Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **10**, No. 6, pp. 1527-1548

DOI: 10.4208/aamm.OA-2018-0073
December 2018

Balanced Truncation Based on Generalized Multiscale Finite Element Method for the Parameter-Dependent Elliptic Problem

Shan Jiang[1,2,*], Anastasiya Protasov[3] and Meiling Sun[4]

[1] *School of Science, Nantong University, Nantong 226019, China*
[2] *Department of Mathematics, Institute for Scientific Computation, Texas A&M University, College Station, TX 77843, USA*
[3] *Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005, USA*
[4] *Department of Public Courses, Nantong Vocational University, Nantong 226007, China*

Received 13 March 2018; Accepted (in revised version) 28 June 2018

Abstract. In this paper, we combine the generalized multiscale finite element method (GMsFEM) with the balanced truncation (BT) method to address a parameter-dependent elliptic problem. Basically, in progress of a model reduction we try to obtain accurate solutions with less computational resources. It is realized via a spectral decomposition from the dominant eigenvalues, that is used for an enrichment of multiscale basis functions in the GMsFEM. The multiscale bases computations are localized to specified coarse neighborhoods, and follow an offline-online process in which eigenvalue problems are used to capture the underlying system behaviors. In the BT on reduced scales, we present a local-global strategy where it requires the observability and controllability of solutions to a set of Lyapunov equations. As the Lyapunov equations need expensive computations, the efficiency of our combined approach is shown to be readily flexible with respect to the online space and an reduced dimension. Numerical experiments are provided to validate the robustness of our approach for the parameter-dependent elliptic model.

AMS subject classifications: 35J25, 65N12, 65N30

Key words: Generalized multiscale method, balanced truncation, parameter dependent, eigenvalue decomposition, Lyapunov equation.

1 Introduction

A variety of research has been devoted to the developments of model reduction for high simulation, optimal control, and engineering design. Many scientific applications in-

*Corresponding author.
Email: jiangshan@ntu.edu.cn (S. Jiang)

clude the flow models in porous media and conduction models in composite materials. These problems are inherently referred to as multiscale in nature, and they exhibit heterogeneous and high contrast behaviors. Traditional methods such as finite difference method, finite element method, or discontinuous Galerkin method, would require the use of very fine meshes to fully resolve the multiscale nature of media. To an end, the practical applications are limited by the computational power. Therefore the interest in developing efficient multiscale and model reduction methods for more computational efficiency and accuracy are of particular interest.

There are many literatures on model reductions, see e.g. [1–3]. Model reduction performs the discretization on a coarse grid, and it may be capable of incorporating fine-scale features into coarse-scale schemes. Multiscale methods are used to solve a variety of computational models. The construction of coarse space is involved in which solutions are sought with the span of multiscale basis functions. Multiscale Finite Element Method (MsFEM, [4]), Heterogeneous Multiscale Method (HMM, [5]) and related approaches [6–10] work for a variety of applications and offer an advantage in parallel computings.

Progress has been made in recent years in developing the multiscale computations. The Generalized Multiscale Finite Element Method (GMsFEM, see [11]) is a generalization of the standard MsFEM, which uses appropriate snapshots and spectral decompositions for the additional enrichment of multiscale basis functions. Its main idea is to systematically enrich the initial coarse space with the eigenvectors of local spectral problems, and gradually accounts for more fine scale details. The enrichment is performed on a spectral decomposition by the fact that its computational efficiency has been validated the online space construction for any input parameter is fast and it can be reused for any force and boundary. Proper Orthogonal Decomposition (POD, see [2,12–14]), is effective to be used to find a low rank approximation to a Hilbert space, which is spanned by the snapshots. In the case of matrix approximation, POD is basically Singular Value Decomposition (SVD). And POD has been applied widely for a number of linear and nonlinear problems.

The methodology of GMsFEM has been used in many recent studies [15–25]. Chung, Efendiev and Li [15] derived an a-posteriori error indicator to develop an adaptive enrichment for high-contrast flow models. A parameter-dependent, single-phase flow is studied in [16], in which GMsFEM is used as a local model reduction, and BT is used as a global model reduction. Hou and Liu [20] provided the harmonic multiscale basis functions with an optimal approximation, and through singular value decompositions of some oversampling operators a good efficiency is achieved. In [22] the GMsFEM is combined with variable-separation techniques, and it presents an iterative algorithm for solving the parameter independent PDEs repeatedly. Jiang and Li [24] proposed a model sparse representation based on the mixed GMsFEM with elliptic random inputs, which improves the online computation and the problem output. Transport flow problem in perforated domains are considered in [25], a mixed Petrov-Galerkin GMsFEM formulation is used to guarantee mass conservation and stability in model reductions.

Balanced Truncation (BT, see [1, 2, 12, 14]) is a global model reduction technique that one can make use of an a-priori error bound and solution stability, from a full system to a reduced system. This skill involves retaining both the most controllable and observable states of the system, then the weak states would be truncated off and the reduced model is solved economically. It recasts the partial differential equations into the system frameworks, in which an input-output mapping is built. To determine the most dominant state variables, a coupled set of Lyapunov equations is to be solved for the observability and controllability Gramians, whose combination is used for constructing a reduced system via the truncated eigenvectors of specified matrices. Since the associated eigenvalues decay rapidly, the measurable outputs can be well approximated by solving a reduced model. However, a disadvantage of BT is that it could be too expensive to compute from a fully-large system directly. As such, an approach that is based on the GMsFEM with coarse-scale and applying the BT with reduced-scale is to be presented for the problem.

In this paper when considering a parameter-dependent model that may require uncertainty quantifications, in order to avoid direct eigenvalue computations the coarse space enrichment is divided into offline-online stages. The main goal is to allow for an efficient online space construction for each fixed parameter. In return, it is not necessary to recompute all the eigenvectors for separate realizations. The offline space is created by initially producing a set of snapshot functions in which local problems are solved on each coarse neighborhood for specified parameters. Then the first offline stage involves forming a larger-dimensional (as compared to the online space) parameter-independent offline space. This offline space accounts for a range of parameters that may be used in the second online stage, and its construction constitutes a one-time preprocessing step. After that at the online stage, we solve the balanced truncation eigenvalue problems for given parameters in order to achieve the desired strength finally.

An outline of our proposed method is presented for the procedure.

1. GMsFEM computations:

    1.1. Offline process.

        1.1.0. Coarse grid generation.

        1.1.1. Construction of snapshot space that will be used to compute an offline space.

        1.1.2. Construction of a small dimensional offline space by performing dimension reduction in the global snapshot space.

    1.2. Online process.

        1.2.1. For each input parameter, compute multiscale basis functions.

        1.2.2. Solution of coarse sub-problems for any force term and boundary condition.

        1.2.3. Offline-online POD coupling completes a coarse model reduction in the GMsFEM.

2. BT computations:

   2.1. Global model reduction further on the basis of available coarse scale, it provides the most important state variables for Lyapunov equations in a input-output mapping.

   2.2. Construction of a reduced system via the truncated eigenvectors for specified and reduced block matrices, and the associated eigenvalues decay rapidly. A local-global model reduction method is well proposed for the measurable outputs in our parameter-dependent PDEs.

The paper is organized as follows. In Section 2 the model is introduced and a preliminary description is given. In Section 3 detailed descriptions in the GMsFEM along with the BT are provided for local-global reductions. Numerical experiments are presented in Section 4 to validate the performance. And conclusions are given in Section 5.

2 A preliminary description of the model

2.1 Parameter-dependent elliptic problem

A two dimensional parameter-dependent elliptic equation with high contrast coefficients is considered. Finding a solution u to

$$-\mathrm{div}\big(\kappa(\mathbf{x};\mu)\nabla u\big)=f \quad \text{in } \Omega, \tag{2.1}$$

where $\kappa(\mathbf{x};\mu)$ is contrast coefficients in 2D space $\mathbf{x}$, μ is used to represent the parameter dependence and may be random uncertainty. The right side is f, and the problem may be subject to different boundary conditions. Ω is a domain in $\mathbb{R}^2$ and its boundary is $\partial\Omega$.

We divide the domain Ω into a set of quadrilateral finite elements on a coarse grid, which is denoted by $\mathcal{T}^H$. $\{y_i\}_{i=1}^{N_v}$ denotes the coarse grid vertices (N_v is the coarse vertices number), and define the neighborhood of a node y_i by

$$\omega_i=\bigcup\{K_j\in\mathcal{T}^H;\ y_i\in\overline{K_j}\}, \tag{2.2}$$

see Fig. 1. Then we refine every coarse element K on the coarse grid $\mathcal{T}^H$ into more fine quadrilateral elements k, and denote a fine grid by $\mathcal{T}^h$. From on, N, M denotes the coarse and fine (of every coarse element) partition number in each direction, and $H(=\frac{1}{N})$, $h(=\frac{H}{M})$ denotes the coarse and fine mesh size, respectively.

The standard finite element space is built through using piecewise bilinear polynomials $\mathbb{Q}^1(k)$ for 2D, and is subordinated to $\mathcal{T}^h$ as

$$V^h=\{v\in H^1(\Omega):\ v_{|_k}\in\mathbb{Q}^1(k),\ \forall k\in\mathcal{T}^h\}. \tag{2.3}$$

In Galerkin weak formulation, we find $u\in V^h$ such that

$$a(u,v)=(f,v),\quad \forall v\in V^h, \tag{2.4}$$

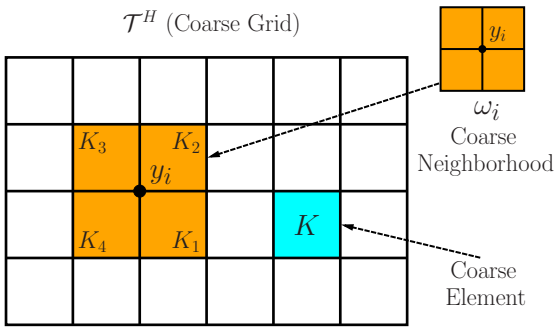

Figure 1: Illustration of coarse grid, element and neighborhood.

where

$$a(u,v)=\int_\Omega \kappa(\mathbf{x};\mu)\nabla u\cdot\nabla v,\quad (f,v)=\int_\Omega fv.$$

With a very fine partition number in the standard finite element method, it results in a huge discrete system for solving expensively as

$$A_f(\mu)u_f=F_f, \tag{2.5}$$

where

$$A_f(\mu)=[a(\mu)_{mn}]=\Sigma\int_k \kappa(\mathbf{x};\mu)\nabla\psi_m\cdot\nabla\psi_n,\quad F_f=[f_m]=\Sigma\int_k f\psi_m,$$

and ψ_m is the standard bilinear basis function defined on each element k. Here the subscript f means that matrices are on a very fine grid, which is different from the subscript c/r for the coarse/reduced case later. Now in this way, the stiffness matrix A_f is assembled from all fine elements, whose size is $N_f\times N_f$, here $N_f=(NM+1)^2$ for the two-dimensional model. We build B, C as the nodal matrices (they are transposed to each other $B=C^T$, the size of B is $N_f\times N^2$), which are allocated to the domain boundary vertices.

A main goal of the paper is to fulfill a suitable model reduction of size N_r such that $N_r\ll N_f$ for computational efficiency, and to preserve a good performance for the model (2.1).

2.2 Local-global reduction approach

First, we recall the standard multiscale finite element method. A homogeneous subproblem is solved for the multiscale basis functions ϕ in coarse neighborhoods from

$$\begin{cases} -\mathrm{div}(\kappa(\mathbf{x};\mu_j)\nabla\phi_{l,j}^{\omega_i})=0 & \text{in } \omega_i,\\ \phi_{l,j}^{\omega_i}=\delta_{l,j} & \text{on } \partial\omega_i, \end{cases} \tag{2.6}$$

where μ_j $(j=1,\cdots,J)$ is a set of specified parameters, and the boundary condition $\delta_{l,j}(y_i)=1$ when $(i=j)$, $\delta_{l,j}(y_i)=0$ when $(i\neq j)$. In this way, some useful local information are tracked through the l-th basis function ϕ_l in the form of coarse matrices. Although this standard multicale method works in a range of situations (see [4, 5, 7, 8]), while for parameter-dependent problems with high contrast it would show infertilities. This encourages us to pursue a powerful strategy for the complicated multiscale reduction.

To be more specific, we would build a so-called mapping matrix which is carrying on the necessary microscopic information. Details are to be provided later in (3.12) of Section 3, the mapping matrix is $R\in\mathbb{R}^{N_c\times N_f}$ (whose row size $N_c\ll N_f$) satisfying

$$A_c(\mu)u_c=F_c, \tag{2.7}$$

where $A_c(\mu)=RA_f(\mu)R^T$, $F_c=RF_f$, and the size of A_c is $N_c\times N_c$. Here subscript c means on the coarse grid we apply the standard multiscale/generalized multiscale method to get a numerical solution. It should be noted that in the latter generalized multiscale scheme we should carry out a necessary eigenvalue computation, which makes the matrix R have a bit larger size. Through enriching the dominant eigenfunctions it guarantees a higher resolution. To this point, on the basis of original fine matrices B,C whose corresponding coarse ones are formed as $B_c=RB$, $C_c=CR^T$.

Next is to carry out the matrices reduction further as

$$A_r(\mu)u_r=F_r, \tag{2.8}$$

where $A_r(\mu)=UA_c(\mu)V$, $F_r=UF_c$, whose size of A_r is $N_r\times N_r$, and N_r $(\ll N_c\ll N_f)$ can be chosen in Section 4. Here U, V are from a singular value decomposition to be defined in (3.28), with $UV=I_{N_r}$. Set $B_r=UB_c$, $C_r=C_cV$, these matrices and vectors are reduced again to save computational costs (especially in Lyapunov equations for observability and controllability). While at the same time, the simulation accuracy is shown from our novel method.

3 GMsFEM and balanced truncation

The GMsFEM [11] is a general framework that generalizes the MsFEM [4] in the sense that multiple basis functions can be systematically added to each coarse region, by flexibly enriching the coarse spaces via necessary eigenvalue computations.

In this generalized method, we first use a coarse partition and calculate a snapshot space in each coarse neighborhood. We should perform an appropriate spectral decomposition and select multiscale basis functions which are corresponding to the dominant eigenvalues. In this way the snapshot space is built to represent the solution space, and they may include all possible local fine-grid functions or harmonic functions. In the following, we just make options for an optimal subspace with respect to the acceptable tolerances.

The generalized multiscale method can be divided into two stages: offline and online. At the offline stage, a small dimensional construction uses the snapshot space and requires solving local spectral decompositions. The space can be re-used to capture the essential fine-scale features. Then a space reduction is performed and the dominant modes are acted as the multiscale basis functions, for any input parameter which is fitting our parameter-dependent problem perfectly. At the online stage, when a force term and boundary condition are given, the above offline basis functions are used to obtain an approximate solution. One can also adaptively select the basis functions in various coarse regions, in order to seek a better efficiency and accuracy.

The theory behind the offline and online spaces is the selection of spectral subproblems and the snapshot space construction. Many multiscale problems with high contrast may have very small eigenvalues and thus, we would like to include the eigenvectors associated with small eigenvalues to be surely enriched in the coarse space. Consequencely, the multiscale basis functions are constructed via multiplying the dominant eigenmodes by the partition of unity functions.

On the other hand, a direct approach of balanced truncation [2, 13] to a parameter-dependent problem would be prohibitively expensive or even not feasible since Lyapunov equations are to be solved. In particular, solving these equations on the fine scale from varying permeability, boundary condition, and forcing term will quickly become prohibitive. In turn, in this paper we propose the uses of a generalized multiscale and offline-online balanced truncation in order to obtain the inexpensive and accurate results.

Our balanced reduction is based on the GMsFEM. It allows to construct an independent set of online basis functions in which a coarse solution may be sought. As a result, the global system would be posed on a coarse scale to avoid to implement the balanced truncation on a fine scale directly. After two stages of the GMsFEM, the process yields a small dimension subspace for each fixed parameter μ in essence. As such, multiscale coarse solutions may be efficiently computed for specified inputs. We note that this contribution is served to further decrease the computational cost, which is associated with the balanced truncation of model reductions.

3.1 Generalized multiscale finite element method

3.1.1 Offline process

At this stage, we first construct a snapshot space subordinated to each coarse neighborhood ω_i in the domain. The snapshot space construction involves solving the local problems from a set of input parameters. Being different from the standard multiscale method for solving the homogeneous local problem (2.6), now an eigenvalue decomposition is introduced in this generalized multiscale method. First, we solve snapshot functions from a spectral problem: to find $(\lambda_{l,j}^{\omega_i,\text{snap}},\phi_{l,j}^{\omega_i,\text{snap}})\in\mathbb{R}\times V^{\omega_i,\text{snap}}$, such that

$$-\text{div}(\kappa(\mathbf{x};\mu_j)\nabla\phi_{l,j}^{\omega_i,\text{snap}})=\lambda_{l,j}^{\omega_i,\text{snap}}\phi_{l,j}^{\omega_i,\text{snap}}\quad\text{in }\ \omega_i, \tag{3.1}$$

note that zero Neumann boundary condition is applied to solve eigenvalue problems, except in the discontinuous case when Dirichlet condition is used on the global domain boundaries. It is well known that solving all the eigenfunctions of a large matrix are exhausting for (3.1), so we keep the first L_i eigenfunctions that correspond to the dominant eigenvalues $\lambda_{l,j}$ $(l=1,\cdots,L_i)$ in order to construct a snapshot space. The snapshot space $V^{\omega_i,\text{snap}}$ related to the neighborhood ω_i, is generated from the multiscale eigenfunctions as

$$V^{\omega_i,\text{snap}}=\text{span}\{\phi_{l,j}^{\omega_i,\text{snap}}:\, l=1,\cdots,L_i,\, j=1,\cdots,J\}. \tag{3.2}$$

Next we reorder the snapshot functions, through using a single index as the matrix columns to create an updated

$$R^{\omega_i,\text{snap}}=\left[\phi_1^{\omega_i,\text{snap}},\cdots,\phi_{L_{\text{snap}}}^{\omega_i,\text{snap}}\right], \tag{3.3}$$

where L_{snap} denotes the functions number that are kept in the snapshot matrix construction. For brevity, we omit the superscript ω_i from V^{snap}, R^{snap}. But it is assumed throughout the paper, the offline and online computations are localized to the respective coarse neighborhoods.

The discretized form of local problem (3.1) is

$$A(\mu_j)\phi_{l,j}^{\omega_i,\text{snap}}=\lambda_{l,j}^{\omega_i,\text{snap}}M(\mu_j)\phi_{l,j}^{\omega_i,\text{snap}} \quad \text{in } \omega_i, \tag{3.4}$$

where

$$A(\mu_j)=\int_{\omega_i}\kappa(\mathbf{x};\mu_j)\nabla\psi_m\cdot\nabla\psi_n, \quad M(\mu_j)=\int_{\omega_i}\widetilde{\kappa}(\mathbf{x};\mu_j)\psi_m\psi_n,$$

and ψ_m denotes the standard bilinear, fine-scale basis functions, $\widetilde{\kappa}$ will be defined in the next Subsection (3.9).

Next, we perform a dimension reduction of the snapshot space in using an auxiliary spectral decomposition in order to construct an offline space. To be specific, we seek a subspace of the snapshot space such that it can approximate any element in an appropriate sense. The main objective is to efficiently construct a set of multiscale basis functions for each parameter μ. We reiterate the offline bilinear forms to be parameter-independent, such that there is no need to reconstruct the offline space for each value μ. It follows that the eigenvalue problem can be described as

$$A^{\text{off}}\Phi_l^{\text{off}}=\lambda_l^{\text{off}}M^{\text{off}}\Phi_l^{\text{off}}, \tag{3.5}$$

where

$$A^{\text{off}}=\int_{\omega_i}\overline{\kappa}(\mathbf{x};\mu)\nabla\phi_m^{\text{snap}}\cdot\nabla\phi_n^{\text{snap}}=R^{\text{snap}}\overline{A}(\mu)(R^{\text{snap}})^T,$$
$$M^{\text{off}}=\int_{\omega_i}\overline{\widetilde{\kappa}}(\mathbf{x};\mu)\phi_m^{\text{snap}}\phi_n^{\text{snap}}=R^{\text{snap}}\overline{M}(\mu)(R^{\text{snap}})^T,$$

here $\overline{\kappa}(\mathbf{x};\mu)$ and $\overline{\widetilde{\kappa}}(\mathbf{x};\mu)$ are domain-based averaged coefficients. $\overline{A}$ and $\overline{M}$ denote similar fine scale matrices as defined in (3.4), except that averaged coefficients are used. The smallest L_{off} eigenvalues are picked up in (3.5) and form the corresponding eigenvectors by setting $\phi_l^{\text{off}}=\sum_j \Phi_{l,j}^{\text{off}}\phi_j^{\text{snap}}$ ($l=1,\cdots,L_{\text{off}}$, $j=1,\cdots,L_{\text{snap}}$), where $\Phi_{l,j}^{\text{off}}$ are the coordinates of the vector Φ_l^{off}. In this way we create the offline matrix

$$R^{\text{off}}=\left[\phi_1^{\text{off}},\cdots,\phi_{L_{\text{off}}}^{\text{off}}\right]. \tag{3.6}$$

3.1.2 Online process

With the available offline space, next we construct the associated online space on each coarse neighborhood for a specified parameter μ. In principle, we assume the online space is a small dimensional subspace for computational efficiency. We reiterate at the online stage, the reduced-order bilinear forms are chosen to be parameter-dependent. The following eigenvalue problem is obtained

$$A^{\text{on}}(\mu)\Phi_l^{\text{on}}=\lambda_l^{\text{on}}M^{\text{on}}(\mu)\Phi_l^{\text{on}}, \tag{3.7}$$

where

$$\begin{aligned}A^{\text{on}}(\mu)&=\int_{\omega_i}\kappa(\mathbf{x};\mu)\nabla\phi_m^{\text{off}}\cdot\nabla\phi_n^{\text{off}}=R^{\text{off}}A(\mu)(R^{\text{off}})^T,\\ M^{\text{on}}(\mu)&=\int_{\omega_i}\widetilde{\kappa}(\mathbf{x};\mu)\phi_m^{\text{off}}\phi_n^{\text{off}}=R^{\text{off}}M(\mu)(R^{\text{off}})^T,\end{aligned}$$

here $\kappa(\mathbf{x};\mu)$ and $\widetilde{\kappa}(\mathbf{x};\mu)$ now depend on the specified μ value, $A(\mu)$ and $M(\mu)$ are specified fine scale matrices in (3.4).

Finally, to generate the online space we choose the smallest L_{on} eigenvalues from (3.7) and finalize the corresponding eigenvectors by setting $\phi_l^{\text{on}}=\sum_j\Phi_{l,j}^{\text{on}}\phi_j^{\text{off}}$ ($l=1,\cdots,L_{\text{on}}$, $j=1,\cdots,L_{\text{off}}$), where $\Phi_{l,j}^{\text{on}}$ are the coordinates of the vector Φ_l^{on}. It should be noted that the matrices size in (3.7) solely depends on the dimension of the offline subspace. This results in the declining cost of the online space in contrast to the more expensive original, one is using the localized fine computations directly.

3.1.3 Offline-online POD coupling

In order to enrich the online basis functions into a reduced-order global formulation of (2.1), we denote χ_i as the standard multiscale partition of unity functions which are from

$$\begin{cases}-\text{div}\big(\kappa(\mathbf{x};\mu)\nabla\chi_i\big)=0 & \text{in } K\in\omega_i,\\ \chi_i=g_i & \text{on } \partial K,\end{cases} \tag{3.8}$$

where g_i is a continuous boundary condition and is bilinear on each edge of ∂K. Using the initial partition of unity we define the summed pointwise energy $\widetilde{\kappa}$ as

$$\widetilde{\kappa}=\kappa\sum_{i=1}^{N_v}H^2|\nabla\chi_i|^2. \tag{3.9}$$

Again, H is the coarse mesh size and N_v is the coarse vertices number. In order to construct the global coarse-grid solution space, we multiply the partition of unity functions by the online eigenfunctions to form the final multiscale basis functions

$$\phi_{i,l}=\chi_i\phi_l^{\text{on}},\quad i=1,\cdots,N_v,\quad l=1,\cdots,L_{\text{on}}. \tag{3.10}$$

Then the online space follows

$$V^{\text{on}}=\text{span}\{\phi_{i,l}:\ i=1,\cdots,N_v,\ l=1,\cdots,L_{\text{on}}\}. \tag{3.11}$$

Using a single index notation we denote $V^{\text{on}}=\text{span}\{\phi_k\}_{k=1}^{N_c}$, where N_c denotes the total number of multiscale basis functions in the coarse scale formulation. Recalling in (2.7) we now finalize the mapping matrix R to be used in the global reduction

$$R=[\phi_1,\cdots,\phi_{N_c}], \tag{3.12}$$

where ϕ_k represents the nodal vector of each generalized multiscale basis function. This matrix R (whose size is $N_c\times N_f$) is now plugged in (2.7), which can be enriched with the most dominant eigenfunctions and meaningful microscopic behaviors. This would lead to simulation advantages later on.

This completes the coarse model reduction in the GMsFEM. In the next subsection, we would like to carry on a further model reduction in a BT.

3.2 Balanced truncation for further reduction

3.2.1 Lyapunov equation and its property

Lyapunov equation is of scientific importance in model reduction, control theory, stochastic analysis, and so on. Large matrices would arise from a large system directly. For example, the discretization by the standard finite element method on very fine grid results in a Lyapunov equation with the stiffness matrix A_f, whose size $N_f\times N_f$ would be very large. A major challenge for solving the problem is that the unknown dense matrix W has N_f^2 entries. The huge storage and solution cost for Lyapnov equations would be problematic, see [26–28].

In this paper, we are focused on a new local-global model reduction method for the combination of GMsFEM and balanced truncation method. Since we start to solve the Lyapunov equation on the basis of GMsFE coarse neighborhoods for A_c, instead of on the very fine elements for A_f, in this way a great deal of storage costs have been saved. Then the balanced truncation is used to solve a unknown matrix W. For simplicity we omit subscripts of above matrices A_f or A_c in the following.

For the standard Lyapunov equation

$$AW+WA^T=Z, \tag{3.13}$$

typically the eigenvalues of W decay very rapidly when the right side Z has a low rank. This decay is strongly related to the approximation error from the balanced truncation.

In general, the matrix W is dense even if the system is sparse. However in the following propositions, we could obtain some useful block sparse matrices.

From the matrix approximation, we set

$$W = U\tilde{W}U^T, \quad A = U\tilde{A}U^T, \tag{3.14}$$

and

$$\tilde{W} = U^T W U, \quad \tilde{A} = U^T A U, \tag{3.15}$$

where $U^T U = I$, then

$$\begin{aligned} U^T Z U &= U^T(AW + WA^T)U = U^T AWU + U^T WA^T U \\ &= U^T AU\tilde{W}U^T U + U^T U\tilde{W}U^T A^T U \\ &= U^T AU\tilde{W} + \tilde{W}U^T A^T U \\ &= \tilde{A}\tilde{W} + \tilde{W}\tilde{A}^T = \tilde{Z}, \end{aligned} \tag{3.16}$$

which means $\tilde{Z}$ is an approximation to Z.

To this point, we provide a joint measure of controllability and observability for solving Lyapunov equations. We point out that these equations are not based on the fine matrices, but are based on the coarse matrices $A_c(\mu)$, B_c, C_c defined in (2.7). For equations

$$A_c(\mu)W_{\text{con}} + W_{\text{con}}\big(A_c(\mu)\big)^T + B_c(B_c)^T = 0, \tag{3.17a}$$

$$\big(A_c(\mu)\big)^T W_{\text{obs}} + W_{\text{obs}}A_c(\mu) + (C_c)^T C_c = 0, \tag{3.17b}$$

where W_{con} is a controllability Gramian, W_{obs} is an observability Gramian [2, 13], and in (3.13) Z is dependent on the forms of B_c, C_c. The solutions of these two Lyapunov equations play an important role in the model reduction and in our balanced truncation particularly.

In general the computations of dense matrices are of order $\mathcal{O}(n^3)$, some works are developed to efficiently compute the solutions by reducing its computational complexity [27, 29]. But these are iterative methods and often low rank approximations, while in our high contrast case it is hard to provide highly satisfactory approximations. In Matlab code lyap, it transforms the matrices to complex Schur forms, computes the solutions of resulting triangular systems, and may transform the solutions back if necessary. It has high rank approximations, and the solutions of Lyapunov equation have numerical connections to the dominant eigenvalues. We apply $W_{\text{con}} = \text{lyap}(A_c, B_c(B_c)^T)$ and $W_{\text{obs}} = \text{lyap}((A_c)^T, (C_c)^T C_c)$ to fulfill these advantages in the experiments. Since these matrices A_c, B_c, C_c are on coarse scale, the sizes of W_{con}, W_{obs} are just $N_c \times N_c$.

Once the two Gramians W_{con}, W_{obs} are available, they may be expressed in the form of Cholesky singular value decompositions to get L_{con}, L_{obs} as

$$W_{\text{con}} = L_{\text{con}}L_{\text{con}}^T, \tag{3.18a}$$

$$W_{\text{obs}} = L_{\text{obs}}L_{\text{obs}}^T, \tag{3.18b}$$

and

$$L_{\text{obs}}^T L_{\text{con}} = U\Sigma V^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & O \\ O & O \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} = U_1 \Sigma_1 V_1^T, \tag{3.19}$$

where U, V are to be specified soon. We set $S_{\text{con}} = L_{\text{con}} V_1 \Sigma_1^{-1/2}$, $S_{\text{obs}} = \Sigma_1^{-1/2} U_1^T L_{\text{obs}}^T$, where $U_1^T U_1 = V_1^T V_1 = I_{N_r}$, and the subscript r means the reduced case. Here some properties and proofs are provided.

Proposition 3.1. Suppose matrices are of full rank n, then both S_{con}, S_{obs} are revertible,

$$S_{\text{con}}^{-1} = S_{\text{obs}}, \tag{3.20}$$

and

$$S_{\text{obs}} W_{\text{con}} S_{\text{obs}}^T = S_{\text{con}}^T W_{\text{obs}} S_{\text{con}} = \Sigma_1, \tag{3.21}$$

furthermore

$$S_{\text{obs}}^T S_{\text{obs}} W_{\text{con}} S_{\text{obs}}^T S_{\text{obs}} = S_{\text{obs}}^T S_{\text{con}}^T W_{\text{obs}} S_{\text{con}} S_{\text{obs}} = M_1, \tag{3.22a}$$

$$S_{\text{con}} S_{\text{obs}} W_{\text{con}} S_{\text{obs}}^T S_{\text{con}}^T = S_{\text{con}} S_{\text{con}}^T W_{\text{obs}} S_{\text{con}} S_{\text{con}}^T = M_2. \tag{3.22b}$$

Proof. Since

$$\begin{aligned} S_{\text{obs}} S_{\text{con}} =& \Sigma_1^{-1/2} U_1^T L_{\text{obs}}^T \cdot L_{\text{con}} V_1 \Sigma_1^{-1/2} \\ =& \Sigma_1^{-1/2} U_1^T U_1 \Sigma_1 V_1^T V_1 \Sigma_1^{-1/2} = I_n, \end{aligned}$$

they are both revertible. For (3.21) the second term

$$\begin{aligned} S_{\text{con}}^T W_{\text{obs}} S_{\text{con}} =& \Sigma_1^{-1/2} V_1^T L_{\text{con}}^T L_{\text{obs}} L_{\text{obs}}^T L_{\text{con}} V_1 \Sigma_1^{-1/2} \\ =& \Sigma_1^{-1/2} V_1^T V_1 \Sigma_1 U_1^T U_1 \Sigma_1 V_1^T V_1 \Sigma_1^{-1/2} = \Sigma_1, \end{aligned}$$

and $S_{\text{obs}} W_{\text{con}} S_{\text{obs}}^T$ is similar. For (3.22a) the second term

$$S_{\text{obs}}^T S_{\text{con}}^T W_{\text{obs}} S_{\text{con}} S_{\text{obs}} = S_{\text{obs}}^T \Sigma_1 S_{\text{obs}} = L_{\text{obs}} U_1 \Sigma_1^{-1/2} \Sigma_1 \Sigma_1^{-1/2} U_1^T L_{\text{obs}}^T := M_1,$$

and others can be obtained similarly. □

Proposition 3.2. Suppose $L_{\text{obs}}^T L_{\text{con}}$ whose rank $r < n$, then there exists matrices $S_{\text{con2}} \in \mathbb{R}^{n\times(n-r)}$, $S_{\text{obs2}} \in \mathbb{R}^{(n-r)\times n}$ such that for

$$X = \begin{bmatrix} S_{\text{con}} & S_{\text{con2}} \end{bmatrix}, \quad Y = \begin{bmatrix} S_{\text{obs}} \\ S_{\text{obs2}} \end{bmatrix},$$

it has

$$X^{-1} = Y, \tag{3.23}$$

and

$$X^T W_{\text{obs}} X = \begin{bmatrix} \Sigma_1 & O \\ O & M_3 \end{bmatrix}, \quad Y W_{\text{con}} Y^T = \begin{bmatrix} \Sigma_1 & O \\ O & M_4 \end{bmatrix}, \tag{3.24}$$

furthermore

$$X^T W_{\text{obs}} W_{\text{con}} Y^T = Y W_{\text{con}} W_{\text{obs}} X = \begin{bmatrix} \Sigma_1^2 & O \\ O & O \end{bmatrix}. \tag{3.25}$$

Proof. Choose S_{con2} whose columns form a basis for the nullspace of S_{obs}, such that $S_{\text{obs}}S_{\text{con2}}=O$. Set S_{obs2} as the last $n-r$ rows of X^{-1}, follows that $S_{\text{obs2}}S_{\text{con}}=O$. Then

$$XY=\begin{bmatrix}S_{\text{con}} & S_{\text{con2}}\end{bmatrix}\begin{bmatrix}S_{\text{obs}}\\ S_{\text{obs2}}\end{bmatrix}=\begin{bmatrix}S_{\text{con}}S_{\text{obs}}+S_{\text{con2}}S_{\text{obs2}}\end{bmatrix}=I_n,$$
$$YX=\begin{bmatrix}S_{\text{obs}}\\ S_{\text{obs2}}\end{bmatrix}\begin{bmatrix}S_{\text{con}} & S_{\text{con2}}\end{bmatrix}=\begin{bmatrix}S_{\text{obs}}S_{\text{con}} & S_{\text{obs}}S_{\text{con2}}\\ S_{\text{obs2}}S_{\text{con}} & S_{\text{obs2}}S_{\text{con2}}\end{bmatrix}=\begin{bmatrix}I_r & O\\ O & I_{n-r}\end{bmatrix}=I_n,$$

from Proposition 3.1 and above that they are revertible.

For (3.24) the first term

$$\begin{aligned}X^TW_{\text{obs}}X&=\begin{bmatrix}S_{\text{con}}^T\\ S_{\text{con2}}^T\end{bmatrix}W_{\text{obs}}\begin{bmatrix}S_{\text{con}} & S_{\text{con2}}\end{bmatrix}\\&=\begin{bmatrix}S_{\text{con}}^TW_{\text{obs}}S_{\text{con}} & S_{\text{con}}^TW_{\text{obs}}S_{\text{con2}}\\ S_{\text{con2}}^TW_{\text{obs}}S_{\text{con}} & S_{\text{con2}}^TW_{\text{obs}}S_{\text{con2}}\end{bmatrix},\end{aligned}$$

where the first entry

$$\begin{aligned}S_{\text{con}}^TW_{\text{obs}}S_{\text{con}}&=\Sigma_1^{-1/2}V_1^TL_{\text{con}}^TL_{\text{obs}}L_{\text{obs}}^TL_{\text{con}}V_1\Sigma_1^{-1/2}\\&=\Sigma_1^{-1/2}V_1^TV_1\Sigma_1U_1^TU_1\Sigma_1V_1^TV_1\Sigma_1^{-1/2}=\Sigma_1,\end{aligned}$$

the second entry

$$\begin{aligned}S_{\text{con}}^TW_{\text{obs}}S_{\text{con2}}&=\Sigma_1^{-1/2}V_1^TL_{\text{con}}^TL_{\text{obs}}L_{\text{obs}}^TS_{\text{con2}}\\&=\Sigma_1^{-1/2}V_1^TV_1\Sigma_1U_1^TL_{\text{obs}}^TS_{\text{con2}}=\Sigma_1S_{\text{obs}}S_{\text{con2}}=O,\end{aligned}$$

the fourth entry

$$S_{\text{con2}}^TW_{\text{obs}}S_{\text{con2}}=S_{\text{con2}}^TL_{\text{obs}}L_{\text{obs}}^TS_{\text{con2}}:=M_3,$$

and $YW_{\text{con}}Y^T$ is similar. For (3.25) the first term

$$\begin{aligned}X^TW_{\text{obs}}W_{\text{con}}Y^T&=\begin{bmatrix}S_{\text{con}}^T\\ S_{\text{con2}}^T\end{bmatrix}L_{\text{obs}}L_{\text{obs}}^T\cdot L_{\text{con}}L_{\text{con}}^T\begin{bmatrix}S_{\text{obs}}^T & S_{\text{obs2}}^T\end{bmatrix}\\&=\begin{bmatrix}S_{\text{con}}^TL_{\text{obs}}L_{\text{obs}}^TL_{\text{con}}L_{\text{con}}^TS_{\text{obs}}^T & S_{\text{con}}^TL_{\text{obs}}L_{\text{obs}}^TL_{\text{con}}L_{\text{con}}^TS_{\text{obs2}}^T\\ S_{\text{con2}}^TL_{\text{obs}}L_{\text{obs}}^TL_{\text{con}}L_{\text{con}}^TS_{\text{obs}}^T & S_{\text{con2}}^TL_{\text{obs}}L_{\text{obs}}^TL_{\text{con}}L_{\text{con}}^TS_{\text{obs2}}^T\end{bmatrix},\end{aligned}$$

where the first entry

$$\begin{aligned}&S_{\text{con}}^TL_{\text{obs}}L_{\text{obs}}^TL_{\text{con}}L_{\text{con}}^TS_{\text{obs}}^T\\=&\Sigma_1^{-1/2}V_1^TL_{\text{con}}^TL_{\text{obs}}L_{\text{obs}}^TL_{\text{con}}L_{\text{con}}^TL_{\text{obs}}U_1\Sigma_1^{-1/2}\\=&\Sigma_1^{-1/2}V_1^TV_1\Sigma_1U_1^TU_1\Sigma_1V_1^TV_1\Sigma_1U_1^TU_1\Sigma_1^{-1/2}=\Sigma_1^2,\end{aligned}$$

the second entry

$$\begin{aligned}
&S_{\text{con}}^T L_{\text{obs}} L_{\text{obs}}^T L_{\text{con}} L_{\text{con}}^T S_{\text{obs2}}^T \\
=&\Sigma_1^{-1/2} V_1^T L_{\text{con}}^T L_{\text{obs}} L_{\text{obs}}^T L_{\text{con}} L_{\text{con}}^T S_{\text{obs2}}^T \\
=&\Sigma_1^{-1/2} V_1^T V_1 \Sigma_1 U_1^T U_1 \Sigma_1 V_1^T L_{\text{con}}^T S_{\text{obs2}}^T \\
=&\Sigma_1^{3/2} V_1^T L_{\text{con}}^T S_{\text{obs2}}^T \\
=&\Sigma_1^2 S_{\text{con}}^T S_{\text{obs2}}^T = O,
\end{aligned}$$

and the fourth entry

$$\begin{aligned}
&S_{\text{con2}}^T L_{\text{obs}} L_{\text{obs}}^T L_{\text{con}} L_{\text{con}}^T S_{\text{obs2}}^T \\
=&S_{\text{con2}}^T L_{\text{obs}} U_1 \Sigma_1 V_1^T L_{\text{con}}^T S_{\text{obs2}}^T \\
=&S_{\text{con2}}^T S_{\text{obs}}^T \Sigma_1^2 S_{\text{con}}^T S_{\text{obs2}}^T = O,
\end{aligned}$$

and the other $YW_{\text{con}}W_{\text{obs}}X$ is similar. □

Remark 3.1. From the above propositions, these block sparse matrices are helpful in the construction of the reduced matrices. Once the approximations are obtained, we can apply for the singular value decomposition and eigenvalue computation further. The reduced model from the coarse version would be shown effective for our parameter-dependent model in the experiments later on.

3.2.2 Balanced reduction

On the basis of coarse matrices (compared with the fine ones), we have a platform to carry on the matrices reduction. In our case, we find the first N_r ordered eigenvalues and normalized eigenvectors of the matrix $L_{\text{con}}^T W_{\text{obs}} L_{\text{con}}$, by solving

$$L_{\text{con}}^T W_{\text{obs}} L_{\text{con}} \xi_i = \lambda_i \xi_i, \quad i = 1, \cdots, N_r. \tag{3.26}$$

The eigenvalues and eigenvectors are given by

$$\{\lambda_i\} \quad \text{and} \quad \{\zeta_i\} = \{\sigma_i^{-1} \xi_i^T L_{\text{con}}^T L_{\text{obs}}^T\}, \tag{3.27}$$

where $\sigma_i = \lambda_i^{1/2}$. With the eigenvalues and eigenvectors in place, we construct the above-mentioned matrices U and V in (2.8) as

$$U = \begin{bmatrix} \sigma_1^{-1/2} \zeta_1 \\ \vdots \\ \sigma_{N_r}^{-1/2} \zeta_{N_r} \end{bmatrix} L_{\text{obs}}, \quad V = L_{\text{con}} \left[\xi_1 \sigma_1^{-1/2} \cdots \xi_{N_r} \sigma_{N_r}^{-1/2} \right], \tag{3.28}$$

whose size is $N_r \times N_c$ and $N_c \times N_r$, respectively.

As a result, the reduced model is formed in the matrices $A_r(\mu)=UA_c(\mu)V$, and $B_r=UB_c$, $C_r=C_cV$, where r depends on the decay of eigenvalues and is specifically selected in the numerical experiment. In particular, we reach the reduced scale since $A_r(\mu)$ has a size of $N_r\times N_r$, compared with the coarse scale $A_c(\mu)$ with a size of $N_c\times N_c$, and the fine scale $A_f(\mu)$ with a size of $N_f\times N_f$.

The balanced truncation has better efficiency in numerical simulations. Their a-priori error estimates simply depend on the decay of eigenvalues in (3.26). And particularly, the error estimates rely on the magnitude of all truncated eigenvalues summation. On the other hand, the BT has its own limitations. Since the Lyapunov equations in (3.17) require costly computations for unknown W_{con} and W_{obs}, and followed by the singular value decompositions, especially for the fine case. In particular, applying BT directly to a huge input-output system may not be feasible in real applications.

Therefore in this paper, a proposed combination of GMsFEM-BT method for the parameter-dependent problem is developed to avoid a huge system computation, while it maintaining a proper accuracy at the same time. Since the approach is applied to an online coarse scale system, which has already been reduced via using the GMsFEM, its computational cost would be significantly reduced.

4 Numerical experiments

In this section numerical results are presented to show the performance of our method. We use Matlab 2014 to address all codes. Computations are performed on a desktop workstation, with Intel Core i5 3.20GHz processor and 8GB RAM. A permeability coefficient is assumed as

$$\kappa(\mathbf{x};\mu)=\mu_1\kappa_1(\mathbf{x})+\mu_2\kappa_2(\mathbf{x}), \tag{4.1}$$

where

$$\kappa_1(\mathbf{x})=10\cdot\left(\frac{2+1.8\sin\left(\frac{2\pi x}{\varepsilon_1}\right)}{2+1.8\cos\left(\frac{2\pi y}{\varepsilon_2}\right)}+\frac{2+1.8\cos\left(\frac{2\pi y}{\varepsilon_1}\right)}{2+1.8\sin\left(\frac{2\pi x}{\varepsilon_2}\right)}\right), \tag{4.2a}$$

$$\kappa_2(\mathbf{x})=10\cdot\left(\frac{2+1.8\sin\left(\frac{2\pi x}{\varepsilon_3}\right)}{2+1.8\cos\left(\frac{2\pi y}{\varepsilon_4}\right)}+\frac{2+1.8\cos\left(\frac{2\pi y}{\varepsilon_3}\right)}{2+1.8\sin\left(\frac{2\pi x}{\varepsilon_4}\right)}\right), \tag{4.2b}$$

and the pair (μ_1,μ_2) may be randomly picked from $[0,1]^2$, and let $\varepsilon_1=0.2$, $\varepsilon_2=0.08$, $\varepsilon_3=0.125$, $\varepsilon_4=0.0078125$.

A resulting permeability field exhibits a periodic contrast structure (and may be extended to randomized structure too), see Fig. 2. And a non-homogeneous boundary condition g is defined, $g=10x$ on lower $x=[0,1]$ and $y=0$; $g=10(1+y)$ on right $x=1$ and $y=[0,1]$; $g=10(1+x)$ on upper $x=[0,1]$ and $y=1$; $g=10y$ on left $x=0$ and $y=[0,1]$.

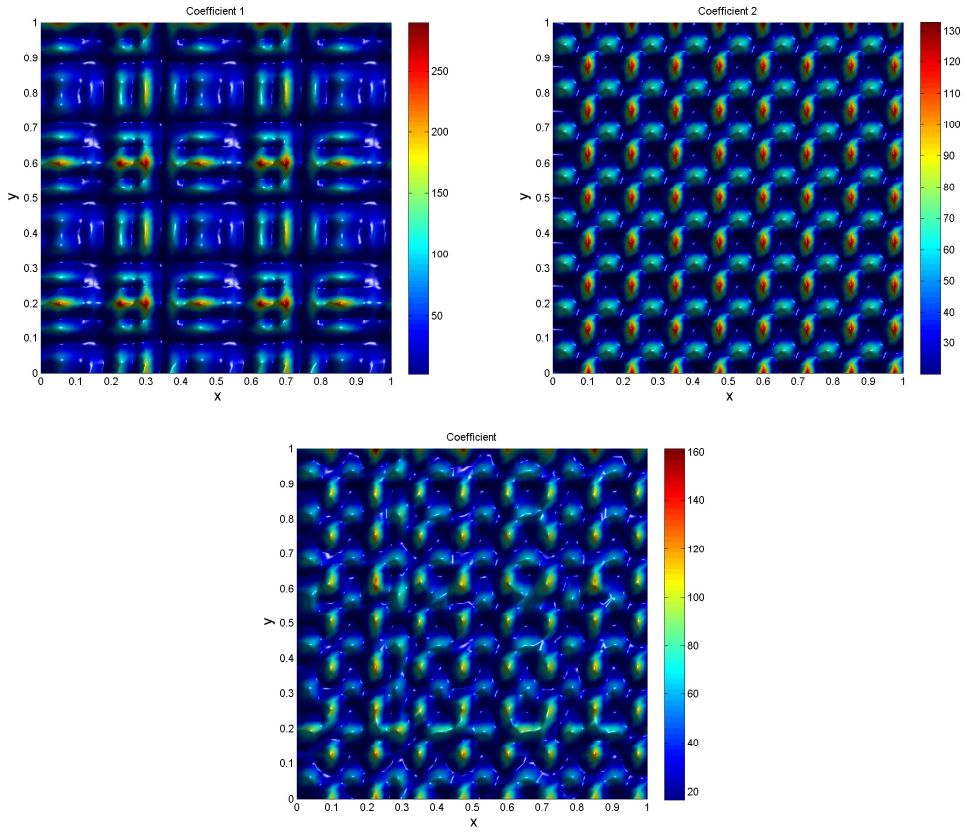

Figure 2: Coefficients $\kappa_1(\mathbf{x})$, $\kappa_2(\mathbf{x})$ and $\kappa(\mathbf{x};\mu)$.

To solve the model (2.1), first we divide the domain $\Omega=[0,1]\times[0,1]$ into $NM\times NM=160\times160$ fine elements and use the FEM to solve a fully-resolved system (2.5). This computation is typically the most expensive, and is used as a reference for testing the performance of the proposed method. Next in a coarse scale scheme, the MsFEM and GMsFEM are applied respectively to obtain the corresponding outputs from (2.7). In both approaches we bisect the coarse partition $N=4,8,16$ and keep the fine partition $M=10$ in each coarse element K. In contrast, with the helps of spectral computation and multiscale basis functions enrichment, the latter generalized approach would show its advantages. Finally, we solve Lyapunov equations on the basis of coarse matrices, and carry on the BT from (2.8) in a further reduced scale to test the abilities of our GMsFEM-BT.

In implementation of the GMsFEM, in order to generate the necessary snapshot space for the offline-online stages we use three equally spaced points in each dimension and keep the eigenfunctions number $L_i=10$ in (3.2), then for two dimensional case it has $L_{\text{snap}}=3^2*10=90$ in (3.3). According to different error tolerances of the experiment setting we may flexibly decide the desired outputs, through selecting the dominant eigenfunctions from the snapshot space V^{snap} to build the mapping matrices. In each coarse neigh-

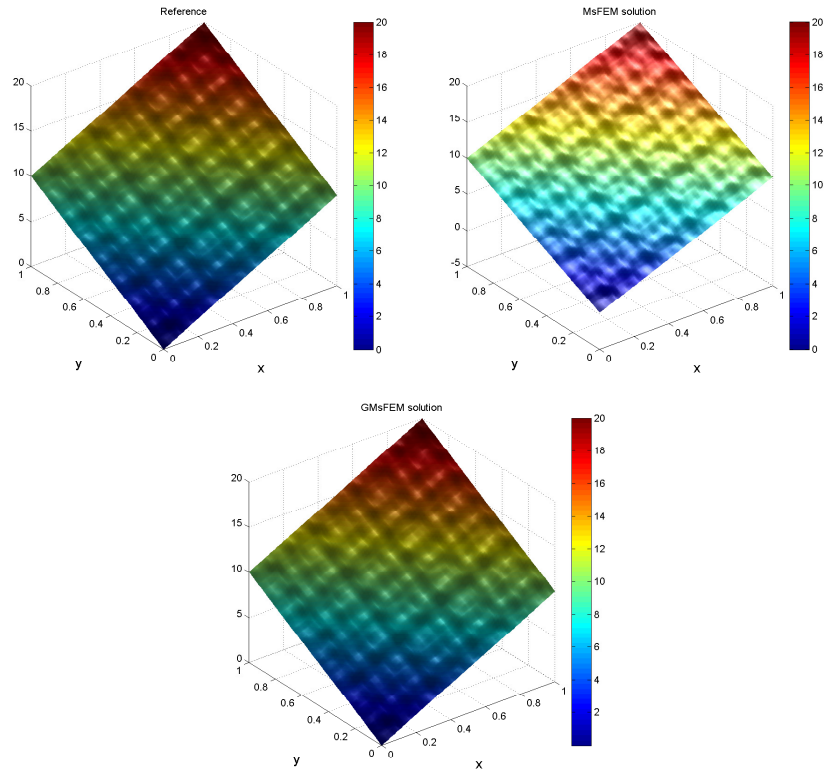

Figure 3: Solutions of reference, MsFEM, GMsFEM on coarse partition $N=8$, respectively.

borhood, the optimal number of eigenfunctions L_{on} is less than the maximum L_{snap}. In the BT with a further reduction, after solving the Lyapunov equations for a controllability W_{con} and a observability W_{obs} on coarse scales, we use the singular value decompositions to build U, V in (3.28), whose size is smaller N_r.

We are interested in comparing the outputs from the fully resolved fine system, the coarse system and the reduced system. Since the coarse partition refinement is $N=4,8,16$ and the same fine partition is $M=10$ in each coarse element, in this way in both directions the coarse vertices number is $N_v=(N+1)^2=25,81,289$. The size of fine system is $N_f=(NM+1)^2=1681,6561,25921$, while the corresponding size of coarse system is $N_c=118,598,2679(\ll N_f)$. We should point out that N_c in the GMsFEM are determined from setting the tolerances of offline and online both at 10^{-3}. In contrast, the size of reduced system is N_r, which is just the specified smaller integers later.

Figs. 3 and 4 show the reference, MsFEM, GMsFEM solutions and errors. The reference is from fully solved fine system, while the standard/generalized MsFEM solutions are from much more coarse system and they are well approximate to the reference. It is verified with the beneficial spectral computation for dominant eigenfunctions and multiscale enrichments, the GMsFEM performs much better. We list the norm errors from

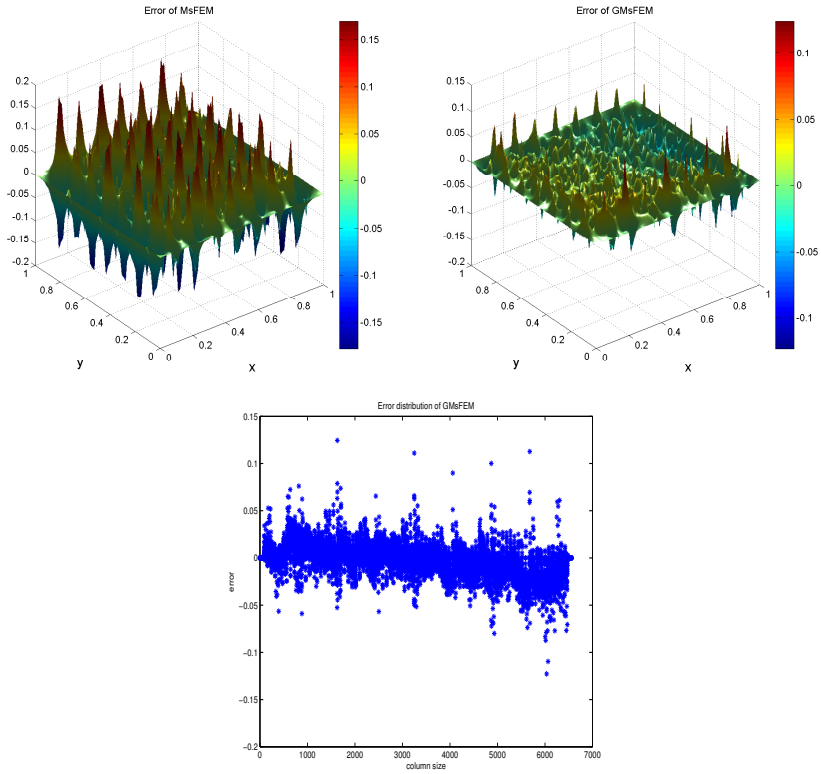

Figure 4: Errors of MsFEM and GMsFEM on coarse partition $N=8$, and error distribution of the latter method.

MsFEM and GMsFEM with the grid refinement in Table 1. These error percentage to the reference show the convergence and stability of GMsFEM, which is superior to the divergence of MsFEM, see Fig. 5.

In Fig. 6 results are presented for the reference on the fine partition $NM=160$, GMsFEM on the coarse partition $N=8$ and BT on the different reductions N_r. We observe that in Table 2 on small N_r such as 20 and 60, the BT does not behave very well. With the increase of N_r it demonstrates the capability to capture more necessary details of the

Table 1: Percentages of norm error from MsFEM and GMsFEM (number of local basis functions per coarse neighborhood is 10), on different partition number N.

coarse partition N	norm	MsFEM error (%)	GMsFEM error (%)
4	L^2	0.29	0.25
	H^1	11.12	9.49
8	L^2	0.38	0.16
	H^1	16.55	9.04
16	L^2	0.54	0.11
	H^1	26.63	8.77

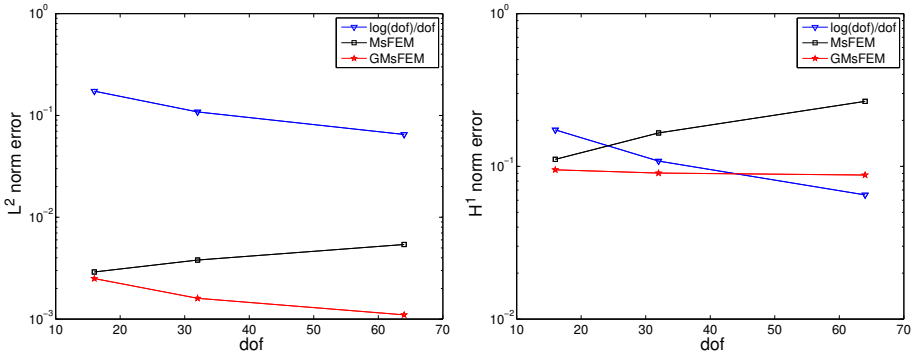

Figure 5: Convergence history of norms from MsFEM and GMsFEM.

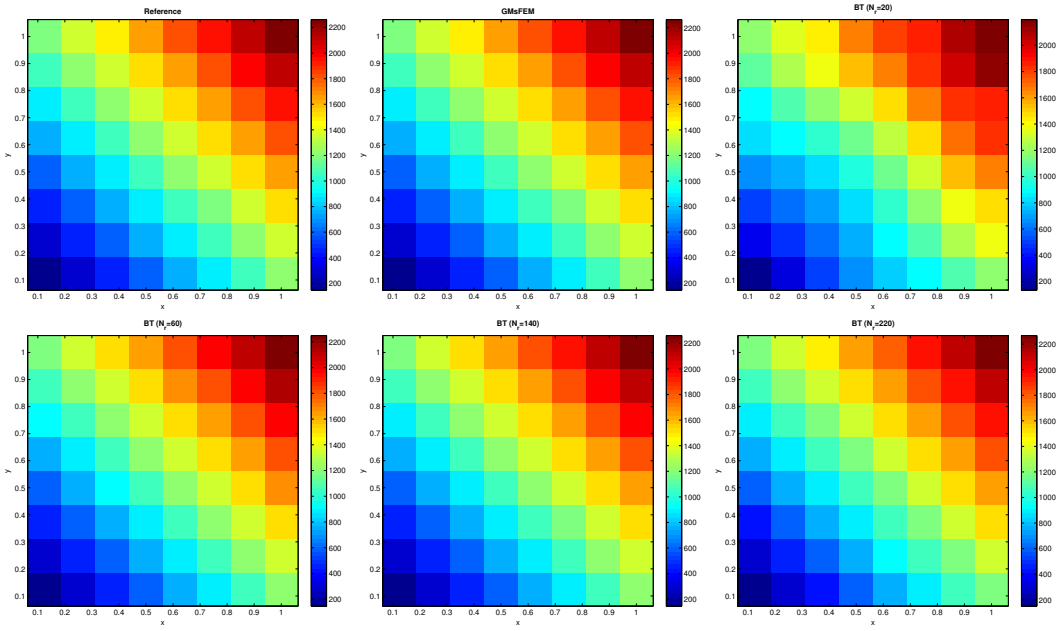

Figure 6: Outputs of reference, GMsFEM coarse and BT reduction with different N_r.

reference, and its error percentage is less than 0.02%.

It should be noted that the size of BT is of the maximum $N_r = 220$, compared to the size of GMsFEM with $N_c = 2679$ or the size of FEM with $N_f = 25921$. At the meantime, we note the fact that the BT works on the basis of the GMsFEM's coarse scale, and it would cost some time for solving Lyapunov equations and singular value decompositions. And if the BT works on the basis of the FEM's very fine scale, the computational cost would be much higher.

Finally in this section, we provide the CPU time for the partition of $N = 8$ and $M = 10$. From Table 3 it is obvious that the FEM spends the most time on a fully fine grid

Table 2: Error percentages from GMsFEM and BT.

GMsFEM error (%)	GMsFEM-BT error on different N_r (%)					
	$N_r=20$	$N_r=60$	$N_r=100$	$N_r=140$	$N_r=180$	$N_r=220$
0.13	0.97	0.33	0.15	0.056	0.025	0.014

Table 3: Comparisons of CPU time.

	FEM	MsFEM	GMsFEM (online)	BT			
size	6561	81	598	20	60	140	220
time (s)	8.60	0.12	0.92	0.00048	0.0020	0.0060	0.0088
			Lyapunov solve	GMsFEM-BT			
time (s)			4.17	0.9205	0.9220	0.9260	0.9288

size of 6561 to obtain a reference. The MsFEM and GMsFEM spend less time to obtain their numerical results. However, the GMsFEM has to construct the snapshot space and couple the offline-online stages with eigenvalue computations. This process definitely consumes resources to bring a good accuracy. In the end, a Lyapunov solver on the basis of GMsFEM size 598 (neither on the FEM, nor on the MsFEM) shows its promise. After that, the reduced size of BT system is available, and the desired balance for both accuracy and efficiency of our proposed GMsFEM-BT is demonstrated with the numerical results.

5 Conclusions

In this paper, a combined approach of generalized MsFEM and balanced truncation is proposed to solve the parameter-dependent elliptic problem. With the benefits of spectral computation from dominant eigenfunctions, the GMsFEM provides a good platform and serves as a local model reduction tool. Then the BT is served as a further global model reduction in which the input-output mapping is used for incorporating the underlying system information, and it requires the singular value decomposition of Lyapunov equations solution. From the theory and experiment, our GMsFEM-BT is shown to be quite flexible with respect to the online space tolerance and reduced matrix size, and it may reach a suitable balance between the accuracy and efficiency for complicated multiscale problems.

Acknowledgements

The Research is supported by NSFC (Grant Nos. 11771224, 11301462), Jiangsu Province Qing Lan Project and Jiangsu Overseas Research Program for University Prominent Teachers to Shan Jiang. We would like to thank Professor Yalchin Efendiev in Texas A&M University for many useful discussions. And we appreciate the referees and editors for their insightful comments and helpful suggestions.

References

[1] S. GUGERCIN AND A. ANTOULAS, *A survey of model reduction by balanced truncation and some new results,* Inter. J. Control, 77(8) (2004), pp. 748–766.

[2] C. W. ROWLEY, *Model reduction for fluids, using balanced proper orthogonal decomposition*, Inter. J. Bifurca. Chaos, 15(3) (2005), pp. 997–1013.

[3] W. H. A. SCHILDERS, H. A. VANDER VORST AND J. ROMMES, Model Order Reduction: Theory, Research Aspects and Applications, Springer, Berlin, 2008.

[4] T. Y. HOU AND X. H. WU, *A multiscale finite element method for elliptic problems in composite materials and porous media*, J. Comput. Phys., 134(1) (1997), pp. 169–189.

[5] W. N. E, P. B. MING AND P. W. ZHANG, *Analysis of the heterogeneous multiscale method for elliptic homogenization problems*, J. Amer. Math. Soc., 18(1) (2004), pp. 121–156.

[6] I. BABUSKA AND R. LIPTON, *Optimal local approximation spaces for generalized finite element methods with application to multiscale problems*, SIAM Multiscale Model. Simul., 9(1) (2011), pp. 373–406.

[7] P. HENNING AND D. PETERSEIM, Oversampling for the multiscale finite element method, SIAM Multiscale Model. Simul., 11(4) (2013), pp. 1149–1175.

[8] W. B. DENG AND H. J. WU, *A combined finite element and multiscale finite element method for the multiscale elliptic problems*, Multiscale Model. Simul., 12 (2014), pp. 1424–1457.

[9] M. L. SUN AND S. JIANG, *Multiscale basis functions for singular perturbation on adaptively graded meshes*, Adv. Appl. Math. Mech., 6(5) (2014), pp. 604–614.

[10] S. JIANG, M. PRESHO AND Y. Q. HUANG, *An adapted Petrov-Galerkin multiscale finite element for singularly perturbed reaction-diffusion problems*, Int. J. Comput. Math., 93(7) (2016), pp. 1200–1211.

[11] Y. EFENDIEV, J. GALVIS AND T. Y. HOU, *Generalized multiscale finite element methods (GMsFEM)*, J. Comput. Phys., 251 (2013), pp. 116–135.

[12] F. CHINESTA, P. LADEVEZE AND E. CUETO, *A short review on model order reduction based on proper generalized decomposition*, Arch. Comput. Methods Eng., 18 (2011), pp. 395–404.

[13] J. R. SINGLER, *Balanced POD for model reduction of linear PDE systems: Convergence theory*, Numer. Math., 121 (2012), pp. 127–164.

[14] D. ABOU JAOUDE AND M. FARHOOD, *Balanced truncation model reduction of nonstationary systems interconnected over arbitrary graphs*, Automatica, 85 (2017), pp. 405–411.

[15] E. T. CHUNG, Y. EFENDIEV AND G. L. LI, *An adaptive GMsFEM for high-contrast flow problems*, J. Comput. Phys., 273 (2014), pp. 54–76.

[16] M. PRESHO, A. PROTASOV AND E. GILDIN, *Local-global model reduction of parameter-dependent, single-phase flow models via balanced truncation*, J. Comput. Appl. Math., 271 (2014), pp. 163–179.

[17] J. J. LIU, L. Q. CAO, N. N. YAN AND J. Z. CUI, *Multiscale approach for optimal design in conductivity of composite materials*, SIAM J. Numer. Anal., 53(3) (2015), pp. 1325–1349.

[18] E. T. CHUNG, Y. EFENDIEV AND W. T. LEUNG, *Residual-driven online generalized multiscale finite element methods*, J. Comput. Phys., 302 (2015), pp. 176-190.

[19] E. T. CHUNG, Y. EFENDIEV AND T. Y. HOU, *Adaptive multiscale model reduction with generalized multiscale finite element methods*, J. Comput. Phys., 320 (2016), pp. 69–95.

[20] T. Y. HOU AND P. F. LIU, *Optimal local multi-scale basis functions for linear elliptic equations with rough coefficients*, Discrete Contin. Dynam. Sys. A, 36(8) (2016), pp. 4451–4476.

[21] V. M. CALO, Y. EFENDIEV, J. GALVIS AND G. L. LI, *Randomized oversampling for generalized multiscale finite element methods*, SIAM Multiscale Model. Simul., 14(1) (2016), pp. 482–501.

[22] L. F. GAO, X. S. TAN AND E. T. CHUNG, *Application of the generalized multiscale finite element method in parameter-dependent PDE simulations with a variable-separation technique*, J. Comput. Appl. Math., 300 (2016), pp. 183–191.
[23] L. J. JIANG AND Q. Q. LI, *Reduced multiscale finite element basis methods for elliptic PDEs with parameterized inputs*, J. Comput. Appl. Math., 301 (2016), pp. 101–120.
[24] L. J. JIANG AND Q. Q. LI, *Model's sparse representation based on reduced mixed GMsFE basis methods*, J. Comput. Phys., 338 (2017), pp. 285–312.
[25] E. T. CHUNG, W. T. LEUNG, M. VASILYEVA AND Y. T. WANG, *Multiscale model reduction for transport and flow problems in perforated domains*, J. Comput. Appl. Math., 330 (2018), pp. 519–535.
[26] B. VANDEREYCKEN AND S. VANDEWALLE, *A Riemannian optimization approach for computing low-rank solutions of Lyapunov equations*, SIAM J. Matrix Anal. Appl., 31(5) (2010), pp. 2553–2579.
[27] M. R. OPMEER, T. REIS AND W. WOLLNER, *Finite-rank ADI iteration for operator Lyapunov equations*, SIAM J. Control Optim., 51(5) (2013), pp. 4084–4117.
[28] L. GRUBISIC AND D. KRESSNER, *On the eigenvalue decay of solutions to operator Lyapunov equations*, Syst. Control Lett., 73 (2014), pp. 42–47.
[29] T. X. LI, C. Y. WENG, K. W. CHU AND W. W. LIN, *Large-scale Stein and Lyapunov equations, Smith method, and applications*, Numer. Algorithms, 63 (2013), pp. 727–752.