An exponential integration generalized multiscale finite element method for parabolic problems

L. F. Contreras[1], **D. Pardo**[3,4,5], **E. Abreu**[2], **J. Muñoz-Matute**[4,6], **C. Diaz**[7], **J. Galvis**[1]

[1] *Departamento de Matemáticas, Universidad Nacional de Colombia, Carrera 45 No. 26-85, Edificio Uriel Gutierréz, Bogotá D.C., Colombia.*

[2] *Department of Applied Mathematics, University of Campinas, Campinas, Brazil*

[3] *University of the Basque Country (UPV/EHU), Leioa, Spain*

[4] *Basque Center for Applied Mathematics (BCAM), Bilbao, Spain*

[5] *Ikerbasque (Basque Foundation For Sciences), Bilbao, Spain.*

[6] *Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, USA*

[7] *Department of Applied Mathematics, Toronto Metropolitan University, 350 Victoria Street Toronto, ON M5B 2K3, Canada*

Abstract

We consider linear and semilinear parabolic problems posed in high-contrast multiscale media in two dimensions. The presence of high-contrast multiscale media adversely affects the accuracy, stability, and overall efficiency of numerical approximations such as finite elements in space combined with some time integrator. In many cases, implementing time discretizations such as finite differences or exponential integrators may be impractical because each time iteration needs the computation of matrix operators involving very large and ill-conditioned sparse matrices. Here, we propose an efficient Generalized Multiscale Finite Element Method (GMsFEM) that is robust against the high-contrast diffusion coefficient. We combine GMsFEM with exponential integration in time to obtain a good approximation of the final time solution. Our approach is efficient and practical because it computes matrix functions of small matrices given by the GMsFEM method. We present representative numerical experiments that show the advantages of combining exponential integration and GMsFEM approximations. The constructions and methods developed here can be easily adapted to three-dimensional domains.

Keywords: Multiscale approximation, time integration, functions of matrices, finite element methods

1. Introduction

Many applications such as modeling environmental issues and subsurface flow have taken on great relevance in current research trends [31, 7, 6]. A central tool in these areas is the correct modeling of diffusion of substances in a heterogeneous porous media with high-contrast multiscale permeability properties. Mathematical modeling and numerical simulations turn relevant for understanding this model but the presence of multiscale fields and high-contrast introduce several challenges regarding the accuracy and computational efficiency of the implemented numerical methods. See [18] and references therein.

*Email address :

A widely used model of diffusion in porous media is given by the following semilinear parabolic problem posed in a high-contrast multiscale media,

$$\begin{cases} \partial_t p - \operatorname{div}(\kappa(x)\nabla p) = f(p), & \text{in } \Omega \times I, \\ p = p_D, & \text{on } \partial\Omega \times I, \\ p(0,x) = \hat{p}(x), & x \in \Omega. \end{cases} \tag{1}$$

Here Ω is a two dimensional convex domain with boundary $\partial\Omega$ and $I = [0,T]$ is the time domain. The field $\kappa(x)$ is a multiscale high-contrast heterogeneous field. Additionally, p is an unknown pressure field satisfying the Dirichlet condition given by p_D and the initial condition given by $\hat{p}$. The constructions and methods developed here can be easily adapted to the three-dimensional domains.

Approximations of solutions of problem (1) and many other interesting questions have been considered in the literature. In particular we mention [14, 26, 2, 33] and references therein. We focus our discussion on the numerical computation of solutions of this problem. In the presence of high-contrast multiscale coefficients, classical methods for the numerical approximation of solutions need to be revisited due to the lack of robustness and efficiency, see [21, 22, 15, 16, 1]. In this paper, we design robust numerical approximation procedures against the presence of multiscale variations and high-contrast in the coefficient κ. We call the attention to two important challenges in order to design efficient and robust numerical methods for equation (1):

1. In the presence of high-contrast multiscale coefficients, the spatial resolution needed to correctly approximate the solution of (1) (or its steady steady-state version) is related to the smallest scale at which we find variations of the coefficient κ. Additionally to the multiscale variations, the discontinuities and high-jumps of the coefficient bring additional difficulties to the numerical approximation of this time-dependent problem. Accuracy and efficiency can be negatively affected by solving large and ill-conditioned linear systems at each time step. See [17, 19, 1, 10].

2. The presence of high-contrast in the coefficients (even without complicated multiscale variations) reduces the stability region of time discretization methods such as Crank–Nicolson ans similar time integrators. See Sections 3 and 4 below.

Extensive research have been devoted to these important issues. It is not our intention to present a complete review of the many works concerning the efforts done to face this issues properly in different contexts. Let us mention first that the challenge 1 above also affects time-independent problems. For time independent problems classical multiscale methods provide good approximations only for moderated-to-low-contrast coefficients. However, Generalized Multiscale Finite Element Methods (GMsFEM) were designed to correctly handle problems with high-contrast in the coefficient where the main ingredient was to use local eigenvalue problems to construct appropriate coarse-mesh approximation spaces. For more details on the construction and analysis related to the the approximation capabilities of the GMSFEM see [17, 19, 1, 10, 35].

GMsFEM have also been applied to time dependent problems, linear and nonlinear parabolic and hyperbolic partial differential equations as well as sampling and inverse problems. In these cases, some time discretization or optimization iteration has to be added on top of the space approximation. We consider the case of time marching

schemes where, in each iteration, a large ill-conditioned linear problem has to be solved. We mention the recent papers [12, 20, 13, 9, 2, 30, 29] where different time dependent problems have been considered within the GMsFEM framework. Unfortunately, the loss of stability (due to the high-contrast in the coefficients) requires the reduction of the time step size (inversely proportional to the contrast in the coefficient) which ends up reducing the gain obtained by using the GMsFEM method to improve the overall computational time to obtain the final time solution. This bring us to face challenge 2 above that is the main topic of this article. Our idea to gain stability and accuracy in time discretizations is to move to exponential integration, see for instance [27, 25, 4, 24, 32]. Exponential integration is a more effective numerical method to overcome stiff problems and to improve the accuracy of numerical computations. The main bottleneck of the computations required by exponential integration is the computation of functions of matrices [23, 28]. This is even more critical for finite elements matrices associated with problem (1) since these are huge and ill-conditioned sparse matrices. In this paper, we show that the function of matrices needed in the exponential integration can be well approximated using a GMsFEM approach, that is, computed by projecting it to the coarse scale-space constructed using the GMsFEM approach. In our numerical experiments, we show that computing matrix functions on the coarse space allows us to advance significant time steps without losing stability and accuracy in the solution.

In summary, we emphasize that the need for exponential integration comes from the lack of stability in time stepping in the presence of high-contrast multiscale coefficient. The usage of GMsFEM approximation improves spatial approximations of the solutions (in terms of efficiency and dependence on the contrast). However, they still need a small enough time step (that scales as the inverse of the contrast in the coefficient) due to stability issues in the time discretization used. For completeness of our paper, we also present a detailed study of the effects of high-contrast and multiscale variations on the stability and accuracy of implicit Euler and Crank–Nicolson methods. We show that the use of GMsFEM can improve spatial approximations of the solutions and the overall efficiency of the method. However, we still need a tiny time step. In comparison the exponential integration allows us to take full advantage of the usage of GMsFEM methods for space discretization with larger time steps.

We remark that in this work we consider GMsFEM approximations for the space variable. We mention that other techniques different from the GMsFEM approach have been proposed for the approximation of equation (1) or similar partial differential equations. A complete review or comparison with these techniques is out of the scope of this paper but we mention the works [5, 33, 14, 14, 29, 30, 3]. These works focus on the quality of the space approximation and use a fine enough time step in order to obtain good approximations. In our work, we pay attention to compute more efficiently as the method advances in time by using exponential integration and we consider only the GMsFEM method for the space discretization. Once more, we mention that the presence of high-contrast reduces stability regions of time discretization. See Sections 3 and 4.

The rest of the paper is organized as follows. In Section 2, we review the GMsFEM methodology. Section 3 considers the finite difference and exponential integrators for the time discretization applied to problem (1). Here we describe how to use GMsFEM downscaling and upscaling operators and solvers to efficiently advance in time avoiding computations of matrix functions with fine-grid matrices. In Section 4, we present a series of numerical experiments to show the efficiency and main issues of the proposed approach. We also include experiments to emphasise the lack of

stability of finite difference approximation due to the high-contrast in the coefficient. In Section 5, we present some conclusions and final comments.

2. Generalized finite element methods (GMsFEMs)

A variational formulation of problem (1) is: Find $p(t) \in H^1(\Omega)$ with $(p(t)-p_D) \in H_0^1 = \left\{w \in H^1(\Omega) : w|_{\partial\Omega} = 0\right\}$ such that

$$(\partial_t p, v) + a(p, v) = F(p; v) \quad \text{for all } v \in H_0^1(\Omega), \tag{2}$$

where $(\cdot, \cdot)$ denotes the usual inner product in $L^2(\Omega)$, the bilinear form a is defined by

$$a(p, v) = \int_\Omega \kappa(x)\nabla p(x)\nabla v(x)dx, \tag{3}$$

and the functional F is defined by

$$F(p; v) = \int_\Omega f(p(x))v(x)dx, \tag{4}$$

for $p, v \in H_0^1(\Omega)$. Let $\mathcal{T}^h$ be a triangulation of the domain Ω. As it is usual in multiscale methods, we assume that h is fine enough to completely describe all the variations of the coefficient κ and therefore we refer to $\mathcal{T}^h$ as the fine mesh. We denote by $V^h(\Omega)$ the usual finite elements discretization of piecewise linear continuous functions with respect to $\mathcal{T}^h$. Denote by $V_0^h(\Omega)$ the subset of $V^h(\Omega)$ made of functions that vanish on $\partial\Omega$. The Galerkin formulation of (2) is to find $(p(t) - p_D) \in V_0^h(\Omega)$ such that

$$\begin{cases} (\partial_t p, v) + a(p, v) = F(p; v) & \text{for all } v \in V_0^h(\Omega), t \in I, \\ (p(0), v) = (\hat{p}, v) & \text{for all } v \in V_0^h(\Omega). \end{cases} \tag{5}$$

We consider the following representation for the solution of (5),

$$p(x, t) = \sum_{i=1}^{n_v} p_i(t)\phi_i(x) \tag{6}$$

where ϕ_i are the usual finite elements basis functions and n_v the number of interior nodes of $\mathcal{T}^h$. Using (5) and taking $v = \phi_j$ for $j = 1, ..., n_v$, we have

$$\begin{cases} \sum_{i=1}^{n_v} p_i'(t)(\phi_i, \phi_j) + \sum_{i=1}^{n_v} p_i(t)a(\phi_i, \phi_j) &= (f(p), \phi_j) \quad j = 1, ..., n_v, t \in I, \\ \sum_{i=1}^{n_v} p_i(0)(\phi_i, \phi_j) &= (\hat{p}, \phi_j) \quad j = 1, ..., n_v. \end{cases} \tag{7}$$

The equivalent continuous-time matrix form of (7) is

$$\begin{aligned} M\partial_t p + Ap &= b(p), \\ Mp(0) &= \hat{p}, \end{aligned} \tag{8}$$

where the vector $\hat{p}$ is $\hat{p} = [\int_\Omega p_D \phi_j]$, and the matrices A, M and the vector b are given by

$$u^T A v = \int_\Omega \kappa \nabla u \nabla v, \quad u^T M v = \int_\Omega \kappa u v \quad \text{and} \quad v^T b = \int_\Omega f(p) v, \quad \text{for all } u, v \in V_0^h(\Omega). \tag{9}$$

We introduce a coarse-scale mesh $\mathcal{T}^H$, where H indicates the coarse-mesh size. In practical applications, the coarse-grid does not resolve all the variations and discontinuities of the coefficient κ. A main goal in multiscale methods is to construct approximation strategies to mimic fine-grid approximation properties but only computing solutions of linear systems at the coarse-scale. The GMsFEM is a multiscale method designed to obtain good approximation of high-contrast multiscale problems. We next review some important aspects in the construction of GMsFEM basis functions. See [17, 1, 9] and references therein for further details.

We denote by $\{y_i\}_{i=1}^{N_v}$ the vertices of the coarse mesh $\mathcal{T}^H$ and define the neighborhood of each node y_i by

$$\omega_i = \bigcup \{K \in \mathcal{T}^H : y_i \in \bar{K}\}.$$

See Figure 1 for an illustration of coarse elements and coarse neighborhoods.

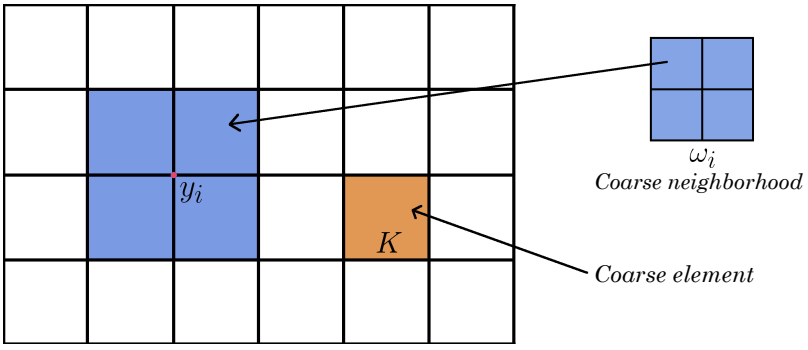


Figure 1: Ilustration of a coarse neighborhood.

Note that $\Omega = \bigcup_{y_i \in \mathcal{T}^H} \{\omega_i\}$. Let $\{\chi_i\}_{i=1}^{N_v}$ be a partition of unity subordinated to the covering $\{\omega_i\}$ and constructed such that $|\nabla \chi_i| \leq \frac{1}{H}$, $i = 1, 2, 3, ..., N_v$, where N_v is the number of nodes in $\mathcal{T}^H$. See [17, 1, 9] for examples of different partition of unity functions that can be used. Now, define the auxiliary coefficient $\widetilde{\kappa}$ by

$$\widetilde{\kappa} = \kappa \sum_{j=1}^{N_v} H^2 |\nabla \chi_j|^2.$$

Coefficient $\widetilde{\kappa}$ can be interpreted as a total pointwise energy for the functions in the partition of unity $\{\chi_i\}_{i=1}^{N_v}$. We define the following local bilinear forms,

$$a^{\omega_i}(p, v) = \int_{\omega_i} \kappa \nabla p \nabla v \quad \text{and} \quad m^{\omega_i}(p, v) = \int_{\omega_i} \widetilde{\kappa} p v \quad \text{for all } p, v \in H^1(\omega_i), \tag{10}$$

for every neighborhood ω_i. Also, define $\widetilde{V}(\omega_i) = \{v \in H^1(\omega_i) : v = 0 \text{ on } \partial\omega_i \cap \partial\Omega\}$ if $\partial\omega_i \cap \partial\Omega$ is non-empty and

$\widetilde{V}(\omega_i) = \left\{ v \in H^1(\omega_i) : \int_{\omega_i} v = 0 \right\}$ otherwise. We consider the local generalized eigenvalue problem

$$a^{\omega_i}(\psi, z) = \sigma^{\omega_i} m^{\omega_i}(\psi, z) \quad \text{for all } z \in \widetilde{V}(\omega_i), \tag{11}$$

with eigenfunction $\psi \in \widetilde{V}(\omega_i)$ and eigenvalue σ. We order eigenvalues as $\sigma_1^{\omega_i} \leq \sigma_2^{\omega_i} \leq ....$ and select the the eigenfunctions corresponding to small eigenvalues. We define the set of GMsFEM coarse basis functions by pointwise multiplication as follows,

$$\Phi_{i,\ell} = \chi_i \psi_\ell^{\omega_i}, \quad \text{for } 1 \leq i \leq N_v \text{ and } 1 \leq \ell \leq L_i, \tag{12}$$

where L_i denotes the number of basis functions on the coarse neighborhood ω_i. The support of $\Phi_{i,\ell}$ is ω_i and we remark that there may be multiple basis functions corresponding to this neighborhood since in general $L_i \geq 1$. See [22, 34, 11] for discussion on how to chose L_i. We define the coarse GMsFEM space by

$$V_0 = \text{span}\{\Phi_{i,\ell} = \chi_i \psi_\ell^{\omega_i}, \quad i = 1, \ldots, N_v, \quad \ell = 1, \ldots, L_i\}. \tag{13}$$

Let p_D^H be a discrete interpolation of the boundary data p_D. We denote the classical multiscale solution by $p_{ms}(t)$ with $p_{ms}(t) - p_D^H \in V_0$ and such that

$$(\partial_t p_{ms}, v) + a(p_{ms}, v) = F(p_{ms}; v) \quad \text{for all } v \in V_0. \tag{14}$$

We construct the coarse-scale matrix of basis functions,

$$R_0^{\mathrm{T}} = [\Phi_{i,1}, ..., \Phi_{i,L_i}]$$

where $\Phi_{i,\ell}$ was introduced in (12). We then define the coarse-scale stiffness matrix and mass matrix by

$$A_0 = R_0 A R_0^{\mathrm{T}} \quad \text{and } M_0 = R_0 M R_0^{\mathrm{T}}, \tag{15}$$

respectively. The coarse-scale load vector is given by $b_{ms} = R_0 b$. With this, we can define the matrix coarse-scale nonlinear system associated to (14) as

$$M_0 \partial_t p_{ms} + A_0 p_{ms} = b_{ms}(p_{ms}). \tag{16}$$

This matrix problem is the GMsFEM coarse-scale version of the fine-scale matrix system (8). Different time stepping methods could be implemented for either (8) or (16). We could solve system (16) and then downscale the final time coarse-scale solution of this matrix problem. However, the approximation may deteriorate as the time advances. We will upscale (project on coarse space) residual vectors and downscale (to the fine-grid) coarse solution at each time step. If h_k denotes the time step, in order to compute the next time solution $p^h(t + h_k)$ from the current time solution $p^h(t)$, we proceed as follows:

1. **Fine-scale residual:** Compute a fine-mesh residual $r^h(t + h_k)$ using the information from the previous time step. Computation of the residual only involves find-grid matrix times find-grid vector products. It does not require solution of fine-grid linear systems neither the computation of functions of fine-grid matrices.

2. **Up-scaling:** Perform up-scaling of the residual vector to obtain a coarse-scale residual: $r_0^H = R_0 r_n^h$.

3. **Coarse-scale solve:** Solve the linear systems and/or function of matrices using coarse-scale matrices A_0, M_0. Here, we obtain a coarse-scale vector representing either the time increment w_0^H or the next time approximation of the solution $p_0^H(t + h_k)$.

4. **Down-scaling:** Compute the next time approximation on the fine grid by downscaling the product of 3. above: here we have, $p^h(t + h_k) = p^h(t) + R_0^T w_0^H$ or $p^h(t + h_k) = R_0^T p_0^H(t + h_k)$.

In the next section, we exemplify this procedure for finite difference and exponential integration time discretizations.

3. Time discretizations combined with GMsFEM spatial approximation

We now solve the ODE matrix systems (8)-(16) in an interval $I = [0, T]$ with the procedure described above. We consider a uniform partition $\{t_0, t_1, \dots, t_M\}$ of I with element size $h_k = T/M$ and approximate the coefficients $p_{ms}(t)$ at the partition points.

3.1. GMsFEM finite difference (GMsFEM-FD)

For the FD method, we use the semi-implicit θ-scheme,

$$M\left(\frac{p^{k+1} - p^k}{h_k}\right) = \theta\left(b^{k+1} - Ap^{k+1}\right) + (1-\theta)\left(b^k - Ap^k\right), \tag{17}$$

where $\theta \in [0, 1]$ and we call $p^k \approx p(t_k)$. We obtain the following solution for p^{k+1}

$$p^{k+1} = (M + h_k\,\theta\,A)^{-1}\left((M - h_k(1-\theta)A)\,p^k + h_k\left((1-\theta)b^k + \theta b^{k+1}\right)\right), \tag{18}$$

where $b^k = b(p^k)$. By taking $\theta = \frac{1}{2}$, we obtain the Crank-Nicholson scheme and by taking $\theta = 1$ we obtain the backward Euler method. The term b^{k+1} is unknown at each time step k so we consider a predictor-corrector algorithm as follows. We set $b^{k+1} = b^k$ to predict p^{k+1} by solving (18) and obtain a new value of b^{k+1} from it, then we use it to correct p^{k+1} by solving (18) again. We can repeat this until we achieve the desired precision.

Remark 1. *From our numerical experiments we found out that, in the presence of high-contrast coefficients, the stability of the θ-scheme is deteriorated with the contrast, specially for $\theta < 1$ due to the presence of the term $(1-\theta)(b^k - Ap^k)$ on the right hand side of the linear system in* (18)*. For this reason we consider only the case $\theta = 1$ with small enough time step size.*

As mentioned before, performing this calculation using fine-scale matrices is not practical because this matrix system has a huge dimension and is very ill-conditioned. In particular, for (18), we see that the computation of the linear system solution, $(M + h_k\,\theta\,A)^{-1}$ is the real bottleneck for this method in terms of computational time. We propose the use of GMsFEM for the approximation of the solution of this linear system. Therefore, we arrive to the GMsFEM-FD method

$$p^{k+1} = R_0^T(M_0 + h_k\,\theta\,A_0)^{-1}R_0\left((M - h_k(1-\theta)A)\,p^k + h_k\left((1-\theta)b^k + \theta b^{k+1}\right)\right). \tag{19}$$

Note that, before using the GMsFEM approximation of the solution of the linear system, we upscale the residue of the previous iteration to the coarse mesh. After computing the solution in the coarse space, we downscale it back to the fine mesh. In summary, we perform the following computation in each time step:

1. Fine scale residue: $r_h = (M - h_k(1-\theta)A)\, p^k + h_k \left((1-\theta)b^k + \theta b^{k+1}\right)$,
2. Up-scaling of residue: $r_H = R_0 r_h$,
3. Coarse-scale solve: $p_H^{k+1} = (M_0 + h_k\, \theta\, A_0)^{-1} r_H$,
4. Downscaling of solution: $p_h^{k+1} = R_0^T p_H^{k+1}$.

3.2. Exponential Integrator (EI)

We know consider the exponential integrator method for integration in time. From (16), we derive the following equivalent system

$$\begin{cases} \partial_t p + Np = F(p) \quad \text{in} \quad I, \\ p(0) = p_0, \end{cases} \tag{20}$$

where $N = M^{-1}A$, $F = M^{-1}b(p)$ and $p_0 = M^{-1}\hat{p}$.
We now use exponential integrators to solve (20). All this methods are based on the following integral representation of the solution of (20) which is called variation-of-constants formula

$$p(t_k) = e^{-h_k N} p(t_{k-1}) + \int_0^{h_k} e^{(\tau - h_k)N} F(p(t_{k-1} + s))d\tau. \tag{21}$$

The main idea here is to find an approximation of the nonlinear term in the variational formula by an algebraic polynomial. In the case of linear problems, the integral in (21) is approximated using exponential quadrature rules. Taking s quadrature points $c_i \in [0, 1]$, we have

$$p^k = e^{-h_k N} p^{k-1} + h_k \sum_{i=0}^{s} b_i(-h_k N) F_i, \tag{22}$$

where $p^k \approx p(t_k)$, $F_i = F(t_{k-1} + c_i h_k)$, and $b_i(z)$, satisfies the following recurrence relations

$$\begin{aligned} z b_0(z) &= e^z - 1 \\ z b_{i+1}(z) + \frac{1}{(i+2)!} &= \left(1 - \frac{1}{(i+2)! z}\right) b_i(z) + ... + (1-z) b_0(z). \end{aligned} \tag{23}$$

The coefficient b_i can be rewritten as a linear combination of the φ-functions,

$$\begin{cases} \varphi_0(z) &= e^z \\ \varphi_p(z) &= \displaystyle\int_0^1 e^{(1-\theta)z} \frac{\theta^{p+1}}{(p-1)!} d\theta, \quad p \geq 1, \end{cases} \tag{24}$$

which satisfy the following recurrence relation

$$\varphi_p(z) = \frac{1}{z}\left(\varphi_p(z) - \frac{1}{p!}\right). \tag{25}$$

We can then construct the following iterative method. If we select one point $c_1 \in [0,1]$, we have $b_0(z) = \varphi_1(z)$ from (25) that $e^z = z\varphi_1(z) + 1$, we obtain the following method

$$p^k = p^{k-1} + h_k\varphi_1(-h_kN)\left(F_1 - Np^{k-1}\right). \tag{26}$$

In the case of semilinear problems, the construction of exponential integrators is more involved. Here, F depends on p so we need internal stages to approximate the the solution in different integration points. We refer to [25] for the construction of Exponential integrators of Runge-Kutta type. In this article, we focus in the lowest order Exponential Runge Kutta method. In (21), we approximate F by the value of the solution in the previous time step, i.e., $F \approx F^{k-1} = F(p^{k-1})$ and we obtain the following time-marching scheme

$$p^k = p^{k-1} + h_k\varphi_1(-h_kN)\left(F^{k-1} - Np^{k-1}\right). \tag{27}$$

which is called, Exponential Euler method.
Observe that the classical eigenvalue problem ([23, 28])

$$-h_kNq = \lambda q$$

is related to the generalized eigenvalue problem

$$-h_kAq = \lambda Mq.$$

Since M and A are symmetric and positive definite and $N = M^{-1}A$ we factor

$$-h_kN = QDQ^{-1}$$

where the columns of Q are the eigenvectors of $-h_kN$ or the generalized eigenvectors of $-h_kA$ with respect to M. The matrix D is the diagonal matrix of eigenvalues of $-h_kN$ that are also the generalized eigenvalues of $-h_kA$ with respect to M. Given that the eigenvectors q_i are orthonormal with respect to the inner product $(u,v)_M = u^TMv$, we have

$$Q^TMQ = I.$$

Now, using $Q^{-1} = Q^TM$, we rewrite

$$-h_kN = QDQ^TM. \tag{28}$$

Using (24) and (28), we obtain

$$\varphi_p(-h_kN) = Q\varphi_p(D)Q^TM. \tag{29}$$

If we use this in (27), we obtain

$$\begin{aligned} p^k &= p^{k-1} + h_k Q\varphi_1(D)Q^T M\left(F^{k-1} - Np^{k-1}\right) \\ &= p^{k-1} + h_k Q\varphi_1(D)Q^T\left(MF^{k-1} - Ap^{k-1}\right). \end{aligned} \tag{30}$$

As before, the implementation of this iteration using fine scale matrices is inadequate due to the very large computational time needed to numerically approximate the matrix functions. Instead, we use a GMsFEM approximation of the eigenvalue problem to speed up the computations of the φ-functions. As a result, we obtain the GMsFEM-EI iteration. In particular we propose the approximation

$$\varphi_p(-h_k N) \approx R_0^T \varphi_p(-h_k N_0) R_0 \tag{31}$$

where $N_0 = M_0^{-1} A_0$ with A_0 and M_0 defined in (15). The associated eigenvalue problem is

$$-h_k A_0 q_0 = \lambda M_0 q_0 \quad \text{or} \quad -h_k N_0 = Q_0 D_0 Q_0^T M_0. \tag{32}$$

Here, Q_0 is the matrix whose columns are GMsFEM-coarse-scale eigenvectors. Note that we apply the approximation in (31) to a fine-scale operator. Given the previous time fine-scale approximation, we first upscale the residual vector to the coarse space, then use the function computed at coarse resolution in the GMsFEM space, and finally, we downscale the result to the fine-grid. For instance, the approximation of the iteration in equation (27) using the GMsFEM-EI iteration is given by

$$p^k = p^{k-1} + h_k R_0^T \varphi_1(-h_k N_0) R_0\left(F^{k-1} - Np^{k-1}\right). \tag{33}$$

Note that we can use any other procedure to compute $\varphi_1(-h_k N_0)$. In terms of the coarse approximation of eigenvectors and eigenvalues in (32) we get

$$p^k = p^{k-1} + h_k R_0^T Q_0 \varphi_1(D_0) Q_0^T R_0\left(MF^{k-1} - Ap^{k-1}\right). \tag{34}$$

In summary we compute as follows

1. Fine scale residue: $r_h = MF^{k-1} - Ap^{k-1}$,
2. Up-scaling of residue: $r_H = R_0 r_h$,
3. Coarse-scale function of matrix: $\delta_H^{k+1} = Q_0\varphi_1(D_0)Q_0^T r_H$,
4. Downscaling of solution: $p_h^{k+1} = p_h^k + R_0^T \delta_H^{k+1}$.

For the computation of φ-functions, we use two different methods. The first one is the procedure described in (34) that requires the computation of a coarse-scale global generalized eigenvaluen problem. Note that the corresponding fine-scale formula in equation (30) will require the computation of a fine-scale global generalized eigenvalue problem involving large sparse and ill conditioned operators. The second method employs the MATLAB package called EXPINT. It is presented in [8] and use Padé approximations.

Remark 2. *Since we are going to solve the time evolution on the coarse space, as initial condition we use the*

orthogonal projection of the initial condition

$$\hat{p}_0 = R_0^T M_0^{-1} R_0 M p_0. \tag{35}$$

That is, we solve the approximated initial condition problem

$$\begin{cases} \partial_t p + Np = F(p) \quad in \quad I, \\ p(0) = \hat{p}_0. \end{cases} \tag{36}$$

4. Numerical results

In this section we present some representative numerical experiments to show the performance of combining GMsFEM with FD and also GMsFEM with EI. We first consider a homogeneous media and later an heterogeneous media with high-contrast. We chose $\Omega = (0,1)^2$ and structured mesh of triangles obtained from dividing Ω in regularly on squares and then dividing each square into two triangles from the bottom-left to the upper-right corner. We consider the following norms,

$$\|p\|_{L^2}^2 = \int_\Omega p^2, \qquad \|p\|_{L_w^2}^2 = \int_\Omega \kappa p^2, \qquad \|p\|_{H_w^1}^2 = \int_\Omega \kappa \nabla p \cdot \nabla p. \tag{37}$$

4.1. Example 1: constant permeability coefficient and linear equation

We consider the problem

$$\begin{cases} \partial_t p - \operatorname{div}(\nabla p) = (5\pi^2 - 1)p & \text{in} \quad \Omega = [0,1]^2 \\ p(0, x_1, x_2) = \sin(2\pi x_1)\sin(\pi x_2), & \\ p(t, x_1, x_2) = 0, & \text{on} \quad \partial\Omega, \end{cases} \tag{38}$$

where the exact solution is $p(x_1, x_2, t) = e^{-t}\sin(2\pi x_1)\sin(\pi x_2)$. We use a coarse mesh made of squares obtaining by dividing Ω into 10×10 squares obtaining 81 interior coarse nodes. We compute numerical solutions $p_{ms}^{(L_i)}$ by taking $L_i = 1, 2, 3, 4, 5, 6$, that is, we use up to 6 multiscale basis functions in each neighborhood.

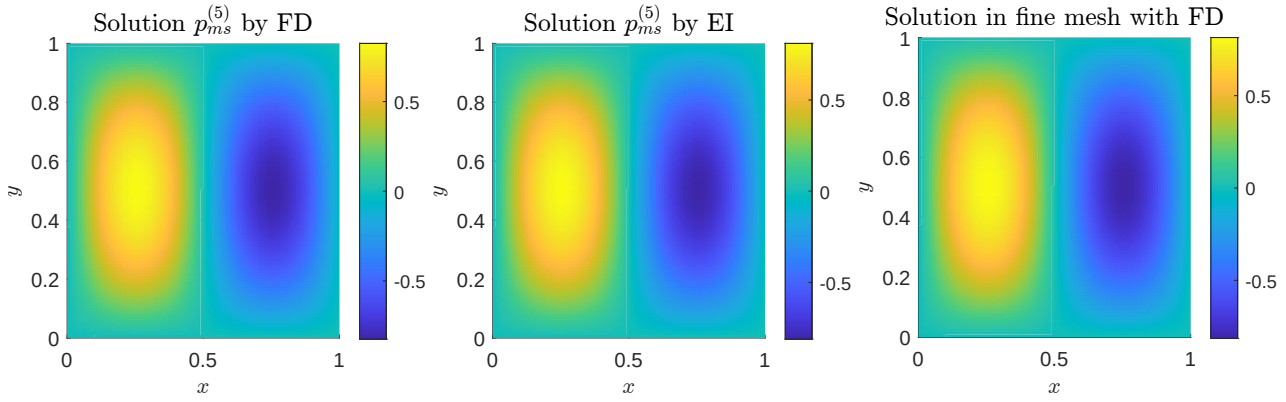


Figure 2: Final time ($T = 0.2$) solution for problem (38). Computed solution using MsFEM-FD with 5 basis functions in each neighborhood and 50 times steps (left). Computed solution using MsFEM-EI with 5 basis functions in each neighborhood and 60 times steps (center). Reference (exact) solution (right).

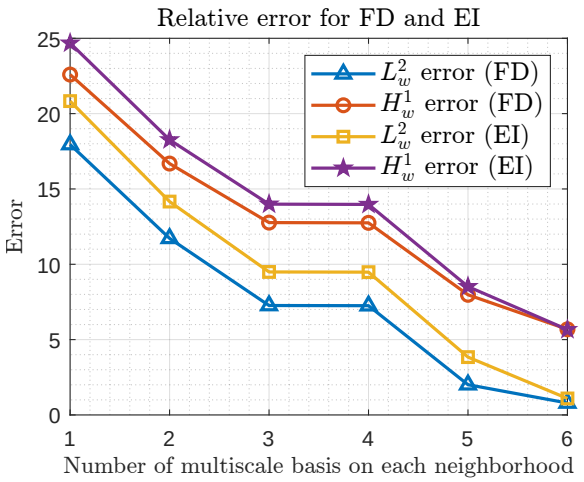


Figure 3: The weighted L^2 and H^1 errors between the reference and the coarse-scale solution at the final time $T = 0.2$ for problem (38). The horizontal axis corresponds to the number of basis functions in each neighborhood used in the GMsFEM coarse spaces.

	FD % Error		EI (eig) % Error		EI (EXPINT) % Error	
L_i	L_w^2	H_w^1	L_w^2	H_w^1	L_w^2	H_w^1
1	22.1	22.8	2.9	3.5	2.9	3.5
2	11.7	12.3	1.6	2.1	1.6	2.1
3	6.2	6.9	0.9	1.3	0.9	1.3
4	6.2	6.9	0.9	1.3	0.9	1.3
5	1.5	2.0	0.3	0.5	0.3	0.5
6	0.3	0.6	0.1	0.2	0.1	0.2

Table 1: The weighted L^2 and H^1 errors between the reference and the coarse-scale solution at the final time $T = 0.2$ for problem (38). In the last column we have added the relative error when the matrix functions are computed using `MatLab expint`.

Our reference solution is the exact solution interpolated to the fine-mesh. In this example, the final time of simulation is $T = 0.5$ and we use 60 time steps for the GMsFEM-FD and the GMsFEM-EI methods. In Figure 2 we depict the solution at the final time. We display the exact solution and also the solution obtained by GMsFEM-FD and GMsFEM-EI methods. In Figure 3 and Table 1 we show the obtained final time weighted H^1 and L^2 errors. From the table and figures we see a good agreement between the coarse scale solutions and the reference solution that in this case is the exact solution. We also observe that the errors, as it is expected, decay as basis functions are added to each neighborhood. We additionally observe that there is not much difference between using the Matlab `expint` routing or the multiscale approximation of eigenvectors and eigenvalues used to computed the matrix functions required in the calculations.

4.2. Example II: low-contrast coefficient and linear equation

We now consider a high-contrast coefficient example. With this example we want to show the effects of the contrast in the stability of the FD approximation. Our reference solution is obtained in the fine-mesh using implicit Euler scheme

with a small enough time step size. We consider the problem,

$$\begin{cases} \partial_t p - \text{div}(\kappa(x)\nabla p) = 0, & \text{in} \quad \Omega = [0,1]^2, \\ p(0, x_1, x_2) = x_1(1-x_1)x_2(1-x_2), & \\ p(t, x_1, x_2) = 0, & \text{on} \quad \partial\Omega, \end{cases} \tag{39}$$

where the high-contrast coefficient κ is depicted in Figure 4. We consider the case where the value of the contrast is set to 10 in this subsection and 1000 in the next subsection. The initial value is given for the following function,

$$u(0, x_1, x_2) = x_1(1-x_1)x_2(1-x_2), \qquad \text{for all } (x_1, x_2) \in \Omega. \tag{40}$$

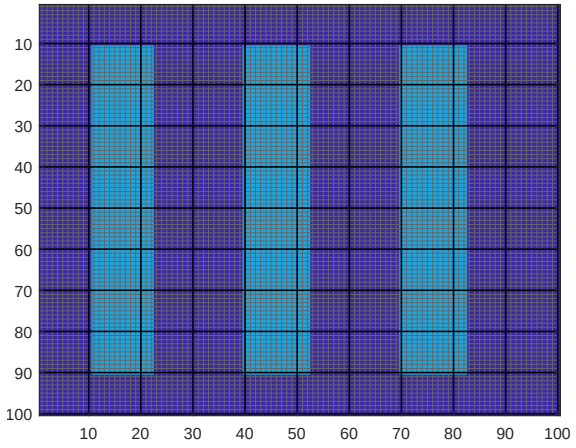


Figure 4: High-contrast coefficients used in the numerical experiment considered in Example 2. We note the presence of three vertical and long high-contrast channel across the domain. In the background we set the value 1 for the coefficient and equal to the contrast inside the 3 channels.

In Table 2 we show the obtained final time weighted H^1 and L^2 errors. We observe an increase in the FD errors due to the presence of a moderated contrast in the coefficient. As motioned before in Remark 1 we need to decrese the time step in order to obtain a better approximation in the final time. See the next section for a more drastic case. We stress that the GMsFEM-EI errors remain similar to the case of constant coefficient.

	FD % Error		EI (eig) % Error		EI (EXPINT) % Error	
L_i	L^2_w	H^1_w	L^2_w	H^1_w	L^2_w	H^1_w
1	10.49	25.20	23.19	28.24	23.19	28.24
2	20.47	28.25	14.53	20.07	14.53	20.07
3	30.15	33.21	6.62	11.98	6.62	11.98
4	30.83	33.61	6.07	11.36	6.07	11.36
5	34.01	35.53	3.50	8.19	3.50	8.19
6	35.23	36.30	2.50	6.72	2.50	6.72

Table 2: The weighted L^2 and H^1 errors between the reference and the coarse-scale solution at the final time $T = 0.2$ for problem (39). In the last column we have added the relative error when the matrix functions are computed using `MatLab expint`. The contrast is 10 in this experiment.

.

4.3. Example III: high-contrast coefficient and linear equation

We consider the same problem as in Example II but now with higher contrast equal to 1000. Recall that we use $\theta = 1$. See Remark 1. Recall also that our reference solution is obtained using implicit Euler in the fine-mesh with a small enough time step size. For this example we used 1000 time steps. The final time of simulation is $T = 0.2$ and we use 50 and 150 time steps for the GMsFEM-FD and the GMsFEM-EI methods. In Figure 5 we depict the solution at the final time. We display the references solution and also the solution obtained by GMsFEM-FD and GMsFEM-EI methods. In Figure 6 and Table 3 we show the obtained final time weighted H^1 and L^2 errors. From the table and figures we see a good agreement between the coarse scale solutions and the reference solution for MsFEM-FD with small enough time step. If the MsFEM-FD time step is not small enough, we observe and increase in error as we add more basis functions in each neighborhood. On the other hand, for MsFEM-EI we observe that the errors decay as basis functions are added to the neighborhood. As before, we observe that there there is not much difference between using the Matlab expint routing or the multiscale approximation of eigenvectors and eigenvalues used to computed the matrix functions required in the calculations.

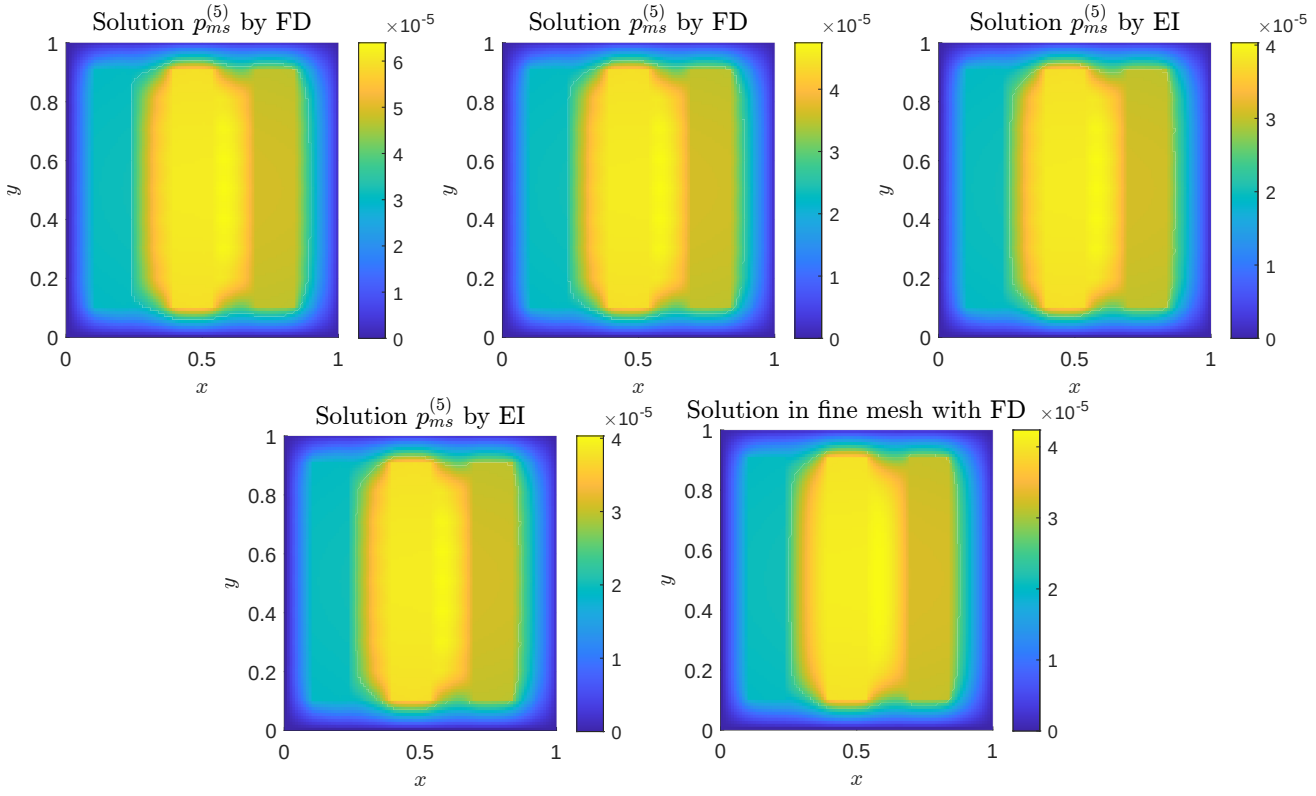
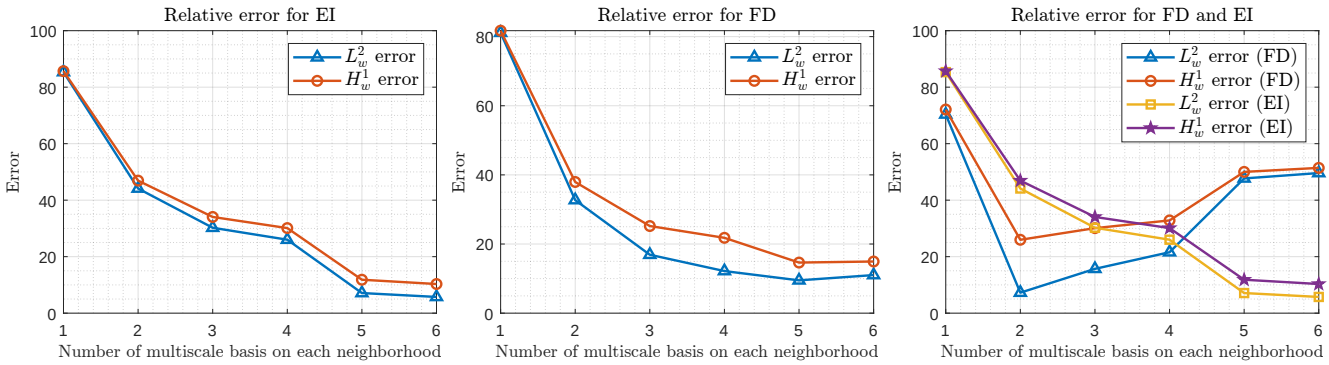


Figure 5: Final time ($T = 0.2$) solution for problem (39). Computed solution using MsFEM-FD with 50 times steps (top-left). Computed solution using MsFEM-FD with 150 times steps (top-right). Computed solution using MsFEM-EI with 50 times steps (center-left). Computed solution using MsFEM-EI with 10 times steps (center-right). Reference fine mesh solution with MsFEM-FD and 30000 times step (bottom-left). In all solutions, we use 5 basis functions in each neighborhood.



(a) MsFEM-EI with 10 steps number.

(b) MsFEM-FD with 150 steps number.

(c) MsFEM-FD and MsFEM-EI with 50 steps number.

Figure 6: The weighted L^2 and H^1 errors between the reference and the coarse-scale solution at the final time $T = 0.2$ for problem (39). The horizontal axis corresponds to the number of basis functions in each neighborhood used in the GMsFEM coarse spaces construction.

	FD % Error		EI (eig) % Error		EI (EXPINT) % Error	
L_i	L_w^2	H_w^1	L_w^2	H_w^1	L_w^2	H_w^1
1	82.80	83.33	86.58	86.90	86.58	86.90
2	35.88	40.70	46.44	49.14	46.44	49.14
3	16.06	24.57	28.91	32.90	28.91	32.90
4	12.37	21.89	25.58	29.80	25.58	29.80
5	9.15	14.22	6.70	11.47	6.70	11.47
6	10.46	14.43	5.50	10.14	5.50	10.14

Table 3: The weighted L^2 and H^1 errors between the reference and the coarse-scale solution at the final time $T = 0.2$ with 50 times step for problem (39). In the last column we have added the relative error when the matrix functions are computed using `MatLab expint`.

4.4. Example IV: high-contrast coefficient and semilinear equation

In this section we consider a semilinear PDE with a high-contrast permeability coefficient. We use the coefficient depicted in the Figure 4 with contrast equal to 1000. We see a behaviour similar to the last section where for GMsFEM-FD we need a small nought time step in order to obtain good approximation at the final time. As expected, the GMsFEM-EI works better than GMsFEM-FD for a larger time step size. See Figures 8 and 9 and Table 4.

$$\begin{cases} \partial_t p - \operatorname{div}(\kappa(x)\nabla p) = -p(1-p)(1+p) & \text{in} \quad \Omega = [0,1]^2 \\ p(0, x_1, x_2) = x_1(1-x_1)x_2(1-x_2), & \\ p(t, x_1, x_2) = 0, & \text{on} \quad \partial\Omega, \end{cases} \tag{41}$$

where κ has a contrast of 1000. The field κ is represented in the Figure 7.

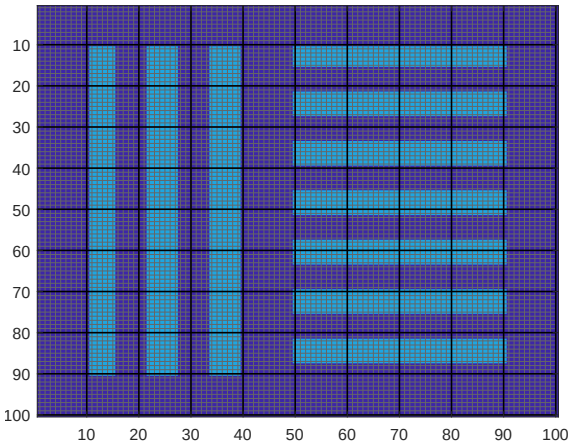


Figure 7: High-contrast coefficients used in the numerical experiment considered in Example 4.4.

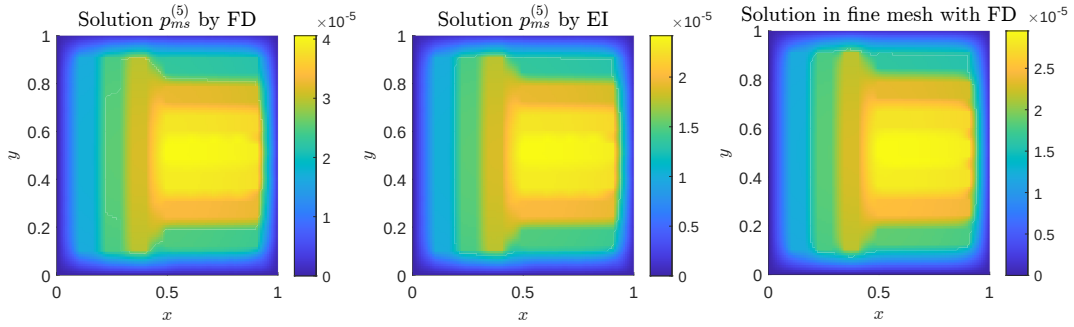


Figure 8: Final time ($T = 0.2$) solution for problem (41). Computed solution using MsFEM-FD with 5 basis functions in each neighborhood and 50 times steps (left). Computed solution using MsFEM-EI with 5 basis functions in each neighborhood and 50 time steps (center). Reference fine mesh solution with 30000 time steps (top-right). (right).

	FD % Error		EI (eig) % Error	
L_i	L_w^2	H_w^1	L_w^2	H_w^1
1	70.8	72.5	85.5	85.8
2	21.6	33.1	25.4	29.7
3	25.6	35.3	22.5	27.0
4	27.5	36.4	21.2	25.8
5	28.7	37.2	20.3	25.1
6	34.8	41.1	16.0	21.0

Table 4: The weighted L^2 and H^1 errors between the reference and the coarse-scale solution at the final time $T = 0.2$ for problem (41).

5. Final comments

In this paper we considered non-linear parabolic problems posed in a high-contrast multiscale media in two dimensions. We explained how the presence of high-contrast multiscale media adversely affects the accuracy, stability and overall efficiency of numerical approximations. We used the GMsFEM in order to implement finite differences and exponential integration in time in an efficient way without the need to compute with large ill-conditioned sparse matrices. We introduced the GMsFEM-FD and the GMsFEM-EI methods. As expected, the GMsFEM-FD requires a

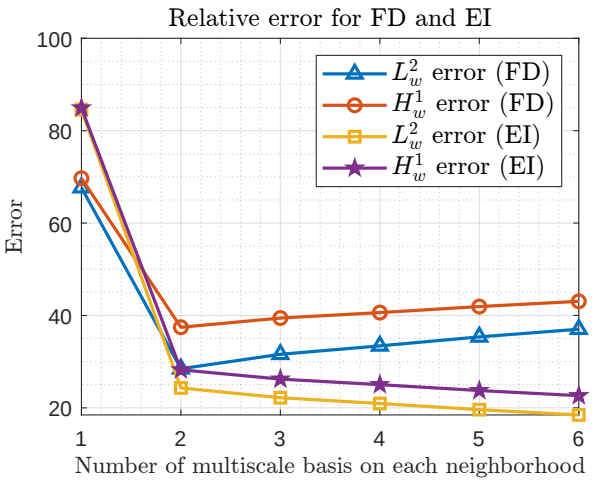


Figure 9: Error of the solutions of the parabolic problem with medium figure 4 for (FD) and (EI) methods with initial condition (41), for $T = 0.2$ with 50 steps, and contrast of 1000.

small-enough time step in order to render a good final time approximation of the solution while the GMsFEM-EI can handle bigger time-step sizes. We presented numerical results that indeed show that the high-contrast affects the stability of the time approximation and that exponential integration can be efficiently used in order to deal with high-contrast time dependent problems.

Acknowledgements

J. Galvis and L. Contreras thank the partial support from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 777778 (MATHROCKS). E. Abreu gratefully acknowledges the financial support of the National Council for Scientific and Technological Development - Brazil (CNPq) (Grant 306385/2019-8). Judit Muñoz-Matute has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie individual fellowship No. 101017984 (GEODPG) and the grant agreement No. 777778 (MATHROCKS). David Pardo has received funding from: the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement No 777778 (MATHROCKS); the European Regional Development Fund (ERDF) through the Interreg V-A Spain-France-Andorra program POCTEFA 2014-2020 Project PIXIL (EFA362/19); the Spanish Ministry of Science and Innovation projects with references PID2019-108111RB-I00 (FEDER/AEI) and PDC2021-121093-I00 (AEI/Next Generation EU), the "BCAM Severo Ochoa" accreditation of excellence (SEV-2017-0718); and the Basque Government through the BERC 2022-2025 program, the three Elkartek projects 3KIA (KK-2020/00049), EXPERTIA (KK-2021/00048), and SIGZE (KK-2021/00095), and the Consolidated Research Group MATHMODE (IT1456-22) given by the Department of Education. All the authors thank the MATHDATA - AUIP Network (*Red Iberoamericana de Investigación en Matemáticas Aplicadas a Datos*) (https://www.mathdata.science/).

References

[1] E. Abreu, C. Diaz, and J. Galvis. A convergence analysis of generalized multiscale finite element methods. *Journal of Computational Physics*, 396:303–324, 2019.

[2] Eduardo Abreu, Ciro Diaz, Juan Galvis, and John Pérez. On the conservation properties in multiple scale coupling and simulation for darcy flow with hyperbolic-transport in complex flows. *Multiscale Modeling & Simulation*, 18(4):1375–1408, 2020.

[3] Eduardo Abreu, Jean François, Wanderson Lambert, and John Pérez. A semi-discrete lagrangian–eulerian scheme for hyperbolic-transport models. *Journal of Computational and Applied Mathematics*, 406:114011, 2022.

[4] Awad H. Al-Mohy and Nicholas J. Higham. Computing the action of the matrix exponential, with an application to exponential integrators. *SIAM Journal on Scientific Computing*, 33(2):488–511, 2011.

[5] Todd Arbogast and Mary F. Wheeler. A nonlinear mixed finite element method for a degenerate parabolic equation arising in flow in porous media. *SIAM Journal on Numerical Analysis*, 33(4):1669–1687, 1996.

[6] Peter Bastian, Johannes Kraus, Robert Scheichl, and Mary Wheeler. *Simulation of flow in porous media: applications in energy and environment*, volume 12. Walter de Gruyter, 2013.

[7] Ilenia Battiato, Peter T Ferrero V, Daniel O'Malley, Cass T Miller, Pawan S Takhar, Francisco J Valdés-Parada, and Brian D Wood. Theory and applications of macroscale models in porous media. *Transport in Porous Media*, 130(1):5–76, 2019.

[8] Håvard Berland, Bård Skaflestad, and Will M. Wright. Expint—a matlab package for exponential integrators. *ACM Trans. Math. Softw.*, 33(1):4–es, mar 2007.

[9] F. Contreras C. Vazquez, J. Galvis. Numerical upscaling of the free boundary dam problem in multiscale high-contrast media. *Journal of Computational and Applied Mathematics*, 367, 2020.

[10] Victor M Calo, Yalchin Efendiev, Juan Galvis, and Guanglian Li. Randomized oversampling for g@articleefendiev2011multiscale, title=Multiscale finite element methods for high-contrast problems using local spectral basis functions, author=Efendiev, Yalchin and Galvis, Juan and Wu, Xiao-Hui, journal=Journal of Computational Physics, volume=230, number=4, pages=937–955, year=2011, publisher=Elsevier @articlecalo2016randomized, title=Randomized oversampling for generalized multiscale finite element methods, author=Calo, Victor M and Efendiev, Yalchin and Galvis, Juan and Li, Guanglian, journal=Multiscale Modeling & Simulation, volume=14, number=1, pages=482–501, year=2016, publisher=SIAM zambrano2021fasteneralized multiscale finite element methods. *Multiscale Modeling & Simulation*, 14(1):482–501, 2016.

[11] Eric Chung, Yalchin Efendiev, and Thomas Y Hou. Adaptive multiscale model reduction with generalized multiscale finite element methods. *Journal of Computational Physics*, 320:69–95, 2016.

[12] Eric Chung, Yalchin Efendiev, Sai-Mang Pun, and Zecheng Zhang. Computational multiscale method for parabolic wave approximations in heterogeneous media. *Applied Mathematics and Computation*, 425:127044, 2022.

[13] Eric T. Chung, Yalchin Efendiev, Wing Tat Leung, and Petr N. Vabishchevich. Contrast-independent partially explicit time discretizations for multiscale flow problems. *Journal of Computational Physics*, 445:110578, 2021.

[14] L. Macul E. Abreu1, P. Ferraz. A multiscale recursive numerical method for semilinear parabolic problems. *CILAMCE, PANACM*, 2021.

[15] 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.

[16] Y. Efendiev and J. Galvis. 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 *Lecture Notes in Computational Science and Engineering*, pages 189–196, Berlin, 2011. Springer-Verlag.

[17] Y. Efendiev, J. Galvis, and T. Hou. Generalized multiscale finite element methods. *Journal of Computational Physics*, 251:116–135, 2013.

[18] Y. Efendiev and T. Hou. *Multiscale Finite Element Methods: Theory and Applications*, volume 4 of *Surveys and Tutorials in the Applied Mathematical Sciences*. Springer, New York, 2009.

[19] Yalchin Efendiev, Juan Galvis, and Xiao-Hui Wu. Multiscale finite element methods for high-contrast problems using local spectral basis functions. *Journal of Computational Physics*, 230(4):937–955, 2011.

[20] Yalchin Efendiev, Sai-Mang Pun, and Petr N. Vabishchevich. Temporal splitting algorithms for non-stationary multiscale problems. *Journal of Computational Physics*, 439:110375, 2021.

[21] J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high contrast media. *SIAM J. Multiscale Modeling and Simulation*, 8:1461–1483, 2010.

[22] J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high contrast media. reduced dimension coarse spaces. *SIAM J. Multiscale Modeling and Simulation*, 8:1621–1644, 2010.

[23] N. Higham. *Functions of matrix theory and computation*. SIAM, University of Manchester, United Kingdom, 2008.

[24] Marlis Hochbruck, Christian Lubich, and Hubert Selhofer. Exponential integrators for large systems of differential equations. *SIAM Journal on Scientific Computing*, 19(5):1552–1574, 1998.

[25] Marlis Hochbruck and Alexander Ostermann. Exponential integrators. *Acta Numerica*, 19:209–286, 2010.

[26] Jianguo Huang, Lili Ju, and Bo Wu. A fast compact exponential time differencing method for semilinear parabolic equations with neumann boundary conditions. *Applied Mathematics Letters*, 94:257–265, 2019.

[27] D. Pardo J. Muñoz and L. Demkowicz. Equivalence between the dpg method and the exponential integrator for linear parabolic problems. *Journal of Computational Physics*, 2020.

[28] J. Galvis J. Olmos and F. Martinez. A geometric mean algorithm of symmetric positive definite matrices. *unpublished*, 2021.

[29] Lijian Jiang, Yalchin Efendiev, and Victor Ginting. Multiscale methods for parabolic equations with continuum spatial scales. *Discrete and Continuous Dynamical Systems - B*, 8(4):833–859, 2007.

[30] Axel Målqvist and Anna Persson. Multiscale techniques for parabolic equations. *Numerische Mathematik*, 138, 01 2018.

[31] Natarajan Narayanan, Berlin Mohanadhas, and Vasudevan Mangottiri. *Flow and Transport in Subsurface Environment*. Springer, 2018.

[32] Michael Presho and Juan Galvis. A mass conservative generalized multiscale finite element method applied to two-phase flow in heterogeneous porous media. *Journal of Computational and Applied Mathematics*, 296:376–388, 2016.

[33] Zheng Sun, José A. Carrillo, and Chi-Wang Shu. A discontinuous galerkin method for nonlinear parabolic equations and gradient flow problems with interaction potentials. *Journal of Computational Physics*, 352:76–104, 2018.

[34] J. Galvis Y. Efendiev and X. Wu. Multiscale finite element methods for high-contrast problems using local spectral basis functions. *Journal of Computational Physics*, 230:937–955, 2011.

[35] Miguel Zambrano, Sintya Serrano, Boyan S Lazarov, and Juan Galvis. Fast multiscale contrast independent preconditioners for linear elastic topology optimization problems. *Journal of Computational and Applied Mathematics*, 389:113366, 2021.