

Adaptive multiscale model reduction with Generalized Multiscale Finite Element Methods

Eric Chung*

Yalchin Efendiev[†]Thomas Y. Hou[‡]

April 29, 2016

Abstract

In this paper, we discuss a general multiscale model reduction framework based on multiscale finite element methods. We give a brief overview of related multiscale methods. Due to page limitations, the overview focuses on a few related methods and is not intended to be comprehensive. We present a general adaptive multiscale model reduction framework, the Generalized Multiscale Finite Element Method. Besides the method's basic outline, we discuss some important ingredients needed for the method's success. We also discuss several applications. The proposed method allows performing local model reduction in the presence of high contrast and no scale separation.

1 Introduction

1.1 Objectives

In this paper, we will present a novel multiscale model reduction technique for solving many challenging problems with multiple scales and high contrast. Our approach systematically and adaptively adds degrees of freedom in local regions as needed and goes beyond scale separation cases. The objectives of the paper are the following: (1) to demonstrate the main concepts of our unified approach for local multiscale model reduction; (2) to discuss the method's main ingredients; and (3) to demonstrate its applications to a variety of challenging multiscale problems.

1.2 The need for a systematic multiscale model reduction approach

Computational meshes. In order to present our approach, we need the notions of coarse and fine meshes, which are illustrated in Figure 1. We assume that the computational domain Ω is partitioned by a coarse grid \mathcal{T}^H , which does not necessarily resolve any multiscale features. We let N_c be the number of nodes in the coarse grid and N_e be the number of coarse edges. We use the notation K to represent a generic coarse element in \mathcal{T}^H . To represent multiscale basis functions, we perform a refinement of \mathcal{T}^H to obtain a fine grid \mathcal{T}^h , with mesh size $h > 0$. The fine grid can essentially resolve all multiscale features of the problem and we perform the computations of the local basis functions on the fine grid.

Scale separation approaches and their limitations. Many current approaches for handling complex multiscale problems typically limit themselves to two or more distinct idealized scales (e.g., pore and Darcy scales). These include homogenization and numerical homogenization methods [64, 37, 68]. To demonstrate some of the main concepts, we consider

$$\mathcal{L}(u) = f, \quad (1)$$

*Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong

[†]Department of Mathematics, Texas A&M University, College Station, TX 77843

[‡]Applied and Computational Math. 9-94, California Institute of Technology, Pasadena 91125

where $\mathcal{L}(u)$ is a differential operator representing the fine-scale process. One can use many different examples for $\mathcal{L}(u)$. To convey our main idea, we consider a simple and well-studied heterogeneous diffusion $\mathcal{L}(u) = -\text{div}(\kappa(x)\nabla u)$, where one assumes $\kappa(x)$ to be a multiscale field representing the media properties. Homogenization and numerical homogenization techniques derive or postulate macroscopic equations and formulate local problems for computing the macroscopic parameters. For example, in the heterogeneous diffusion example, one computes the effective properties $\kappa^*(x)$ on a coarse grid as a constant tensor, κ_{ij}^* , (see Figure 1 for illustration of coarse and fine grids) via solving local problems (see Section 2). Using $\kappa^*(x)$, one solves the global problem (1). *The number of macroscopic parameters represents the effective dimension of the local solution space.* Consequently, these (numerical homogenization) approaches cannot represent many important local features of the solution space unless they are identified apriori in a modeling step. In this paper, we propose a novel approach that avoids these limitations and determines necessary local degrees of freedom as needed.

Global model reduction approaches. The proposed approaches share some common features with global model reduction techniques [49, 5, 12], which construct global basis functions. Our approach uses local dimension reduction techniques. However, there are many important differences as we will discuss. First, global model reduction approaches, though powerful in reducing the degrees of freedom, lack local adaptivity and numerical discretization properties (e.g., conservations of local mass and energy,...) that local approaches enjoy. Many successful macroscopic laws (e.g., Darcy’s law, and so on) are possible because the solution space admits a large compression *locally*. We will present several distinct examples to demonstrate this. For this reason, it is important to construct local multiscale model reduction techniques that can identify local degrees of freedom and be consistent with homogenization when there is scale separation. The proposed method is a first systematic step in developing such approaches.

Multiscale Finite Element Methods and some related methods. The proposed methods take their origin in Multiscale Finite Element Methods [50, 54] and Generalized Finite Element Methods [60]. The main idea of MsFEM is to construct local multiscale basis functions, ϕ^{ω_i} for each coarse block ω_i , and replace macroscopic equations by using a limited number of basis functions. More precisely, for each coarse node i (see Figure 1), we construct multiscale basis function ϕ^{ω_i} and seek an approximate solution of (1) $u_{\text{approx}} = \sum_i c_i \phi^{\omega_i}$. These approaches motivate our new techniques as MsFEMs are the first methods that replace macroscopic equation concepts with carefully designed multiscale basis functions within finite element methods. Multiscale finite element approaches are shown to be powerful and have many advantages over numerical homogenization methods as MsFEMs can recover fine-scale information and be flexible in terms of gridding. However, these methods do not contain a systematic way to add degrees of freedom locally and adaptively, which are the proposed method’s main contributions. Approaches, such as variational multiscale methods [53] (see also, [47]), multiscale finite volume [54], mortar multiscale method [4], localized reduced basis [6, 28] can be related to the MsFEM and seek to approximate the solution when there is no scale separation. Other classes of approaches built on numerical homogenization methods (e.g., [29, 58, 48, 41]) are limited to problems with scale separation and when macroscopic equations can be formulated. Due to the lack of space, we will briefly discuss a few of these methods in the paper.

1.3 The basic concepts of Generalized Multiscale Finite Element Method

General idea of GMsFEM. In this paper, we will develop novel solution strategies centered around adaptive multiscale model reduction. The GMsFEM was first presented in [32] and later investigated in several other papers (e.g., [43, 44, 33, 35, 8, 25, 26, 23, 24]). The GMsFEM is motivated by our earlier works on designing multiscale coarse spaces for domain decomposition preconditioners [42, 31, 34], the studies in MsFEM, and the concepts of global model reduction methods (e.g., [49, 5] and references therein), which employ global snapshots and Proper Orthogonal Decomposition (POD). The proposed approach is a generalization of the MsFEM and defines appropriate local snapshots and local spectral decompositions. The need for such systematic methods will be discussed in the examples of various applications. Our proposed approaches add local degrees of freedom as needed and provide *numerical* macroscopic equations for problems without scale separation and high contrast. Because of the proposed multiscale model reduction’s local nature of, one can adaptively add the degrees of freedom based on error estimators and rigorously estimate the errors.

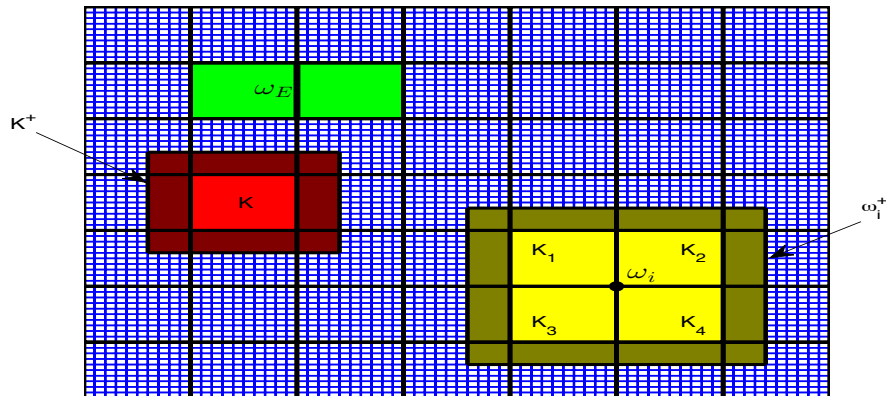


Figure 1: Illustration of fine grid, coarse grid, coarse neighborhood and oversampled domain.

To our best knowledge, this is one of the first local systematic multiscale model reduction frameworks that can be easily adopted for different applications.

Multiscale basis functions and snapshot spaces. The main idea of our multiscale approach is to systematically select important degrees of freedom for the solution in each coarse block (see Figure 1 for coarse and fine grid illustration). More precisely, for each coarse block ω_i (or K), we identify local multiscale basis functions $\phi_j^{\omega_i}$ ($j = 1, \dots, N_{\omega_i}$) and seek the solution in the span of these basis functions. For problems with scale separation, one needs a limited number of degrees of freedom. However, as the heterogeneities get more complicated, one needs a systematic approach to find the additional degrees of freedom. In each coarse grid, we first build the snapshot space, $V_{\text{snap}}^{\omega_i} = \text{span}\{\psi_j^{\omega_i}\}$. The choice of the snapshot space depends on the global discretization and the particular application. The appropriate snapshot space (1) yields faster convergence, (2) imposes problem relevant restrictions on the coarse spaces (e.g., divergence free solutions) and (3) reduces the computational cost of building the offline spaces. Each snapshot can be constructed, for example, using random boundary conditions or source terms [9].

Reducing the degrees of freedom. Once we construct the snapshot space $V_{\text{snap}}^{\omega_i}$, we identify the offline space, which is a principal component subspace of the snapshot space and derived based on analysis. To obtain the offline space, we perform a dimension reduction of the snapshot space. This reduction identifies dominant modes, which we use to construct a multiscale space. If the snapshot space and the local spectral problems are appropriately chosen, we obtain reduced dimensional spaces corresponding to numerical homogenization in the case of scale separation. When there is no scale separation, we have a constructive method to add the necessary extra degrees of freedom that capture the relevant interactions between scales.

Adaptivity and nonlinearities. The algorithmic framework proposed above is general and can be used for different multiscale, high contrast, and perforated problems [22, 26, 43, 8]. The multiscale basis functions are constructed locally and using an adaptive criteria. Thus, in different regions, we expect a different number of basis functions depending on the local features of the problem, such as heterogeneities and high contrast. For example, in the regions with scale separation, we expect only a limited number of degrees of freedom. The adaptivity will be achieved using error indicators. In nonlinear problems, one performs local nonlinear interpolation to approximate the Jacobian or other nonlinear terms. The above multiscale procedure can be complemented with online basis functions which helps to converge to the fine-scale solution by constructing multiscale basis functions in the simulations [23, 24] (also, see Figure 2 for the ingredients of proposed multiscale method).

Limitations. Though the proposed approaches can be used in many applications, it is limited to problems where the solution space locally has a low dimensional structure. If the latter is not the case, our approaches will use many degrees of freedom and the computational gain may not be significant in

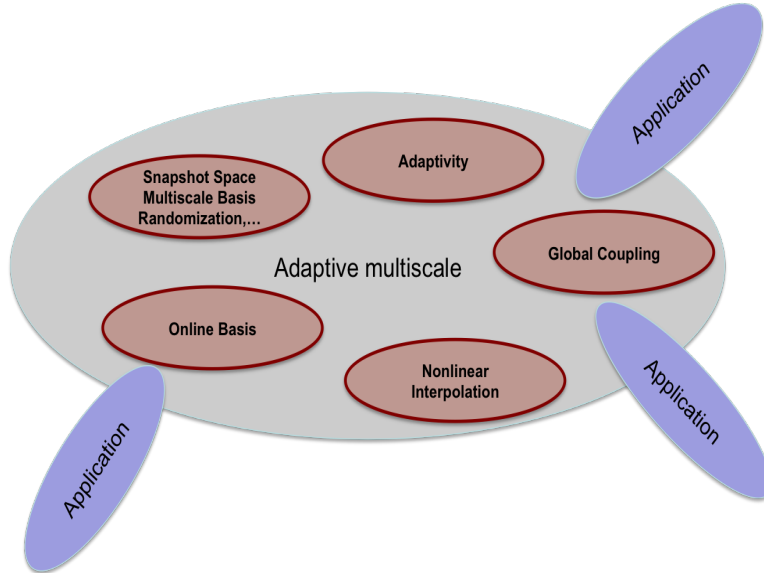


Figure 2: Multiscale model reduction. Ingredients and Applications.

those regions. Many multiscale application problems we have encountered can benefit from local adaptive multiscale model reduction.

1.3.1 Methodological ingredients of GMsFEM

Our computational framework will rely on several important ingredients (see Figure 2) that are general for various discretizations (such as mixed methods, discontinuous Galerkin, and etc.) and applications. These ingredients include (1) a procedure for identifying local snapshot spaces and multiscale basis, (2) developing global coupling mechanisms for multiscale basis functions, (3) adaptivity strategies, (4) local nonlinear interpolation tools, and (5) online basis functions. These ingredients are building blocks that are needed for constructing an accurate and robust multiscale model reduction. We will motivate the use of multiscale model reduction by showing how it can take us beyond conventional macroscopic modeling and discuss some relevant ingredients.

The proposed framework provides a promising tool and capability for solving a large class of multiscale problems without scale separation and high contrast. This framework is tested and applied in various applications where we have followed the general framework to construct local multiscale spaces. We identify main ingredients of the framework, which can be further investigated for speed-up and accuracy.

1.4 Applications and organization of the paper

We apply the proposed adaptive multiscale framework to several challenging and distinct applications where one deals with a rich hierarchy of scales. These include (1) flows in heterogeneous porous media (2) diffusion in fractured media (3) multiscale processes in perforated regions (4) wave propagation in heterogeneous media (5) nonlinear diffusion equations in heterogeneous media. We will discuss some of the applications here.

In Section 2, we describe numerical homogenization and its limitations. In Section 3, we discuss Multiscale Finite Element Method. In Section 4, we describe the GMsFEM and its basic principles. In Section 5, adaptive strategies for the GMsFEM are described. Section 6 is devoted to the construction of the online basis functions. We describe the selected global couplings in Section 7. The GMsFEM using sparsity in the snapshot space is discussed in Section 8. We discuss the GMsFEM for space-time heterogeneous problems

in Section 9. In Section 10, we discuss the GMSFEM for problems in perforated domains. We present some selected applications in Section 11. In Section 12, we discuss some remaining aspects of the GMSFEM, such as applications to nonlinear problems and its use in global model reduction techniques.

2 A brief introduction to numerical homogenization

2.1 Numerical homogenization

The *main idea of numerical homogenization* is to identify the homogenized coefficients in each coarse-grid block. The basic underlying principle is to compute these upscaled quantities such that they preserve some averages for a given set of local boundary conditions. We discuss it on an example.

We consider a simple example

$$\frac{\partial}{\partial x_i} \left(\kappa_{ij}(x) \frac{\partial}{\partial x_j} u \right) = f, \quad (2)$$

where $\kappa_{ij}(x)$ is a heterogeneous field. The objective is to define an upscaled (or homogenized) conductivity for each coarse block without directly using the periodicity. Following the homogenization technique, the local problems are solved in each coarse block

$$\frac{\partial}{\partial x_i} \left(\kappa_{ij}(x) \frac{\partial}{\partial x_j} \mathcal{N}_l \right) = 0 \text{ in } K, \quad (3)$$

where K is a coarse block. The boundary condition needs to be chosen such that it represents the local heterogeneities. We limit ourselves to Dirichlet boundary conditions $\mathcal{N}_l = x_l$ on ∂K . One can use other boundary conditions [68].

We note that if the local problem is homogenized and the homogenized coefficients are constant, then the solution of the homogenized equation is $\mathcal{N}_l^* = x_l$ in K . The upscaled coefficients $\kappa_{ij}^{*,nh}$ can be defined by averaging the fluxes:

$$\int_K \kappa_{ij}^{*,nh} \frac{\partial}{\partial x_j} \mathcal{N}_l^* = \int_K \kappa_{ij}(x) \frac{\partial}{\partial x_j} \mathcal{N}_l. \quad (4)$$

The motivation behind this upscaling is to state that the average flux response for the fine-scale local problem with prescribed boundary conditions is the same as that for the upscaled solution. If we take $\mathcal{N}_l^* = x_l$ in K , we have

$$\kappa_{il}^{*,nh} = \frac{1}{|K|} \int_K \kappa_{ij}(x) \frac{\partial}{\partial x_j} \mathcal{N}_l. \quad (5)$$

Remark 1 (Convergence). *Assuming that $\kappa_{ij}(x) = \kappa_{ij}(x/\epsilon)$, one can study the convergence of the numerical homogenization technique and show ([68]) that*

$$|\kappa_{il}^{*,nh} - \kappa_{il}^*| \leq C \sqrt{\frac{\epsilon}{H}},$$

where $\kappa_{il}^{*,nh}$ is the numerical homogenization and κ_{il}^* is the “correct” homogenized coefficients.

Remark 2 (Oversampling). *The convergence analysis suggests that the boundary layers due to artificial linear boundary conditions, $\mathcal{N}_l = x_l$, cause the resonance errors. To reduce these resonance errors, oversampling technique has been proposed. The main idea of this method is to solve local problems in a larger domain. In oversampling method [50], the local problem that is analogous to (3) is solved in a larger domain (see Figure 1). In particular, if we denote the large domain by K^+ while the target coarse block by K (see Figure 1), then*

$$\frac{\partial}{\partial x_i} \left(\kappa_{ij}(x) \frac{\partial}{\partial x_j} \mathcal{N}_l^{ovs} \right) = 0 \text{ in } K^+. \quad (6)$$

For simplicity, we can use $\mathcal{N}_l^{ovs} = x_l$ on ∂K^+ . The upscaled conductivity $\kappa_{ij}^{*,ovs}$ is computed by equating the fluxes on the target coarse block:

$$\int_K \kappa_{ij}^{*,ovs} \frac{\partial}{\partial x_j} \mathcal{N}_l^{*,ovs} = \int_K \kappa_{ij}(x) \frac{\partial}{\partial x_j} \mathcal{N}_l^{ovs}. \quad (7)$$

Noting that the domain K^+ is only slightly larger, we can still claim that $\mathcal{N}_l^{*,ovs} = x_l$, and thus,

$$\kappa_{il}^{*,ovs} = \frac{1}{|K|} \int_K \kappa_{ij}(x) \frac{\partial}{\partial x_j} \mathcal{N}_l^{ovs}. \quad (8)$$

The advantage of this approach is that one reduces the effects of the oscillatory boundary conditions. One can show ([68]) that the error for the residual is small and scales as ϵ/H instead of $\sqrt{\epsilon/H}$.

2.2 Limitations

The above numerical homogenization concepts (for linear and nonlinear problems) are often used to derive macroscopic equations. These techniques assume that the solution space has a reduced dimensional structure. For example, as we can observe from the above example and derivations, the solution in each coarse block is approximated by three local fields (in 3D). In fact, this “effective” dimension is related to the number of elements (constants) in the effective properties. In general, for many macroscopic equations, one implicitly assumes the local effective dimension for the solution space related to the number of macroscopic parameters. In many cases, the assumptions on the limited effective dimension of microscale problem break down and the local solution space may need more degrees of freedom. This requires general approaches, where we do not rely on macroscale equations, and approximate the solution space via multiscale basis functions. Next, we give a brief overview of these concepts.

3 Multiscale Finite Element Method

In this section, we will give a brief overview of MsFEM as a method for solving a problem on a coarse grid. MsFEMs consist of two major ingredients: (1) multiscale basis functions and (2) a global numerical formulation which couples these multiscale basis functions. Multiscale basis functions are designed to capture the fine-scale features of the solution. Important multiscale features of the solution need to be incorporated into these localized basis functions which contain information about the scales which are smaller (as well as larger) than the local numerical scale defined by the basis functions. In particular, we need to incorporate the features of the solution that can be localized and use additional basis functions to capture the information about the features that need to be separately included in the coarse space. A global formulation couples these basis functions to provide an accurate approximation of the solution.

As before, we consider the second order elliptic equations with heterogeneous coefficients

$$\mathcal{L}(u) = f, \quad (9)$$

where $\mathcal{L}(u) = -\frac{\partial}{\partial x_i} \left(\kappa_{ij}(x) \frac{\partial}{\partial x_j} u \right)$ with appropriate boundary conditions. We seek multiscale basis functions supported in each domain ω_j , denoted by ϕ^{ω_j} . Then, the coarse-grid solution is represented by

$$u_H = \sum_i c_i \phi^{\omega_i},$$

where c_i are determined from

$$a(u_H, v_H) = (f, v_H), \quad \text{for all } v_H \in V_0. \quad (10)$$

In the above formulation, we define $V_0 = \text{span}\{\phi^{\omega_i}\}$,

$$a(u, v) = \int_{\Omega} \kappa(x) \nabla u(x) \nabla v(x) \quad \text{and} \quad (f, v) = \int_{\Omega} f(x) v(x). \quad (11)$$

One can also view MsFEM in the discrete setting. Assume that the basis functions are defined on a fine grid as Φ^{ω_i} with i varying from 1 to N_c , where N_c is the number of multiscale basis functions. Given coarse-scale basis functions, the coarse space is given by

$$V_0 = \text{span}\{\Phi^{\omega_i}\}_{i=1}^{N_c}, \quad (12)$$

and the coarse matrix is given by $A_H = R_0^T A R_0$ where A is the fine-scale stiffness matrix and

$$R_0 = [\Phi^{\omega_1}, \dots, \Phi^{\omega_{N_c}}].$$

Here Φ^{ω_i} 's are discrete coarse-scale basis functions defined on a fine grid (i.e., column vectors). Multiscale finite element solution is the finite element projection of the fine-scale solution into the space V_0 . More precisely, multiscale solution U_H is given by

$$A_H u_H = f_H,$$

where $f_H = R_0^T b$. Next, we discuss some coarse spaces.

Linear boundary conditions. Let χ_i^0 be the standard piecewise linear or piecewise polynomial basis function supported in ω_i . We define multiscale finite element basis functions that coincide with χ_i^0 on the boundaries of the coarse partition. In particular,

$$\text{div}(\kappa \nabla \chi_i^{ms}) = 0 \text{ in } K \in \omega_i, \quad \chi_i^{ms} = \chi_i^0 \text{ in } \partial K, \quad \forall K \subset \omega_i, \quad (13)$$

where K is a coarse grid block within ω_i . Note that multiscale basis functions coincide with standard finite element basis functions on the boundaries of coarse grid blocks, while are oscillatory in the interior of each coarse grid block.

Remark 3. *We would like to remark that the MsFEM formulation allows one to take advantage of scale separation. In particular, K can be chosen to be a volume smaller than the coarse grid and the integrals in the stiffness matrix computations need to be re-scaled (see [37] for discussions).*

Remark 4 (The relation between MsFEM and numerical homogenization). *It can be shown ([37]) that the MsFEM with one basis function on triangular elements yields the same coarse-grid stiffness matrix as the numerical homogenization. This is due to the fact that the numerical homogenization uses the local solutions of PDEs as in the MsFEM. However, the MsFEM has several advantages, which include fine-scale information recovery, adaptivity based on the residual, the use of global information and so on ([37]).*

Remark 5 (The relation between MsFEM and some other multiscale techniques). *The relation between the MsFEM and other multiscale methods is discussed in [37]. It can be shown that the MsFEM can use an approximation of multiscale basis functions if there is a periodicity. As a result, the MsFEM can recover a similar approximation and the computational cost as Heterogeneous Multiscale Method ([29]) for elliptic equations in the presence of scale separation. The relation of the MsFEM and variational multiscale method ([53]) is also discussed in [37]. The variational multiscale method (VMM) recovers the fine-scale information via the equation for the residual. The localization of this equation and the choice of the initial multiscale spaces are important for the VMM. It can be shown that with a simple choice, one can recover the MsFEM. On the other hand, one can use more sophisticated coarse spaces and the residual recovery, to improve the accuracy of the VMM [10].*

Oversampling technique. Because of linear boundary conditions, the basis functions do not capture the fine-scale features of the solution along the boundaries. This can lead to large errors. When coefficients have a single physical scale ϵ , it has been shown that the error (see [52]) is proportional to ϵ/H , and thus can be large when H is close to ϵ . Motivated by such examples, Hou and Wu in [50] proposed an oversampling technique for multiscale finite element method. Specifically, let ψ^{+, ω_i} be the basis functions satisfying the homogeneous elliptic equation in the larger domain $K^+ \supset K$ (see Figure 1). We then form the actual basis ϕ^{+, ω_i} by linear combination of ψ^{+, ω_i} , $\phi^{+, \omega_i} = \sum_j \alpha_{ij} \psi^{+, \omega_j}$, and restricting them to K . The coefficients

α_{ij} are determined by condition $\phi^{+, \omega_i}(x_j) = \delta_{ij}$, where x_j are nodal points. Other conditions can also be imposed (e.g., α_{ij} are determined based on homogenized parts of ψ^{+, ω_i}). Note that this method is non-conforming. One can also multiply the oversampling functions by linear basis functions to restrict them onto ω_i and have a conforming method. Numerical results and more discussions on oversampling methods can be found in [37]. Many other boundary conditions are introduced and analyzed in the literature. For example, reduced boundary conditions are found to be efficient in many porous media applications ([54]).

The use of limited global information. Previously, we discussed multiscale methods which employ local information in computing basis functions with the exception of energy minimizing basis functions. The accuracy of these approaches depends on local boundary conditions. Though effective in many cases, multiscale methods that only use local information may not accurately capture the local features of the solution. In a number of previous papers, multiscale methods that employ limited global information is introduced. The main idea of these multiscale methods is to incorporate some fine-scale information about the solution that can not be computed locally and that is given globally. More precisely, in these approaches, we assume that the solution can be represented by a number of fields p_1, \dots, p_N , such that

$$u \approx G(p_1, \dots, p_N), \quad (14)$$

where G is sufficiently smooth function, and p_1, \dots, p_N are global fields. These fields typically contain the essential information about the heterogeneities at different scales and can also be local fields defined in larger domains. In the above assumption (14), p_i are solutions of elliptic equations. These global fields are used to construct multiscale basis functions (often multiple basis functions per a coarse node). Finding p_1, \dots, p_N , in general, can be a difficult task and we refer to Owhadi and Zhang [62] as well as to [36] where various choices of global information are proposed.

4 Generalized Multiscale Finite Element Method. Basic Concepts

4.1 Overview

Previous multiscale finite element concepts focus on constructing one basis function per node. As we discussed earlier these approaches are similar to numerical homogenization (upscaling), i.e., one can use one basis function to localize the effects of local heterogeneities. However, for many complex heterogeneities, multiple basis functions are needed to represent the local solution space. For example, if the coarse region contains several high-conductivity regions, one needs multiple multiscale basis functions to represent the local solution space. In this section, we discuss how to systematically construct multiscale basis functions in a general framework Generalized Multiscale Finite Element Methods.

Generalized Multiscale Finite Element Method (GMsFEM) incorporates complex input space information and the input-output relation. It systematically enriches the coarse space through our local construction. Our approach, as in many multiscale and model reduction techniques, divides the computation into two stages (see Figure 3): offline and online. In the offline stage, we construct a small dimensional space that can be efficiently used in the online stage to construct multiscale basis functions. These multiscale basis functions can be re-used for any input parameter to solve the problem on a coarse-grid. Thus, this provides a substantial computational saving in the online stage. Below, we present an outline of the algorithm and will discuss its main ideas on the example of (9).

Offline computations: (1) Coarse grid generation; (2) Construction of snapshot space that will be used to compute an offline space. (3) Construction of a small dimensional offline space by performing dimension reduction in the space of snapshots.

Online computations: (1) For each input parameter, compute multiscale basis functions (for parameter dependent problems); (2) Solution of a coarse-grid problem for any force term and boundary condition; (3) Iterative solvers, if needed.

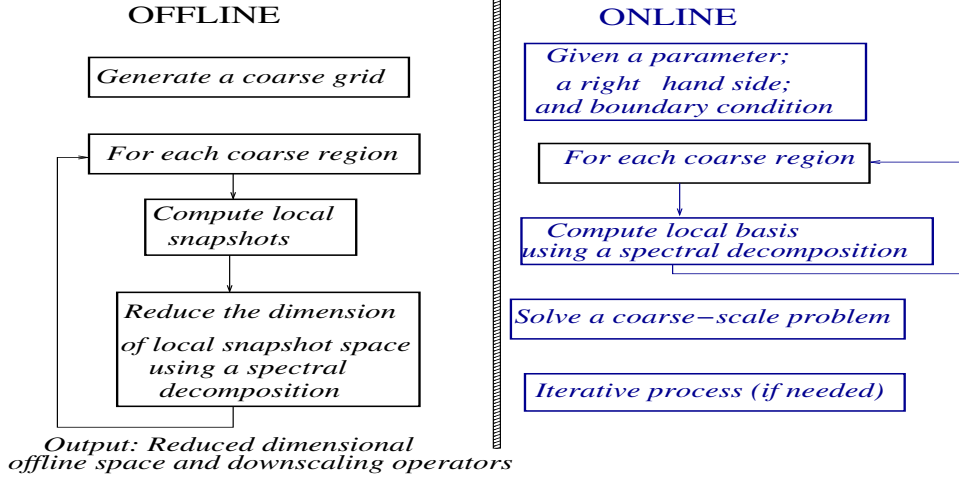


Figure 3: Flow chart

4.2 Examples of snapshot spaces. Oversampling and non-oversampling

The snapshot space, denoted by $V_{H,\text{snap}}^{\omega_i}$ for a generic domain ω_i , is defined for each coarse region ω_i (with elements of the space denoted $\psi_l^{\omega_i}$) and can approximate the solution space with a prescribed accuracy. The appropriate snapshot space (1) yields faster convergence, (2) imposes problem relevant restrictions on the coarse spaces (e.g., divergence free solutions) and (3) reduces the computational cost of building the offline spaces. In particular, we emphasize that the use of oversampling in the snapshot spaces can improve the convergence.

4.2.1 All fine-grid functions

We can use local fine-scale spaces consisting of fine-grid basis functions within a coarse region. More precisely, the snapshot vectors consists of unit vectors defined on a fine grid within a coarse region. In this case, the offline spaces will be computed on a fine grid directly ([31]).

4.2.2 Harmonic extensions

This choice of snapshot space consists of harmonic extension of fine-grid functions defined on the boundary of ω_i . More precisely, for each fine-grid function, $\delta_l^h(x)$, which is defined by $\delta_l^h(x_k) = \delta_{l,k}$, $\forall x_k \in J_h(\omega_i)$, where $J_h(\omega_i)$ denotes the set of fine-grid boundary nodes on $\partial\omega_i$, we obtain a snapshot function $\psi_l^{\omega_i}$ by

$$\mathcal{L}(\psi_l^{\omega_i}) = 0 \quad \text{in } \omega_i$$

subject to the boundary condition, $\psi_l^{\omega_i} = \delta_l^h(x)$ on $\partial\omega_i$. Here $\delta_{l,k} = 1$ if $l = k$ and $\delta_{l,k} = 0$ if $l \neq k$.

4.2.3 Oversampling approaches

To describe the oversampling approach, we consider the harmonic extension as described above in the oversampled region (see, e.g., Figure 1). This choice of snapshot space consists of harmonic extension of fine-grid functions defined on the boundary of ω_i^+ . More precisely, for each fine-grid function, $\delta_l^h(x)$, which is defined by $\delta_l^h(x_k) = \delta_{l,k}$, $\forall x_k \in J_h(\omega_i^+)$, where $J_h(\omega_i^+)$ denotes the set of fine-grid boundary nodes on $\partial\omega_i^+$, we obtain a snapshot function ψ_l^{+, ω_i} by

$$\mathcal{L}(\psi_l^{+, \omega_i}) = 0 \quad \text{in } \omega_i^+$$

subject to the boundary condition, $\psi_l^{+, \omega_i} = \delta_l^h(x)$ on $\partial\omega_i^+$.

4.2.4 Randomized boundary conditions

In the above construction of snapshot vectors, many local problems are solved. This is not necessary and one can only solve a relatively small number of snapshot vectors. The number of snapshot vectors is defined by the number of offline multiscale basis functions. In this case, we will use random boundary conditions.

More precisely, for each fine-grid function, $r_l^h(x)$, which is defined by $r_l^h(x_k) = r_{l,k}$, $\forall x_k \in J_h(\omega_i^+)$, where $r_{l,k}$ are random numbers, we obtain a snapshot function ψ_l^{+,ω_i} by

$$\mathcal{L}(\psi_l^{+,\omega_i}) = 0 \quad \text{in } \omega_i^+$$

subject to the boundary condition, $\psi_l^{+,\text{snap}} = r_l^h(x)$ on $\partial\omega_i^+$.

4.3 Offline spaces

The offline space, denoted by $V_{H,\text{off}}^{\omega_i}$ for a generic domain ω_i , is defined for each coarse region ω_i (with elements of the space denoted $\phi_l^{\omega_i}$) and used to approximate the solution. The offline space is constructed by performing a spectral decomposition in the snapshot space and selecting the dominant eigenvectors (corresponding to the smallest eigenvalues). The choice of the spectral problem is important for the convergence and is derived from the analysis as it is described below. The convergence rate of the method is proportional to $1/\Lambda_*$, where Λ_* is the smallest eigenvalue among all coarse blocks whose corresponding eigenvector is not included in the offline space. Our goal is to select the local spectral problem to remove as many small eigenvalues as possible so that we can obtain smaller dimensional coarse spaces to achieve a higher accuracy.

4.3.1 General concept and example

The construction of the offline space requires solving an appropriate local spectral eigenvalue problem. The local spectral problem is derived from the analysis. A first step in the analysis is to decompose the energy functional corresponding to the error into coarse subdomains. For simplicity, we denote the energy functional corresponding to the domain Ω by $a_\Omega(u, u)$, e.g., $a_\Omega(u, u) = \int_\Omega \kappa \nabla u \cdot \nabla u$. Then,

$$a_\Omega(u - u_H, u - u_H) \preceq \sum_{\omega} a_\omega(u^\omega - u_H^\omega, u^\omega - u_H^\omega), \quad (15)$$

where ω are coarse regions (ω_i), u^ω is the localization of the solution using the snapshot vectors defined for ω and u_H^ω is the component of the solution u_H spanned by the basis in ω .

The local spectral decomposition is chosen to bound $a_\omega(u^\omega - u_H^\omega, u^\omega - u_H^\omega)$. We seek the subspace $V_{H,\text{off}}^\omega$ (since $u^\omega \in V_{H,\text{snap}}^\omega$) such that for any $\psi \in V_{H,\text{snap}}^\omega$, there exists $\psi_0 \in V_{H,\text{off}}^\omega$ with,

$$a_\omega(\psi - \psi_0, \psi - \psi_0) \preceq \delta s_\omega(\psi - \psi_0, \psi - \psi_0), \quad (16)$$

where $s_\omega(\cdot, \cdot)$ is an auxiliary bilinear form, which needs to be chosen and δ is a threshold related to eigenvalues. First, we would like the fine-scale solution to be bounded in $s_\omega(\cdot, \cdot)$. We note that, in computations, (16) involves solving a generalized eigenvalue problem with a mass matrix defined using $s_\omega(\cdot, \cdot)$ and the basis functions are selected based on dominant eigenvalues as described above. The threshold value δ is chosen based on the eigenvalue distribution. Secondly, we would like the eigenvalue problem to have a fast decay in the spectrum (this typically requires using oversampling techniques). Thirdly, we would like to bound

$$\sum_{\omega} s_\omega(u^\omega - u_H^\omega, u^\omega - u_H^\omega) \preceq \tilde{a}_\Omega(u, u),$$

where $\tilde{a}_\Omega(u, u)$ can be bounded independent of physical parameters and the mesh sizes. In this step, one can need energy minimizing snapshots (See Remark 8 and [18]).

Next, we discuss a choice for $a_\omega(\cdot, \cdot)$ and $s_\omega(\cdot, \cdot)$. Recall that for (16), we need a local spectral problem, which is to find a real number λ and $v \in V_{H,\text{snap}}^{\omega_i}$ such that

$$a_{\omega_i}(v, w) = \lambda s_{\omega_i}(v, w), \quad \forall w \in V_{H,\text{snap}}^{\omega_i}, \quad (17)$$

where a_{ω_i} is a symmetric non-negative definite bilinear operator and s_{ω_i} is a symmetric positive definite bilinear operators defined on $V_{H,\text{snap}}^{\omega_i} \times V_{H,\text{snap}}^{\omega_i}$. Based on our analysis, we can choose

$$a_{\omega_i}(v, w) = \int_{\omega_i} \kappa \nabla v \cdot \nabla w, \quad s_{\omega_i}(v, w) = \int_{\omega_i} \tilde{\kappa} v w,$$

where $\tilde{\kappa} = \sum_{i=1}^{N_c} \kappa \nabla \chi_i^{ms} \cdot \nabla \chi_i^{ms}$ and χ_i^{ms} are multiscale basis functions (see (13)). We let $\lambda_j^{\omega_i}$ be the eigenvalues of (17) arranged in ascending order. We will use the first l_i eigenfunctions to construct our offline space $V_{H,\text{off}}^{\omega_i}$. The choice of the eigenvalue problem is motivated by the convergence analysis. The convergence rate is proportional to $1/\Lambda_*$ (with the proportionality constant depending on H), where Λ_* is the smallest eigenvalue that the corresponding eigenvector is not included in the offline space. We would like to remove as many small eigenvalues as possible. The eigenvectors of the corresponding small eigenvalues represent important features of the solution space. By choosing the multiscale basis functions, χ_i^{ms} , in the construction of local spectral problem, some localizable important features are taken into consideration through χ_i^{ms} and, as a result, we have fewer small eigenvalues (see [31] for more discussions).

For constructing conforming multiscale basis functions, the selected eigenfunctions are multiplied by the partition of unity functions, such as χ_i^{ms} or χ_i^0 (cf., [60]). The multiplication by the partition of unity functions modifies the multiscale nature of the selected multiscale eigenfunctions. We will discuss some other important discretizations such as mixed, discontinuous Galerkin discretizations, which can be more suitable for the coupling of the multiscale eigenfunctions and which avoid the multiplication by the partition of unity functions. The global offline space $V_{H,\text{off}}$ is formed as the union of all $V_{H,\text{off}}^{\omega_i}$. Once the offline space is constructed, we solve (10) and find $u_H = \sum c_{i,j} \phi_j^{\omega_i} \in V_{H,\text{off}}$ such that

$$a(u_H, v_H) = (f, v_H), \quad \forall v_H \in V_{H,\text{off}}. \quad (18)$$

4.3.2 An implementation view

Next, we present an implementation view of the local spectral decomposition. The space formed by the snapshot vectors is

$$V_{H,\text{snap}}^{\omega_i} = \text{span}\{\Psi_l^{\omega_i} : 1 \leq l \leq L_i\}, \quad V_{H,\text{snap}}^{+,\omega_i} = \text{span}\{\Psi_l^{+,\omega_i} : 1 \leq l \leq L_i^+\}$$

for each coarse neighborhood ω_i and for each oversampled coarse neighborhood ω_i^+ , respectively, where L_i and L_i^+ are the dimensions of the snapshot spaces $V_{H,\text{snap}}^{\omega_i}$ and $V_{H,\text{snap}}^{+,\omega_i}$. We note that in the case when ω_i is adjacent to the global boundary, no oversampled domain is used. We can put all snapshot functions using a matrix representation

$$R_{\text{snap}}^{\omega_i} = [\Psi_1^{\omega_i}, \dots, \Psi_{L_i}^{\omega_i}], \quad R_{\text{snap}}^{+,\omega_i} = [\Psi_1^{+,\omega_i}, \dots, \Psi_{L_i^+}^{+,\omega_i}],$$

The local spectral problems (17) can be written in a matrix form as

$$A^{\omega_i} \Psi_k^{\omega_i} = \lambda_k^{\omega_i} S^{\omega_i} \Psi_k^{\omega_i}, \quad A^{+,\omega_i} \Psi_k^{+,\omega_i} = \lambda_k^{+,\omega_i} S^{+,\omega_i} \Psi_k^{+,\omega_i} \quad (19)$$

for the k -th eigenpair. We present two choices: one with no oversampling and one with oversampling, though one can consider various options [35]. In (19), with no oversampling, we can choose $A^{\omega_i} = [a_{mn}^{\omega_i}]$ and $S^{\omega_i} = [s_{mn}^{\omega_i}]$, where $a_{mn}^{\omega_i} = \int_{\omega_i} \kappa \nabla \psi_m^{\omega_i} \cdot \nabla \psi_n^{\omega_i} = (R_{\text{snap}}^{\omega_i})^T A R_{\text{snap}}^{\omega_i}$ and $s_{mn}^{\omega_i} = \int_{\omega_i} \tilde{\kappa} \psi_m^{\omega_i} \psi_n^{\omega_i} = (R_{\text{snap}}^{\omega_i})^T S R_{\text{snap}}^{\omega_i}$. With oversampling, we can choose $A^{+,\omega_i} = [a_{mn}^{+,\omega_i}]$ and $S^{+,\omega_i} = [s_{mn}^{+,\omega_i}]$, where $a_{mn}^{+,\omega_i} = \int_{\omega_i^+} \kappa \nabla \psi_m^{+,\omega_i} \cdot \nabla \psi_n^{+,\omega_i} = (R_{\text{snap}}^{+,\omega_i})^T A R_{\text{snap}}^{+,\omega_i}$ and $s_{mn}^{+,\omega_i} = \int_{\omega_i^+} \tilde{\kappa} \psi_m^{+,\omega_i} \psi_n^{+,\omega_i} = (R_{\text{snap}}^{+,\omega_i})^T S R_{\text{snap}}^{+,\omega_i}$. To generate

the offline space we then choose the smallest l_i eigenvalues and form the corresponding eigenvectors in the respective space of snapshots by setting $\phi_k^{\omega_i} = \sum_{j=1}^{L_i} \Psi_{kj}^{\omega_i} \psi_j^{\omega_i}$ or $\phi_k^{+, \omega_i} = \sum_{j=1}^{L_i^+} \Psi_{kj}^{+, \omega_i} \psi_j^{+, \omega_i}$ (for $k = 1, \dots, L_i$ or $k = 1, \dots, L_i^+$), where $\Psi_{kj}^{\omega_i}$ and Ψ_{kj}^{+, ω_i} are the coordinates of the vector $\Psi_k^{\omega_i}$ and Ψ_k^{+, ω_i} respectively. Collecting all offline basis functions and using a single index notation, we then create the offline matrices $R_{\text{off}}^+ = [\psi_1^{+, \text{off}}, \dots, \psi_{M_{\text{off}}}^{+, \text{off}}]$ and $R_{\text{off}} = [\psi_1^{\text{off}}, \dots, \psi_{M_{\text{off}}}^{\text{off}}]$ to be used in the online space construction, where M_{off} is the total number of offline basis functions. The discrete system corresponding to (18) is

$$R_{\text{off}}^T A R_{\text{off}} u_H^{\text{discrete}} = R_{\text{off}}^T f$$

where u_H^{discrete} is the discrete version of u_H .

4.4 A numerical example

We present a numerical result that demonstrate the convergence of the GMsFEM. More detailed numerical studies can be found in the literature. We consider the permeability field, $\kappa(x)$ (see (9)) that is shown in Figure 4, the source term $f = 0$, and the boundary condition to be x_1 . The fine grid is 100×100 , coarse grid is 10×10 . We consider the snapshot space spanned by harmonic functions in the oversampled domain (with randomized boundary conditions) and vary the number of basis functions per node. The numerical results are shown in Table 1 in which e_a and e_2 denote errors in energy and L^2 norms respectively. As we observe from these numerical results the GMsFEM converges as we increase the number of basis functions. We also show the convergence when using polynomial basis functions. It is clear that the FEM method with polynomial basis functions does not converge and one can only observe this convergence only after a very large number of basis functions are chosen so that we cross the fine-scale threshold. We note that the convergence with one basis function per node does not perform well for the GMsFEM because of the high contrast. The first, second, and third smallest eigenvalues, Λ_* , among all coarse blocks are 0.0024, 24.0071, 35.6941, respectively. We emphasize that if the eigenvalue problem (and the snapshot space) is not chosen appropriately, one can get more very small eigenvalues ([31]). As we see that the first smallest eigenvalue is very small and, as a result, the error is large, when using one basis function. The eigenvalue distribution is important for online basis construction, as will be discussed later.

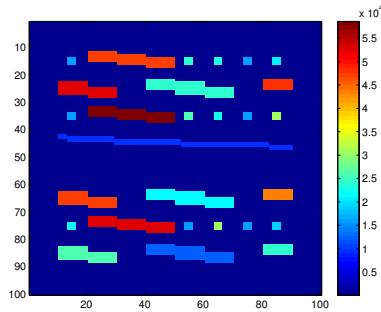


Figure 4: The permeability field $\kappa(x)$.

5 Adaptivity in GMsFEM

The success of the GMsFEM depends on adaptive implementation and appropriate error indicators. In this section, we discuss an a-posteriori error indicator for the GMsFEM framework. We will demonstrate the main idea using a continuous Galerkin formulation; however, this concept can be generalized to other discretizations as we will discuss later on. This error indicator is further used to develop an adaptive

| #basis (DOF) | e_a | e_2 | DOF | e_a | e_2 |
|--------------|--------|--------|-----|--------|-------|
| 1 (81) | 69.05% | 12.19% | 81 | 103.2% | 23% |
| 2 (162) | 22.55% | 1.19% | 361 | 100% | 23% |
| 3 (243) | 19.86% | 0.99% | 841 | 80.33% | 15% |
| 4 (324) | 16.31% | 0.70% | | | |
| 5 (405) | 14.20% | 0.65% | | | |

Table 1: Left table: The convergence for the GMsFEM using multiscale basis functions. Right table: The convergence for FEM using polynomials basis functions.

enrichment algorithm for the linear elliptic equation with multiscale high-contrast coefficients. Rigorous a-posteriori error indicators are needed to perform an adaptive enrichment. We would like to point out that there are many related activities in designing a-posteriori error estimates for global reduced models. The main difference is that the error estimators presented in this section are based on a special local eigenvalue problem and use the eigenstructure of the offline space.

We can consider various kinds of error indicators that are based on the L^2 -norm of the local residual and the other is based on the weighted H^{-1} -norm (we will also call it H^{-1} -norm based) of the local residual where the weight is related to the coefficient of the elliptic equation. The latter will be studied in the paper and we refer to [25] for more details.

Let $u_H \in V_{H,\text{off}}$ be the solution obtained in (18). Consider a given coarse neighborhood ω_i . We define a space $V_i = H_0^1(\omega_i) \cap V$ which is equipped with the norm $\|v\|_{V_i}^2 = \int_{\omega_i} \kappa(x) |\nabla v|^2$. We also define the following linear functional on V_i by

$$R_i(v) = \int_{\omega_i} f v - \int_{\omega_i} \kappa \nabla u_H \cdot \nabla v. \quad (20)$$

This is called the H^{-1} -residual on ω_i , The functional norm of R_i , denoted by $\|R_i\|_{V_i^*}$, gives a measure of the size of the residual. The first important result states that these residuals give a computable indicator of the error $u - u_H$ in the energy norm. We have

$$\|u - u_H\|_V^2 \leq C_{\text{err}} \sum_{i=1}^{N_c} \|R_i\|_{V_i^*}^2 (\lambda_{l_i+1}^{\omega_i})^{-1}, \quad (21)$$

where C_{err} is a uniform constant, and $\lambda_{l_i+1}^{\omega_i}$ denotes the $(l_i + 1)$ -th eigenvalue for the problem (19) in the coarse neighborhood ω_i , and corresponds to the first eigenvector that is not included in the construction of $V_{H,\text{off}}^{\omega_i}$.

The above error indicator allows one to construct an adaptive enrichment algorithm. It is an iterative process, and basis functions are added in each iteration based on the current solution. We use the index $m \geq 1$ to represent the enrichment level. At the enrichment level m , we use $V_{H,\text{off}}^m$ to denote the corresponding GMsFEM space and u_H^m the corresponding solution obtained in (18) using the space $V_{H,\text{off}}^m$. Furthermore, we use l_i^m to denote the number of basis functions used in the coarse neighborhood ω_i . We will present the strategy for getting the space $V_{H,\text{off}}^{m+1}$ from $V_{H,\text{off}}^m$. Let $0 < \theta < 1$ be a given number independent of m . First of all, we compute the local residuals for every coarse neighborhood ω_i :

$$\eta_i^2 = \|R_i\|_{V_i^*}^2 (\lambda_{l_i^m+1}^{\omega_i})^{-1}, \quad i = 1, 2, \dots, N_c,$$

where $R_i(v)$ is defined using u_H^m , namely,

$$R_i(v) = \int_{\omega_i} f v - \int_{\omega_i} \kappa \nabla u_H^m \cdot \nabla v, \quad \forall v \in V_i.$$

Next, we will add basis functions for the coarse neighborhoods with large residuals. To do so, we re-enumerate the coarse neighborhoods so that the above local residuals η_i^2 are arranged in decreasing order

$\eta_1^2 \geq \eta_2^2 \geq \dots \geq \eta_N^2$. We then select the smallest integer k such that

$$\theta \sum_{i=1}^{N_c} \eta_i^2 \leq \sum_{i=1}^k \eta_i^2. \quad (22)$$

For those coarse neighborhoods $\omega_1, \dots, \omega_k$ (in the new enumeration) chosen in the above procedure, we will add basis functions by using the next eigenfunctions $\Psi_{l_i+1}^{\omega_i}, \Psi_{l_i+2}^{\omega_i}, \dots$. We remark that the number of new additional basis depends on the eigenvalue decay. The resulting space is called V_{off}^{m+1} . We remark that the choice of k defined in (22) is called the Dorfler's bulk marking strategy [27].

As we mentioned the local basis functions do not contain any global information, and thus they cannot be used to efficiently capture these global behaviors. We will therefore present in the next section a fundamental ingredient of GMsFEM - the development of online basis functions that is necessary to obtain a coarse representation of the fine-scale solution and gives a rapid convergence of the corresponding adaptive enrichment algorithm.

Remark 6 (Implementation). *The algorithm above can be described as follows. We start with an initial space with a small number of basis functions for each coarse grid block. Then we solve the problem and compute the error estimator. We locate the coarse grid blocks with large errors and add more basis functions for these coarse grid blocks. This procedure is repeated until the error goes below a certain tolerance. We remark that the adaptive strategy belongs to the online process, because it is the actual simulation. On the other hand, the generation of basis functions belongs to the offline process. About stopping criteria for this algorithm, one can stop the algorithm when the total number of basis functions reach a certain level. On the other hand, one can stop the algorithm when the value of the error indicator goes below a certain tolerance.*

Remark 7 (Goal Oriented Adaptivity). *For some practical problems, one is interested in approximating some function of the solution, known as the quantity of interest, rather than the solution itself. Examples include an average or weighted average of the solution over a particular subdomain, or some localized solution response. In these cases, goal-oriented adaptive methods yield a more efficient approximation than standard adaptivity, as the enrichment of degrees of freedom is focused on the local improvement of the quantity of interest rather than across the entire solution. In [30], we study goal-oriented adaptivity for multiscale methods, and in particular the design of error indicators to drive the adaptive enrichment based on the goal function. In this methodology, one seeks to determine the number of multiscale basis functions adaptively for each coarse region to efficiently reduce the error in the goal functional. Two estimators are studied. One is a residual based strategy and the other uses dual weighted residual method for multiscale problems. The method is demonstrated on high-contrast problems with heterogeneous multiscale coefficients, and is seen to outperform the standard residual based strategy with respect to efficient reduction of error in the goal function.*

5.1 Numerical Results

We show numerical results for the H^{-1} adaptivity. We consider the permeability field shown in Figure 4 for (9) and the forcing term shown in Figure 5. In Table 2, we present the numerical results and in Figure 5 (right plot), we compare the convergence of the adaptive GMsFEM and the GMsFEM, which uses a uniform number of basis functions. As we observe that the adaptive GMsFEM converges faster.

| #DOF | e_a | e_2 | #DOF | e_a | e_2 |
|------|--------|--------|------|--------|--------|
| 81 | 75.04% | 42.48% | 81 | 75.04% | 42.48% |
| 151 | 30.47% | 7.84% | 162 | 33.39% | 6.74% |
| 245 | 22.65% | 4.70% | 243 | 27.00% | 5.52% |
| 334 | 18.76% | 3.59% | 324 | 25.11% | 4.72% |
| 395 | 16.84% | 3.08% | 405 | 21.68% | 3.50% |

Table 2: Errors for the adaptive GMsFEM. Left: Adaptive enrichment. Right: Uniform enrichment.

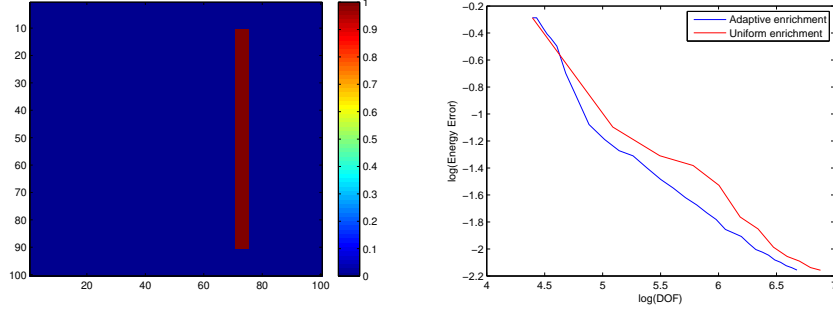


Figure 5: Left: Source function f . Right: The convergence of adaptive vs. uniform enrichment.

6 Residual-based online procedure

Previously, we discussed adaptive enrichment procedures and derived an a-posteriori error indicator, which gives an estimate of the local error over coarse grid regions. We developed the error indicators based on the L^2 -norm of the local residual and on the weighted H^{-1} -norm of the local residual, where the weight is related to the coefficient of the elliptic equation. Adaptivity is important for local multiscale methods as it identifies regions with large errors. However, after adding some initial basis functions, one needs to take into account some global information as the distant effects can be important. In this section, we discuss the development of online basis functions that substantially accelerate the convergence of GMsFEM. The online basis functions are constructed based on a residual and motivated by the analysis. We show that one needs to have a sufficient number of initial (offline) basis functions to guarantee an error decay independent of the contrast. In particular, the error decay in one adaptive iteration is proportional to $1 - C\Lambda_*$ (see (25) for more precise estimate), where Λ_* is the smallest eigenvalue among all coarse blocks that the corresponding eigenvector is not included in the offline space. Thus, it is important to include all eigenvectors corresponding to very small eigenvalues in the offline space. In general, we would like to apply one iteration of the online procedure and reach a desired error threshold. Numerical results are presented to demonstrate that one needs to have a sufficient number of initial basis functions in the offline space before constructing online multiscale basis functions.

6.1 Residual-based online adaptive GMsFEM

We use the index $m \geq 1$ to represent the enrichment level. At the enrichment level m , we use $V_{H,\text{off}}^m$ to denote the corresponding GMsFEM space and u_H^m the corresponding solution. The sequence of functions $\{u_H^m\}_{m \geq 1}$ will converge to the fine-scale solution. We emphasize that the space $V_{H,\text{off}}^m$ can contain both offline and online basis functions. We will construct a strategy for getting the space $V_{H,\text{off}}^{m+1}$ from $V_{H,\text{off}}^m$. The online basis functions are computed based on some local residuals for the current multiscale solution, that is, the function $u_{H,\text{off}}^m$.

Suppose that we need to add a basis function $\phi^{\text{on},\omega_i} \in V_i$ on the i -th coarse neighbourhood ω_i . Let $V_{H,\text{off}}^{m+1} = V_{H,\text{off}}^m + \text{span}\{\phi^{\text{on},\omega_i}\}$ be the new approximation space, and $u_H^{m+1} \in V_{H,\text{off}}^{m+1}$ be the corresponding GMsFEM solution. It is easy to see that u_H^{m+1} satisfies

$$\|u - u_H^{m+1}\|_V^2 = \inf_{v \in V_{H,\text{off}}^{m+1}} \|u - v\|_V^2.$$

Taking $v = u_H^{m+1} + \alpha\phi^{\text{on},\omega_i}$, where α is a scalar to be determined, we have

$$\|u - u_H^{m+1}\|_V^2 \leq \|u - u_H^m - \alpha\phi^{\text{on},\omega_i}\|_V^2 = \|u - u_H^m\|_V^2 - 2\alpha a(u - u_H^m, \phi^{\text{on},\omega_i}) + \alpha^2 a(\phi^{\text{on},\omega_i}, \phi^{\text{on},\omega_i}). \quad (23)$$

The last two terms in the above inequality measure the amount of reduction in error when the new basis function $\phi^{\text{on},\omega_i}$ is added to the space $V_{H,\text{off}}^m$. To determine $\phi^{\text{on},\omega_i}$, we first assume that the basis function

$\phi^{\text{on},\omega_i}$ is normalized so that $a(\phi^{\text{on},\omega_i}, \phi^{\text{on},\omega_i}) = 1$. In order to maximize the reduction in error, one can show that $\alpha = a(u - u_H^m, \phi^{\text{on},\omega_i})$. Using this choice of α , we have

$$\|u - u_H^{m+1}\|_V^2 \leq \|u - u_H^m\|_V^2 - |a(u - u_H^m, \phi^{\text{on},\omega_i})|^2.$$

We can show that

$$a(\phi^{\text{on},\omega_i}, v) = (f, v) - a(u_H^m, v), \quad \forall v \in V_i \quad (24)$$

and $\|\phi^{\text{on},\omega_i}\|_{V_i} = \|R_i\|_{V_i^*}$. Hence, the new online basis function $\phi^{\text{on},\omega_i} \in V_i$ can be obtained by solving (24). In addition, the residual norm $\|R_i\|_{V_i^*}$ provides a measure on the amount of reduction in energy error.

In [23, 24], we have studied the convergence of the above online adaptive procedure. To simplify notations, we write $r_i = \|R_i\|_{V_i^*}$. We have shown that

$$\|u - u_H^{m+1}\|_V^2 \leq \left(1 - \frac{\lambda_{n_i+1}^{\omega_i}}{C_{\text{err}}} \frac{r_i^2 (\lambda_{n_i+1}^{\omega_i})^{-1}}{\sum_{j=1}^{N_c} r_j^2 (\lambda_{n_j+1}^{\omega_j})^{-1}}\right) \|u - u_H^m\|_V^2, \quad (25)$$

where n_i is the number of offline basis in ω_i . The above inequality gives the convergence of the online adaptive GMsFEM with a precise convergence rate for the case when one online basis function is added per iteration. The estimate (25) shows that the eigenvectors corresponding to very small eigenvalues need to be included in the offline space in order to achieve a significant error reduction in one iteration. To enhance the convergence and efficiency of the online adaptive GMsFEM, we consider enrichment on non-overlapping coarse neighborhoods. Let $I \subset \{1, 2, \dots, N\}$ be the index set of some non-overlapping coarse neighborhoods. For each $i \in I$, we can obtain a basis function $\phi^{\text{on},\omega_i} \in V_i$ using (24). We define $V_{H,\text{off}}^{m+1} = V_{H,\text{off}}^m + \text{span}\{\phi^{\text{on},\omega_i}, i \in I\}$.

We remark that the error will decrease independent of physical parameters such as the contrast and scales if the offline space is appropriately chosen. We will demonstrate the effectiveness of this method by a numerical example. We remark that one can also derive apriori error estimate for $\|u - u_{H,\text{off}}^{m+1}\|_V^2$.

6.2 Numerical result

In Table 3, we present numerical results for online enrichment for the GMsFEM. We use the permeability field shown in Figure 4 and the forcing term shown in Figure 5 (left figure). As we observe the online enrichment does not improve the error if we have only one offline basis function. We remind that the first, second, and third smallest eigenvalues, Λ_* , among all coarse blocks are 0.0024, 24.0071, 35.6941, respectively. Because the first eigenvalue is small, the error decrease in one online iteration is small. In particular, for each online iteration, the error decreases slightly. It is important that one adaptive online iteration can decrease the error substantially. As we increase the number of offline basis functions, the convergence is very fast and one online iteration is sufficient to reduce the error significantly. In one iteration, the error drops below 1% for the energy error. We note that this procedure can be implemented adaptively and we add online basis functions only in some regions.

| DOF \ #initial basis | 1 | 2 | 3 | | 1 | 2 | 3 |
|----------------------|--------|-----------|-----------|--|--------|-----------|-----------|
| 81 | 75.06% | - | - | | 42.49% | - | - |
| 162 | 32.77% | 33.50% | - | | 16.29% | 6.76% | - |
| 243 | 21.59% | 1.13% | 27.14% | | 7.33% | 0.079% | 5.54 |
| 324 | 3.46% | 0.019% | 1.35% | | 1.21% | 0.0018% | 0.081 |
| 405 | 2.45% | 2.61e-04% | 0.018% | | 0.68% | 1.81e-05% | 0.0011 |
| 486 | 1.18% | 2.88e-06% | 2.49e-04% | | 0.31% | 1.74e-07% | 1.60e-05% |

Table 3: The errors for the GMsFEM with the online procedure (left table: H^1 errors; right table: L^2 errors).

7 Selected global discretizations and energy minimizing oversampling

In previous sections, we used continuous Galerkin coupling for multiscale basis functions. In many applications, one needs to use various discretizations. For example, for flows in porous media, mass conservation is very important and, thus, it is advantageous to use mixed methods. In seismic wave applications involving the time-explicit discretization of wave equations, one needs block-diagonal mass matrices, which can be obtained using discontinuous Galerkin approaches. In this section, we present two global couplings, mixed finite element and Interior Penalty Discontinuous Galerkin (IPDG) couplings. We refer to [39] for Hybridized Discontinuous Galerkin (HDG) coupling and to [18] for comparisons between using IPDG and HDG couplings with multiscale basis functions.

For each discretization, we define snapshot spaces and local spectral decompositions in the snapshot spaces. As we emphasized earlier the construction of the snapshot spaces and local multiscale spaces depends on the global discretization. For example, for mixed GMsFEM, the multiscale spaces are constructed for the velocity field using two neighboring coarse elements. We will present some ingredients of GMsFEM introduced earlier in the construction of multiscale basis functions, e.g., oversampling techniques and online multiscale basis functions. We also mention a new ingredient for multiscale basis construction - energy minimizing snapshots.

In a mixed formulation, the problem can be formulated as

$$\kappa^{-1}v + \nabla u = 0 \text{ in } \Omega, \quad \text{div}(v) = f \text{ in } \Omega, \quad (26)$$

with Neumann boundary condition $v \cdot n = v_\Omega$ on $\partial\Omega$. Depending on a discretization technique, the coarse grid configuration will vary. For a mixed formulation, coarse grid blocks that share common edges (faces) will be used in constructing multiscale basis functions. In a discontinuous Galerkin formulation, the support of multiscale basis functions is limited to coarse blocks.

7.1 Discretizations

7.1.1 The mixed GMsFEM

For the mixed GMsFEM, we will construct basis functions whose supports are ω_E , which are the two coarse elements that share a common edge (face) E . In particular, we let \mathcal{E}^H be the set of all edges (faces) of the coarse grid and let N_e be the total number of edges (faces) of the coarse grid. We define the coarse grid neighborhood ω_E of a face $E \in \mathcal{E}^H$ as

$$\omega_E = \bigcup \{K \in \mathcal{T}^H : E \in \partial K\}, \quad i = 1, 2, \dots, N_e,$$

which is a union of two coarse grid blocks if E_i is an interior edge (face) (see Figure 1). For a coarse edge E_i , we write $\omega_{E_i} = \omega_i$ to unify the notations.

Next, we define the notations for the solution spaces for the pressure u and the velocity v . Let $Q_{H,\text{off}}$ be the space of functions which are constant on each coarse grid block. We will use this space to approximate u . For the multiscale approximation of the velocity v , we will follow the general framework of the GMsFEM and construct a multiscale space $V_{H,\text{off}}$ for the velocity. Using the pressure space $Q_{H,\text{off}}$ and the velocity space $V_{H,\text{off}}$, we solve for $u_H \in Q_{H,\text{off}}$ and $v_H \in V_{H,\text{off}}$ such that

$$\int_{\Omega} \kappa^{-1}v_H \cdot w - \int_{\Omega} \text{div}(w)u_H = 0, \quad \forall w \in \dot{V}_{H,\text{off}}, \quad \int_{\Omega} \text{div}(v_H)q = \int_{\Omega} fq, \quad \forall q \in Q_{H,\text{off}}, \quad (27)$$

with boundary condition $v_H \cdot n = v_{\Omega,H}$ on $\partial\Omega$, where $\dot{V}_{H,\text{off}} = \{v \in V_{H,\text{off}} : v \cdot n = 0 \text{ on } \partial\Omega\}$, and $v_{\Omega,H}$ is the projection of v_Ω in the multiscale space. We remark that we will define the snapshot and offline spaces so that the functions in $V_{H,\text{off}}$ are globally $H(\text{div})$ -conforming, and as a result the normal components of the basis are continuous across coarse grid edges.

7.1.2 DG GMsFEM (GMsDGM)

For the GMsDGM, the general methodology for the construction of multiscale basis functions is similar to the above mixed GMsFEM. We will construct multiscale basis functions for the approximation of u in (9). The main difference is that the functions in the snapshot space $V_{H,\text{snap}}$ and the offline space $V_{H,\text{off}}$ are supported in coarse element K , instead of the coarse neighborhood ω_i . Moreover, in the oversampling approach, the oversampled regions K^+ are defined by enlarging coarse elements K (see Figure 1). The construction of basis functions will be given in the next section. When the offline space $V_{H,\text{off}}$ is available, we can find the solution $u_H \in V_{H,\text{off}}$ such that (see [15, 24])

$$a_{\text{DG}}(u_H, q) = (f, q), \quad \forall q \in V_{H,\text{off}}, \quad (28)$$

where the bilinear form a_{DG} is defined as

$$a_{\text{DG}}(u, q) = a_H(u, q) - \sum_{E \in \mathcal{E}^H} \int_E \left(\{\kappa \nabla u \cdot n_E\} [q] + \{\kappa \nabla q \cdot n_E\} [u] \right) + \sum_{E \in \mathcal{E}^H} \frac{\gamma}{h} \int_E \bar{\kappa} [u] [q] \quad (29)$$

with $a_H(p, q) = \sum_{K \in \mathcal{T}^H} a_H^K(p, q)$, $a_H^K(p, q) = \int_K \kappa \nabla p \cdot \nabla q$, where $\gamma > 0$ is a penalty parameter, n_E is a fixed unit normal vector defined on the coarse edge $E \in \mathcal{E}^H$. Note that, in (29), the average and the jump operators are defined in the classical way. Specifically, consider an interior coarse edge $E \in \mathcal{E}^H$ and let K_L and K_R be the two coarse grid blocks sharing the edge E . For a piecewise smooth function G , we define $\{G\} = \frac{1}{2}(G_R + G_L)$, $[[G]] = G_R - G_L$, on E , where $G_R = G|_{K_R}$ and $G_L = G|_{K_L}$ and we assume that the normal vector n_E is pointing from K_R to K_L . Moreover, on the edge E , we define $\bar{\kappa} = (\kappa_{K_R} + \kappa_{K_L})/2$ where κ_{K_L} is the maximum value of κ over K_L and κ_{K_R} is defined similarly. For a coarse edge E lying on the boundary ∂D , we define $\{G\} = [[G]] = G$, and $\bar{\kappa} = \kappa_K$ on E , where we always assume that n_E is pointing outside of D . We note that the DG coupling (28) is the classical interior penalty discontinuous Galerkin (IPDG) method with the multiscale basis functions as the approximation space.

7.2 Basis construction

7.2.1 Multiscale basis functions in mixed GMsFEM. Non-oversampling

First, we present the construction of the snapshot space for the approximation of the velocity v . The space is a large function space containing basis functions whose normal traces on coarse grid edges are resolved up to the fine grid level. Let $E_i \in \mathcal{E}^H$ be a coarse edge. We can write the edge $E_i = \bigcup_{j=1}^{J_i} e_j$, where the e_j 's are the fine grid edges (faces) contained in E_i and J_i is the total number of those fine grid edges. To construct the snapshot vectors, for each $j = 1, 2, \dots, J_i$, we will solve the following local problem

$$\kappa^{-1} \psi_{i,j} + \nabla \eta_{i,j} = 0 \text{ in } K \subset \omega_i, \quad \text{div}(\psi_{i,j}) = \alpha_{i,j} \text{ in } K \subset \omega_i, \quad (30)$$

subject to the Neumann boundary conditions $\psi_{i,j} \cdot n_i = 0$ on $\partial \omega_i$ and $\psi_{i,j} \cdot n_{E_i} = \delta_j^h$ on E_i , where n_{E_i} is a unit normal vector on E_i , δ_j^h is a fine-scale discrete delta function defined by

$$\delta_j^h = \{1 \text{ on } e_j; 0 \text{ on } E_i \setminus e_j\} \quad (31)$$

and n is the outward unit-normal vector on $\partial \omega_i$. The function $\alpha_{i,j}$ is constant on each coarse grid block within ω_i and it should satisfy the condition $\int_K \alpha_{i,j} = \int_{\partial K} \psi_{i,j} \cdot n_i$ for each coarse grid $K \in \omega_i$. We denote the number of snapshots from (30) by L_i and the space spanned by these functions by $V_{H,\text{snap}}^{\omega_i}$.

Now, we define the local spectral problem for $V_{H,\text{snap}}^{\omega_i}$ for the construction of the offline space $V_{H,\text{off}}^{\omega_i}$. The local spectral problem is to find real number λ and $v \in V_{H,\text{snap}}^{\omega_i}$ such that

$$a_i(v, w) = \lambda s_i(v, w), \quad \forall w \in V_{H,\text{snap}}^{\omega_i}, \quad (32)$$

where a_i is a symmetric non-negative definite bilinear operator and s_i is a symmetric positive definite bilinear operators defined on $V_{H,\text{snap}}^{\omega_i} \times V_{H,\text{snap}}^{\omega_i}$. Motivated by the analysis ([21]), we choose

$$a_i(v, w) = \int_{E_i} (v \cdot n_{E_i})(w \cdot n_{E_i}), \quad s_i(v, w) = \int_{\omega_i} \kappa^{-1} v \cdot w + \text{div}(v)\text{div}(w).$$

We let $\lambda_j^{\omega_i}$ be the eigenvalues of (32) arranged in ascending order, and $\psi_j^{\omega_i}$ be the corresponding eigenfunctions. We will use the first l_i eigenfunctions to construct our offline space $V_{H,\text{off}}^{\omega_i}$. We note that it is important to keep the eigenfunctions with small eigenvalues in the offline space. The global offline space $V_{H,\text{off}}$ is the union of all $V_{H,\text{off}}^{\omega_i}$.

The *oversampling* can be performed by using the snapshots in the regions surrounding the edge and computing boundary conditions. We refer to [21] for the details.

7.2.2 Multiscale basis functions in GMsDGM. Oversampling

For each $K_i \in \mathcal{T}^H$, we consider an oversampled region $K_i^+ \supset K_i$. We define $\psi_{i,j}^+$ such that

$$-\nabla \cdot (\kappa \nabla \psi_{i,j}^+) = 0 \quad \text{in } K_i^+, \quad \psi_{i,j}^+ = \delta_j^h \quad \text{on } \partial K_i^+, \quad (33)$$

where δ_j^h is the delta function defined in Section 4. We denote the number of local snapshots by L_i^+ and the space spanned by these functions by $V_{H,\text{snap}}^{+,K_i}$.

These basis functions $\psi_{i,j}^+$ are supported in K_i^+ . We denote the restriction of $\psi_{i,j}^+$ on K_i by $\psi_{i,j}$. Then we remove the linear dependence of $\psi_{i,j}$ by performing Proper Orthogonal Decomposition (POD) ([49]). Next, we will define the local spectral problem as finding λ and $v \in V_{H,\text{snap}}^{+,\omega_i}$ such that

$$a_i(v, w) = \lambda s_i(v, w) \quad \forall w \in V_{H,\text{snap}}^{+,\omega_i}, \quad (34)$$

where a_i is a symmetric non-negative definite bilinear operator and s_i is a symmetric positive bilinear operators defined on $V_{H,\text{snap}}^{+,\omega_i} \times V_{H,\text{snap}}^{+,\omega_i}$, $a_i(v, w) = \int_{K_i^+} \kappa \nabla v \cdot \nabla w$ and $s_i(v, w) = \int_{\partial K_i} \kappa v w$.

We let $\lambda_j^{\omega_i}$ be the eigenvalues of (34) arranged in ascending order, and ψ_j^{+,ω_i} be the corresponding eigenfunctions. We denote $\psi_j^{\omega_i}$ as the restriction of ψ_j^{+,ω_i} on K_i . We will use the first l_i restricted eigenfunctions to construct the offline space $V_{H,\text{off}}^{\omega_i}$. We note that it is important to keep the eigenfunctions with small eigenvalues in the offline space. The global offline space $V_{H,\text{off}}$ is the union of all $V_{H,\text{off}}^{\omega_i}$.

Remark 8 (Energy minimizing oversampling). *For efficient online simulations, one needs to develop energy minimizing snapshot vectors [18]. In these approaches, the snapshot vectors are computed by solving local constrained minimization problems. More precisely, first, we use an oversampled region to construct the snapshot space. We denote them $\psi_1^+, \dots, \psi_N^+$ for simplicity and consider one coarse block. Next, we consider the restriction of these snapshot vectors in a target coarse block and identify linearly independent components, ψ_1, \dots, ψ_M , $M \leq N$. A next important ingredient is to construct minimum energy snapshot vectors that represent the snapshot functions in the target coarse region. These snapshot vectors are constructed by solving a local minimization problem. We denote them by $\tilde{\psi}_1, \dots, \tilde{\psi}_M$. In the final step, we perform a local spectral decomposition to compute multiscale basis functions ([18]). Energy minimizing snapshots are important for online basis computations and are used to show the stable decomposition.*

7.3 Numerical results

We present the numerical results for the mixed GMsFEM. The permeability field is as shown in Figure 4 and $f = 1$. The coarse grid 10×10 and we use zero Dirichlet boundary conditions. In Table 4, the relative errors measured in the norm $\|v\|^2 = \int_{\Omega} \kappa^{-1} |v|^2$ are shown as we increase the number of basis functions. As we observe from this table, the mixed GMsFEM has a very good convergence rate. In general, we have observed a good accuracy when the mixed formulation of the GMsFEM is used. With only 3 basis functions per node, the error in the velocity field is less than 5%. We note that the total degrees of freedom for the fine-grid solution is 20200.

| | | | | | |
|-------------------------|---------|---------|---------|---------|----------|
| #basis | 1 (220) | 2 (440) | 3 (660) | 4 (880) | 5 (1100) |
| $\ v_h - v_H\ /\ v_h\ $ | 16.50% | 6.41% | 3.91% | 2.65% | 1.64% |

Table 4: Convergence history with increasing number of offline basis functions for the mixed GMsFEM.

8 Discussions on the sparsity in the snapshot space

In previous approaches, multiscale basis functions are constructed using local snapshot spaces, where a snapshot space is a large space that represents the solution behavior in a coarse block. In a number of applications, one may have a sparsity in the snapshot space for an appropriate choice of a snapshot space. More precisely, the solution may only involve a portion of the snapshot space. In this case, one can use sparsity techniques to identify multiscale basis functions. We briefly discuss two such sparse local multiscale model reduction approaches (see [14] for details).

In the first approach (which is used for parameter-dependent multiscale PDEs), we use local minimization techniques, such as sparse POD, to identify multiscale basis functions, which are sparse in the snapshot space. These minimization techniques use l_1 minimization to find local multiscale basis functions, which are further used for finding the solution. In the second approach (which is used for the Helmholtz equation), we directly apply l_1 minimization techniques to solve the underlying PDEs. This approach is more expensive as it involves a large snapshot space; however, in this example, we cannot identify a local minimization principle, such as local generalized SVD.

We consider $\mathcal{L}u = f$, where \mathcal{L} is a differential operator. For example, we consider parameter-dependent heterogeneous flows, $\mathcal{L}u = -\text{div}(\kappa(x; \mu)\nabla u)$, and the Helmholtz equation, $\mathcal{L}u = -\text{div}(\kappa(x)\nabla u) - \zeta^2 n(x)u$. Previous approaches use all snapshot vectors when seeking multiscale basis functions. In a number of applications, the solution is sparse in the snapshot space, which implies that in the expansion

$$u = \sum_{i,j} c_{i,j} \psi_i^{\omega_j},$$

many coefficients $c_{i,j}$ are zeros. In this case, one can save computational effort by employing sparsity techniques.

The main challenge in these applications is to construct a snapshot space, where the solution is sparse. In the first example, this can be achieved, because an online parameter value μ can be close to some of the pre-selected offline values of μ 's, and thus, the multiscale basis functions (and the solution) can have a sparse representation in the snapshot space. In the second example, we select cases where the solution u contains only a few snapshot vectors corresponding to some wave directions. We note that if the snapshot space is not chosen carefully, one may not have the sparsity. We can consider two distinct cases.

- First approach: “Local-Sparse Snapshot Subspace Approach”. Determining the online sparse space locally via local spectral sparse decomposition in the snapshot space (motivated by parameter-dependent problems).
- Second approach: “Sparse Snapshot Subspace Approach”. Determining the online space globally via a global solve (motivated by using plane wave snapshot vectors and the Helmholtz equation).

See Figure 6 for illustration. We use sparsity techniques (e.g., [11, 67]) to identify local multiscale basis functions and solve the global problem. The numerical results and more discussions can be found in [14].

9 Space-time GMsFEM

Many multiscale processes vary over multiple space and time scales. In many applications, the heterogeneities change due to the time can be significant and it needs to be taken into account in reduced-order models. Some well-known approaches for handling *separable* spatial and temporal scales are homogenization techniques

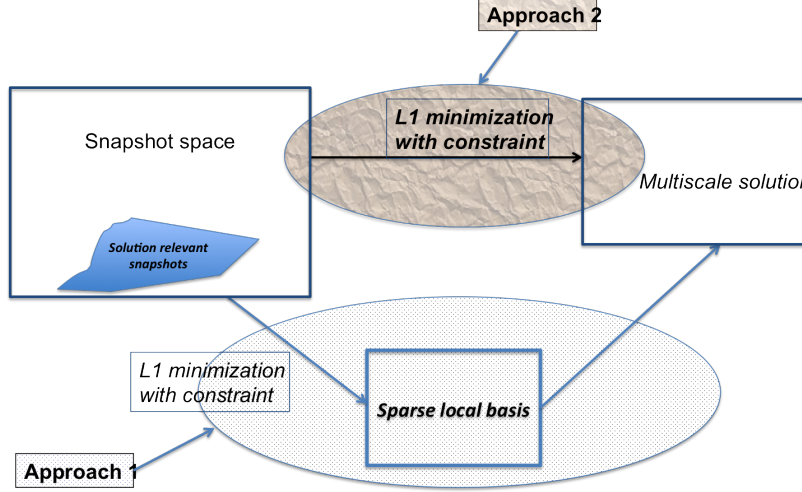


Figure 6: Illustration of the approaches.

[55, 63, 65, 40]. In these methods, one solves local problems in space and time. We remind a well-known case of the parabolic equation

$$\partial u / \partial t - \operatorname{div}(\kappa(x, t) \nabla u) = f, \quad (35)$$

subject to smooth initial and boundary conditions. The homogenized equation has the same form as (35), but with the smooth coefficients $\kappa^*(x, t)$. One can compute the coefficients using the solutions of local space-time parabolic equations in the periodic cell. This localization is possible thanks to the scale separation. One can extend this homogenization procedure to numerical homogenization type methods [61, 41], where one solves the local parabolic equations in each coarse block.

Previous approaches within GMsFEM focused on constructing multiscale spaces and relevant ingredients in space only. The extension of the GMsFEM to space-time heterogeneous problems requires a modification of the space only problems because of (1) the parabolic nature of cell solutions, (2) extra degrees of freedom associated with space-time cells, and (3) local boundary conditions in space-time cells. In the approach that we are going to discuss, we construct snapshot spaces in space-time local domains. We construct the snapshot solutions by solving local problems. We can construct a complete snapshot space by taking all possible boundary conditions; however, this can lead to very high computational cost. For this reason, we use randomized boundary conditions for local snapshot vectors. Computing multiscale basis functions employs local spectral problems in space-time domain.

We consider the parabolic differential equation (35) in a space-time domain $\Omega \times (0, T)$ and assume $u = 0$ on $\partial\Omega \times (0, T)$ and $u(x, 0) = \beta(x)$ in Ω . The proposed method follows the space-time finite element framework, where the time dependent multiscale basis functions are constructed on the coarse grid. Therefore, compared with the time independent basis structure, it gives a more efficient numerical solver for the parabolic problem in complicated media.

We would like to compute the solution u_H in the whole time interval $(0, T)$. In fact, if we assume the solution space $V_{H,\text{off}}^{(0,T)}$ is a direct sum of the spaces only containing the functions defined on one single coarse time interval (T_{n-1}, T_n) , we can decompose the problem into a sequence of problems and find the solution u_H in each time interval sequentially. The coarse space will be constructed in each time interval and $V_{H,\text{off}}^{(0,T)} = \bigoplus_{n=1}^N V_{H,\text{off}}^{(T_{n-1}, T_n)}$, where $V_{H,\text{off}}^{(T_{n-1}, T_n)}$ only contains the functions having zero values in the time interval $(0, T)$ except (T_{n-1}, T_n) , namely $\forall v \in V_{H,\text{off}}^{(T_{n-1}, T_n)}$, $v(\cdot, t) = 0$ for $t \in (0, T) \setminus (T_{n-1}, T_n)$.

The coarse-grid equation consists of finding $u_H^{(n)} \in V_{H,\text{off}}^{(T_{n-1}, T_n)}$ (where $V_{H,\text{off}}^{(T_{n-1}, T_n)}$ will be defined later)

satisfying

$$\int_{T_{n-1}}^{T_n} \int_{\Omega} \frac{\partial u_H^{(n)}}{\partial t} v + \int_{T_{n-1}}^{T_n} \int_{\Omega} \kappa \nabla u_H^{(n)} \cdot \nabla v + \int_{\Omega} u_H^{(n)}(x, T_{n-1}^+) v(x, T_{n-1}^+) = \int_{T_{n-1}}^{T_n} \int_{\Omega} f v + \int_{\Omega} g_H^{(n)}(x) v(x, T_{n-1}^+),$$

for all $v \in V_{H, \text{off}}^{(T_{n-1}, T_n)}$, where $g_H^{(n)}(\cdot) = \{u_H^{(n-1)}(\cdot, T_{n-1}^-) \text{ for } n \geq 1; \beta(\cdot) \text{ for } n = 0\}$, and $F(\alpha^+)$ and $F(\alpha^-)$ denote the right hand and left hand limits of F at α respectively. Then, the solution u_H of the problem in $(0, T)$ is the direct sum of all these $u_H^{(n)}$'s, that is $u_H = \bigoplus_{n=1}^N u_H^{(n)}$.

Next, we motivate the use of space-time multiscale basis functions by comparing it to space multiscale basis functions. In particular, we discuss the savings in the reduced models when space-time multiscale basis functions are used compared to space multiscale basis functions. We denote $\{t_{n1}, \dots, t_{np}\}$ as the p fine time steps in (T_{n-1}, T_n) . When we construct space-time multiscale basis functions, the solution can be represented as $u_H^{(n)} = \sum_{l,i} c_{l,i} \psi_l^{\omega_i}(x, t)$ in the interval (T_{n-1}, T_n) . In this case, the number of coefficients $c_{l,i}$ is related to the size of the reduced system in space-time interval. On the other hand, if we use only space multiscale basis functions, we need to construct these multiscale basis functions at each fine time instant t_{nj} , denoted by $\psi_l^{\omega_i}(x, t_{nj})$. The solution u_H spanned by these basis functions will have a much larger dimension because each time instant is represented by multiscale basis functions. Thus, performing space-time multiscale model reduction can provide a substantial CPU savings.

9.1 Construction of offline basis functions

9.1.1 Snapshot space

Let ω be a given coarse neighborhood in space. We omit the coarse node index to simplify the notations. The construction of the offline basis functions on coarse time interval (T_{n-1}, T_n) starts with a snapshot space $V_{H, \text{snap}}^{\omega}$ (or $V_{H, \text{snap}}^{\omega, (T_{n-1}, T_n)}$). We also omit the coarse time index n to simplify the notations. The snapshot space $V_{H, \text{snap}}^{\omega}$ is a set of functions defined on ω and contains all or most necessary components of the fine-scale solution restricted to ω . A spectral problem is then solved in the snapshot space to extract the dominant modes in the snapshot space. These dominant modes are the offline basis functions and the resulting reduced space is called the offline space. There are two choices of $V_{H, \text{snap}}^{\omega}$ that are commonly used.

The first choice is to use all possible fine-grid functions in $\omega \times (T_{n-1}, T_n)$. This snapshot space provides an accurate approximation for the solution space; however, this snapshot space can be very large. The second choice for the snapshot spaces consists of solving local problems for all possible boundary conditions, which we present here. As before, we denote by ω^+ the oversampled space region of $\omega \subset \omega^+$, defined by adding several fine- or coarse-grid layers around ω . Also, we define (T_{n-1}^*, T_n) as the left-side oversampled time region for (T_{n-1}, T_n) . In the following, we generate inexpensive snapshots using random boundary conditions on the oversampled space-time region $\omega^+ \times (T_{n-1}^*, T_n)$. That is for each fine boundary node on $\partial(\omega^+ \times (T_{n-1}^*, T_n))$, we solve a small number of local problems imposed with random boundary conditions

$$\partial \psi_j^{+, \omega} / \partial t - \text{div}(\kappa(x, t) \nabla \psi_j^{+, \omega}) = 0 \quad \text{in } \omega^+ \times (T_{n-1}^*, T_n), \quad \psi_j^{+, \omega}(x, t) = r_l \quad \text{on } \partial(\omega^+ \times (T_{n-1}^*, T_n)),$$

where r_l are independent identically distributed (i.i.d.) standard Gaussian random vectors on the fine-grid nodes of the boundaries $t = T_{n-1}^*$ and on $\partial \omega^+ \times (T_{n-1}^*, T_n)$.

Then the local snapshot space on $\omega^+ \times (T_{n-1}^*, T_n)$ is

$$V_{H, \text{snap}}^{+, \omega} = \text{span}\{\psi_j^{+, \omega}(x, t) | j = 1, \dots, l^{\omega} + p_{\text{bf}}^{\omega}\},$$

where l^{ω} is the number of local offline basis we want to construct in ω and p_{bf}^{ω} is the buffer number. Later on, we use the same buffer number for all ω 's and simply use the notation p_{bf} . In the following sections, if we specify one special coarse neighborhood ω_i , we use the notation l_i to denote the number of local offline basis. With these snapshots, we follow the procedure in the following subsection to generate offline basis functions by using an auxiliary spectral decomposition.

9.1.2 Offline space

To obtain the offline basis functions, we need to perform a space reduction by appropriate spectral problems. Motivated by a convergence analysis, we adopt the following spectral problem on $\omega^+ \times (T_{n-1}, T_n)$. Find $(\phi, \lambda) \in V_{H, \text{snap}}^{+, \omega} \times \mathbb{R}$ such that

$$A_n(\phi, v) = \lambda S_n(\phi, v), \quad \forall v \in V_{\text{snap}}^{\omega^+}, \quad (36)$$

where the bilinear operators $A_n(\phi, v)$ and $S_n(\phi, v)$ are defined by

$$\begin{aligned} A_n(\phi, v) &= \frac{1}{2} \left(\int_{\omega^+} \phi(x, T_n) v(x, T_n) + \int_{\omega^+} \phi(x, T_{n-1}) v(x, T_{n-1}) \right) + \int_{T_{n-1}}^{T_n} \int_{\omega^+} \kappa(x, t) \nabla \phi \cdot \nabla v, \\ S_n(\phi, v) &= \int_{\omega^+} \phi(x, T_{n-1}) v(x, T_{n-1}) + \int_{T_{n-1}}^{T_n} \int_{\omega^+} \tilde{\kappa}^+(x, t) \phi v, \end{aligned} \quad (37)$$

where the weight function $\tilde{\kappa}^+(x, t)$ is defined by $\tilde{\kappa}^+(x, t) = \kappa(x, t) \sum_{i=1}^{N_c} |\nabla \chi_i^+|^2$, $\{\chi_i^+\}_{i=1}^{N_c}$ is a partition of unity associated with the oversampled coarse neighborhoods $\{\omega_i^+\}_{i=1}^{N_c}$ and satisfies $|\nabla \chi_i^+| \geq |\nabla \chi_i|$ on ω_i , where χ_i is the standard multiscale basis function for the coarse node x_i , $-\text{div}(\kappa(x, T_{n-1}) \nabla \chi_i) = 0$, in K , $\chi_i = g_i$ on ∂K , for all $K \in \omega_i$, where g_i is linear on each edge of ∂K .

We arrange the eigenvalues $\{\lambda_j^\omega | j = 1, 2, \dots, L_\omega + p_{\text{bf}}^\omega\}$ from (36) in the ascending order, and select the first L_ω eigenfunctions, which are corresponding to the first L_ω ordered eigenvalues, and then we can obtain the dominant modes $\psi_j^\omega(x, t)$ on the target region $\omega \times (T_{n-1}, T_n)$ by restricting $\psi_j^{+, \omega}(x, t)$ onto $\omega \times (T_{n-1}, T_n)$. Finally, the offline basis functions on $\omega \times (T_{n-1}, T_n)$ are defined by $\phi_j^\omega(x, t) = \chi^\omega \psi_j^\omega(x, t)$, where χ^ω is the standard multiscale basis function for a generic coarse neighborhood ω . This product gives conforming basis functions (Discontinuous Galerkin discretizations can also be used). We also define the local offline space on $\omega \times (T_{n-1}, T_n)$ as

$$V_{H, \text{off}}^\omega = \text{span}\{\phi_j^\omega(x, t) | j = 1, \dots, l^\omega\}.$$

Note that one can take $V_{H, \text{off}}^{(T_{n-1}, T_n)}$ in the coarse-grid equation as $V_{H, \text{off}}^{(T_{n-1}, T_n)} = \text{span}\{\phi_j^{\omega_i}(x, t) | 1 \leq i \leq N_c, 1 \leq j \leq l_i\}$.

9.2 Numerical result

We start with a high-contrast permeability field shown in Figure 4, which is translated uniformly in x_2 direction after every other fine time step. The total translation is 10% of the global domain. The number of local offline basis that will be used in each ω_i , is denoted by l_i , and the buffer number p_{bf} needs to be chosen in advance since they determine how many local snapshots are used. Then, we can construct the lower dimensional offline space by performing space reduction on the snapshot space. In the experiments, we use the same buffer number and the same number of local offline basis for all coarse neighborhood ω_i 's. Our numerical experiments show that the error does not change beyond $p_{\text{bf}} = 4$. In our numerical simulations, we take $p_{\text{bf}} = 8$ and vary l_i . The errors are displayed in Table 5. These are L^2 and energy errors. We observe that with a fixed buffer number, the relative errors are decreasing as using more offline basis. We can also observe that the values $1/\Lambda_*$'s and energy errors are correlated as predict our theory [19].

| L_i | $\dim(V_{\text{off}})$ | $L^2([0, T], \Omega)$ | energy |
|-------|------------------------|-----------------------|--------|
| 2 | 162 | 23.05% | 87.86% |
| 6 | 486 | 7.86% | 43.29% |
| 10 | 810 | 5.93% | 35.23% |
| 14 | 1134 | 3.43% | 25.66% |
| 18 | 1458 | 1.57% | 17.42% |
| 22 | 1782 | 0.88% | 12.89% |

Table 5: Relative errors using different l_i 's.

10 GMsFEM in perforated domains

One important class of multiscale problems consists in perforated domains. In these problems, differential equations are formulated in perforated domains. These domains can be considered to be the outside of inclusions or connected bodies of various sizes. Due to the variable sizes and geometries of these perforations, solutions to these problems have multiscale features. One solution approach involves posing the problem in a domain without perforations but with a very high contrast penalty term representing the domain heterogeneities ([45]). However, the void space can be a small portion of the whole domain and, thus, it is computationally expensive to enlarge the domain substantially.

The main difference in developing multiscale methods for problems in perforated domains is the complexity of the domains and that many portions of the domain are excluded in the computational domain. This poses a challenging task. For typical upscaling and numerical homogenization (e.g., [66, 48]), the macroscopic equations do not contain perforations and one computes the effective properties.

Several multiscale methods have been developed for problems in perforated domains. The use of GMsFEM for solving multiscale problems in perforated domains is motivated by recent works [59, 56, 48]. Using snapshot spaces is essential in problems with perforations, because the snapshots contain necessary geometry information. In the snapshot space, we perform local spectral decomposition to identify multiscale basis functions. In this section, we show an example for Stokes equations and refer to [26] for more discussions and extensions to other problems.

10.1 Problem setting

In this section, we present the underlying problem as stated in [20] and the corresponding fine-scale and coarse-scale discretization. Let $\Omega \subset \mathbb{R}^d$ ($d = 2, 3$) be a bounded domain covered by inactive cells (for Stokes flow and Darcy flow) or active cells (for elasticity problem) \mathcal{B}^ε . We will consider $d = 2$ case, though the results can be extended to $d > 2$. We use the superscript ε to denote quantities related to perforated domains. The active cells are where the underlying problem is solved, while inactive cells are the rest of the region. Suppose the distance between inactive cells (or active cells) is of order ε . Define $\Omega^\varepsilon := \Omega \setminus \mathcal{B}^\varepsilon$. See Figure 7 (left) for an illustration of the perforated domain. We consider the following problem defined in a perforated domain Ω^ε

$$\mathcal{L}^\varepsilon(w) = f \text{ in } \Omega^\varepsilon, \quad w = 0 \text{ or } \frac{\partial w}{\partial n} = 0 \text{ on } \partial\Omega^\varepsilon \cap \partial\mathcal{B}^\varepsilon, \quad (38)$$

$w = g$ on $\partial\Omega \cap \partial\Omega^\varepsilon$, where \mathcal{L}^ε denotes a linear differential operator, n is the unit outward normal to the boundary, f and g denote given functions with sufficient regularity. We will focus on the Dirichlet problem, namely, $w = 0$ in (38).

Denote by $V(\Omega^\varepsilon)$ the appropriate solution space, and $V_0(\Omega^\varepsilon) = \{v \in V(\Omega^\varepsilon), v = 0 \text{ on } \partial\Omega^\varepsilon\}$. The variational formulation of (38) is to find $w \in V(\Omega^\varepsilon)$ such that

$$\langle \mathcal{L}^\varepsilon(w), v \rangle_{\Omega^\varepsilon} = \langle f, v \rangle_{\Omega^\varepsilon} \quad \text{for all } v \in V_0(\Omega^\varepsilon),$$

where $\langle \cdot, \cdot \rangle_{\Omega^\varepsilon}$ denotes a specific inner product over Ω^ε for either scalar functions or vector functions and $(f, v)_{\Omega^\varepsilon}$ is the L^2 inner product. Some specific examples for the above abstract notations are given below.

Laplace: For the Laplace operator with homogeneous Dirichlet boundary conditions on $\partial\Omega^\varepsilon$, we have $\mathcal{L}^\varepsilon(u) = -\Delta u$, and $V(\Omega^\varepsilon) = H_0^1(\Omega^\varepsilon)$, $\langle \mathcal{L}^\varepsilon(u), v \rangle_{\Omega^\varepsilon} = (\nabla u, \nabla v)_{\Omega^\varepsilon}$.

Elasticity: For the elasticity operator with a homogeneous Dirichlet boundary condition on $\partial\Omega^\varepsilon$, we assume the medium is isotropic. Let $u \in (H^1(\Omega^\varepsilon))^2$ be the displacement field. The strain tensor $\varepsilon(u) \in (L^2(\Omega^\varepsilon))^{2 \times 2}$ is defined by $\varepsilon(u) = \frac{1}{2}(\nabla u + \nabla u^T)$. Thus, the stress tensor $\sigma(u) \in (L^2(\Omega^\varepsilon))^{2 \times 2}$ relates to the strain tensor $\varepsilon(u)$ such that $\sigma(u) = 2\mu\varepsilon + \xi\nabla \cdot u I$, where $\xi > 0$ and $\mu > 0$ are the Lamé coefficients. We have $\mathcal{L}^\varepsilon(u) = -\nabla \cdot \sigma$, where $V(\Omega^\varepsilon) = (H_0^1(\Omega^\varepsilon))^2$ and $\langle \mathcal{L}^\varepsilon(u), v \rangle_{\Omega^\varepsilon} = 2\mu(\varepsilon(u), \varepsilon(v))_{\Omega^\varepsilon} + \xi(\nabla \cdot u, \nabla \cdot v)_{\Omega^\varepsilon}$.

Stokes: For Stokes equations, we have

$$\mathcal{L}^\varepsilon(u, p) = (\nabla p - \Delta u, \nabla \cdot u)^T, \quad (39)$$

where μ is the viscosity, p is the fluid pressure, u represents the velocity, $V(\Omega^\varepsilon) = (H_0^1(\Omega^\varepsilon))^2 \times L_0^2(\Omega^\varepsilon)$, and

$$\langle \mathcal{L}^\varepsilon(u, p), (v, q) \rangle_{\Omega^\varepsilon} = \begin{pmatrix} (\nabla u, \nabla v)_{\Omega^\varepsilon} & -(\nabla \cdot v, p)_{\Omega^\varepsilon} \\ (\nabla \cdot u, q)_{\Omega^\varepsilon} & 0 \end{pmatrix}.$$

We recall that $L_0^2(\Omega^\varepsilon)$ contains functions in $L^2(\Omega^\varepsilon)$ with zero average in Ω^ε . We will illustrate our ideas and results using the Stokes equations.

For the numerical approximation of the above problems, we first introduce the notations of fine and coarse grids pertinent to perforated domains. It follows similar constructions as before with the exception of domain geometries, which can intersect the boundaries of coarse regions. Let \mathcal{T}_H be a coarse-grid partition of the domain Ω^ε with mesh size H . Notice that, the edges of the coarse elements do not necessarily have straight edges because of the perforations (see Figure 7, left). By conducting a conforming refinement of the coarse mesh \mathcal{T}^H , we can obtain a fine mesh \mathcal{T}^h of Ω^ε with mesh size h . Typically, we assume that $0 < h \ll H < 1$, and that the fine-scale mesh \mathcal{T}_h is sufficiently fine to fully resolve the small-scale information of the domain, and \mathcal{T}_H is a coarse mesh containing many fine-scale features. Let N_v and N_e be the number of nodes and edges in coarse grid respectively. We denote by $\{x_i | 1 \leq i \leq N_e\}$ the set of coarse nodes, and $\{E_j | 1 \leq j \leq N_e\}$ the set of coarse edges. We define a coarse neighborhood ω_i^ε for each coarse node x_i by $\omega_i^\varepsilon = \cup \{K_j^\varepsilon \in \mathcal{T}^H; x_i \in \bar{K}_j^\varepsilon\}$, which is the union of all coarse elements having the node x_i . For the Stokes problem, additionally, we define a coarse neighborhood ω_m^ε for each coarse edge E_m by

$$\omega_m^\varepsilon = \cup \{K_j^\varepsilon \in \mathcal{T}^H; E_m \in \bar{K}_j^\varepsilon\}, \quad (40)$$

which is the union of all coarse elements having the edge E_m . We let V_h be the fine scale space for velocity u and Q_h be the fine scale space for the pressure p . One can choose V_h to be piecewise quadratic and Q_h to be piecewise constant with respect to the fine mesh \mathcal{T}^h .

We will then obtain the fine-scale solution $(u, p) \in V_h \times Q_h$ for the Stokes system by solving the following variational problem

$$\langle \mathcal{L}^\varepsilon(u, p), (v, q) \rangle_{\Omega^\varepsilon} = ((f, 0), (v, q))_{\Omega^\varepsilon}, \quad \text{for all } (v, q) \in V_h \times Q_h. \quad (41)$$

These solutions are used as reference solutions to test the performance of the schemes.

10.2 The construction of offline basis functions

In this section, we describe the construction of offline basis for the Stokes problem in perforated domains. We refer to [16] for online basis construction and the analysis, as well as GMsFEM for the elliptic equation and the elasticity equations in perforated domains.

10.2.1 Snapshot space

To compute snapshot functions, we solve the following problem on the coarse neighborhood ω_i^ε : find (u_l^i, p_l^i) (on a fine grid) such that

$$\int_{\omega_i^\varepsilon} \nabla u_l^i : \nabla v - \int_{\omega_i^\varepsilon} p_l^i \operatorname{div}(v) = 0, \quad \forall v \in V_{h,0}^i, \quad \int_{\omega_i^\varepsilon} q \operatorname{div}(u_l^i) = \int_{\omega_i^\varepsilon} cq, \quad \forall q \in Q_h^i, \quad (42)$$

with boundary conditions $u_l^i = (0, 0)$, on $\partial \mathcal{B}^\varepsilon$, $u_l^i = (\delta_l^h, 0)$ or $(0, \delta_l^h)$, on $\partial \omega_i^\varepsilon \setminus \partial \mathcal{B}^\varepsilon$. Here, we write $\omega_i^\varepsilon \setminus \partial \mathcal{B}^\varepsilon = \cup_{l=1}^{S_i} e_l$, where e_l are the fine-grid edges and S_i is the number of these fine grid edges on $\omega_i^\varepsilon \setminus \partial \mathcal{B}^\varepsilon$. Moreover, δ_l^h is a fine-scale delta function such that it has value 1 on e_l and value 0 on other fine-grid edges. In (42), we define V_h^i and Q_h^i as the restrictions of the fine grid space in ω_i^ε and $V_{h,0}^i \subset V_h^i$ are functions that vanish on $\partial \omega_i^\varepsilon$. We remark that we choose the constant c in (42) by compatibility condition, $c = \frac{1}{|\omega_i^\varepsilon|} \int_{\partial \omega_i^\varepsilon / \partial \mathcal{B}^\varepsilon} u_l^i \cdot n_i \, ds$. We emphasize that, for the Stokes problem, we will solve (42) in both node-based coarse neighborhoods and

edge-based coarse neighborhoods (40). Collecting the solutions of the local problems generates the snapshot space, $\psi_l^{\omega_i^\varepsilon} = u_l^i$ in ω_i^ε :

$$V_{H,\text{snap}}^{\omega_i^\varepsilon} = \{\psi_l^{\omega_i^\varepsilon} : 1 \leq l \leq 2S_i, 1 \leq i \leq (N_e + N_v)\}.$$

One can reduce the cost of solving local problems by using the randomization techniques [9].

10.2.2 Offline Space

We perform a space reduction in the snapshot space through using a local spectral problem in ω_i^ε . We consider the following local eigenvalue problem in the snapshot space

$$A^{\omega_i^\varepsilon} \Psi_k^{\omega_i^\varepsilon} = \lambda_k^{\omega_i^\varepsilon} S^{\omega_i^\varepsilon} \Psi_k^{\omega_i^\varepsilon}, \quad (43)$$

where $(A^{\omega_i^\varepsilon})_{mn} = a_i(\psi_m^{\omega_i^\varepsilon}, \psi_n^{\omega_i^\varepsilon})$, $(S^{\omega_i^\varepsilon})_{mn} = s_i(\psi_m^{\omega_i^\varepsilon}, \psi_n^{\omega_i^\varepsilon})$ and $a_i(u, v) = \int_{\omega_i^\varepsilon} \nabla u : \nabla v$, $s_i(u, v) = \int_{\omega_i^\varepsilon} |\nabla \chi_i|^2 u \cdot v$ with χ_i specified below. Note that we solve the above spectral problem in the local snapshot space corresponding to the neighborhood domain ω_i^ε . We arrange the eigenvalues in an increasing order, choosing the first l_i eigenvalues and taking the corresponding eigenvectors $\Psi_k^{\omega_i^\varepsilon}$, for $k = 1, 2, \dots, l_i$, to form the basis functions, i.e., $\tilde{\Phi}_k^{\omega_i^\varepsilon} = \sum_j \Psi_{kj}^{\omega_i^\varepsilon} \psi_j^{\omega_i^\varepsilon}$, where $\Psi_{kj}^{\omega_i^\varepsilon}$ are the coordinates of the vector $\Psi_k^{\omega_i^\varepsilon}$. Then we define

$$\tilde{V}_{H,\text{off}}^{\omega_i^\varepsilon} = \text{span}\{\tilde{\Phi}_k^{\omega_i^\varepsilon}, \quad k = 1, 2, \dots, l_i\}. \quad (44)$$

For constructing the conforming offline space, we multiply the functions $\tilde{\Phi}_k^{\omega_i^\varepsilon} = (\tilde{\Phi}_{x_1,k}^{\omega_i^\varepsilon}, \tilde{\Phi}_{x_2,k}^{\omega_i^\varepsilon})$ by a partition of unity function χ_i . Note that discontinuous Galerkin methods will eliminate the need for χ_i . We remark that we define the partition of unity functions $\{\chi_i\}$ with respect to the coarse nodes and the mid-points of coarse edges. One can choose $\{\chi_i\}$ to be the standard multiscale finite element basis. However, upon multiplying by partition of unity functions, the resulting basis functions no longer have constant divergence, which affects the scheme's stability. Moreover, the multiplication by partition of unity functions may not honor the perforation boundary conditions as discussed before. To resolve the problem with the divergence, for each $\tilde{\Phi}_k^{\omega_i^\varepsilon}$ we find two functions $\Phi_{1,k}^{\omega_i^\varepsilon}$ and $\Phi_{2,k}^{\omega_i^\varepsilon}$ (these are vector functions) that solve two local optimization problems in every coarse-grid element $K_j^i \subset \omega_i^\varepsilon$:

$$\min \left\| \nabla \Phi_{1,k}^{\omega_i^\varepsilon} \right\|_{L^2(K_j^i)} \quad \text{such that} \quad \text{div}(\Phi_{1,k}^{\omega_i^\varepsilon}) = \frac{1}{|K_j^i|} \int_{\partial K_j^i} (\chi_i \tilde{\Phi}_{x_1,k}^{\omega_i^\varepsilon}, 0) \cdot n_{ij} \, ds \quad \text{in } K_j^i \quad (45)$$

with n_{ij} the outward normal vector to ∂K_j^i and the boundary condition $\Phi_{1,k}^{\omega_i^\varepsilon} = (\chi_i \tilde{\Phi}_{x_1,k}^{\omega_i^\varepsilon}, 0)$, on ∂K_j^i ; and

$$\min \left\| \nabla \Phi_{2,k}^{\omega_i^\varepsilon} \right\|_{L^2(K_j^i)} \quad \text{such that} \quad \text{div}(\Phi_{2,k}^{\omega_i^\varepsilon}) = \frac{1}{|K_j^i|} \int_{\partial K_j^i} (0, \chi_i \tilde{\Phi}_{x_2,k}^{\omega_i^\varepsilon}) \cdot n_{ij} \, ds \quad \text{in } K_j^i, \quad (46)$$

with the boundary condition $\Phi_{2,k}^{\omega_i^\varepsilon} = (0, \chi_i \tilde{\Phi}_{x_2,k}^{\omega_i^\varepsilon})$, on ∂K_j^i .

Combining them, we obtain the global offline space:

$$V_{H,\text{off}} = \text{span}\{\Phi_{1,k}^{\omega_i^\varepsilon} \text{ and } \Phi_{2,k}^{\omega_i^\varepsilon} : \quad 1 \leq i \leq (N_e + N_v) \text{ and } 1 \leq k \leq l_i\}.$$

Using a single index notation, we can write $V_{H,\text{off}} = \text{span}\{\Phi_i^{\text{off}}\}_{i=1}^{N_u}$, where $N_u = \sum_{i=1}^{N_e + N_v} l_i$. We will use this as an approximation space for the velocity. For coarse approximation of the pressure, we will take $Q_{H,\text{off}}$ to be the space of piecewise constant functions on the coarse mesh. Note that for many applications, the size of the coarse problem can be substantially small compared to the fine-scale problem. For example, for problems with scale separation, we can only use 1-2 basis functions. Once multiscale basis functions are constructed, the global coarse-grid problem can be solved on the coarse grid with pre-computed multiscale basis functions.

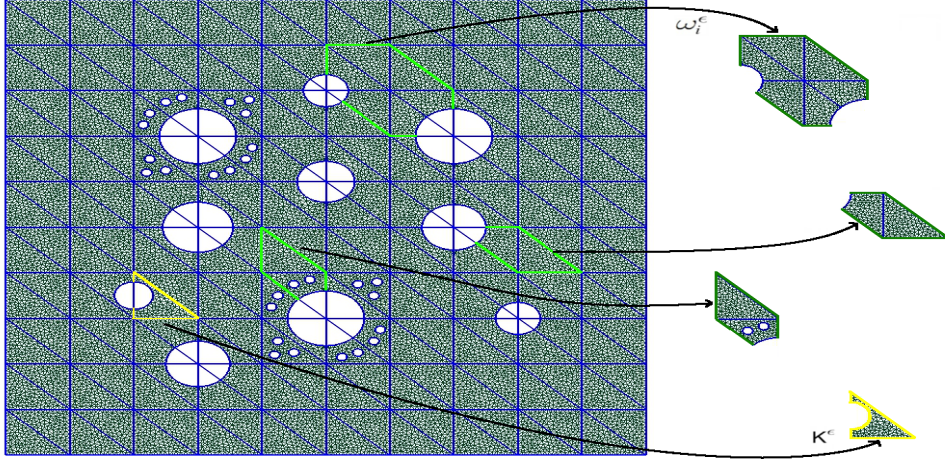


Figure 7: Heterogeneous perforated medium and the description of coarse and fine meshes.

10.2.3 Numerical results

We present a numerical result for Stokes equations in $\Omega = [0, 1] \times [0, 1]$. We consider a perforated domain as illustrated in Figure 7, where the perforated regions \mathcal{B}^ε are circular. We coarsely discretize the computational domain using uniform triangulation, where the coarse mesh size $H = \frac{1}{5}$ for Stokes problem. Note that our approach does not impose any geometry restriction on coarse meshes. Furthermore, we use nonuniform triangulation inside each coarse triangular element to obtain a finer discretization.

We consider the Stokes operator with zero velocity $u = (0, 0)$ on $\partial\Omega^\varepsilon \cap \partial\mathcal{B}^\varepsilon$ and $\frac{\partial u}{\partial n} = (0, 0)$ on $\partial\Omega$. For the fine-scale approximation of the Stokes problem, we use P^2 -elements for velocity and piecewise constants for pressure. The x_1 -component of the velocity solution is shown in Figure 8. We present the offline solution with one basis function and three basis functions. We observe that the offline velocity solution with one basis function (DOF=1082) is not able to capture the solution accurately. The L^2 -error is 10.1%. If we use three multiscale basis functions per coarse element (DOF=2846), we obtain a more accurate solution, the L^2 -error is only 1.02%. These results show that one needs a systematic and rigorous approach for enriching coarse-grid spaces in order to obtain accurate solutions using only a small number of basis functions.

11 Selected Applications

Previously, we discussed the main concept of the GMsFEM and some important ingredients. The GMsFEM can be used in various applications. Below, we discuss a few applications.

11.1 Two-phase flow

We present simulation results for two-phase flow and transport problems. We consider the two-phase flow problem with zero Neumann boundary condition

$$-\eta(S)\kappa\nabla p = v \text{ in } \Omega, \quad \text{div } v = f \text{ in } \Omega, \quad v \cdot n = 0 \text{ on } \partial\Omega, \quad (47)$$

where $\eta(S) = \frac{\kappa_{rw}(S)}{\mu_w} + \frac{\kappa_{ro}(S)}{\mu_o}$ and $\kappa_{rw}(S) = S^2$, $\kappa_{ro}(S) = (1-S)^2$, $\mu_w = 1$, $\mu_o = 5$. The saturation equation is given by

$$S_t + v \cdot \nabla F(S) = r,$$

where $F(S) = \frac{\kappa_{rw}(S)/\mu_w}{\kappa_{rw}(S)/\mu_w + \kappa_{ro}(S)/\mu_o}$. The above flow equation (47) is solved by the mixed GMsFEM, and the saturation equation is solved on the fine grid by the finite volume method. In our simulations, we take f to

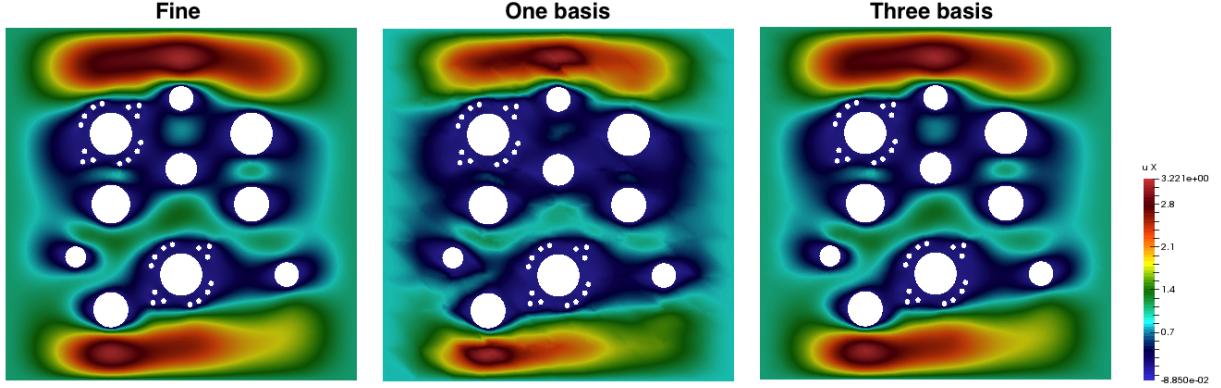


Figure 8: Stokes problem. Fine-scale and multiscale solutions for velocity u_1 in perforated domain (Figure 7). Left: fine-scale solution, $DOF = 171688$. Middle: multiscale solutions using 1 multiscale basis function for the velocity (L_2 error is 10.1%). Right: multiscale solutions with 3 multiscale basis function for the velocity (L_2 error is 1.02%).

be zero except for the top-left and bottom-right fine-grid elements, where f takes the values of 1 and -1 , respectively. Moreover, we set the initial value of S to be zero. For the source r , we also take it as zero except for the top-left fine element where $r = 1$.

For the simulations of two-phase flow, the mixed GMsFEM is used for the flow equations. The multiscale basis functions are computed at time zero using the unit mobility, i.e., $\eta = 1$. These multiscale basis function for the velocity field is used without any modification to compute the fine-scale velocity field. The fine-scale velocity field is further used to update the saturation file. We note that each time the flow equation is solved on a coarse grid, which provides a substantial computational saving. We present the numerical results for the saturation in Figure 9 for the permeability field shown in Figure 4. As we observe from this figure that with three basis functions, we obtain very good agreement (L^2 error is 1.8%).

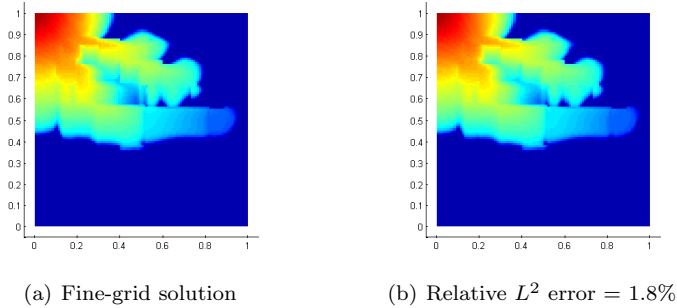


Figure 9: Saturation solution obtained by using v_o (10×10 coarse grid, 3 basis per coarse edge)

11.2 Flow in fractured media

In this section, we briefly discuss the application of the GMsFEM to flows in fractured media, in particular, the application to shale gas transport in fractured media. We remark that the study of flows in fractured

media has a long history (see, e.g., [57, 46]). We study nonlinear gas transport in fractured media (cf., [2])

$$a_z(c) \frac{\partial c}{\partial t} = \operatorname{div}(b_z(c, x) \nabla c), \quad (48)$$

where $z = m, f$ with m referring to the matrix phase and f referring to the fracture phase. The fractures have “zero” thickness and are represented on a fine-scale variational formulation as edge elements. More precisely, the domain Ω can be represented by $\Omega = \Omega_m \oplus_i d_i \Omega_{f,i}$. Here, d_i is the aperture of the i th fracture and i is the index of the fractures (see Figure 10). Then, the fine-scale equations have the form

$$\int_{\Omega_m} a_m(c) \frac{\partial c}{\partial t} v + \sum_i d_i \int_{\Omega_{f,i}} a_f(c) \frac{\partial c}{\partial t} v + \int_{\Omega_m} b_m(c, x) \nabla c \cdot \nabla v + \sum_i d_i \int_{\Omega_{f,i}} b_m(c, x) \nabla c \cdot \nabla v = 0. \quad (49)$$

In our numerical simulations, we solve the above problem using the GMsFEM framework for some specific parameter values that are specified below. We note that the fractures are modeled within snapshot space via harmonic extensions as we discussed above (see [1] for more details). In the method, the basis functions are constructed using the steady state with $b_z = 1$ in (48) and taking into account the fracture distribution. The problem (48) is solved by using implicit discretization and by linearizing c at the previous time step.

We use following parameters: $a_m(c) = \phi + (1 - \phi) \partial F / \partial c$; $b_m(c, x) = \phi D + (1 - \phi) D_s \partial F / \partial c + \phi \kappa RTc / \mu$; $a_f(c) = \phi_f$; $b_f(c, x) = \kappa_f RTc / \mu$; $D = 510^{-7} [m^2/s]$, $\phi = 0.04$, $T = 413 [K]$, $\mu = 2 \cdot 10^{-5} [kg/(m s)]$ and for fractures $k_f = 10^{-12} [m^2]$, $\phi_f = 0.001$. As for permeability κ , we use constant $\kappa = 10^{-18} [m^2]$. $p = RTc$, $p_c = 10^9 [Pa]$, $p_1 = 1.8 \cdot 10^9 [Pa]$, $\alpha = 0.5$ and $\mathcal{M} = 0.5$. For the sorbed gas, we use Langmuir model $F(c) = c_{\mu s} \frac{s}{(1+sc)^2}$, where $s = 0.26 \cdot 10^{-3}$ and $c_{\mu s} = 0.25 \cdot 10^{-5} [mol/m^3]$. We present the numerical results in Figure 10. In the left figure, we depict the fracture distributions, and then, show the fine-scale and the coarse-scale solution. We observe that with only 4 basis functions, we obtain a good agreement between the fine-scale and multiscale solution (L^2 error 0.34%, H^1 error 4.69%).

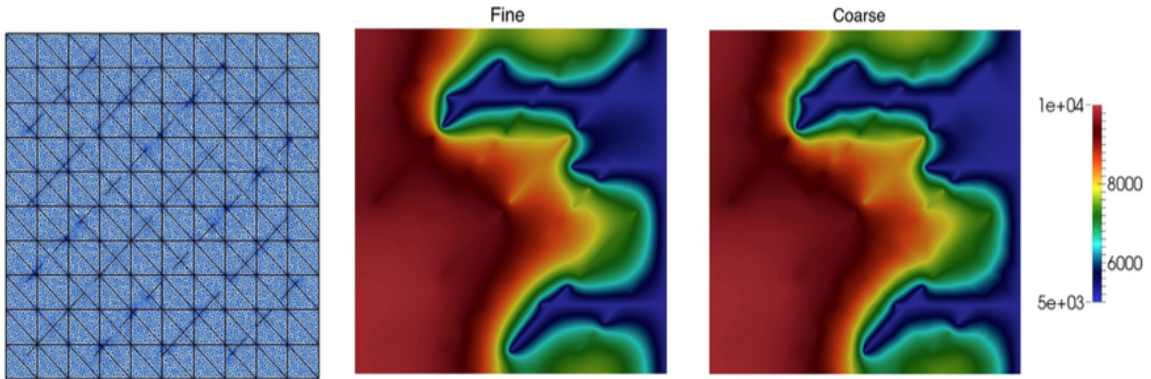


Figure 10: Left: the fracture distributions. Middle: the fine-scale solution. Right: the coarse-grid solution.

11.3 Other applications

We have demonstrated two applications of the GMsFEM. The GMsFEM can be used for other complex multiscale problems. For example, in [13], we have applied the GMsFEM for elastic wave propagation in fractured media. We have also used the GMsFEM for acoustic wave propagation [22]. The GMsFEM can also be effectively used in inverse problem, where the forward problem is solved many times. In [38], we apply the GMsFEM in uncertainty quantification in inverse problems.

12 Discussions

In this paper, we give an overview of multiscale finite element methods and discuss their generalizations. Due to page limitations, we do not present a comprehensive overview of multiscale methods and mention only some important aspects of the GMSFEM as a tool for performing adaptive multiscale model reduction. The application of the method to nonlinear problems is not discussed in the paper. We refer to ([17]) for the application to nonlinear problems, where we consider nonlinear eigenvalue problems, hybridization, and nonlinear interpolation ideas to efficiently solve these problems.

The proposed local reduced-order models can be used in performing global model reduction. The GMSFEM can be employed to construct the snapshots for the global model reduction methods. In this case, using reduced representation of the global snapshots, we can perform POD or another model reduction for the global modes. Moreover, we can use adaptively local approaches to update the global snapshots when online computations are needed. These global-local approaches can be effective in many applications [3].

Optimal multiscale basis functions can be obtained using the oversampling techniques, optimization, and singular value decomposition [51]. In this paper, we focus on a general adaptive strategy that can be easily adapted for various applications and discretization, as demonstrated. One can use the ideas of optimized multiscale basis functions as in [51] in the proposed framework.

The stochastic multiscale problems are challenging due to additional degrees of freedom due to randomness. For stochastic problems, one can use multiscale finite element framework with Monte Carlo techniques or separation of variables. A promising approach for solving multiscale stochastic problems is to use data driven stochastic method concepts as shown in [69].

Multiscale framework proposed in the paper can also be employed for the stabilization. For example, in [7], the authors use the GMSFEM to stabilize the convection-dominated problems by constructing multiscale test spaces.

Finally, we would like to remark that the proposed methods are well suited for the parallel computation, in particular, the task of multiscale basis computations is embarrassingly parallel. Our numerical results (using up to 1000 processors) on 3D perforated problems show that one can achieve almost perfect scaling in the parallel computations.

Acknowledgements. TYH's research was in part supported by DOE Grant DE-FG02-06ER257, and NSF Grant No. DMS-1318377, DMS-1159138. EC's research was in part supported by Hong Kong RGC General Research Fund Project No. 400813.

References

- [1] IY Akkutlu, Yalchin Efendiev, and Maria Vasilyeva. Multiscale model reduction for shale gas transport in fractured media. *arXiv preprint arXiv:1507.00113*, 2015.
- [2] Y Akkutlu, Y Efendiev, and V Savatarova. Multiscale asymptotic analysis of gas transport in shale matrix. *Transport in Porous Media*, pages 1–26, 2015.
- [3] M Alotaibi, VM Calo, Y Efendiev, JC Galvis, and M Ghommem. Global–local nonlinear model reduction for flows in heterogeneous porous media. *Computer Methods in Applied Mechanics and Engineering*, 292:122–137, 2015.
- [4] T. Arbogast, G. Pencheva, M.F. Wheeler, and I. Yotov. A multiscale mortar mixed finite element method. *SIAM J. Multiscale Modeling and Simulation*, 6(1):319–346, 2007.
- [5] M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera. An ‘Empirical Interpolation’ Method: Application to Efficient Reduced-Basis Discretization of Partial Differential Equations. *C.R. Acad. Sci. Paris, Ser. I*, 339:667–672, 2004.
- [6] S. Boyaval, C. LeBris, T. Lelièvre, Y. Maday, N. Nguyen, and A. Patera. Reduced basis techniques for stochastic problems. *Archives of Computational Methods in Engineering*, 17:435–454, 2010.

- [7] V. Calo, E. Chung, Y. Efendiev, and W. T. Leung. Multiscale stabilization for convection-dominated diffusion in heterogeneous media. *arXiv preprint arXiv:1509.06833*. to appear in CMAME.
- [8] VM Calo, Y Efendiev, J Galvis, and M Ghommem. Multiscale empirical interpolation for solving nonlinear PDEs. *Journal of Computational Physics*, 278:204–220, 2014.
- [9] VM Calo, Y Efendiev, J Galvis, and G Li. Randomized oversampling for generalized multiscale finite element methods. <http://arxiv.org/pdf/1409.7114.pdf>. to appear in SIAM MMS.
- [10] VM Calo, Y Efendiev, and JC Galvis. A note on variational multiscale methods for high-contrast heterogeneous porous media flows with rough source terms. *ADWR*, 34(9):1177–1185, 2011.
- [11] E. Candès et al. Compressive sampling. *Proceedings of the ICM*, 3:1433–1452, 2006.
- [12] S. Chaturantabut and D. C. Sorensen. Application of POD and DEIM to dimension reduction of nonlinear miscible viscous fingering in porous media. *Mathematical and Computer Modeling of Dynamical Systems*, 17:337–353, 2011.
- [13] E. Chung, Y. Efendiev, R. Gibson, and M. Vasilyeva. A generalized multiscale finite element method for elastic wave propagation in fractured media. 2015. submitted.
- [14] E. Chung, Y. Efendiev, T. W. Leung, and G. Li. Sparse generalized multiscale finite element methods and their applications. *arXiv preprint arXiv:1506.08509*. to appear in IJ Multiscale Computational Engineering.
- [15] E. Chung, Y. Efendiev, and W. T. Leung. An adaptive generalized multiscale discontinuous galerkin method (GMsDGM) for high-contrast flow problems. *arXiv preprint arXiv:1409.3474*, 2014.
- [16] E. Chung, Y. Efendiev, W.T. Leung, M. Vasilyeva, and Y. Wang. Adaptive online generalized multiscale finite element method for multiscale problems in perforated domain. *submitted*.
- [17] E. Chung, Y. Efendiev, K. Shi, and S. Ye. A multiscale model reduction method for nonlinear monotone elliptic equations in heterogeneous media. *submitted*.
- [18] E. Chung, Y. Efendiev, and T. Wing. Generalized multiscale finite element methods with energy minimizing oversampling. *submitted*.
- [19] E. Chung, Y. Efendiev, T. Wing, and S. Ye. Generalized multiscale finite element methods for space-time heterogeneous parabolic equations. *submitted*.
- [20] E. T. Chung, Y. Efendiev, G. Li, and M. Vasilyeva. Generalized multiscale finite element method for problems in perforated heterogeneous domains. *Applicable Analysis*, 255:1–15, 2015.
- [21] ET Chung, Y Efendiev, and CS Lee. Mixed generalized multiscale finite element methods and applications. *Multiscale Modeling & Simulation*, 13(1):338–366, 2015.
- [22] E.T. Chung, Y. Efendiev, and W.T. Leung. Generalized multiscale finite element method for wave propagation in heterogeneous media. to appear in SIAM MMS.
- [23] ET Chung, Y Efendiev, and WT Leung. Residual-driven online generalized multiscale finite element methods. *To appear in J. Comput. Phys.*
- [24] ET Chung, Y Efendiev, and WT Leung. An online generalized multiscale discontinuous Galerkin method (GMsDGM) for flows in heterogeneous media. *arXiv preprint arXiv:1504.04417*, 2015.
- [25] ET Chung, Y Efendiev, and G Li. An adaptive GMsFEM for high-contrast flow problems. *Journal of Computational Physics*, 273:54–76, 2014.

- [26] ET Chung, Y Efendiev, G Li, and M Vasilyeva. Generalized multiscale finite element methods for problems in perforated heterogeneous domains. *Applicable Analysis*, to appear, 2015.
- [27] W. Dorfler. A convergent adaptive algorithm for poisson’s equation. *SINUM*, 33:1106 – 1124, 1996.
- [28] M. Drohmann, B. Haasdonk, and M. Ohlberger. Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation. *SIAM J. Sci. Comput.*, 34(2):A937–A969, 2012.
- [29] W. E and B. Engquist. Heterogeneous multiscale methods. *Comm. Math. Sci.*, 1(1):87–132, 2003.
- [30] W.T. Leung E. Chung and S. Pollock. Goal-oriented adaptivity for GMsFEM. *Journal of Computational and Applied Mathematics*, 296:625–637, 2016.
- [31] Y. Efendiev and J. Galvis. A domain decomposition preconditioner for multiscale high-contrast problems. In Y. Huang, R. Kornhuber, O. Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering XIX*, volume 78 of *Lect. Notes in Comput. Science and Eng.*, pages 189–196. Springer-Verlag, 2011.
- [32] Y. Efendiev, J. Galvis, and T. Hou. Generalized multiscale finite element methods. *Journal of Computational Physics*, 251:116–135, 2013.
- [33] Y Efendiev, J Galvis, R Lazarov, M Moon, and M Sarkis. Generalized multiscale finite element method. Symmetric interior penalty coupling. *Journal of Computational Physics*, 255:1–15, 2013.
- [34] Y. Efendiev, J. Galvis, R. Lazarov, and J. Willems. Robust domain decomposition preconditioners for abstract symmetric positive definite bilinear forms. *ESIAM : M2AN*, 46:1175–1199, 2012.
- [35] Y. Efendiev, J. Galvis, G. Li, and M. Presho. Generalized multiscale finite element methods. Oversampling strategies. *International Journal for Multiscale Computational Engineering*, *accepted*, 2013.
- [36] Y. Efendiev, V. Ginting, T. Hou, and R. Ewing. Accurate multiscale finite element methods for two-phase flow simulations. *Journal of Computational Physics*, 220:155–174, 2006.
- [37] Y. Efendiev and T. Hou. *Multiscale Finite Element Methods: Theory and Applications*. Springer, 2009.
- [38] Y. Efendiev, B. Jin, M. Presho, and X. Tan. Multilevel markov chain monte carlo method for high-contrast single-phase flow problems. *Communications in Computational Physics*, 17(01):259–286, 2015.
- [39] Y. Efendiev, R. Lazarov, M. Moon, and K. Shi. A spectral multiscale hybridizable discontinuous Galerkin method for second order elliptic problems. *Computer Methods in Applied Mechanics and Engineering*, 292:243–256, 2015.
- [40] Y Efendiev, A Pankov, et al. Homogenization of nonlinear random parabolic operators. *Advances in Differential Equations*, 10(11):1235–1260, 2005.
- [41] Jacob Fish. *Practical multiscaleing*. John Wiley & Sons, 2013.
- [42] J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high contrast media. Reduced dimensional coarse spaces. *SIAM J. Multiscale Modeling and Simulation*, 8:1621–1644, 2010.
- [43] J Galvis, G Li, and K Shi. A generalized multiscale finite element method for the Brinkman equation. *Journal of Computational and Applied Mathematics*, 280:294–309, 2015.
- [44] J. Galvis and J. Wei. Ensemble level multiscale finite element and preconditioner for channelized systems and applications. *Journal of Computational and Applied Mathematics*, 255:456–467, 2014.

- [45] M. Griebel and M. Klitz. Homogenization and numerical simulation of flow in geometries with textile microstructures. *Multiscale Modeling & Simulation*, 8(4):1439–1460, 2010.
- [46] H. Hajibeygi, D. Kavounis, and P. Jenny. A hierarchical fracture model for the iterative multiscale finite volume method. *Journal of Computational Physics*, 230(4):8729–8743, 2011.
- [47] Patrick Henning, Axel Malqvist, and Daniel Peterseim. A localized orthogonal decomposition method for semi-linear elliptic problems. *arXiv preprint arXiv:1211.3551*, 2012.
- [48] Patrick Henning and Mario Ohlberger. The heterogeneous multiscale finite element method for elliptic homogenization problems in perforated domains. *Numerische Mathematik*, 113(4):601–629, 2009.
- [49] M. Hinze and S. Volkwein. POD surrogate models for nonlinear dynamical systems: error estimates and suboptimal control. volume 45 of *Lecture Notes in CSE*, pages 261–306. 2005.
- [50] T. Hou and X.H. Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, 134:169–189, 1997.
- [51] Thomas Y Hou and Pengfei Liu. Optimal local multi-scale basis functions for linear elliptic equations with rough coefficient. *arXiv preprint arXiv:1508.00346*, 2015.
- [52] T.Y. Hou, X.H. Wu, and Z. Cai. Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients. *Math. Comp.*, 68:913–943, 1999.
- [53] TJR Hughes, G Feijoo, L Mazzei, and J Quincy. The variational multiscale method—a paradigm for computational mechanics. *Comput. Methods Appl. Mech. Engrg.*, 166:3–24, 1998.
- [54] P. Jenny, S.H. Lee, and H. Tchelep. Multi-scale finite volume method for elliptic problems in subsurface flow simulation. *J. Comput. Phys.*, 187:47–67, 2003.
- [55] V. Jikov, S. Kozlov, and O. Oleinik. *Homogenization of differential operators and integral functionals*. Springer Science & Business Media, 2012.
- [56] C. Le Bris, F. Legoll, and A. Lozinski. An MsFEM type approach for perforated domains. *Multiscale Modeling & Simulation*, 12(3):1046–1077, 2014.
- [57] S.H. Lee, M.F. Lough, and C.L. Jensen. Hierarchical modeling of flow in naturally fractured formations with multiple length scales. *Water resources research*, 37(3):443–455, 2001.
- [58] Ana-Maria Matache and Christoph Schwab. Two-scale fem for homogenization problems. *ESAIM: Mathematical Modelling and Numerical Analysis*, 36(04):537–572, 2002.
- [59] V. Maz`ya, A. Movchan, and M. Nieves. *Green Kernels and Meso-Scale Approximations in Perforated Domains*. Springer-Berlin, Lecture Notes in Mathematics, 2077, 2013.
- [60] Jens M Melenk and Ivo Babuška. The partition of unity finite element method: basic theory and applications. *Computer methods in applied mechanics and engineering*, 139(1):289–314, 1996.
- [61] P. Ming and P. Zhang. Analysis of the heterogeneous multiscale method for parabolic homogenization problems. *Mathematics of Computation*, 76(257):153–177, 2007.
- [62] H. Owhadi and L. Zhang. Metric-based upscaling. *Comm. Pure. Appl. Math.*, 60:675–723, 2007.
- [63] Aleksandr Andreevich Pankov. *G-convergence and homogenization of nonlinear partial differential operators*, volume 422. Springer Science & Business Media, 2013.
- [64] G Papanicolau, A Bensoussan, and J-L Lions. *Asymptotic analysis for periodic structures*. Elsevier, 1978.

- [65] Grigorios A Pavliotis and Andrew Stuart. *Multiscale methods: averaging and homogenization*. Springer Science & Business Media, 2008.
- [66] Enrique Sánchez-Palencia. Non-homogeneous media and vibration theory. In *Non-homogeneous media and vibration theory*, volume 127, 1980.
- [67] Hayden Schaeffer, Russel Caffisch, Cory D Hauck, and Stanley Osher. Sparse dynamics for partial differential equations. *Proceedings of the National Academy of Sciences*, 110(17):6634–6639, 2013.
- [68] X.H. Wu, Y. Efendiev, and T.Y. Hou. Analysis of upscaling absolute permeability. *Discrete and Continuous Dynamical Systems, Series B.*, 2:158–204, 2002.
- [69] Zhiwen Zhang, Maolin Ci, and Thomas Y Hou. A multiscale data-driven stochastic method for elliptic pdes with random coefficients. *Multiscale Modeling & Simulation*, 13(1):173–204, 2015.