\|_E \geq \|u_h(x) - u(x)\|_E, \tag{1.10}$$

because the trial space in (1.9),  $V_H$  defined in (1.7), is a subspace of  $V_h$ ,

$$V_H \subset V_h.$$

Thus to obtain faithful numerical results in the multi-scale numerical solution  $u_H^{MS}(x)$  (1.8), we require that the underlying fine mesh resolve the small scale variation of the coefficient  $a(x)$  in (1.1), and  $\|u_h(x) - u(x)\|$  be much smaller than our desired accuracy of the numerical computation,

$$\|u(x) - u_h(x)\| \ll 1. \tag{1.11}$$

In this paper, we neglect this part of error, and do not distinguish between  $u(x)$  and  $u_h(x)$ .

Convergence analysis of MsFEM in the periodic homogenization setting, where  $a(x) = A(\frac{x}{\epsilon})$  with  $\epsilon = o(1)$ , was given in [48,29,17]. To construct the multi-scale basis function  $\phi_{i_j}(x)$  on  $D_i$  in (1.6), appropriate local boundary conditions on  $\partial D_i$  are needed. We will review in section 2 that the local boundary conditions of the basis functions are crucial and determine the accuracy of the MsFEM numerical solution. With the optimal choice of local boundary conditions for the basis and some local bubble corrections, MsFEM can indeed recover the underlying fine-mesh solution. However, due to the nonlocal nature of the solution operator to elliptic equations, the local solution to (1.1) depends on the coefficient  $a(x)$  and  $f(x)$  on the whole domain  $D$ . And to exactly recover the underlying fine-mesh solution, one needs to incorporate the global information of the equation into the local boundary conditions of the basis functions in (1.6).

### 1.2. Main contributions of this work

In this paper we propose an iteratively adaptive method, Algorithm 1, to construct the multi-scale basis. The global information of the elliptic equation is transferred to the local boundary conditions of the multi-scale basis through iterative coarse-mesh solving and fine-mesh smoothing operations.

In the  $s$ -th step of the iteration, we first construct basis functions  $\phi_i^s(x)$ ,  $i = 1, 2, \dots, n$ , based on the approximate numerical solutions from the previous several iteration steps, namely,  $u^k(x)$ ,  $k \leq s - 1$ . We require that these multi-scale basis functions can exactly recover  $u^{s-1}(x)$  on the boundary of the coarse-mesh elements,  $\Gamma = \cup_i \partial D_i$ . Then we solve the equation (1.1) using the basis functions  $\phi_i^s(x)$  and a Galerkin projection, and correct the numerical solution  $u_H^{(s)}(x)$  by solving a series of local bubble problems. Finally, we apply several steps of smoothing operations on the fine mesh to the corrected numerical solution to get the updated approximate solution  $u^s(x)$ . Throughout the iteration process, the approximation property of the basis functions  $\phi_i^s(x)$  to the solution  $u(x)$  gets improved. Our numerical results suggest that the numerical solutions  $u^s(x)$  converge to the underlying fine-mesh solution exponentially fast. The coarse-mesh solving and the fine-mesh smoothing operations are the two key steps of the proposed algorithm.

Algorithm 2 is derived as a stabilized variant of Algorithm 1, and is essentially a two-level multi-grid method that employs multi-scale basis functions for the restriction and prolongation operations. By comparing Algorithm 1 with Algorithm 2, we show that even without explicit restriction operation, Algorithm 1 can indeed be viewed as a nonlinear version of the two-level multi-grid method where the restriction and prolongation operators are adapted to the approximate solutions of the previous iteration steps.

Convergence analysis of the proposed algorithms is carried out under the framework of the two-level multi-grid method. And the harmonic coordinates [62] are employed to establish the approximation property of the adaptively constructed multi-scale basis functions to the solution space of equation (1.1).

There exist some multi-scale methods in the literature that adaptively update the boundary conditions of the multi-scale basis. In [25,57], the adaptive multi-scale basis functions in each iteration step are constructed using the local oversampling technique, and the local boundary conditions on the oversampling domain are constructed based on approximate solutions of the previous steps. There exist no smoothing operations in these methods, and they cannot obtain fine mesh accuracy as the present work. In [41], an adaptive correction function is introduced in each iteration step, and the resulting method is similar to Algorithm 2 in this work. No convergence analysis has been provided for the previous adaptive multi-scale methods [41,25,57]. After suitable modifications, these methods may be analyzed under the same two-level multi-grid method framework presented in the current paper.

To demonstrate the applicability of our proposed methods for a variety of applications with multi-scale features, we consider three classes of PDE problems, including elliptic problems with rough coefficients [41,42,47,48], elliptic interface problems with high-contrast discontinuous coefficients [18,19,52,57], and convection-dominated diffusion equations [20,31,33,34,69]. In all the three types of problems that we consider, we observe significant improvement of the proposed methods over classical methods.

### 1.3. Review of multi-scale methods

Considerable amount of effort has been devoted to analyzing and numerically solving the multi-scale PDE (1.1), and we list a few previous works related to the present paper below.

The classical homogenization theories, including the periodic homogenization where  $a(x) = A(\frac{x}{\epsilon})$  [11,50,65,3,2], and the H, G,  $\Gamma$ -convergence theories [59,22,21,71,70,58,36], consider the convergence (in an appropriate sense) of a sequence of operators parameterized by  $\epsilon$  as  $\epsilon \rightarrow 0$ .

Numerical methods for the multi-scale elliptic PDE (1.1) can be divided into two categories. Methods in the first category can be viewed as fast solvers of the equation on the underlying fine mesh and alternatives of the multi-grid methods [32,13,38,39]. Multi-grid methods are known as the fastest solvers for elliptic boundary value problems with linear computational complexity. However, due to the lack of regularity of the solutions, classical multi-grid methods in general would fail for the multi-scale elliptic PDE (1.1) [23,30,26]. Methods in the first category include the algebraic multi-grid methods (AMG) [1,15,14,67,72,51], the hierarchical matrix methods [40,10], AMG based on energy minimization [54,76,78,77,74], stabilized hierarchical basis methods [73,75], wavelet based methods [16,12,4,24], and the multi-scale domain decomposition methods for problems with high contrast coefficient [37,35].

Methods in the second category, including the MsFEM, are model reduction methods. They employ special basis functions adapted to the elliptic operator instead of piecewise polynomials to discretize the equation to reduce the total degrees of freedom. These methods generally employ a two-level discretization as shown in Fig. 1, and are only able to achieve coarse-mesh accuracy. Methods in the second category also include the generalized finite element methods [56,8,5], oversampling methods [7,6,45], localization methods [63,53,43,66,44,46], harmonic coordinates transformation [62], the polyharmonic splines [64,60], and a recent method that depends on the multi-resolution decomposition of the solution space [61].

The proposed Algorithm 1 in this paper is able to achieve fine-mesh accuracy by iteratively incorporating the global information of the equation into the local boundary conditions of the basis functions. Thus it falls into the first category of multi-scale methods. Namely, it can be viewed as a fast fine-mesh solver and an alternative of the classical multi-grid methods. On the other hand, the proposed Algorithm 1 employs special basis to discretize the equations on the coarse mesh, and from this perspective it is similar to the model reduction method in the second category. In multi-scale methods of the second category, building basis with robust approximation property is in general expensive. The proposed method bypasses this difficulty by adaptively improving the approximation property of the multi-scale basis in the iteration.

The rest of the paper is organized as follows. In section 2, we briefly review multi-scale basis functions and the importance of their local boundary conditions. In section 3, we detail the proposed method to solve the multi-scale PDE (1.1) and derive a variant of this method. In section 4, we provide convergence analysis of the proposed algorithms under the two-level multi-grid method framework. In section 5, we present some numerical results employing the proposed method. Concluding remarks are made in section 6.

## 2. Motivation: multi-scale basis and the importance of their local boundary conditions

Given a coarse-mesh discretization of the domain  $D = \cup_i D_i$  as shown in Fig. 1, we consider the following decomposition of the solution to (1.1),  $u(x) = u^1(x) + u^2(x)$ ,

$$\begin{cases} -\operatorname{div}(a(x)\nabla u^1(x)) = 0, & x \in D_i; \\ u^1(x)|_{\partial D_i} = u(x)|_{\partial D_i}; \\ -\operatorname{div}(a(x)\nabla u^2(x)) = f(x), & x \in D_i; \\ u^2(x)|_{\partial D_i} = 0. \end{cases} \quad (2.1)$$

The  $u^1(x)$  part of the solution is  $a(x)$ -harmonic on each coarse element and only depends on the trace of the solution on the edges of coarse elements,

$$\Gamma = \cup_i \partial D_i. \tag{2.2}$$

We call  $u^1(x)$  the  $a(x)$ -harmonic part of the solution.  $u^2(x)$  depends on the forcing term  $f(x)$  on each coarse element, and it can be constructed by solving the local elliptic problems with homogeneous boundary conditions (2.1). In this paper, we refer to the local problems (2.1) as *local bubble problems*. We call  $u^2(x)$  the bubble part of the solution.  $u^1(x)$  and  $u^2(x)$  are  $a(x)$ -orthogonal,

$$a(u^1(x), u^2(x)) = 0.$$

Besides, the local bubble part of the solution  $u^2(x)$  is small on each coarse element,

$$\|u^2(x)|_{D_i}\|_{H^1_0(D_i)} \leq CH \|f(x)\|_{L^2(D_i)}, \quad \|u^2(x)|_{D_i}\|_{L^2(D_i)} \leq CH^2 \|f(x)\|_{L^2(D_i)}, \tag{2.3}$$

which can be obtained using a scaling argument. Summing (2.3) up over each coarse element, we obtain

$$\|u^2(x)\|_{H^1_0(D)} \leq CH \|f(x)\|_{L^2(D)}, \quad \|u^2(x)\|_{L^2(D)} \leq CH^2 \|f(x)\|_{L^2(D)}. \tag{2.4}$$

Motivated by the above orthogonal decomposition of the solution, a special type of basis functions, the *multi-scale basis functions* were introduced in [45], which generalize the basis functions employed in MsFEM. They are  $a(x)$ -harmonic within each coarse element, but are not necessarily nodal basis.

**Definition ([45]).** (Multi-scale basis) Consider basis functions  $\phi_1(x), \phi_2(x), \dots, \phi_n(x) \in H^1_0(D)$ , if they are  $a(x)$ -harmonic on each coarse element  $D_j$  of the discretization (but not across the boundary), namely,

$$-\text{div}(a(x)\nabla\phi_i(x)) = 0, \quad x \in D_j,$$

then they are called multi-scale basis functions.

Due to the smallness of  $u^2(x)$  (2.4), one can get the following optimal property of multi-scale basis.

**Proposition ([45]).** Consider a set of basis functions  $\psi_i(x) \in H^1_0(D)$ ,  $i = 1, 2, \dots, m$ , and a set of multi-scale basis  $\phi_i(x)$ ,  $i = 1, 2, \dots, n$ . Denote the Galerkin solutions using these basis functions as  $u_H^\psi(x)$  and  $u_H^{MS}(x)$  respectively, and let the union of coarse element boundaries be  $\Gamma = \cup_i \partial D_i$ . If

$$\text{span}\{\psi_1(x)|_\Gamma, \psi_2(x)|_\Gamma, \dots, \psi_m(x)|_\Gamma\} = \text{span}\{\phi_1(x)|_\Gamma, \phi_2(x)|_\Gamma, \dots, \phi_n(x)|_\Gamma\}, \tag{2.5}$$

then

$$\|u(x) - u_H^{MS}(x)\|_E^2 \leq \|u(x) - u_H^\psi(x)\|_E^2 + CH^2 \|f(x)\|_{L^2(D)}^2. \tag{2.6}$$

The proposition says that for the same boundary conditions on  $\Gamma$ , (2.5), the multi-scale basis functions perform at least as well as other basis, if only  $O(H)$  accuracy in the energy norm is desired.

Moreover, the multi-scale basis functions  $\phi_i(x)$ ,  $i = 1, 2, \dots, n$ , are  $a(x)$ -orthogonal to the local bubble part of the solution  $u^2(x)$ , then according to the Galerkin projection formulation (1.8) and the optimal property (1.9), we have the following characterization of the numerical solution  $u_H^{MS}(x)$ ,

$$\|u^1(x) - u_H^{MS}(x)\|_E = \inf_{v(x) \in V_H} \|u^1(x) - v(x)\|_E.$$

Recall that we can construct the local bubble part of the solution  $u^2(x)$  by solving a series of local bubble problems (2.1) on each coarse element, independently from the Galerkin projection (1.8). We can then add  $u^2(x)$  to  $u_H^{MS}(x)$  to get the *corrected multi-scale solution*  $u_H(x)$ ,

$$u_H(x) = u_H^{MS}(x) + u^2(x).$$

We have the following approximation property for the corrected multi-scale solution,

$$\|u(x) - u_H(x)\|_E = \|u^1(x) - u_H^{MS}(x)\|_E = \inf_{v(x) \in V_H} \|u^1(x) - v(x)\|_E. \tag{2.7}$$

The right hand side of (2.7) only depends on  $u(x)|_\Gamma$  and the local boundary conditions of the multi-scale basis (2.5). If the multi-scale basis can exactly recover the solution on the boundaries of the coarse elements,  $\Gamma$ , the corrected multi-scale solution  $u_H(x)$  can exactly recover the fine-mesh solution.

For a specific solution  $u(x)$  to (1.1), one can choose local boundary conditions in (1.6) to make (2.7) vanish. However, these choices will depend on  $u(x)|_\Gamma$ , which is unknown in advance. For elliptic equations (1.1), the local solution depends on the elliptic coefficient and the forcing term on the whole domain. Thus to exactly recover the underlying fine-mesh solution, one needs to incorporate the global information of the elliptic equation into the local boundary conditions of the multi-scale basis functions. In this paper, we propose an iteratively adaptive approach that achieves this global-to-local information transfer through the combination of coarse-mesh solving and fine-mesh smoothing operations.

**3. An iteratively adaptive MsFEM for multi-scale elliptic PDEs**

We propose an iteratively adaptive multi-scale finite element method in this section, which iteratively updates the multi-scale basis functions based on the approximate numerical solutions in previous iteration steps. We also derive a stabilized variant of the proposed method, and it is essentially a two-level multi-grid method that employs a set of multi-scale basis for the restriction and prolongation operations.

*3.1. An iteratively adaptive MsFEM*

In this algorithm, we iteratively update the multi-scale basis functions  $\phi_i^s(x)$  and the approximate numerical solution  $u^s(x)$  simultaneously, where  $s$  is the iteration step.

Assume that at the  $s$ -th iteration step we have got a set of multi-scale basis functions and the corresponding trial space (these basis functions will be updated in each step, and initially at  $s = 0$ , we can choose them to be the multi-scale basis functions with linear local boundary conditions (1.6)),

$$V_H^s = \text{span}\{\phi_i^s(x), i = 1, \dots, n\}, \quad \phi_i^s(x_j) = \delta_{ij}, \tag{3.1}$$

where  $n$  is the total number of interior coarse-mesh node points. At the  $s$ -th step, we first solve the equation (1.1) on the coarse mesh using trial space  $V_H^s$  (3.1) and the Galerkin projection (1.8) to get  $u_H^{(s)}(x)$ ,

$$a(u_H^{(s)}(x), v_H(x)) = \langle f(x), v_H(x) \rangle, \quad \text{for all } v_H(x) \in V_H^s. \tag{3.2}$$

We call  $u_H^{(s)}(x)$  the *multi-scale numerical solution* at step  $s$ .

Then we solve the local bubble problems (2.1) to get the local bubble part of the solution  $u^2(x)$ . In our numerical implementation, we use the underlying fine mesh to numerically solve the bubble problems (2.1), and get the numerical solution  $u_h^2(x)$ . Since we have assumed that the underlying fine mesh can resolve the small scale variation of the problem, and the error in fine mesh solution can be neglected, we remove the subscript  $h$ , and do not distinguish between  $u^2(x)$  and  $u_h^2(x)$ . We add the local bubble part  $u^2(x)$  to  $u_H^{(s)}(x)$  to get  $u^{(s)}(x)$ . Recall that we can compute  $u_H^{(s)}(x)$  and  $u^2(x)$  independently because  $\phi_i^s(x)$  are multi-scale basis functions, and they are  $a(x)$ -orthogonal to the local bubble part of the solution. We obtain

$$u^{(s)}(x) = u_H^{(s)}(x) + u^2(x). \tag{3.3}$$

We call  $u^{(s)}(x)$  the *corrected multi-scale numerical solution* at step  $s$ .

These local bubble problems (2.1) need to be solved for only once in the iteration, and the obtained bubble part of the solution  $u^2(x)$  will be used to correct  $u_H^{(s)}(x)$  in each step of the iteration.

Recall that the multi-scale basis functions  $\phi_i^s(x)$  and the local bubble part of the solution  $u^2(x)$  are all constructed and stored using the piecewise linear basis functions on the underlying fine mesh. Then the approximate numerical solution  $u^{(s)}(x)$  will also be stored as linear combination of the fine mesh piecewise linear basis. We denote the interior node points of the underlying fine mesh as

$$x^1, x^2, \dots, x^i, \dots, x^N,$$

and denote the associated fine mesh piecewise linear basis functions as

$$\psi_1(x), \psi_2(x), \dots, \psi_i(x), \dots, \psi_N(x), \quad \psi_i(x^j) = \delta_{ij}. \tag{3.4}$$

Discretizing the equation (1.1) using the fine mesh basis, we get the underlying fine mesh linear system

$$A\vec{u} = \vec{f}, \tag{3.5a}$$

where  $A$  is the stiffness matrix, and  $\vec{f}$  is the load vector,

$$A(i, j) = a(\psi_i(x), \psi_j(x)), \quad \vec{f}(i) = \langle f(x), \psi_i(x) \rangle. \tag{3.5b}$$

Next, we apply  $\nu$  steps of iteration, e.g., weighted Jacobi iteration, to the underlying fine mesh linear system (3.5) using  $u^{(s)}(x)$  as initial guess to get the updated  $u^s(x)$ ,

$$u^s(x) = S^{(v)}u^{(s)}(x). \tag{3.6}$$

We call  $u^s(x)$  the approximate numerical solution at step  $s$ .

The purpose of this step is to eliminate the high frequency modes of the error in  $u^{(s)}(x)$  (3.3). Here the high-frequency modes correspond to the eigenfunctions of the elliptic operator with large eigenvalues.

Then we employ the approximate numerical solutions of previous iteration steps,  $u^k(x)$ ,  $k \leq s$ , to construct the updated multi-scale basis functions for the next iteration step,

$$\phi_i^{s+1}(x), \quad i = 1, \dots, n. \quad \phi_i^{s+1}(x_j) = \delta_{ij}. \tag{3.7}$$

Here we require that the updated multi-scale basis functions (3.7) can exactly recover  $u^s(x)|_\Gamma$ ,

$$u^s(x) - \sum_i u^s(x_i)\phi_i^{s+1}(x) = 0, \quad x \in \Gamma. \tag{3.8}$$

With the updated multi-scale basis  $\phi_i^{s+1}(x)$ , we can move to the next iteration step.

There are several choices to construct the multi-scale basis (3.7), and we describe two of them below.

1. One possible choice of the local boundary conditions for the multi-scale basis based on sampled solutions was proposed in [27] and later employed in [25,57]. On the coarse element  $D_i$  as shown in Fig. 1, we choose  $\phi_{i_1}^{s+1}(x) = 0$  on the edge  $x_{i_2}x_{i_3}$ . And on the edges  $x_{i_1}x_{i_j}$ ,  $j = 2, 3$ , we choose

$$\phi_{i_1}^{s+1}(x) = \begin{cases} \frac{u^s(x) - u^s(x_{i_j})}{u^s(x_{i_1}) - u^s(x_{i_j})}, & u^s(x_{i_1}) \neq u^s(x_{i_j}); \\ \text{linear}, & u^s(x_{i_1}) = u^s(x_{i_j}). \end{cases} \tag{3.9}$$

One can show that in the case that the values of  $u(x)$  on any two adjacent node points are not equal,

$$u^s(x) = \sum_{i=1}^n u^s(x_i)\phi_i^{s+1}(x), \quad x \in \Gamma.$$

Namely, the basis functions can exactly recover the trace of the solution on  $\Gamma$ .

2. Another choice is employing the oversampling technique [47]. In the oversampling method, one solves two local boundary value problems (1.6) on an oversampling domain  $K_i$  that encloses the coarse element  $D_i$ , and uses the linear combination of these local sampled solutions to get the multi-scale basis on  $D_i$ .

Now given two sampled solutions  $u^s(x)$  and  $u^{s-1}(x)$ , we want to build local boundary conditions for the multi-scale basis functions on the element  $D_i$ , which is shown in Fig. 1. Note that  $D_i$  has three node points  $x_{i_1}$ ,  $x_{i_2}$  and  $x_{i_3}$ , and  $\phi_{i_j}(x)$ ,  $j = 1, 2, 3$  are the multi-scale basis functions associated with the three node points. We build local boundary condition for  $\phi_{i_j}$  using the linear combination of  $u^s(x)$ ,  $u^{s-1}(x)$  and constant,

$$\bar{\phi}_{i_j}^{s+1}(x) = c_{i,j,0}^{s+1} + c_{i,j,1}^{s+1}u^s(x) + c_{i,j,2}^{s+1}u^{s-1}(x), \quad x \in \partial D_i, \tag{3.10}$$

where the coefficients  $c_{i,j,k}^{s+1}$ ,  $k = 0, 1, 2$  are determined by the condition that

$$\bar{\phi}_{i_j}^{s+1}(x_{i_l}) = \delta_{jl}. \tag{3.11}$$

In (3.10), the subscripts  $i$  and  $j$  indicate that we are constructing local boundary conditions for the  $j$ -th multi-scale basis function on element  $D_i$ ,  $\phi_{i_j}(x)$ ; the third subscript of  $c_{i,j,k}^{s+1}$  indicates that  $c_{i,j,k}^{s+1}$  is the coefficient before the constant basis function ( $k = 0$ ), the approximate solution at the current step ( $k = 1$ ), or the approximate solution at the previous step ( $k = 2$ ). The superscript  $s + 1$  indicates that we are constructing local boundary conditions for the multi-scale basis functions of the  $(s + 1)$ -th step.

The basis functions constructed in (3.10) are in general non-conforming and  $\bar{\phi}_i(x)$  may have jump across the coarse-mesh edges. This is because the local boundary conditions on a coarse mesh edge  $e$  determined from its two neighbor coarse elements may be different. We can use a simple averaging technique to obtain conforming basis. For edge  $e$ , we denote its two neighbor coarse elements as  $D_{i_1}$ ,  $D_{i_2}$ . We modify the local boundary conditions of  $\bar{\phi}_j(x)$  on  $e$ , and get the averaged basis  $\phi_j(x)$ ,

$$\phi_j(x)|_e = \frac{1}{2}(\bar{\phi}_{j,D_{i_1}}(x)|_e + \bar{\phi}_{j,D_{i_2}}(x)|_e), \tag{3.12}$$

where  $\bar{\phi}_{j,D_{i_1}}(x) = \bar{\phi}_j(x)|_{D_{i_1}}$ ,  $\bar{\phi}_{j,D_{i_2}}(x) = \bar{\phi}_j(x)|_{D_{i_2}}$ . The price we pay for this averaging process is that the resulting basis  $\phi_j(x)$  has enlarged support. This is shown in Fig. 2. The same averaging technique to make the basis functions conforming has been employed in [25,19,57].

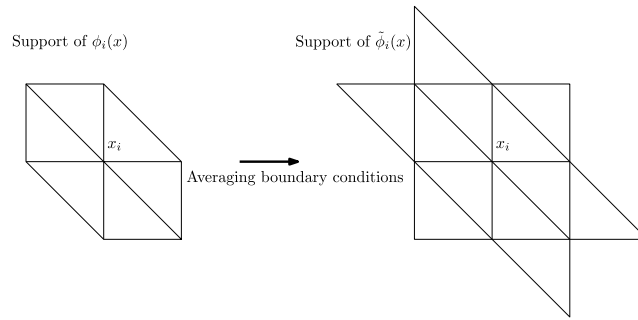


Fig. 2. Averaging on the coarse mesh edge.

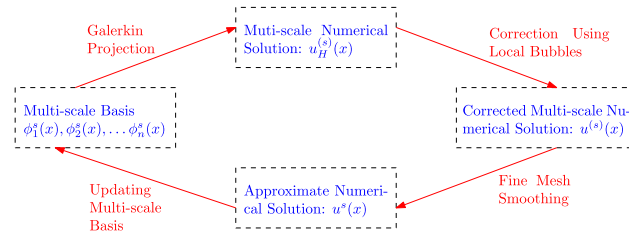


Fig. 3. Flow chart of Algorithm 1.

Note that to determine the coefficients  $c_{i,j,k}^{s+1}$  according to the condition (3.11), one needs to solve a  $3 \times 3$  linear system. If the system is singular, we choose linear local boundary conditions for the multi-scale basis functions on element  $D_i$ . If the linear system is not singular, one can show that the resulting multi-scale basis functions (3.12) can recover  $u^s(x)$  on  $\Gamma = \cup \partial D_i$ . Convergence analysis of Algorithm 1 with the choice of updated multi-scale basis (3.12) will be carried out in section 4.

Iterative implementation of the above coarse mesh solving (3.2) and fine mesh smoothing (3.6) operations leads to the proposed iteratively adaptive multi-scale finite element method, Algorithm 1.

A flow chart of Algorithm 1 is depicted in Fig. 3.

---

**Algorithm 1** Iteratively adaptive multi-scale finite element algorithm.

---

- 1: **for**  $s = 1$  to  $s_{\max}$  **do**
  - 2:   **for**  $i = 1$  to  $n$  **do**
  - 3:     Construct the local boundary conditions for the multi-scale basis  $\phi_i^s(x)$  based on the approximate solutions of the previous steps,  $u^k(x), k \leq s - 1$ . For example, employing formula (3.9) or (3.12).
  - 4:     Construct the multi-scale basis  $\phi_i^s(x)$  by solving local boundary value problems (1.6).
  - 5:   **end for**
  - 6:   Employ  $V_H^s$  (3.1) and the Galerkin projection to solve (1.1) and get  $u_H^{(s)}(x)$ .
  - 7:   Correct the coarse-mesh solution using the local bubble part  $u^2(x)$ ,
 
$$u^{(s)}(x) = u_H^{(s)}(x) + u^2(x).$$
  - 8:   Solve the fine mesh linear system  $A\tilde{u} = \tilde{f}$  using an iteration method (e.g. Jacobi iteration) for  $\nu$  times with initial guess  $u^{(s)}(x)$  to get  $u^s(x)$ ,  $u^s(x) = S^{(\nu)}u^{(s)}(x)$ .
  - 9:   Check the user-defined stopping criteria.
  - 10: **end for**
- 

**Remark 1.** There are similar multi-scale methods in the literature, for example [25,57], which iteratively update the local boundary conditions of the multi-scale basis functions. In those works, the adaptive multi-scale basis functions are constructed using the oversampling technique. And the local boundary conditions of basis functions on the oversampling domain are constructed based on the approximate numerical solution of the previous iteration step,  $u^s(x)$ , and the rule (3.9). These works are different from our proposed method in the following aspects. In [25,57], the multi-scale basis functions cannot recover  $u^{s-1}(x)$  on the boundaries of coarse elements,  $\Gamma$ , because the local bubble part of  $u^{s-1}(x)$  on  $K_i$  is not taken into account in the boundary conditions of the basis functions on  $\partial D_i$ . Thus these methods [25,57] cannot obtain the underlying fine mesh solution even the iteration converges. Besides, there are no explicit smoothing operations in the previous methods [25,57], and the transfer of the global information of the equation to the local boundary conditions of the multi-scale basis is achieved through oversampling and coarse-mesh solving.



We will prove in section 4 under certain assumptions that the approximate numerical solutions  $u^s(x)$  in Algorithm 1 converge to the underlying fine mesh solution exponentially fast, which means that the iteratively adaptive approach proposed in Algorithm 1 can effectively incorporate the global information of the equation into the local boundary conditions of the multi-scale basis (2.5).

Before we end this subsection, we make the following comments on the proposed Algorithm 1.

1. In the construction of the updated multi-scale basis functions, for certain  $u^s(x)$ , there may not exist nodal basis functions  $\phi_i^{s+1}(x)$ ,  $i = 1, \dots, n$ , which can recover the solution on  $\Gamma$  in (3.8). An extreme case is that  $u^s(x_i) = 0$ ,  $i = 1, 2, \dots, n$ , but  $u^s(x)$  is not a zero function.
2. In each step of the iteration, one needs to construct the updated multi-scale basis functions by solving local boundary value problems on the fine mesh and assembling the updated stiffness matrix.
3. In constructing the local boundary conditions, one may encounter dividing by a small number in (3.9) or solving a nearly singular system in (3.11), thus suffer from some instability issue.

A stabilized variant of the proposed method, Algorithm 2, is derived in the next subsection, which introduces an adaptive correction term and keeps the multi-scale basis fixed throughout the iteration. Algorithm 2 is less expensive in each iteration step and avoids the potential instability issue in Algorithm 1.

### 3.2. A stabilized variant of the proposed algorithm

In Algorithm 1, the multi-scale basis functions  $\phi_i^s(x)$  are constructed according to the approximate numerical solutions in the previous iteration steps,  $u^k(x)$ ,  $k < s$ . Then we employ  $\phi_i^s(x)$  and a Galerkin projection to get the approximate numerical solution  $u^{(s)}(x)$  (3.3). We consider simplifying this step by using a set of multi-scale basis functions constructed before the iteration, which we denote as

$$\phi_i(x), \quad i = 1, 2, \dots, n, \quad \phi_i(x_j) = \delta_{ij}. \tag{3.13}$$

And we denote the space spanned by this multi-scale basis as

$$V_H = \text{span}\{\phi_1(x), \phi_2(x), \dots, \phi_n(x)\}. \tag{3.14}$$

We have the following decomposition of  $u^{s-1}(x)$  and  $u^{(s)}(x)$  using  $\phi_i(x)$ ,

$$u^{s-1}(x) = \sum_i u^{s-1}(x_i)\phi_i(x) + R^{s-1}(x), \tag{3.15a}$$

$$u^{(s)}(x) = \sum_i u^{(s)}(x_i)\phi_i(x) + R^{(s)}(x), \tag{3.15b}$$

where  $\sum_i u^{s-1}(x_i)\phi_i(x)$  and  $\sum_i u^{(s)}(x_i)\phi_i(x)$  are simply the interpolation of  $u^{s-1}(x)$  and  $u^{(s)}(x)$  on coarse mesh node points  $x_i$  using multi-scale basis functions  $\phi_i(x)$ ,

$$\phi_i(x_j) = \delta_{ij},$$

and thus  $R^{s-1}(x)$  and  $R^{(s)}(x)$  are the corresponding interpolation residuals.

Since both  $u^{s-1}(x)$  and  $u^{(s)}(x)$  approximate  $u(x)$ , we assume that the interpolation residuals  $R^{s-1}(x)$  and  $R^{(s)}(x)$  are close,  $R^{s-1}(x) \approx R^{(s)}(x)$ . Then  $u^{(s)}(x)$  can be written as

$$u^{(s)}(x) \approx \sum_i u^{(s)}(x_i)\phi_i(x) + u^{s-1}(x) - \sum_i u^{s-1}(x_i)\phi_i(x) = \sum_i c_i\phi_i(x) + u^{s-1}(x). \tag{3.16}$$

Based on (3.16), we construct an approximation of  $u^{(s)}(x)$  by solving the following optimization problem,

$$\min_{c_i} \|u(x) - \sum_i c_i\phi_i(x) - u^{s-1}(x)\|_E.$$

The resulting  $e_H^s(x) = \sum_{i=1}^n c_i\phi_i(x)$  is the Galerkin finite element solution of equation

$$-\text{div}(a(x)\nabla e(x)) = f(x) + \text{div}(a(x)\nabla u^{s-1}(x)), \quad e(x)|_{\partial D} = 0, \tag{3.17}$$

using the multi-scale basis functions  $\phi_i(x)$ ,  $i = 1, \dots, n$ . And we get an approximation of  $u^{(s)}(x)$ ,

$$u^{(s)}(x) \approx e_H^s(x) + u^{s-1}(x). \tag{3.18}$$

To reduce the error in the bubble part of  $u^{(s)}(x)$  (3.18), we construct the bubble part of the solution to (3.17) by solving a series of local bubble problems as in (2.1). We denote the bubble part of the solution to (3.17) as  $e^{s,2}(x)$ . We use  $e^{s,2}(x)$  to correct the approximate numerical solution in (3.18) and obtain

---

**Algorithm 2** Iteratively adaptive multi-scale finite element method.

---

- 1: Construct multi-scale basis functions  $\phi_i(x)$ ,  $i = 1, \dots, n$ .  
 Let  $u^0(x) = 0$  be the initial guess of the solution.
  - 2: **for**  $s = 1$  to  $s_{\max}$  **do**
  - 3: Solve equation (3.17) using multi-scale basis functions  $\phi_i(x)$  to get  $e_H^s(x)$
  - 4: Solve the local bubble part of the solution to (3.17),  $e^{s,2}(x)$ .
  - 5: Obtain the approximate numerical solution
 
$$u^{(s)}(x) = u^{s-1}(x) + e_H^s(x) + e^{s,2}(x).$$
  - 6: Solve the fine mesh linear system  $A\bar{u} = \bar{f}$  using iteration method (e.g. Jacobi iteration) for  $\nu$  times with initial guess  $u^{(s)}(x)$  to get  $u^s(x)$ .  $u^s(x) = S^{(\nu)}u^{(s)}(x)$ .
  - 7: Check the user-defined stopping criteria.
  - 8: **end for**
- 

$$u^{(s)}(x) \approx u^{s-1}(x) + e_H^s(x) + e^{s,2}(x). \tag{3.19}$$

Replacing the construction of  $u^{(s)}(x)$  in Algorithm 1 by the approximation (3.19), we get Algorithm 2. The interpolation residual  $R^{s-1}(x)$  is introduced to motivate Algorithm 2 in (3.16), and in numerical implementation, we only need to solve equation (3.17) to get the approximate  $u^{(s)}(x)$  (3.19).

Because there is no need to construct updated multi-scale basis in each iteration step according to (3.9) or (3.10), the potential instability issue in the rules (3.9) and (3.11) is avoided, and this is the reason that we call Algorithm 2 a stabilized variant of Algorithm 1. Moreover, the simplification procedure (3.19) also brings in computational savings in each iteration step of the Algorithm, since we do not need to construct the updated multi-scaled basis and the updated stiffness matrix in each iteration step.

The choice of multi-scale basis  $\phi_i(x)$  (3.13) determines the convergence of Algorithm 2, and we will come back to the issue about the choice of the multi-scale basis in the convergence analysis in section 4.

**Remark 2.** The method in [41] is similar to our proposed Algorithm 2 but is very different from Algorithm 1. The method in [41] uses an adaptive correction function to correct the coarse-mesh multi-scale solution, which is similar to the interpolation residual that we introduced in (3.15). Algorithm 2 is different from the method in [41] in the following minor aspects. In [41], the finite element volume formulation is employed to construct the coarse-mesh linear system, and a set of dual coarse grids in addition to the primary coarse grids are required in the discretization. While in Algorithm 2, we use the Galerkin projection. In [41], the correction term is explicitly constructed in each step of the iteration by solving a series of local boundary value problems with boundary conditions obtained from solving local 1D problems. While in Algorithm 2, the interpolation residual (3.15) is only introduced to motivate the Algorithm, and there is no need to construct the residual explicitly in implementation. Thus the method in [41] has an additional step than Algorithm 2.

**4. Convergence of the proposed algorithms**

In this section, we carry out the convergence analysis of the proposed Algorithms 1 and 2. We first show that Algorithm 2 is a two-level multi-grid method that employs a set of multi-scale basis for the restriction and prolongation operations. Then by comparing with Algorithm 2, we show that Algorithm 1 is a nonlinear version of the two-level multi-grid method with the restriction and prolongation operators adapted to the approximate solutions of the previous iteration steps. To establish the convergence of proposed Algorithm 1, one needs certain approximation property of the adaptive multi-scale basis to the solution space of (1.1) with  $L^2(D)$  forcing  $f(x)$ , and this is achieved by employing the harmonic coordinates.

*4.1. Reformulation of the two algorithms as two-level multi-grid methods*

Recall that we denote the piecewise linear basis functions on the underlying fine mesh (3.4) as

$$\psi_k(x), \quad k = 1, 2, \dots, N, \quad \psi_k(x^l) = \delta_{kl}, \tag{4.1}$$

where  $x^l$  are the node points of the fine mesh. Since the fine mesh is a refinement of the coarse mesh with node points  $x_i$ ,  $i = 1, \dots, n$ , we define  $l_i$  as the index of coarse mesh node point  $x_i$  on the fine mesh,

$$x_i = x^{l_i}.$$

The multi-scale basis functions  $\phi_i^s(x)$  in Algorithm 1 (3.1) and  $\phi_i(x)$  (3.13) in Algorithm 2 are all represented as linear combinations of  $\psi_k(x)$ . We denote these linear combinations as

$$\phi_i^s(x) = \sum_{k=1}^N D_{ki}^s \psi_k(x), \quad \phi_i(x) = \sum_{k=1}^N D_{ki} \psi_k(x),$$

then  $D^s$  and  $D$  are the matrix representations of the multi-scale basis functions using  $\psi_k(x)$  in (4.1).

Consider fine mesh basis functions  $\psi_l(x)$  in (4.1) that satisfy

$$\psi_l(x)|_\Gamma = 0, \tag{4.2}$$

where  $\Gamma$  is defined as the union of the element boundaries in (2.2),  $\Gamma = \cup \partial D_i$ .

We denote the part of fine mesh basis satisfying the above condition (4.2) as

$$\phi_j^b(x), \quad j = 1, \dots, N_b,$$

and refer to them as *bubble basis functions*.

The bubble basis functions are the fine mesh basis functions that we employ to solve the bubble part of solution in (2.1). We denote  $k_j$  as the index of  $\phi_j^b(x)$  in the fine mesh basis functions (4.1), namely,

$$\phi_j^b(x) = \psi_{k_j}(x), \quad \psi_{k_j}(x)|_\Gamma = 0. \tag{4.3}$$

Then we denote  $D^b$  as the matrix representation of  $\phi_j^b(x)$  using  $\psi_k(x)$ , and have

$$D^b(i, j) = \delta_{i, k_j}.$$

We denote  $V_H^b$  as the space spanned by these bubble basis functions

$$V_H^b = \text{span}\{\phi_1^b, \phi_2^b, \dots, \phi_{N_b}^b\}. \tag{4.4}$$

When we solve equation (1.1) in Algorithm 1 or (3.17) in Algorithm 2 using the multi-scale basis (3.1) or (3.13), and add the corresponding local bubble parts back to the numerical solutions, we are essentially solving the equations using the multi-scale basis functions enriched with the bubble basis (4.3)

$$\tilde{V}_H^s = V_H^s \oplus V_H^b, \quad \tilde{V}_H = V_H \oplus V_H^b. \tag{4.5}$$

Here  $V_H^s$  is the adaptive multi-scale trial space in Algorithm 1 defined in (3.1);  $V_H$  is the multi-scale trial space in Algorithm 2 defined in (3.14);  $V_H^b$  is the bubble part of the solution space defined in (4.4). This procedure can be separated into two steps, i.e., the coarse-mesh solving using  $V_H$  and local bubble correction using  $V_H^b$ , because the bubble basis is  $a(x)$ -orthogonal to the multi-scale basis.

We denote  $\tilde{D}$  and  $\tilde{D}^s$  as

$$\tilde{D} = [D, D^b], \quad \tilde{D}^s = [D^s, D^b],$$

which are the matrix representations of the enriched basis in (4.5).

Recall that we denote the underlying fine-mesh linear system for equation (1.1) as

$$A\vec{u} = \vec{f}, \quad A(i, j) = a(\psi_i(x), \psi_j(x)), \quad \vec{f}(i) = \langle f(x), \psi_i(x) \rangle. \tag{4.6}$$

With the above notations, we derive the matrix representation of the iteration relation in Algorithm 1 and Algorithm 2. We consider Algorithm 2 first, which is a linear two-level multi-grid method.

Let  $\vec{u}^s$  be the fine-mesh discrete solution in the  $s$ -th step, namely,

$$u^s(x) = \sum_{i=1}^N \vec{u}^s(i) \psi_i(x).$$

In Algorithm 2, we employ the enriched trial space  $\tilde{V}_H$  in (4.5) to solve (3.17) and get

$$e_H^s(x) + e^{s,2}(x).$$

We first compute the corresponding load vector  $\langle \phi_i(x), f(x) + \text{div}(a(x)\nabla u^s(x)) \rangle$ , which is

$$\tilde{D}^T(\vec{f} - A\vec{u}^s). \tag{4.7}$$

The corresponding stiffness matrix employing the trial space (4.5) is

$$\tilde{D}^T A \tilde{D}, \tag{4.8}$$

and we can get the coarse-mesh solution based on (4.8) and (4.7),

$$(\tilde{D}^T A \tilde{D})^{-1} \tilde{D}^T(\vec{f} - L_h \vec{u}^s). \tag{4.9}$$

Here we have used  $L_h$  as the fine mesh stiffness matrix defined in (4.6),  $L_h = A$ .

Next we represent the coarse-mesh solution (4.9) using the fine mesh basis  $\psi_k(x)$ , we get

$$\tilde{e}^s + \tilde{e}^{s,2} = \tilde{D}(\tilde{D}^T A \tilde{D})^{-1} \tilde{D}^T (\tilde{f} - L_h \tilde{u}^s). \tag{4.10}$$

We denote  $L_H^{-1}$  as the coarse-mesh solution operator,

$$L_H^{-1} = \tilde{D}(\tilde{D}^T A \tilde{D})^{-1} \tilde{D}^T, \tag{4.11}$$

then we have  $\tilde{e}^s + \tilde{e}^{s,2} = L_H^{-1}(\tilde{f} - L_h \tilde{u}^s)$  in (4.10). We add it back to  $\tilde{u}^s$  in (3.19), and get

$$\tilde{u}^{(s)} = \tilde{u}^s + L_H^{-1}(\tilde{f} - L_h \tilde{u}^s), \tag{4.12}$$

where  $\tilde{u}^{(s)}$  is the vector representation of the corrected multi-scale solution (3.19) using the fine mesh basis. Finally, we apply  $\nu$  steps of smoothing operations (3.6) to the approximate solution (4.12). Let  $S_M$  be the iteration matrix in the smoothing operation, then we get the following iteration relation for Algorithm 2,

$$\tilde{u}^{s+1} = S_M^\nu (I - L_H^{-1} L_h) \tilde{u}^s + \tilde{M}, \tag{4.13}$$

where  $\tilde{M}$  is a constant vector that does not depend on  $\tilde{u}^s$ ,

$$\tilde{M} = S_M^\nu L_H^{-1} \tilde{f} + (I - S_M^\nu) L_h^{-1} \tilde{f}.$$

We can see that Algorithm 2 is a two-level multi-grid method employing the multi-scale basis (3.13) and the bubble basis (4.3) for the restriction and prolongation operations, which are  $\tilde{D}^T$  and  $\tilde{D}$  in (4.11).

Next we consider the proposed Algorithm 1. The only difference between Algorithm 1 and 2 is that in Algorithm 1, the multi-scale basis in the coarse-mesh solution operator  $L_H^{-1}$  is adapted to the approximate numerical solutions in previous iteration steps. To show this we consider replacing the multi-scale basis in the coarse-mesh correction (3.17) by  $\phi_i^s(x)$ , which are obtained according to (3.9) or (3.12).

Consider solving (3.17) using trial space  $\tilde{V}_H^s$  (4.5) and a Galerkin projection to get  $e_H^s(x) + e^{s,2}(x)$ . Then

$$a(u(x) - u^{s-1}(x) - e_H^s(x) - e^{s,2}(x), v(x)) = 0, \quad \text{for } v(x) \in V_H^s. \tag{4.14}$$

Recall that in our construction of the adaptive multi-scale basis  $\phi_i^s(x)$ , we require that they can recover the approximate solution  $u^{s-1}(x)$  on  $\Gamma$  (3.8). Therefore we have  $u^{s-1}(x) \in \tilde{V}_H^s$ , and

$$u^{(s)}(x) = u^{s-1}(x) + e_H^s(x) + e^{s,2}(x) \in \tilde{V}_H^s. \tag{4.15}$$

The representation of  $u^{(s)}(x)$  (4.15) and the orthogonality condition (4.14) together imply that  $u^{(s)}(x)$  is just the Galerkin numerical solution to the original equation (1.1) employing the updated trial space  $\tilde{V}_H^s$ , which is the same as the approximate numerical solution  $u^{(s)}(x)$  that we obtain in Algorithm 1, (3.3). Namely, if we replace the coarse mesh solution operator  $L_H^{-1}$  in Algorithm 2 using  $(L_H^s)^{-1}$ , we get Algorithm 1. So we conclude that Algorithm 1 is a nonlinear version of the two-level multi-grid method with the restriction and prolongation operators adapted to approximate solutions of the previous iteration steps.

Note that there is no explicit restriction operation in the proposed Algorithm 1, namely, we do not need to compute the fine mesh residual and pass it to coarse mesh as in (4.7). The reason that we can skip the restriction operation in Algorithm 1 is that we have required that the multi-scale basis  $\phi_i^s(x)$  can recover the approximate numerical solution of the previous iteration step,  $u^{s-1}(x)$ , on  $\Gamma$ .

Similar to (4.13) in Algorithm 2, we get the following iteration relation for Algorithm 1,

$$\tilde{u}^s = S_M^\nu (I - (L_H^s)^{-1} L_h) \tilde{u}^{s-1} + \tilde{M}_s. \tag{4.16}$$

Multi-grid method is known as the fastest elliptic solver, but the classical multi-grid methods that use piecewise linear basis functions on the coarse mesh for the interpolation and prolongation operations may fail for problems (1.1) with rough coefficient due to the lack of regularity of the solution. There are many multi-scale methods in the literature that employ problem-dependent interpolation operations to recover the optimal efficiency of the multi-grid. Among these methods, the Algebraic Multi-grid Methods (AMG) have achieved some success. However, most of these AMG methods are based on heuristics, and there exist no convergence analysis. The AMG methods introduce the notion of “strength of connection” for the node points, and build the interpolation basis automatically based on the “strength matrix”. For our proposed methods, the interpolation basis functions are constructed based on sampled solutions of the elliptic equations, and there is a clear underlying geometric picture. In the next subsection, we carry out the convergence analysis for our proposed algorithms using the theory of Harmonic coordinates.

#### 4.2. Estimates of the error decay in the two proposed algorithms

The exact fine-mesh solution  $\bar{u} = A^{-1}\bar{f}$  is a fixed point of the iteration schemes (4.16) and (4.13). To unify the notation for the two algorithms, we will write  $L_H^{-1}$  in (4.13) as  $(L_H^s)^{-1}$ .

We have the following relation between the errors in two consecutive iteration steps,

$$\bar{u}^{s+1} - \bar{u} = S_M^\nu (I - (L_H^s)^{-1}L_h)(\bar{u}^s - \bar{u}). \tag{4.17}$$

We will provide convergence analysis of the proposed algorithms under the framework of the two-level multi-grid method, see [39], in which the iteration relation (4.17) is decomposed as (with  $A = L_h$ )

$$A(\bar{u}^{s+1} - \bar{u}) = (AS_M^\nu)[L_h^{-1} - (L_H^s)^{-1}][A(\bar{u}^s - \bar{u})]. \tag{4.18}$$

Then we have the following estimate on the decay of the  $l^2$  norm of the residual,

$$\|A(\bar{u}^{s+1} - \bar{u})\| \leq \|AS_M^\nu\| \|L_h^{-1} - (L_H^s)^{-1}\| \|A(\bar{u}^s - \bar{u})\|.$$

In this section, we use  $\|\cdot\|$  to denote the discrete  $l^2$  norm and the induced operator norm. We need

$$\|AS_M^\nu\| \|L_h^{-1} - (L_H^s)^{-1}\| < \rho < 1 \tag{4.19}$$

for the robust convergence of the iteration scheme (4.18). Classical convergence analysis of the two-level multi-grid methods estimates the two parts in (4.19) separately.

For the first part, one needs the following smoothing property,

$$\|AS_M^\nu\| = O\left(\frac{1}{\nu}\right), \quad \nu \rightarrow +\infty, \tag{4.20}$$

which requires that the smoothing operation can effectively damp the high frequency modes in the error.

For the second part in (4.19), classical convergence analysis requires the approximation property

$$\|L_h^{-1} - (L_H^s)^{-1}\| \leq CH^2h^{-2}. \tag{4.21}$$

We have assumed that the underlying fine mesh is of size  $O(h)$  and has bounded aspect ratio, then in the computation of the load vector (4.6), we have

$$\|\bar{f}\| = O(h\|f(x)\|_{L^2(D)}). \tag{4.22}$$

And for the error in the solution  $u(x) - u^s(x)$ , we have

$$\|u(x) - u^s(x)\|_{L^2(D)} = O(h\|\bar{u} - \bar{u}^s\|), \tag{4.23}$$

where  $\bar{u}$  and  $\bar{u}^s$  are the vector representation of  $u(x)$  and  $u^s(x)$  using the fine mesh basis functions  $\psi_k(x)$  (4.1). With (4.22) and (4.23), we can see that (4.21) requires  $O(H^2)$  accuracy of the Galerkin solution (1.8),

$$\|u(x) - u_H^{MS}(x)\|_{L^2(D)} \leq CH^2\|f(x)\|_{L^2(D)}. \tag{4.24}$$

To obtain (4.21) from (4.24), we use (4.22), (4.23), and have

$$\|(L_h^{-1} - (L_H^s)^{-1})\bar{f}\| = \|\bar{u} - \bar{u}_H^{MS}\| \leq h^{-1}\|u(x) - u_H^{MS}(x)\|_{L^2(D)} \leq Ch^{-1}H^2\|f(x)\|_{L^2(D)} \leq Ch^{-2}H^2\|\bar{f}\|.$$

We have used (4.23) in the first estimate, (4.24) in the second estimate, and (4.22) in the last estimate.

For the multi-scale PDE (1.1) with rough coefficient  $a(x)$ , the desired smoothing property (4.20) still holds for several types of smoothing operators, which only depends on the uniform ellipticity of the equation (1.1), see [39] in Chapter 6. However, due to the lack of regularity of the solution, the approximation property (4.21) fails if piecewise linear basis functions are used in the coarse-mesh solving.

To recover the robust convergence (4.19) as for smooth coefficient problems, one needs the approximation property of the multi-scale basis functions  $\phi_i^s(x)$  and  $\phi_i(x)$  to the solution space of the equation, (4.24).

1. In Algorithm 2, if we have constructed a set of multi-scale basis functions  $\phi_i(x)$  that can achieve the desired approximation property (4.24) to the solution space of (1.1), then there exists  $\nu^*$  that depends on  $H/h$ , such that for  $\nu > \nu^*$ , one can obtain the robust convergence (4.19) for Algorithm 2.

In the periodic homogenization setting, it was shown in [47] that the multi-scale basis functions (1.6) with linear local boundary conditions have good approximation property to the solution space of (1.1). We will see in the numerical results section that even for some rough coefficient problems without clear scale-separation, Algorithm 2 employing multi-scale basis functions  $\phi_i(x)$  with linear local boundary conditions also performs significantly better than the classical two-level multi-grid method.

2. In Algorithm 1, the multi-scale basis functions  $\phi_i^S(x)$  are constructed based on the approximate solutions in previous iteration steps, which can be viewed as sampled solutions to (1.1). Success of multi-scale methods that employ sampled solutions to construct local multi-scale basis can be explained using the harmonic coordinates. We will prove the approximation property (4.24) for multi-scale basis constructed according to (3.12) under certain assumptions of the harmonic coordinates in the next subsection.

One key idea of Algorithm 1 is using approximate solutions of previous iteration steps as sampled solutions to construct the multi-scale basis, instead of generating sampled solutions by solving (1.1) with random forcing terms. The latter has the same computational complexity as the original problem.

3. Once we have generated multi-scale basis functions in Algorithm 1 that can achieve good approximation property to the solution space of the equation, (4.24), we can then switch to Algorithm 2 with this set of multi-scale basis, since Algorithm 2 is computationally more efficient for each iteration step.

**Remark 3.** The adaptive multi-scale method in [41] can be directly put into the two-level multi-grid method framework as (4.13). The multi-scale methods in [25,57] cannot obtain the fine mesh solution, thus these methods are not fine mesh solvers of the equation as two-level multi-grid method. However, the oversampling method employed there is similar to blockwise smoothing operations. With suitable modifications, these multi-scale methods may be analyzed as nonlinear two-level multi-grid methods as (4.16).

4.3. Approximation property of the multi-scale basis  $\phi_i^S(x)$

We employ the harmonic coordinates to establish the approximation property of multi-scale basis functions constructed from (3.12). The  $a(x)$ -harmonic coordinates

$$F(x) = (F_1(x), F_2(x)), \tag{4.25}$$

are defined as the solutions to the following equations,

$$\begin{cases} -\operatorname{div}(a(x)\nabla F_i(x)) = 0; \\ F_i(x)|_{\partial D} = x_i. \end{cases} \tag{4.26}$$

A remarkable property proved in [62] is that under certain assumptions about  $F(x)$ , the solution to (1.1),  $u(x)$ , gains an order of regularity with respect to the harmonic coordinates,

$$\|u \circ F^{-1}(x)\|_{W^{2,p}(D)} \leq C(p)\|f \circ F^{-1}(x)\|_{L^p(D)}, \quad p > 2. \tag{4.27}$$

We consider two sampled solutions of the equation (1.1),  $v_1(x)$  and  $v_2(x)$ ,

$$-\operatorname{div}(a(x)\nabla v_1(x)) = g_1(x), \quad -\operatorname{div}(a(x)\nabla v_2(x)) = g_2(x), \quad v_{1,2}(x)|_{\partial D} = 0, \tag{4.28}$$

where  $v_1(x)$  and  $v_2(x)$  are normalized based on

$$\|g_1(x)\|_{L^p(D)} = 1, \quad \|g_2(x)\|_{L^p(D)} = 1. \tag{4.29}$$

We build multi-scale basis based on  $v_1(x)$ ,  $v_2(x)$  following the procedure outlined in (3.10) and (3.12), and show the approximation property (4.24) of the resulting multi-scale basis  $\phi_i(x)$ ,  $i = 1, \dots, n$ .

These basis functions are approximately local linear combinations of  $v_1(x)$  and  $v_2(x)$ , (3.10). Our basic idea is the following: we consider approximating the solution  $u(x)$  using the multi-scale basis  $\phi_i(x)$  in the harmonic coordinates; on each  $\tilde{D}_i = F(D_i)$ ,  $\tilde{v}_1(x) = v_1 \circ F^{-1}(x)$  and  $\tilde{v}_2(x) = v_2 \circ F^{-1}(x)$  has  $W^{2,p}$  regularity, thus can be approximated by linear functions; the multi-scale basis functions in the harmonic coordinates, namely,  $\tilde{\phi}_i(x) = \phi_i \circ F^{-1}(x)$ , are approximately linear combinations of  $\tilde{v}_1(x)$  and  $\tilde{v}_2(x)$ , thus are approximately local linear functions;  $\tilde{u}(x) = u \circ F^{-1}(x)$  has  $W^{2,p}$  regularity and can be approximated by the constructed  $\tilde{\phi}_i(x)$ . Then we pass the estimates in the harmonic coordinates to the physical space to get (4.24). To facilitate the error estimation procedure, we make the following two relatively strong assumptions on the stability of the harmonic coordinates (4.25) and the sampled solutions (4.28).

1. The solution in the harmonic coordinates,  $u \circ F^{-1}(x)$ , satisfies a non-divergence form elliptic equation, and to obtain the regularity result (4.27), one needs the Cordes condition [55]. This requires estimates of  $\nabla F(x)$ . And to pass the estimates in the harmonic coordinates to the physical space, one also needs estimates of  $\nabla F(x)$ . Here for simplicity, we make a strong assumption that

$$\nabla F(x), \quad (\nabla F(x))^{-1} \in L^\infty(D). \tag{4.30}$$

For each coarse element  $D_i$  with node points  $x_{i_j}$ ,  $j = 1, 2, 3$ , we denote the triangle of  $\tilde{x}_{i_j} = F(x_{i_j})$  as  $\Delta\tilde{x}_{i_1}\tilde{x}_{i_2}\tilde{x}_{i_3}$ . To obtain the desired estimates in the harmonic coordinates, we assume that

$$\Delta\tilde{x}_{i_1}\tilde{x}_{i_2}\tilde{x}_{i_3} \text{ has bounded aspect ratio.} \tag{4.31}$$

2. On each coarse element  $D_i$ , the interpolation of  $u(x)$  using the basis  $\bar{\phi}_{ij}(x)$  can be written as

$$c_i^0 + c_i^1 v_1(x) + c_i^2 v_2(x). \tag{4.32}$$

To obtain the estimates in the harmonic coordinates, we need the stability of the interpolation and some estimates on  $c_i^j$ . We denote the derivatives of the linear interpolation of  $u \circ F^{-1}(x)$ ,  $v_1 \circ F^{-1}(x)$  and  $v_2 \circ F^{-1}(x)$  on  $\bar{D}_i$  as  $k_i^u$ ,  $k_i^{v_1}$  and  $k_i^{v_2}$  respectively.

Then the coefficients  $c_i^j$  in (4.32) can be obtained as

$$(c_i^1, c_i^2)^T = [k_i^{v_1}, k_i^{v_2}]^{-1} k_i^u. \tag{4.33}$$

For the stability of the interpolation, we assume that

$$C_i = \|[k_i^{v_1}, k_i^{v_2}]^{-1}\| \leq C. \tag{4.34}$$

Recall that to find the coefficients in the local multi-scale basis (3.10), one needs to solve a  $3 \times 3$  linear system based on condition (3.11), thus requires that the linear system is non-singular. The above assumption (4.34) can be viewed as quantitative version of that requirement.

**Remark 4.** For arbitrarily rough coefficient  $a(x)$ , one can only get that  $F(x)$  is Hölder continuous and cannot get (4.30). However, under some integrability condition of  $\nabla F(x)$ , the same estimate procedure in this subsection still carries through, but we can only get weaker approximation property than (4.24).

With the major assumptions made above (4.30), (4.31), (4.34), and some other technical assumptions (A.5), (A.13), (A.14) made in the appendix, we can show that using the multi-scale basis (3.10) obtained from  $v_1(x)$ ,  $v_2(x)$ , the Galerkin numerical solution  $u_H^{MS}(x)$  satisfies

$$\|u(x) - u_H^{MS}(x)\|_{L^2(D)} \leq CH^2 \|f(x)\|_{L^2(D)}. \tag{4.35}$$

The proof of the above approximate property is technical and given in the appendix. If the approximate numerical solutions  $u^s(x)$  and  $u^{s-1}(x)$  after rotation and normalization (4.29) satisfy the same stability condition as  $v_1(x)$ ,  $v_2(x)$  in (4.34), then using the basis  $\phi_i^s(x)$  constructed following the procedure outlined in (3.10) and (3.12) we can recover the desired approximation property (4.21) as for smooth coefficient problems. Following the decomposition (4.19) and using (4.21) and (4.20), we can conclude that in Algorithm 1, there exists  $\nu^*$  that depends on  $H/h$ , such that for  $\nu > \nu^*$ , we can achieve the convergence (4.19),

$$\|A\bar{u}^s - \bar{f}\| \leq \rho^s \|A\bar{u}^0 - \bar{f}\|, \quad \rho < 1. \tag{4.36}$$

If there exist a set of multi-scale basis  $\phi_1(x), \dots, \phi_n(x)$ , which has the desired approximation property (4.24) to the solution space of (1.1). Then employing this set of multi-scale basis in Algorithm 2, we can also get the robust convergence result for Algorithm 2 as in (4.36),

$$\|A\bar{u}^s - \bar{f}\| \leq \rho^s \|A\bar{u}^0 - \bar{f}\|, \quad \rho < 1.$$

**Remark 5.** Recall that  $C_i = \|[k_i^{v_1}, k_i^{v_2}]^{-1}\|^{-1}$ .  $C_i$  will be large if (1)  $k_i^{v_1}$  and  $k_i^{v_2}$  are close to being align to each other; or (2)  $k_i^{v_j}$  is small. To avoid the first case, we want  $v_1(x)$  and  $v_2(x)$  to be independent sampled solutions to (1.1). To avoid the second case, we want  $\|\nabla v_j(x)\|$  to be large.

**Remark 6.** Each iteration step in Algorithm 2 has the same computational cost as the classical two-level multi-grid method. Algorithm 1 is more expensive than Algorithm 2 since one needs to construct the adaptive multi-scale basis functions and the corresponding stiffness matrix in each iteration step.

### 5. Numerical results

In this section, to demonstrate the efficiency of the proposed iteratively adaptive multi-scale finite element method, we apply it to several problems with multi-scale features, including a rough coefficient problem, a high-contrast interface problem, and a convection-dominated diffusion equation. In particular, we will show its advantage over the classical two-level multi-grid method as fine mesh solvers.

We employ a two-level discretization of the domain  $D = [0, 1] \times [0, 1]$  as shown in Fig. 1. The coarse mesh size is chosen to be  $H = \frac{1}{32}$  and the underlying fine mesh size is chosen to be  $h = \frac{1}{1024}$ .

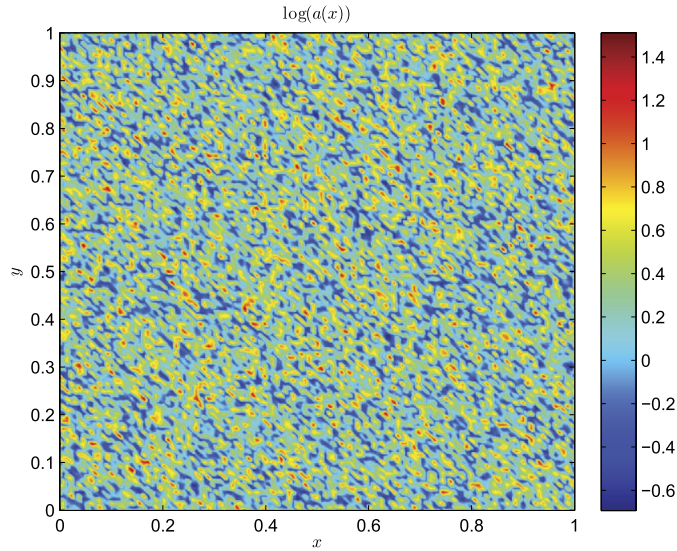


Fig. 4. Rough coefficient  $a(x, y)$  without scale-separation.

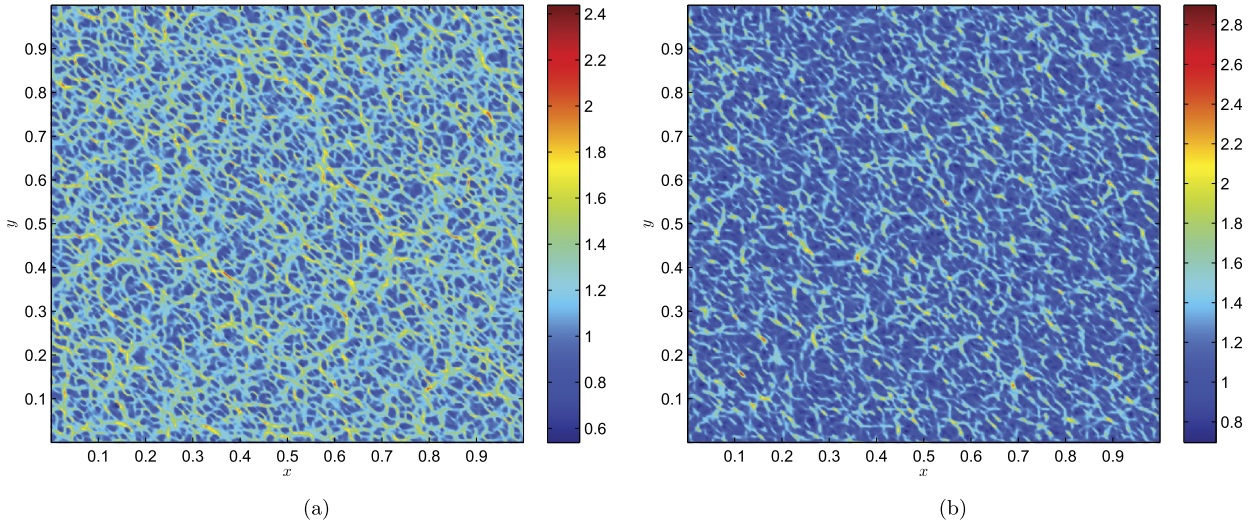


Fig. 5.  $\|\nabla F(x)\|$  and  $\|(\nabla F(x))^{-1}\|$ .

5.1. An elliptic PDE with rough coefficient

In this subsection, we consider an example of the multi-scale PDE (1.1) where the coefficient  $a(x, y)$  is rough and does not have clear scale-separation.  $a(x, y)$  and  $f(x, y)$  are given by

$$a(x, y) = |\tilde{a}(x, y)| + 0.5, \quad f(x, y) = 1.$$

The values of  $\tilde{a}(x, y)$  on the node points of an intermediate mesh of size  $\frac{1}{128}$  are i.i.d. Gaussian random variables. And the values of  $a(x, y)$  on the fine mesh node points are obtained using piecewise linear interpolation based on its values on the intermediate mesh node points. A typical realization of the coefficient  $a(x, y)$  is shown in Fig. 4, and we will solve equation (1.1) with this realization of  $a(x, y)$ .

We first numerically verify our stability assumption on the harmonic coordinates (4.30). We solve the harmonic coordinates (4.26) on the underlying fine mesh and numerically compute  $\|\nabla F(x)\|$  and  $\|(\nabla F(x))^{-1}\|$ . They are plotted in Fig. 5. We can see that  $\|\nabla F(x)\|$  and  $\|(\nabla F(x))^{-1}\|$  are both bounded, namely, the stability assumption (4.30) holds for this rough coefficient elliptic problem.

To start the iteration in Algorithm 1, we need two sampled solutions to generate the multi-scale basis  $\phi_i^1(x)$ . We first solve the equation (1.1) using piecewise linear basis on the coarse mesh to get  $u^{-1}(x)$ . Then we apply 10 steps of Jacobi



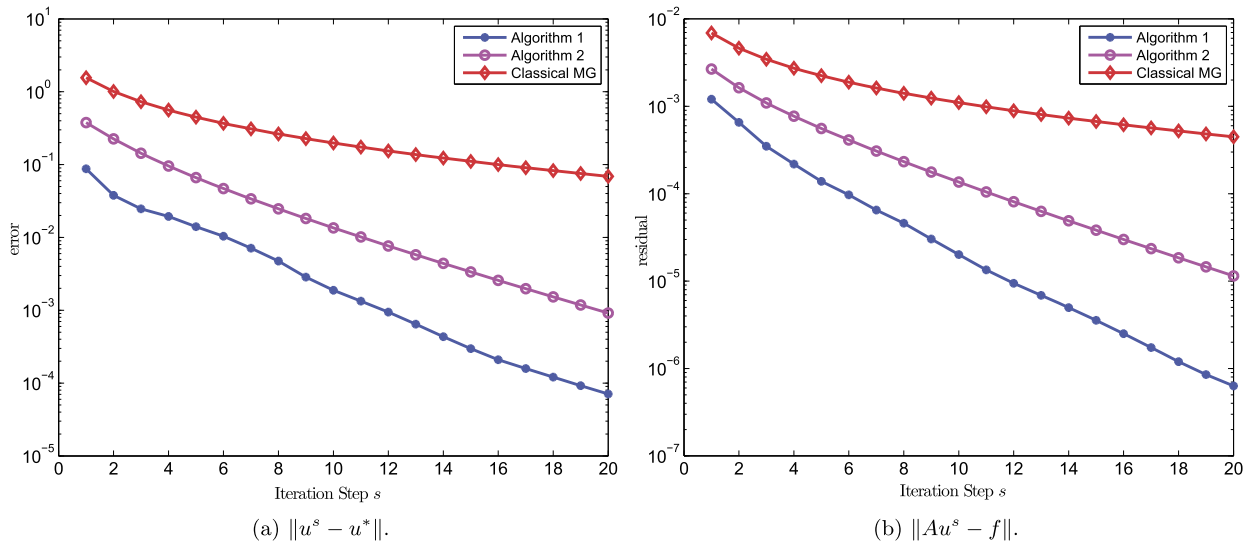


Fig. 6. Decay of the error  $\|u^s - u^*\|$  and the residual  $\|Au^s - f\|$ .

iteration on the fine mesh using  $u^{-1}(x)$  as initial guess to obtain  $u^0(x)$ . With  $u^{-1}(x)$  and  $u^0(x)$ , we build multi-scale  $\phi_i^1(x)$  following the construction (3.10) and (3.12).

In the iteration of both Algorithm 1 and Algorithm 2, we choose  $\nu = 10$  and the Gauss–Seidel iteration for the smoothing operations in step (3.6). There exist other smoothing methods that may perform better than the Gauss–Seidel iteration. However, for the purpose of comparing the proposed methods with the classical two-level multi-grid method, we will employ the simple Gauss–Seidel smoothing operation in implementation. We keep track of the  $l^2$  norms of the error  $\|\bar{u}^s - \bar{u}\|$  and the residual  $\|A\bar{u}^s - \bar{f}\|$  in the iteration. We solve the multi-scale equation (1.1) using Algorithm 1, Algorithm 2, and the classical two-level multi-grid method under the same discretization. Note that the only difference between the three methods is the choice of the finite element basis functions in the coarse mesh solution operator  $(L_H^s)^{-1}$ . In Algorithm 1, we employ the adaptive multi-scale basis functions  $\phi_i^s(x)$  and the bubble basis functions  $\phi_j^b(x)$ ; in Algorithm 2, we employ the multi-scale basis functions with linear local boundary conditions  $\phi_i(x)$  and  $\phi_j^b(x)$ ; in the classical two-level multi-grid method we employ the coarse-mesh piecewise linear basis functions. The decay of  $\|\bar{u}^s - \bar{u}\|$  and  $\|A\bar{u}^s - \bar{f}\|$  for the three methods is shown in Fig. 6.

According to the comparisons made in Fig. 6, we can get the following conclusions.

1. The  $l^2$  norms of the error and the residual both decay exponentially fast for Algorithm 1 and Algorithm 2. This agrees with our convergence analysis under the two-level multi-grid method framework.
2. The error and residual in Algorithm 1 and Algorithm 2 decay faster than the classical two-level multi-grid method. This is because the multi-scale basis functions employed in Algorithms 1 and 2 have better approximation property to the solution space of (1.1) than the coarse mesh piecewise linear basis.

In practical applications, even though the approximate solution  $u^s(x)$  in Algorithms 1 and 2 converge to the underlying fine mesh solution  $u_h(x)$  exponentially fast, we do not need to wait until

$$\|u^s(x) - u_h(x)\|_{L^2(D)} = O(h^2) \tag{5.1}$$

to stop the iteration. This is because the fine mesh size  $h$  is not chosen according to the desired order of accuracy, but according to the small-scale variations of the coefficient.

Even if we terminate the iteration based on the condition (5.1), we cannot get  $O(h)$  accuracy in the numerical solution  $u^s(x)$ . Due to the oscillation in the coefficient  $a(x)$ , we have

$$\|u(x) - u_h(x)\|_{L^2(D)} \gg h^2.$$

If our desired order of accuracy is the coarse mesh size  $H$ , as in the multi-scale methods of the second category described in section 1.3, [7,53,45,64], then we can terminate the iteration much earlier.

For this specific problem, with the Gauss–Seidel iteration for the smoothing operation, we obtain  $\frac{H^2}{10}$  in  $L^2(D)$  accuracy after 1 step of iteration in Algorithm 1 and 4 steps of iteration in Algorithm 2.

In the multi-scale methods, for example, [7,53,45,64], constructing multi-scale basis functions with robust approximation property could be very expensive. Our method, Algorithm 1, bypasses this difficulty by adaptively updating the multi-scale basis to improve the approximation property of the basis.

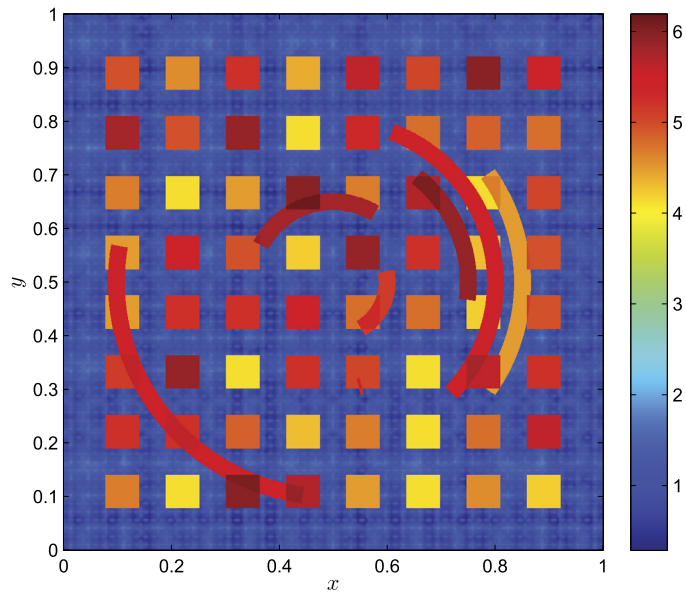


Fig. 7. The high-contrast coefficient.  $\log_{10} a(x)$ .

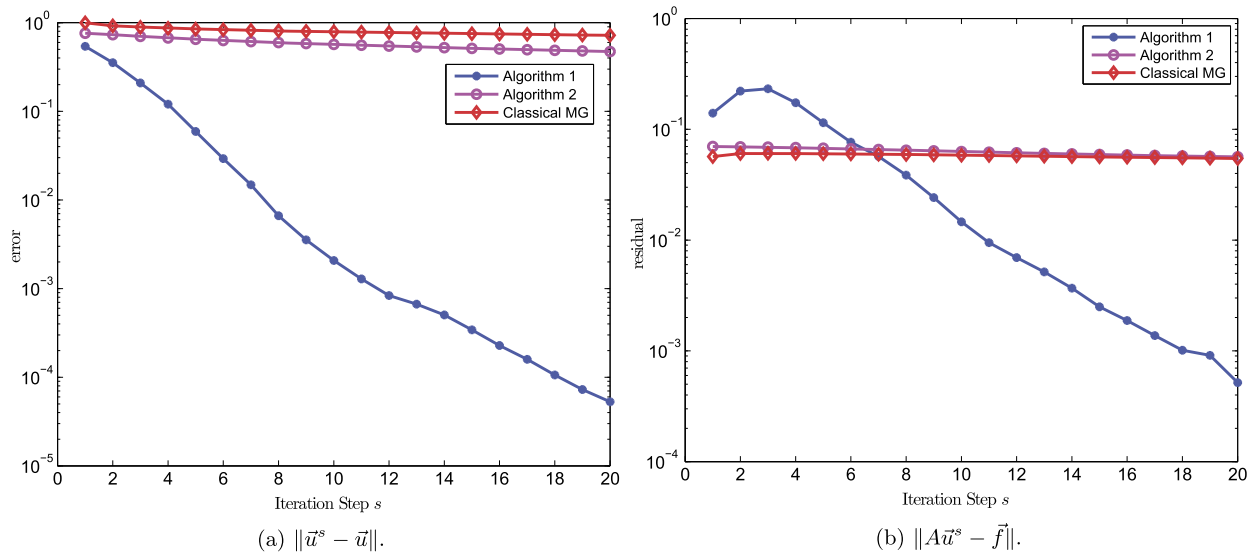


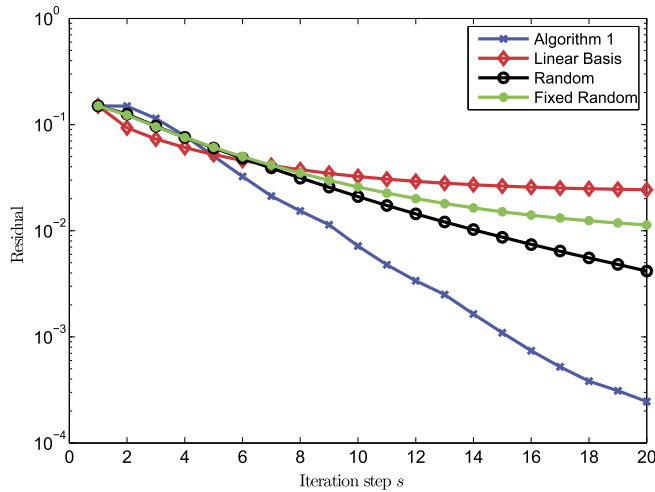
Fig. 8. Decay of  $l^2$  error and the residual for the high-contrast interface problem.

5.2. A high-contrast interface problem

In this subsection, we consider an elliptic problem with a high-contrast interface. Note that the high-contrast inclusions violate the uniform ellipticity assumption (1.2). However, the proposed Algorithm 1 still works well and performs much better than the classical two-level multi-grid method.

The high-contrast coefficient  $a(x, y)$  that we consider in this example is illustrated in Fig. 7, and the right hand side function is chosen as  $f(x, y) = 1$ . As in the previous numerical example, we solve this high-contrast interface problem using Algorithm 1, Algorithm 2 and the classical two-level multi-grid method under the same discretization. We choose  $\nu = 10$  and the Gauss–Seidel iteration in the smoothing operation. The decay of the error and the residual in the iteration is plotted in Fig. 8.

For this high-contrast interface problem, no convergence is observed for the proposed Algorithm 2 employing multi-scale basis with linear local boundary conditions and the classical two-level multi-grid method. Due to the high contrast inclusions, the derivatives of the solution has large jump across the boundaries of the inclusions. Multi-scale basis with linear local boundary conditions cannot accurately capture these jumps, thus has very poor approximation property to the solution space of (1.1). This explains the failure of Algorithm 2 and the classical two-level multi-grid method observed in Fig. 8.



**Fig. 9.** Comparison of Algorithm 1 with Algorithm 2 using multi-scale basis functions generated from random sampled solutions of (1.1). The line labeled “Fixed Random” corresponds the first approach described above for Algorithm 2, and the line labeled “Random” corresponds to the second approach for Algorithm 2. The line labeled “Linear Basis” corresponds to the classical two-level multi-scale method.

For this high-contrast interface problem, the  $l^2$  norms of the error  $\bar{u}^s - \bar{u}$  and the residual  $A\bar{u}^s - \bar{f}$  in the iteration of Algorithm 1 both converge exponentially fast. The adaptive multi-scale basis functions in Algorithm 1 are constructed based on sampled solutions of the equation, which can capture this jump of derivatives, thus have much better approximation property to the solution space of (1.1).

Next we consider constructing multi-scale basis functions  $\phi_i(x), i = 1, 2, \dots, n$  based on randomly generated sampled solutions of (1.1) following (3.10) and (3.12), and employing them in Algorithm 2 to solve the equation (1.1). For this purpose, we consider the following finite dimensional subspace of  $L^p(D)$ ,

$$\{1, \sin(2\pi x), \cos(2\pi x), \dots, \sin(10\pi x), \cos(10\pi x)\} \otimes \{1, \sin(2\pi y), \cos(2\pi y), \dots, \sin(10\pi y), \cos(10\pi y)\},$$

and generate random forcing functions  $f(x)$  from this finite-dimensional subspace of  $L^p(D)$ .

We remark that solving (1.1) with randomly generated forcing has the same computational complexity as the original problem, and we design this example only to compare the performance of Algorithm 1 and Algorithm 2. We consider two approaches for Algorithm 2. In the first approach, we generate a set of multi-scale basis functions before the iteration, and employ them in the coarse mesh solving step of Algorithm 2. In the second approach, we generate a new set of multi-scale basis functions for the coarse mesh solving operation in each iteration step of Algorithm 2. For this high-contrast interface problem, the convergence of the solutions is plotted in Fig. 9. We observe that the iterative error corresponding to Algorithm 2 using the second approach indeed decays faster than that using the first approach. This shows that updating the multi-scale basis functions during the iteration indeed helps to reduce the iteration error. However, the iterative error in Algorithm 1 decays much faster than the iterative errors of Algorithm 2 even if we update multi-scale basis functions based on randomly generated sampled solutions (the second approach).

### 5.3. A convection-dominated diffusion equation

The proposed iteratively adaptive multi-scale finite element method, Algorithm 1, can also be naturally applied to solve convection dominated diffusion problems. In this subsection, we consider equation

$$\begin{cases} -\kappa \Delta u(x) + \mathbf{a} \cdot \nabla u(x) = f, & x \in \Omega, \\ u(x) = g(x), & x \in \partial\Omega, \end{cases}$$

where  $\kappa = 10^{-6}$  is the diffusivity,  $\mathbf{a} = (a_1, a_2)^T$  is the velocity field,  $f(x)$  is a given source term, and  $g$  is given Dirichlet boundary condition. We consider the following four test cases as in [34]

1. Convective field with a fixed angle.  $\Omega = [0, 1] \times [0, 1]$ ,  $\mathbf{a} = (\sqrt{2}/2, \sqrt{2}/2)^T$ ,  $f = 0$ .
2. Thermal boundary layer problem.  $\Omega = [0, 1] \times [0, 0.5]$ ,  $\mathbf{a} = (2y, 0)^T$ ,  $f = 0$ .
3. Double ramp problem,  $\Omega = [0, 0.5] \times [0.5, 1] \cup [0.5, 1.5] \times [0, 1]$ ,  $\mathbf{a} = (1, 0)^T$ , and  $f = 1$ .
4. Smith–Hutton problem [72],  $\Omega = [-1, 1] \times [0, 1]$ ,  $\mathbf{a} = (2y(1 - x^2), -2x(1 - y^2))^T$ , and  $f = 0$ .

The problem setting and boundary conditions for the four cases are illustrated in Fig. 10.

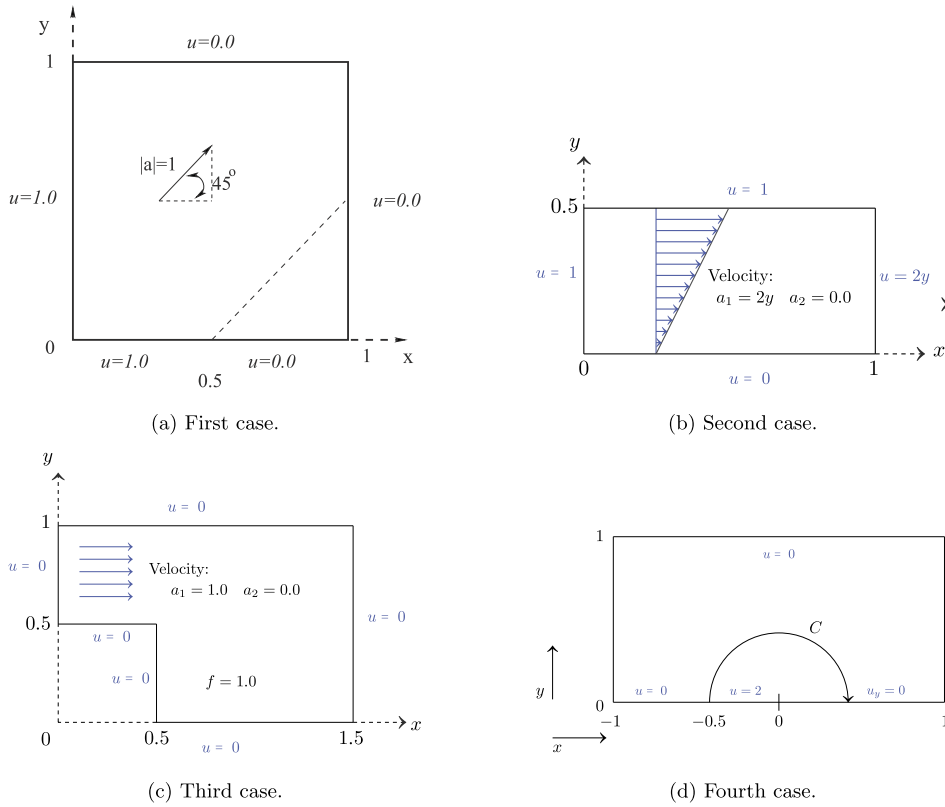


Fig. 10. Boundary conditions for the four convection-diffusion problems.

In the discretization of the problem, the coarse mesh size is  $H = \frac{1}{16}$  and fine mesh size is  $h = \frac{1}{128}$ . The streamline-crosswind diffusion method (SD/CD) [69] with a set of uniform rectangular meshes is used to construct the multi-scale basis with boundary conditions given by (3.9). And in the coarse mesh solving, we employ the following discretization of the equation: find  $u^{(s)}(x) \in V_H^s$ , such that

$$a_c(u^{(s)}(x), v(x)) = (f(x), v(x))_c, \quad \forall v(x) \in V_H^s,$$

where

$$a_c(u(x), v(x)) = a(u(x), v(x)) + \sum_i \tau_i (\mathbf{a} \cdot \nabla u, \mathbf{a} \cdot \nabla v)_{D_i} + \sum_i (\kappa_m - \kappa) (\mathbf{b} \cdot \nabla u, \mathbf{b} \cdot \nabla v)_{D_i}$$

and

$$(f, v)_c = (f, v) + \sum_i \tau_i (f, \mathbf{b} \cdot \nabla v)_{D_i}.$$

Here  $a(u, v) = \kappa(\nabla u, \nabla v) + (\mathbf{a} \cdot \nabla u, \nabla v)$  is a bilinear form and  $(f, v)$  is a linear form.

The stabilization parameter  $\tau_i$  is defined as

$$\tau_i = \frac{H_{D_i}^2}{6\kappa \max\{1, Pe_i\} + 6\kappa},$$

with

$$Pe_i = \frac{|\mathbf{a}|_a H_{D_i}}{3\kappa}.$$

The artificial crosswind diffusion coefficient  $\kappa_m$  is given as

$$\kappa_m = \begin{cases} \kappa, & \text{if } \kappa \geq H_{D_i}^{3/2}, \\ H_{D_i}^{3/2}, & \text{if } \kappa < H_{D_i}^{3/2}. \end{cases}$$

**Table 1**  
Number of iteration steps to obtain convergence for the four tested cases.

$\nu$	Fixed angle	Thermal	Double ramp	Smith–Hutton
1	5	11	10	7
2	4	7	8	6
3	4	6	8	5
4	3	5	6	5
5	3	5	5	5

And  $\mathbf{b} = (-a_2, a_1)$  is the crosswind vector.

The initial and final MsFEM solutions for the 4 cases are plotted in Fig. 11.

The initial MsFEM solutions are obtained using multi-scale basis with linear local boundary conditions. We observe that the accuracies of the MsFEM numerical solutions are significantly improved in the end of the iteration: the overshooting problem within the boundary layer regions is completely eliminated.

The smoothing operation (3.6) is chosen as the incomplete LU factorization without fill-in, namely, the ILU(0) method [68]. We choose  $\nu = 1, 2, \dots, 5$ , and the stopping criteria for the Algorithm is  $\|u^{s-1}(x) - u^s(x)\|_{L^2(\Omega)} \leq 10^{-4}$ . To obtain the convergence, the steps of iterations required for the four cases are summarized in Table 1. For  $\nu \geq 4$ , the iteration converges within 6 steps for all the 4 cases.

### 6. Concluding remarks

An iteratively adaptive multi-scale finite element is proposed, in which the multi-scale basis functions and the approximate solutions are updated simultaneously in each iteration step. The global information of the elliptic equation is transferred to the local boundary conditions of the multi-scale basis functions through coarse mesh solving using adaptive multi-scale basis and fine mesh smoothing operations.

A stabilized variant of the proposed algorithm is also derived, which is a two-level multi-grid method that employs a set of multi-scale basis functions for the restriction and prolongation operations. The proposed iteratively adaptive MsFEM can be viewed as a nonlinear version of the two-level multi-grid method.

Convergence analysis of the proposed algorithm is carried out in the two-level multi-grid method framework. And we employ the harmonic coordinates to establish the approximation property of the adaptively constructed multi-scale basis functions to the solution space of the equation.

Numerical results are presented for a rough coefficient problem, an elliptic interface problem and several convection dominated diffusion problems. Significant improvements over classical methods are observed for all these three types of PDE problems. In particular, the proposed Algorithm 1 has significant advantage over the classical two-level multi-grid method as a fine mesh solver of the multi-scale equation.

### Acknowledgements

This research was in part supported by NSF Grants DMS-1613861, DMS-1318377. The second author was also supported in part by the Ministry of Science and Technology of Taiwan, MOST-104-2115-M-008-011, MOST-105-2115-M-008-007.

### Appendix A

Let  $\phi_i(x)$ ,  $i = 1, 2, \dots, n$  be the multi-scale basis functions constructed using the sampled solutions  $v_1(x)$ ,  $v_2(x)$  obtained from (4.28) and following the procedure outlined in (3.10) and (3.12). Further, we denote  $u_H^{MS}(x)$  as the Galerkin finite element solution of (1.1) using the resulting trial space

$$V_h = \text{span}\{\phi_1(x), \phi_2(x), \dots, \phi_n(x)\}.$$

We will prove that

$$\|u(x) - u_H^{MS}(x)\|_{L^2(D)} \leq CH^2 \|f(x)\|_{L^2(D)}, \tag{A.1}$$

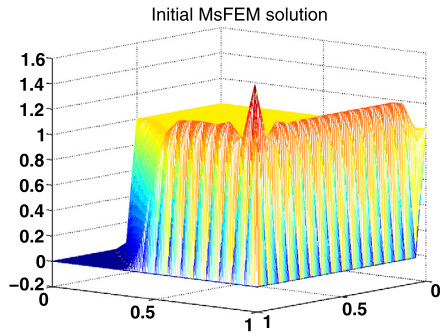
under the two assumptions in subsection 4.3, and some additional technical assumptions (A.5), (A.13), (A.14).

**Proof.** We first consider the interpolation of the solution  $u(x)$  using the non-conforming basis  $\bar{\phi}_i(x)$ . On the boundary of the coarse element  $D_i$ , the interpolation residual  $e_i(x) = u(x) - \sum_{j=1}^3 u(x_j) \bar{\phi}_j(x)$  is

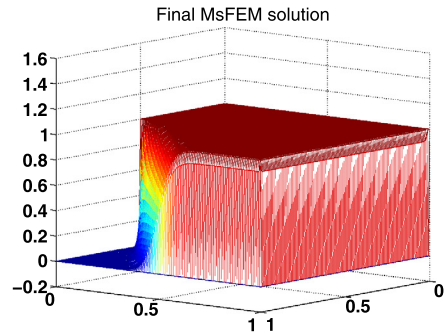
$$e_i(x) = u(x) - c_i^0 - c_i^1 v_1(x) - c_i^2 v_2(x), \quad x \in \partial D_i.$$

Then employing the optimality property of multi-scale basis functions, (2.6), we have

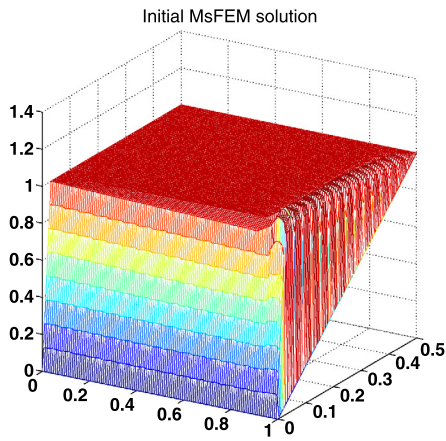
$$\|e_i(x)\|_E^2 \leq \|u(x) - c_i^0 - c_i^1 v_1(x) - c_i^2 v_2(x)\|_E^2 + CH^2 \|f(x)\|_{L^2(D_i)}^2, \tag{A.2}$$



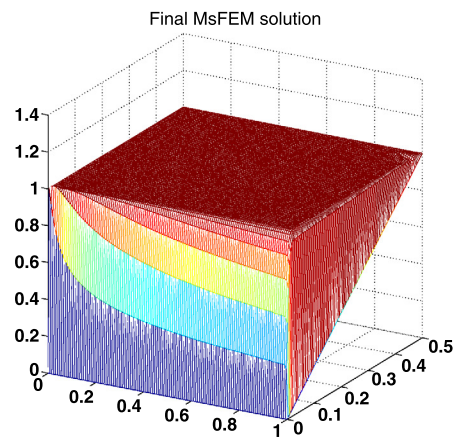
(a) Initial MsFEM solution. First case.



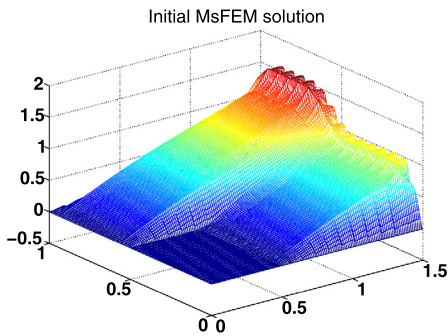
(b) Final MsFEM solution. First case.



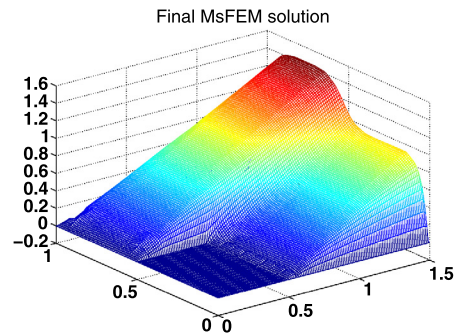
(c) Initial MsFEM solution. Second case.



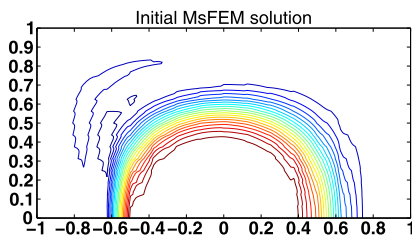
(d) Final MsFEM solution. Second case.



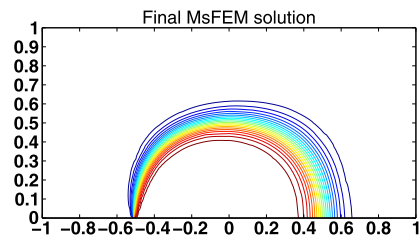
(e) Initial MsFEM solution. Third case.



(f) Final MsFEM solution. Third case.



(g) Initial MsFEM solution. Fourth case.



(h) Final MsFEM solution. Fourth case.

Fig. 11. Initial and final MsFEM solutions for the convection-dominated diffusion problems.

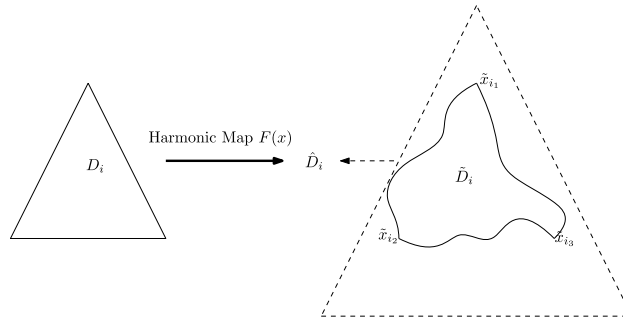


Fig. 12. Coarse-mesh element in Harmonic Coordinate.

where  $\|\cdot\|_E$  is the energy norm (1.9). To bound the energy norm of  $E_i(x) = u(x) - c_i^0 - c_i^1 v_1(x) - c_i^2 v_2(x)$  in the above estimate, (A.2), we express this term in the harmonic coordinates,  $\tilde{E}_i(x) = E_i \circ F^{-1}(x)$ ,

$$\tilde{E}_i(x) = \tilde{u}(x) - c_i^0 - c_i^1 \tilde{v}_1(x) - c_i^2 \tilde{v}_2(x), \quad x \in \tilde{D}_i. \tag{A.3}$$

We denote  $\hat{v}_{i,1}(x)$ ,  $\hat{v}_{i,2}(x)$  and  $\hat{u}_i(x)$  as the linear interpolation of  $\tilde{v}_1(x)$ ,  $\tilde{v}_2(x)$  and  $\tilde{u}(x)$  on the three node points,  $\tilde{x}_{i,1}$ ,  $\tilde{x}_{i,2}$ ,  $\tilde{x}_{i,3}$ . Then we can decompose (A.3) as

$$\tilde{E}_i(x) = [\tilde{u}(x) - \hat{u}_i(x) - c_i^1 \hat{v}_{i,1}(x) - c_i^2 \hat{v}_{i,2}(x)] + [c_i^1 (\hat{v}_{i,1}(x) - \tilde{v}_1(x)) + c_i^2 (\hat{v}_{i,2}(x) - \tilde{v}_2(x))]. \tag{A.4}$$

The first term in (A.4) is simply  $\tilde{u}(x) - \hat{u}_i(x)$ , and to get an estimate of it on  $\tilde{D}_i$ , we need to put  $\tilde{D}_i$  in a regular domain. Let  $\hat{D}_i$  be the triangle which is the scaling of  $\Delta \tilde{x}_{i,1} \tilde{x}_{i,2} \tilde{x}_{i,3}$  by a factor of 2, and has the same center as  $\Delta \tilde{x}_{i,1} \tilde{x}_{i,2} \tilde{x}_{i,3}$ . Then we assume that  $\tilde{D}_i \subset \hat{D}_i$  and each  $x \in D$  is covered by finite  $\hat{D}_i$ ,

$$\tilde{D}_i \subset \hat{D}_i, \quad \max_{x \in D} \#\{i : x \in \hat{D}_i\} \leq C. \tag{A.5}$$

The element  $D_i$ , its image  $\tilde{D}_i$ , and the triangle  $\hat{D}_i$  are illustrated in Fig. 12.

Then using the regularity result (4.27) and the assumption that  $\hat{D}_i$  has bounded aspect ratio, we have

$$\|\tilde{u}(x) - c_i^0 - c_i^1 \hat{v}_{i,1}(x) - c_i^2 \hat{v}_{i,2}(x)\|_{H^1(\tilde{D}_i)} \leq \|\tilde{u}(x) - c_i^0 - c_i^1 \hat{v}_{i,1}(x) - c_i^2 \hat{v}_{i,2}(x)\|_{H^1(\hat{D}_i)} \leq CH \|\tilde{u}(x)\|_{H^2(\hat{D}_i)}. \tag{A.6}$$

Using the same assumptions as in (A.6), we can bound the second term in (A.4) by

$$\|c_i^1 (\hat{v}_{i,1}(x) - \tilde{v}_1(x)) + c_i^2 (\hat{v}_{i,2}(x) - \tilde{v}_2(x))\|_{H^1(\tilde{D}_i)} \leq |c_i^1| H \|\tilde{v}_1(x)\|_{H^2(\hat{D}_i)} + |c_i^2| H \|\tilde{v}_2(x)\|_{H^2(\hat{D}_i)}. \tag{A.7}$$

Putting estimates (A.6) and (A.7) together in the decomposition (A.4), we have

$$\|\tilde{E}_i(x)\|_{H^1(\tilde{D}_i)}^2 \leq CH^2 \|\tilde{u}(x)\|_{H^2(\hat{D}_i)}^2 + C(c_i^1)^2 H^2 \|\tilde{v}_1(x)\|_{H^2(\hat{D}_i)}^2 + C(c_i^2)^2 H^2 \|\tilde{v}_2(x)\|_{H^2(\hat{D}_i)}^2. \tag{A.8}$$

Adding up the estimate (A.8) over each coarse element, we have

$$\sum_i \|E_i(x)\|_{H^1(D_i)}^2 \leq CH^2 \|\tilde{u}(x)\|_{H^2(D)}^2 + C \max_i (c_i^1)^2 H^2 \|\tilde{v}_1(x)\|_{H^2(D)}^2 + C \max_i (c_i^2)^2 H^2 \|\tilde{v}_2(x)\|_{H^2(D)}^2. \tag{A.9}$$

Using the regularity result (4.27) and the normalization condition (4.29), we have

$$\|\tilde{u}(x)\|_{H^2(D)} \leq C \|f(x)\|_{L^p(D)}; \quad \|\tilde{v}_1(x)\|_{H^2(D)}, \|\tilde{v}_2(x)\|_{H^2(D)} \leq C. \tag{A.10}$$

For the coefficients  $c_i^1$  and  $c_i^2$ , according to (4.33), we have

$$c_i^1, c_i^2 \leq C \|\tilde{u}(x)\|_{C^1(D)} \leq C \|f(x)\|_{L^p(D)}. \tag{A.11}$$

Putting the estimates (A.10), (A.11) and (A.9) in (A.2), we have

$$\sum_i \|e_i(x)\|_{H^1(D_i)}^2 \leq CH^2 \|f(x)\|_{L^p(D)}^2. \tag{A.12}$$

To obtain the approximation property of the conforming basis (3.12), we simply assume that the error will not be amplified due to the averaging,

$$\|u(x) - \sum_i u(x_i) \phi_i(x)\|_{H^1(D)}^2 \leq C \sum_i \|e_i(x)\|_{H^1(D_i)}^2. \tag{A.13}$$

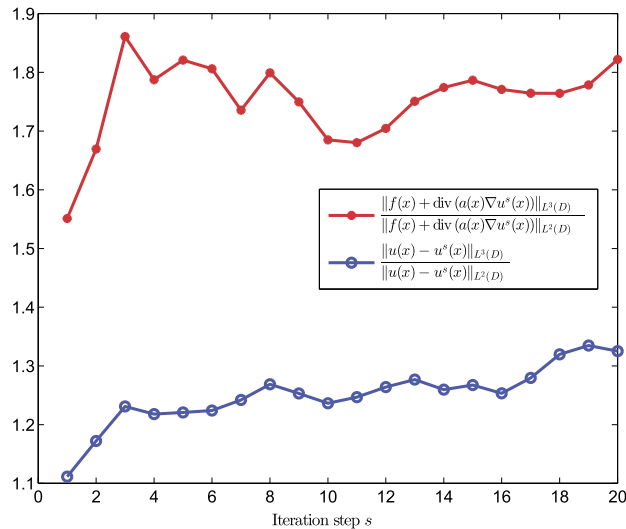


Fig. 13. The constants in (A.14) and (A.15).

Then using the optimal property (1.9), (A.12) and (A.13), we have

$$\|u_H^{MS}(x) - u(x)\|_{H^1(D)} \leq CH \|f(x)\|_{L^p(D)}.$$

Finally, we assume that for all the  $f(x)$  that occur in the iteration of Algorithm 1, we have

$$\|f(x)\|_{L^p(D)} \leq C \|f(x)\|_{L^2(D)}, \quad (\text{A.14})$$

which requires that the  $f(x)$  is wide spread. With (A.14), we get

$$\|u(x) - u_H^{MS}\|_{H^1(D)} \leq CH \|f(x)\|_{L^2(D)}.$$

Then using a Aubin–Nitsche duality argument, we get (A.1). The assumption (A.14) requires

$$\|u(x) - u_H^{MS}\|_{L^p(D)} \leq C \|u(x) - u_H^{MS}(x)\|_{L^3(D)} \quad (\text{A.15})$$

in the duality argument. We keep track of the constants  $C$  in (A.14) and (A.15) in the iteration of Algorithm 1 for the rough coefficient problem in subsection (5.1), and they are plotted in Fig. 13.

## References

- [1] R.E. Alcouffe, A. Brandt, J.E. Dendy Jr., J.W. Painter, The multi-grid method for the diffusion equation with strongly discontinuous coefficients, *SIAM J. Sci. Stat. Comput.* 2 (4) (1981) 430–454.
- [2] G. Allaire, Homogenization and two-scale convergence, *SIAM J. Math. Anal.* 23 (6) (1992) 1482–1518.
- [3] G. Allaire, *Shape Optimization by the Homogenization Method*, vol. 146, Springer Science & Business Media, 2012.
- [4] A. Averbuch, G. Beylkin, R. Coifman, M. Israeli, Multiscale inversion of elliptic operators, *Wavelet Anal. Appl.* 7 (1998) 341–359.
- [5] I. Babuška, G. Caloz, J. Osborn, Special finite element methods for a class of second order elliptic problems with rough coefficients, *SIAM J. Numer. Anal.* 31 (4) (1994) 945–981.
- [6] I. Babuška, X. Huang, R. Lipton, Machine computation using the exponentially convergent multiscale spectral generalized finite element method, *ESAIM: Math. Model. Numer. Anal.* 48 (02) (2014) 493–515.
- [7] I. Babuska, R. Lipton, Optimal local approximation spaces for generalized finite element methods with application to multiscale problems, *Multiscale Model. Simul.* 9 (1) (2011) 373–406.
- [8] I. Babuška, J. Osborn, Generalized finite element methods: their performance and their relation to mixed methods, *SIAM J. Numer. Anal.* 20 (3) (1983) 510–536.
- [9] I. Babuška, J. Osborn, Can a finite element method perform arbitrarily badly?, *Math. Comput. AMS* 69 (230) (2000) 443–462.
- [10] M. Bebendorf, *Hierarchical Matrices*, Springer, 2008.
- [11] A. Bensoussan, J.L. Lions, G. Papanicolaou, *Asymptotic Analysis for Periodic Structures*, vol. 374, American Mathematical Soc., 2011.
- [12] G. Beylkin, N. Coult, A multiresolution strategy for reduction of elliptic PDEs and eigenvalue problems, *Appl. Comput. Harmon. Anal.* 5 (2) (1998) 129–155.
- [13] A. Brandt, Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problems, in: *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics*, Springer, 1973, pp. 82–89.
- [14] A. Brandt, Algebraic multigrid theory: the symmetric case, *Appl. Math. Comput.* 19 (1) (1986) 23–56.
- [15] A. Brandt, S. McCormick, J. Huges, Algebraic multigrid (AMG) for sparse matrix equations, in: *Sparsity and Its Applications*, 1985, p. 257.
- [16] M.E. Brewster, G. Beylkin, A multiresolution strategy for numerical homogenization, *Appl. Comput. Harmon. Anal.* 2 (4) (1995) 327–349.
- [17] Z. Chen, T.Y. Hou, A mixed multiscale finite element method for elliptic problems with oscillating coefficients, *Math. Comput.* 72 (242) (2003) 541–576.
- [18] I.L. Chern, Y.C. Shu, A coupling interface method for elliptic interface problems, *J. Comput. Phys.* 225 (2) (2007) 2138–2174.



- [19] C.C. Chu, I. Graham, T.Y. Hou, A new multiscale finite element method for high-contrast elliptic interface problems, *Math. Comput.* 79 (272) (2010) 1915–1955.
- [20] R. Codina, A discontinuity-capturing crosswind-dissipation for the finite element solution of the convection–diffusion equation, *Comput. Methods Appl. Mech. Eng.* 110 (1993) 325–342.
- [21] E. De Giorgi, New problems in  $\gamma$ -convergence and  $g$ -convergence, in: *Free Boundary Problems*, vol. 2, 1980, pp. 183–194.
- [22] E. De Giorgi, Sulla convergenza di alcune successioni d'integrali del tipo dell'area, *Ennio De Giorgi*, 2006, p. 414.
- [23] P.M. De Zeeuw, Matrix-dependent prolongations and restrictions in a blackbox multigrid solver, *J. Comput. Appl. Math.* 33 (1) (1990) 1–27.
- [24] M. Dorobantu, B. Engquist, Wavelet-based numerical homogenization, *SIAM J. Numer. Anal.* 35 (2) (1998) 540–559.
- [25] L.J. Durlofsky, Y. Efendiev, V. Ginting, An adaptive local–global multiscale finite volume element method for two-phase flow simulations, *Adv. Water Resour.* 30 (3) (2007) 576–588.
- [26] Y. Efendiev, J. Galvis, P.S. Vassilevski, Spectral element agglomerate algebraic multigrid methods for elliptic problems with high-contrast coefficients, in: *Domain Decomposition Methods in Science and Engineering XIX*, Springer, 2011, pp. 407–414.
- [27] Y. Efendiev, V. Ginting, T.Y. Hou, R. Ewing, Accurate multiscale finite element methods for two-phase flow simulations, *J. Comput. Phys.* 220 (1) (2006) 155–174.
- [28] Y. Efendiev, T.Y. Hou, V. Ginting, Multiscale finite element methods for nonlinear problems and their applications, *Commun. Math. Sci.* 2 (4) (2004) 553–589.
- [29] Y. Efendiev, T.Y. Hou, X.H. Wu, Convergence of a nonconforming multiscale finite element method, *SIAM J. Numer. Anal.* 37 (3) (2000) 888–910.
- [30] B. Engquist, E. Luo, Convergence of a multigrid method for elliptic equations with highly oscillatory coefficients, *SIAM J. Numer. Anal.* 34 (6) (1997) 2254–2273.
- [31] C. Farhat, I. Kalashnikova, R. Tezaur, A higher-order discontinuous enrichment method for the solution of high Péclet advection–diffusion problems on unstructured meshes, *Int. J. Numer. Methods Eng.* 81 (2010) 604–636.
- [32] R.P. Fedorenko, A relaxation method for solving elliptic difference equations, *USSR Comput. Math. Math. Phys.* 1 (4) (1962) 1092–1096.
- [33] L.P. Franca, S.L. Frey, T.J.R. Hughes, Stabilized finite element methods: I. Application to the advective–diffusive model, *Comput. Methods Appl. Mech. Eng.* 95 (1992) 253–276.
- [34] L.P. Franca, F.N. Hwang, Refining the submesh strategy in the two-level finite element method: application to the advection–diffusion equation, *Int. J. Numer. Methods Fluids* 39 (2002) 161–187.
- [35] J. Galvis, Y. Efendiev, Domain decomposition preconditioners for multiscale flows in high-contrast media, *Multiscale Model. Simul.* 8 (4) (2010) 1461–1483.
- [36] A. Gloria, An analytical framework for the numerical homogenization of monotone elliptic operators and quasiconvex energies, *Multiscale Model. Simul.* 5 (3) (2006) 996–1043.
- [37] I.G. Graham, P.O. Lechner, Robert Scheichl, Domain decomposition for multiscale PDEs, *Numer. Math.* 106 (4) (2007) 589–626.
- [38] W. Hackbusch, A fast iterative method for solving Poisson's equation in a general region, in: *Numerical Treatment of Differential Equations*, Springer, 1978, pp. 51–62.
- [39] W. Hackbusch, *Multi-grid Methods and Applications*, vol. 4, Springer Science & Business Media, 2013.
- [40] W. Hackbusch, L. Grasedyck, An introduction to hierarchical matrices, *Math. Bohem.* 127 (2) (2002) 229–241.
- [41] H. Hajibeygi, G. Bonfigli, M.A. Hesse, P. Jenny, Iterative multiscale finite-volume method, *J. Comput. Phys.* 227 (19) (2008) 8604–8621.
- [42] H. Hajibeygi, P. Jenny, Adaptive iterative multiscale finite volume method, *J. Comput. Phys.* 230 (2011) 628–643.
- [43] P. Henning, D. Peterseim, Oversampling for the multiscale finite element method, *Multiscale Model. Simul.* 11 (4) (2013) 1149–1175.
- [44] T.Y. Hou, P. Liu, A heterogeneous stochastic FEM framework for elliptic PDEs, *J. Comput. Phys.* 281 (2015) 942–969.
- [45] T.Y. Hou, P. Liu, Optimal local multi-scale basis functions for linear elliptic equations with rough coefficient, *Discrete Contin. Dyn. Syst.* 36 (8) (2016) 4451–4476.
- [46] T.Y. Hou, P. Liu, Z. Zhang, A model reduction method for elliptic PDEs with random input using the heterogeneous stochastic FEM framework, *Bull. - Inst. Math.* 11 (2016) 179–216.
- [47] T.Y. Hou, X.H. Wu, A multiscale finite element method for elliptic problems in composite materials and porous media, *J. Comput. Phys.* 134 (1) (1997) 169–189.
- [48] T.Y. Hou, X.H. Wu, Z. Cai, Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients, *Math. Comput. AMS* 68 (227) (1999) 913–943.
- [49] T.Y. Hou, X.H. Wu, Y. Zhang, Removing the cell resonance error in the multiscale finite element method via a Petrov–Galerkin formulation, *Commun. Math. Sci.* 2 (2) (2004) 185–205.
- [50] V.V. Jikov, S.M. Kozlov, O.A. Oleinik, *Homogenization of Differential Operators and Integral Functionals*, Springer Science & Business Media, 2012.
- [51] Jim E. Jones, Panayot S. Vassilevski, Amge based on element agglomeration, *SIAM J. Sci. Comput.* 23 (1) (2001) 109–133.
- [52] Z. Li, The immersed interface method using a finite element formulation, *Appl. Numer. Math.* 27 (1998) 253–267.
- [53] A. Målqvist, D. Peterseim, Localization of elliptic multiscale problems, *Math. Comput.* 83 (290) (2014) 2583–2603.
- [54] J. Mandel, M. Brezina, P. Vaněk, Energy optimization of algebraic multigrid bases, *Computing* 62 (3) (1999) 205–228.
- [55] A. Maugerí, K. Palagachev, L.G. Softova, *Elliptic and Parabolic Equations with Discontinuous Coefficients*, vol. 109, Wiley, 2000.
- [56] J.M. Melenk, I. Babuška, The partition of unity finite element method: basic theory and applications, *Comput. Methods Appl. Mech. Eng.* 139 (1) (1996) 289–314.
- [57] R. Millward, *A New Adaptive Multiscale Finite Element Method with Applications to High Contrast Interface Problems*, PhD thesis, University of Bath, 2011.
- [58] F. Murat, Compacité par compensation, *Ann. Sc. Norm. Super. Pisa, Cl. Sci.* 5 (3) (1978) 489–507.
- [59] F. Murat, L. Tartar, *H-Convergence*, Springer, 1997.
- [60] H. Owhadi, Bayesian numerical homogenization, *Multiscale Model. Simul.* 13 (13) (2015) 812–828.
- [61] H. Owhadi, Multi-grid with rough coefficients and multiresolution operator decomposition from hierarchical information games, arXiv:1503.03467, 2015.
- [62] H. Owhadi, L. Zhang, Metric-based upscaling, *Commun. Pure Appl. Math.* 60 (5) (2007) 675–723.
- [63] H. Owhadi, L. Zhang, Localized bases for finite-dimensional homogenization approximations with nonseparated scales and high contrast, *Multiscale Model. Simul.* 9 (4) (2011) 1373–1398.
- [64] H. Owhadi, L. Zhang, L. Berlyand, Polyharmonic homogenization, rough polyharmonic splines and sparse super-localization, *ESAIM: Math. Model. Numer. Anal.* 48 (02) (2014) 517–552.
- [65] N. Panasenko, N.S. Bakhvalov, *Homogenization: Averaging Processes in Periodic Media: Mathematical Problems in the Mechanics of Composite Materials*, 1989.
- [66] D. Peterseim, Variational multiscale stabilization and the exponential decay of fine-scale correctors, arXiv:1505.07611, 2015.
- [67] J.W. Ruge, K. Stüben, Algebraic multigrid, *Multigrid Methods* 3 (1987) 73–130.
- [68] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, 2003.

- [69] Y.T. Shih, H.C. Elman, Iterative methods for stabilized discrete convection-diffusion problems, *IMA J. Numer. Anal.* 20 (2000) 333–358.
- [70] S. Spagnolo, Sulla convergenza di soluzioni di equazioni paraboliche ed ellittiche, *Ann. Sc. Norm. Super. Pisa, Cl. Sci.* 22 (4) (1968) 571–597.
- [71] S. Spagnolo, Convergence in energy for elliptic operators, in: *Numerical Solutions of Partial Differential Equations III*, Acad. Press, New York, 1976.
- [72] U. Trottenberg, C. Oosterlee, A. Schüller, *Multigrid*, Academic Press, 2000.
- [73] P.S. Vassilevski, On two ways of stabilizing the hierarchical basis multilevel methods, *SIAM Rev.* 39 (1) (1997) 18–53.
- [74] P.S. Vassilevski, General constrained energy minimization interpolation mappings for AMG, *SIAM J. Sci. Comput.* 32 (1) (2010) 1–13.
- [75] P.S. Vassilevski, J. Wang, Stabilizing the hierarchical basis by approximate wavelets II: implementation and numerical results, *SIAM J. Sci. Comput.* 20 (2) (1998) 490–514.
- [76] W.L. Wan, T.F. Chan, B. Smith, An energy-minimizing interpolation for robust multigrid methods, *SIAM J. Sci. Comput.* 21 (4) (1999) 1632–1649.
- [77] J. Xu, Y. Zhu, Uniform convergent multigrid methods for elliptic problems with strongly discontinuous coefficients, *Math. Models Methods Appl. Sci.* 18 (01) (2008) 77–105.
- [78] J. Xu, L. Zikatanov, On an energy minimizing basis for algebraic multigrid methods, *Comput. Vis. Sci.* 7 (3–4) (2004) 121–127.