Journal of Scientific Computing (2024) 99:77
https://doi.org/10.1007/s10915-024-02539-9

Multilevel Monte Carlo Methods for Stochastic Convection–Diffusion Eigenvalue Problems

Tiangang Cui[1] · Hans De Sterck[2] · Alexander D. Gilbert[3] · Stanislav Polishchuk[4] · Robert Scheichl[5]

Received: 4 March 2023 / Revised: 21 December 2023 / Accepted: 14 March 2024 /
Published online: 3 May 2024

Abstract
We develop new multilevel Monte Carlo (MLMC) methods to estimate the expectation of the smallest eigenvalue of a stochastic convection–diffusion operator with random coefficients. The MLMC method is based on a sequence of finite element (FE) discretizations of the eigenvalue problem on a hierarchy of increasingly finer meshes. For the discretized, algebraic eigenproblems we use both the Rayleigh quotient (RQ) iteration and implicitly restarted Arnoldi (IRA), providing an analysis of the cost in each case. By studying the variance on each level and adapting classical FE error bounds to the stochastic setting, we are able to bound the total error of our MLMC estimator and provide a complexity analysis. As expected, the complexity bound for our MLMC estimator is superior to plain Monte Carlo. To improve the efficiency of the MLMC further, we exploit the hierarchy of meshes and use coarser approximations as starting values for the eigensolvers on finer ones. To improve the stability of the MLMC method for convection-dominated problems, we employ two additional strategies. First, we consider the streamline upwind Petrov–Galerkin formulation

✉ Tiangang Cui
tiangang.cui@sydney.edu.au

Hans De Sterck
hdesterck@uwaterloo.ca

Alexander D. Gilbert
alexander.gilbert@unsw.edu.au

Stanislav Polishchuk
stanislav.polishchuk@monash.edu

Robert Scheichl
r.scheichl@uni-heidelberg.de

1 School of Mathematics and Statistics, The University of Sydney, Sydney, NSW 2006, Australia

2 Department of Applied Mathematics, University of Waterloo, Waterloo, ON N2L 3G1, Canada

3 School of Mathematics and Statistics, The University of New South Wales, Sydney, NSW 2052, Australia

4 School of Mathematics, Monash University, Melbourne, VIC 3800, Australia

5 Institute of Applied Mathematics and Interdisciplinary Center for Scientific Computing (IWR), Universität Heidelberg, Im Neuenheimer Feld 205, 69120 Heidelberg, Germany

of the discrete eigenvalue problem, which allows us to start the MLMC method on coarser meshes than is possible with standard FEs. Second, we apply a homotopy method to add stability to the eigensolver for each sample. Finally, we present a multilevel quasi-Monte Carlo method that replaces Monte Carlo with a quasi-Monte Carlo (QMC) rule on each level. Due to the faster convergence of QMC, this improves the overall complexity. We provide detailed numerical results comparing our different strategies to demonstrate the practical feasibility of the MLMC method in different use cases. The results support our complexity analysis and further demonstrate the superiority over plain Monte Carlo in all cases.

Keywords Convection–diffusion eigenvalue problems · Multilevel Monte Carlo · Uncertainty quantification · Homotopy

1 Introduction

We consider the following convection–diffusion eigenvalue problem with random coefficients: Find a non-trivial eigenpair $(\lambda, u) \in \mathbb{C} \times H_0^1(D; \mathbb{C})$ such that

$$-\nabla \cdot \big(\kappa(\mathbf{x}, \boldsymbol{\omega})\nabla u(\mathbf{x}, \boldsymbol{\omega})\big) + \mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u(\mathbf{x}, \boldsymbol{\omega}) = \lambda(\boldsymbol{\omega}) u(\mathbf{x}, \boldsymbol{\omega}). \tag{1}$$

The PDE is considered for the *physical variable* $\mathbf{x}$ in a bounded Lipschitz domain $D \in \mathbb{R}^d$ with $d = 1, 2$, or 3, and for the *stochastic variable* $\boldsymbol{\omega}$ from a given probability space $(\Omega, \mathcal{F}, \pi)$. For π-almost all $\boldsymbol{\omega} \in \Omega$, we assume Dirichlet conditions on the boundary $\Gamma = \partial D$,

$$u(\mathbf{x}, \boldsymbol{\omega}) = 0 \quad \text{for } \mathbf{x} \in \Gamma.$$

The conductivity $\kappa(\mathbf{x}, \boldsymbol{\omega}) : D \times \Omega \to \mathbb{R}$ is a log-uniform random field (as used in, e.g., [19]), defined using the process convolution approach in [37], such that

$$\log \kappa(\mathbf{x}, \boldsymbol{\omega}) = Z(\mathbf{x}, \boldsymbol{\omega}) = \sum_i \omega_i k(\mathbf{x} - \mathbf{c}_i), \tag{2}$$

with $k(\mathbf{x} - \mathbf{c}_i)$ a kernel centered at a certain number of points $\mathbf{c}_i \in D$ and i.i.d. uniform random variables $\omega_i \sim \mathcal{U}[0, 1]$. Similarly, the convection velocity $\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) : D \times \Omega \to \mathbb{R}^d$ can also be some bounded random variable, which also depends on uniform random variables $\omega_i \sim \mathcal{U}[0, 1]$ and is additionally assumed to be divergence-free, i.e.,

$$\nabla \cdot \mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) = 0. \tag{3}$$

The purpose of this paper is to compute the expectation of the smallest eigenvalue of (1),

$$\mathbb{E}[\lambda] = \int_\Omega \lambda(\boldsymbol{\omega}) \, \mathrm{d}\pi(\boldsymbol{\omega}), \tag{4}$$

using multilevel Monte Carlo methods.

Stochastic eigenvalue problems arise in a variety of physical and scientific applications and their numerical simulations. Factors such as measurement noise, limitations of mathematical models, the existence of hidden variables, the randomness of input parameters, and other factors contribute to uncertainties in the modelling and prediction of many phenomena. Applications of uncertainty quantification (UQ) specifically related to eigenvalue problems include: nuclear reactor criticality calculations [2, 3, 25], the derivation of the natural frequencies of an aircraft or a naval vessel [41], band gap calculations in photonic crystals [22, 27, 55], the computation of ultrasonic resonance frequencies to detect the presence of gas hydrates

[51], the analysis of the elastic properties of crystals with the use of rapid measurements [52, 61], or the calculation of acoustic vibrations [12, 66]. Stochastic convection–diffusion equations are used to describe simple cases of turbulent [24, 44, 54, 63] or subsurface flows [64, 67].

Monte Carlo sampling is one of the most popular methods for quantifying uncertainties in quantities of interest coming from stochastic PDEs. Although simple and robust, Monte Carlo methods can be severely inefficient when applied to UQ problems, because their slow convergence rate often requires a large number of samples to meet the desired accuracy. To improve the efficiency, the multilevel Monte Carlo (MLMC) method was developed, where the key idea is to reduce the computational cost by spreading the samples over a hierarchy of discretizations. The main idea was introduced by Heinrich [36] for path integration, then generalized by Giles [30] for SDEs. More recently, MLMC methods have been applied with great success to stochastic PDEs, see, e.g., [6, 7, 14, 28, 29, 53, 60, 65] specifically for eigenproblems. A general overview of MLMC is presented by Giles [31].

In this paper, we present a MLMC method to approximate (4), which, motivated by the use of MLMC for source problems described above, is based on a hierarchy of discretizations of the eigenvalue problem (1) and which is much more efficient in practice than a Monte carlo approximation. We consider two discretization methods, a standard Galerkin finite element method (FEM) and a streamline upwind Petrov–Galerkin (SUPG) method. The SUPG method improves the stability of the approximation for cases with high convection and also allows us to start the MLMC method from a coarser discretization. To further reduce the cost of our MLMC method, we again exploit the hierarchy of discretizations by using approximations on coarse levels as the starting values for the eigensolver on the fine level. We also present the two extensions of MLMC that aim to improve different aspects of the method. First, to improve the stability of the eigensolver for each sample we include a homotopy method for solving convection–diffusion eigenvalue problems in the MLMC algorithm. The homotopy method computes the eigenvalue of the convection–diffusion operator by following a continuous path starting from the pure diffusion operator. Second, to improve the overall complexity we present a multilevel quasi-Monte Carlo method that aims to speed up the convergence of the variance on each level by replacing the Monte Carlo samples with a quasi-Monte Carlo (QMC) quadrature rule.

The structure of the paper is as follows. Section 2 introduces the variational formulation of (1), along with necessary background material on stochastic convection–diffusion eigenvalue problems. Two discrete formulations of the eigenvalue problem are introduced: the Galerkin FEM and the SUPG method. Section 3 introduces the MLMC method and presents the corresponding complexity analysis. In particular, this section details how to efficiently use each eigensolver, the Rayleigh quotient and implicitly restarted Arnoldi iterations, within the MLMC algorithm. In Sect. 4, we present the two extensions of our MLMC algorithm: a homotopy MLMC and a multilevel quasi-Monte Carlo method. Section 5 presents numerical results for finding the smallest eigenvalue of the convection–diffusion operator in a variety of settings. In particular, we present examples for difficult cases with high convection.

To ease notation, for the remainder of the paper we combine the random variables in the convection and diffusion coefficients into a single uniform random vector of dimension $s < \infty$, denoted by $\boldsymbol{\omega} = (\omega_i)_{i=1}^s$ with $\omega_i \sim \mathcal{U}[0, 1]$. In this case, π is the product uniform measure on $\Omega := [0, 1]^s$.

2 Variational Formulation

The eigenvalue problem (1) needs to be discretized, because its solution is not analytically tractable for arbitrary geometries and parameters. As such, we apply the standard finite element method to (1) to obtain an approximation of the desired eigenpair (λ, u).

Before deriving the variational form of (1), we first establish certain assumptions about the problem domain, the random field $\kappa(\boldsymbol{\omega})$ and the velocity field $\mathbf{a}(\boldsymbol{\omega})$ for $\boldsymbol{\omega} \in \Omega$, which, in particular, ensure that the solution is in $H^2(D)$ [33] as well as incompressibility.

Assumption 1 Assume that $D \subset \mathbb{R}^d$, for $d = 1, 2$, or 3, is a bounded, convex domain with Lipschitz continuous boundary Γ.

Assumption 2 The diffusion coefficient is bounded from above and from below for almost all $\boldsymbol{\omega} \in \Omega$, i.e., there exist two constants $\kappa_{\min}, \kappa_{\max}$ such that $0 < \kappa_{\min} \leq \kappa(\mathbf{x}, \boldsymbol{\omega}) \leq \kappa_{\max} < \infty$. In addition, we assume that also $\|\kappa(\cdot, \boldsymbol{\omega})\|_{W^{1,\infty}} \leq \kappa_{\max}$ for almost all $\boldsymbol{\omega} \in \Omega$.

Assumption 3 The convection coefficient is divergence free, $\nabla \cdot \mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) = 0$ for all $\mathbf{x} \in D$, and uniformly bounded, $\|\mathbf{a}(\cdot, \boldsymbol{\omega})\|_{L^\infty} \leq \mathbf{a}_{\max}$, for almost all $\boldsymbol{\omega}$.

A simple example of a random convection term is a homogeneous convection, $\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) = [a_1\omega_1, \ldots, a_d\omega_d]^\top$ for $a_1, \ldots, a_d \in \mathbb{R}$, which are independent of $\mathbf{x}$. Another example is the curl of random vector field, e.g., $\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) = \nabla \times \boldsymbol{Z}(\mathbf{x}, \boldsymbol{\omega})$ where $\boldsymbol{Z}$ is a vector-valued random field similar to that defined in (2). Both of these examples satisfy Assumption 3.

Next we introduce the variational form of (1). Whenever it does not lead to confusion, we drop the spatial coordinate of (stochastic) functions for brevity—for example, $u(\mathbf{x}, \boldsymbol{\omega})$ is also written as $u(\boldsymbol{\omega})$. Let $V = H_0^1(\Omega)$ be the first-order Sobolev space of complex-valued functions with vanishing trace on the boundary with norm $\|v\|_V = \|\nabla v\|_{L^2}$. Then let V^* denote the dual space of V. Multiplying (1) by a test function $v \in V$ and then performing integration by parts, noting that we have no Neumann boundary condition term since $u(\mathbf{x}, \boldsymbol{\omega}) = 0$ on Γ, we obtain

$$\int_D \mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u(\mathbf{x}, \boldsymbol{\omega}) v(\mathbf{x}) \, \mathrm{d}\mathbf{x} + \int_D \kappa(\mathbf{x}, \boldsymbol{\omega}) \nabla u(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla v(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \lambda(\boldsymbol{\omega}) \int_D u(\mathbf{x}, \boldsymbol{\omega}) v(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

The variational eigenvalue problem corresponding to (1) is then: Find a non-trivial eigenpair $(\lambda(\boldsymbol{\omega}), u(\boldsymbol{\omega})) \in \mathbb{C} \times V$ with $\|u(\boldsymbol{\omega})\|_{L^2} = 1$ such that

$$\mathcal{A}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) + \mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) = \lambda(\boldsymbol{\omega}) \langle u(\boldsymbol{\omega}), v \rangle \quad \forall v \in V, \tag{5}$$

where

$$\mathcal{A}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) := \int_D \kappa(\mathbf{x}, \boldsymbol{\omega}) \nabla u(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla \overline{v(\mathbf{x})} \, \mathrm{d}\mathbf{x},$$
$$\mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) := \int_D \mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u(\mathbf{x}, \boldsymbol{\omega}) \overline{v(\mathbf{x})} \, \mathrm{d}\mathbf{x},$$

and $\langle \cdot, \cdot \rangle$ denotes the $L^2(D)$ inner product

$$\langle u(\boldsymbol{\omega}), v \rangle := \int_D u(\mathbf{x}, \boldsymbol{\omega}) \overline{v(\mathbf{x})} \, \mathrm{d}\mathbf{x}.$$

Since the velocity $\mathbf{a}$ is divergence free, $\nabla \cdot \mathbf{a} = 0$, the sesquilinear form in (5) is uniformly coercive, i.e.,

$$\mathcal{A}(\boldsymbol{\omega}; v, v) + \mathcal{B}(\boldsymbol{\omega}; v, v) \geq a_{\min} \|v\|_V^2, \quad \forall v \in V, \tag{6}$$

with $a_{\min} > 0$ independent of $\boldsymbol{\omega}$. It is also uniformly bounded, i.e.,

$$\mathcal{A}(\boldsymbol{\omega}; v, z) + \mathcal{B}(\boldsymbol{\omega}; v, z) \leq a_{\max} \|v\|_V \|z\|_V, \quad \forall v, z \in V, \tag{7}$$

with $a_{\max} < \infty$ independent of $\boldsymbol{\omega}$.

For each $\boldsymbol{\omega} \in \Omega$, the eigenvalue problem (5) admits a countable sequence of eigenvalues $(\lambda_k(\boldsymbol{\omega}))_{k=1}^{\infty} \subset \mathbb{C}$, which has no finite accumulation points, and the smallest eigenvalue, $\lambda_1(\boldsymbol{\omega})$, is real and simple, see, e.g., [4]. The eigenvalues are enumerated in order of increasing magnitude, counting multiplicity, such that

$$0 < \lambda_1(\boldsymbol{\omega}) < |\lambda_2(\boldsymbol{\omega})| \leq |\lambda_3(\boldsymbol{\omega})| \leq \cdots$$

with corresponding eigenfunctions $(u_k(\cdot, \boldsymbol{\omega}))_{k=1}^{\infty}$, enumerated accordingly.

In addition to the primal form (5), to facilitate our analysis later on we also consider the dual eigenproblem: Find a non-trivial dual eigenpair $(\lambda^*(\boldsymbol{\omega}), u^*(\boldsymbol{\omega})) \in \mathbb{C} \times V$ with $\|u^*(\boldsymbol{\omega})\|_{L^2} = 1$ such that

$$\mathcal{A}(\boldsymbol{\omega}; v, u^*(\boldsymbol{\omega})) + \mathcal{B}(\boldsymbol{\omega}; v, u^*(\boldsymbol{\omega})) = \overline{\lambda^*(\boldsymbol{\omega})} \langle v, u^*(\boldsymbol{\omega}) \rangle \quad \forall v \in V. \tag{8}$$

The primal and dual eigenvalues are related to each other via $\lambda(\boldsymbol{\omega}) = \overline{\lambda^*(\boldsymbol{\omega})}$.

Proposition 1 *For all $\boldsymbol{\omega} \in \Omega$, the smallest eigenvalue $\lambda_1(\boldsymbol{\omega})$ of (5) is simple and the gap is uniformly bounded, i.e., there exists $\rho > 0$, independent of $\boldsymbol{\omega}$, such that*

$$|\lambda_2(\boldsymbol{\omega}) - \lambda_1(\boldsymbol{\omega})| \geq \rho. \tag{9}$$

Proof For each $\boldsymbol{\omega} \in \Omega$, the Krein–Rutman Theorem implies that $\lambda_1(\boldsymbol{\omega})$ is simple. It remains to show that the gap is uniformly bounded for $\boldsymbol{\omega} \in \Omega$. Since the eigenvalues are continuous in $\boldsymbol{\omega}$, it follows that the gap is also continuous. Hence, there exists a strictly positive minimum on the compact domain Ω and we can take

$$\rho := \min_{\boldsymbol{\omega} \in \Omega} |\lambda_2(\boldsymbol{\omega}) - \lambda_1(\boldsymbol{\omega})| > 0.$$

□

Theorem 1 *Suppose Assumptions 1–3 hold. For $\boldsymbol{\omega} \in \Omega$, let $(\lambda(\boldsymbol{\omega}), u(\cdot, \boldsymbol{\omega}))$ be an eigenpair of the EVP (5) and let $(\lambda^*(\boldsymbol{\omega}), u^*(\cdot, \boldsymbol{\omega}))$ be the corresponding dual eigenpair of the adjoint EVP (8), i.e., $\lambda(\boldsymbol{\omega}) = \overline{\lambda^*(\boldsymbol{\omega})}$. Then, the primal and the dual eigenfunctions satisfy $u(\cdot, \boldsymbol{\omega}),\ u^*(\cdot, \boldsymbol{\omega}) \in V \cap H^2(D)$ with*

$$\|u(\boldsymbol{\omega})\|_{H^2} \leq C_{\lambda,2} |\lambda(\boldsymbol{\omega})| \quad \text{and} \quad \|u^*(\boldsymbol{\omega})\|_{H^2} \leq C_{\lambda^*,2} |\lambda^*(\boldsymbol{\omega})|, \tag{10}$$

for $C_{\lambda,2} < \infty$ and $C_{\lambda^*,2} < \infty$ independent of $\boldsymbol{\omega}$.

Proof Rearranging (1), we can write the Laplacian of $u(\cdot, \boldsymbol{\omega})$ as

$$\begin{aligned} -\Delta u(\mathbf{x}, \boldsymbol{\omega}) &= \frac{1}{\kappa(\mathbf{x}, \boldsymbol{\omega})} \big(\nabla \kappa(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u(\mathbf{x}, \boldsymbol{\omega}) - \mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u(\mathbf{x}, \boldsymbol{\omega}) + \lambda(\boldsymbol{\omega}) u(\mathbf{x}, \boldsymbol{\omega}) \big) \\ &=: f_{\boldsymbol{\omega}}(\mathbf{x}), \end{aligned}$$

which holds for almost all $\mathbf{x} \in D$. Since $\kappa(\cdot, \boldsymbol{\omega}) \in W^{1,\infty}(D)$, $\mathbf{a}(\cdot, \boldsymbol{\omega}) \in L^\infty(D)^d$, $u(\cdot, \boldsymbol{\omega}) \in V$ and $1/\kappa(\mathbf{x}, \boldsymbol{\omega}) \leq 1/\kappa_{\min} < \infty$ it follows that $f_{\boldsymbol{\omega}} \in L^2(D)$ with

$$\begin{aligned} \|f_{\boldsymbol{\omega}}\|_{L^2} &\leq \frac{1}{\kappa_{\min}} \left(\|\kappa(\boldsymbol{\omega})\|_{W^{1,\infty}} \|u(\boldsymbol{\omega})\|_V + \|\mathbf{a}(\boldsymbol{\omega})\|_{L^\infty} \|u(\boldsymbol{\omega})\|_V + |\lambda(\boldsymbol{\omega})| \right) \\ &\leq \frac{1}{\kappa_{\min}} \left(\left(\kappa_{\max} + \mathbf{a}_{\max} \right) \|u(\boldsymbol{\omega})\|_V + |\lambda(\boldsymbol{\omega})| \right), \end{aligned}$$

where in the last step we have used that $\|u(\boldsymbol{\omega})\|_{L^2} = 1$, as well as Assumptions 2 and 3. Since $\lambda(\boldsymbol{\omega})$, $u(\cdot, \boldsymbol{\omega})$ satisfy (5) with $\|u(\boldsymbol{\omega})\|_{L^2} = 1$ and the sesquilinear form is coercive, it follows from (6) that

$$|\lambda(\boldsymbol{\omega})| = |\mathcal{A}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), u(\boldsymbol{\omega})) + \mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), u(\boldsymbol{\omega}))| \geq a_{\min} \|u(\boldsymbol{\omega})\|_V^2 \geq a_{\min} C_{\text{Poin}}^2,$$

where in the last inequality we have used Poincaré's inequality, as well as $\|u(\boldsymbol{\omega})\|_{L^2} = 1$ again. The first inequality also implies $\|u(\boldsymbol{\omega})\|_V \leq \sqrt{|\lambda(\boldsymbol{\omega})|/a_{\min}}$. Thus, substituting these two bounds, the L^2-norm of $f_{\boldsymbol{\omega}}$ is bounded by

$$\|f_{\boldsymbol{\omega}}\|_{L^2} \leq \frac{1}{\kappa_{\min}} \left(\frac{\kappa_{\max} + \mathbf{a}_{\max}}{a_{\min} C_{\text{Poin}}} + 1 \right) |\lambda(\boldsymbol{\omega})|, \tag{11}$$

where the constant is independent of λ.

Finally, using classical results in Grisvard [33] it follows that

$$\|u(\boldsymbol{\omega})\|_{H^2} \leq C_D \|\Delta u(\boldsymbol{\omega})\|_{L^2} = C_D \|f_{\boldsymbol{\omega}}\|_{L^2},$$

where C_D depends only on the domain D. Finally, substituting in the bound on $\|f_{\boldsymbol{\omega}}\|_{L^2}$ (11) gives the desired upper bound (10).

The result for the dual eigenfunction follows analogously. □

2.1 Finite Element Formulation

Let $\{\mathcal{T}_h\}_{h>0}$ be a family of (quasi-)uniform, shape-regular, conforming meshes on the spatial domain D, where each $\mathcal{T}_h$ is parameterised by its mesh width $h > 0$. For $h > 0$, we approximate the infinite-dimensional space V by a finite-dimensional subspace V_h. In this paper, we consider piecewise linear finite element (FE) spaces, but the method will work also for more general spaces.

The resulting discrete variational problem is to find non-trivial primal and dual eigenpairs $(\lambda(\boldsymbol{\omega}), u_h(\boldsymbol{\omega})) \in \mathbb{C} \times V_h$ and $(\lambda^*(\boldsymbol{\omega}), u_h^*(\boldsymbol{\omega})) \in \mathbb{C} \times V_h$ such that

$$\mathcal{A}(\boldsymbol{\omega}; u_h(\boldsymbol{\omega}), v_h) + \mathcal{B}(\boldsymbol{\omega}; u_h(\boldsymbol{\omega}), v_h) = \lambda_h(\boldsymbol{\omega}) \langle u_h(\boldsymbol{\omega}), v_h \rangle, \quad \forall v_h \in V_h, \tag{12}$$

and

$$\mathcal{A}(\boldsymbol{\omega}; v_h, u_h^*(\boldsymbol{\omega})) + \mathcal{B}(\boldsymbol{\omega}; v_h, u_h^*(\boldsymbol{\omega})) = \overline{\lambda_h^*}(\boldsymbol{\omega}) \langle v_h, u_h^*(\boldsymbol{\omega}) \rangle, \quad \forall v_h \in V_h. \tag{13}$$

For each $\boldsymbol{\omega}$, it is well-known that for h sufficiently small the FE eigenvalue problem (12) admits $M_h := \dim(V_h)$ eigenpairs, denoted by

$$\left(\lambda_{h,1}(\boldsymbol{\omega}), u_{h,1}(\boldsymbol{\omega})\right), \left(\lambda_{h,2}(\boldsymbol{\omega}), u_{h,2}(\boldsymbol{\omega})\right), \ldots, \left(\lambda_{h,M_h}(\boldsymbol{\omega}), u_{h,M_h}(\boldsymbol{\omega})\right) \in \mathbb{C} \times V_h, \tag{14}$$

which approximate the first M_h eigenpairs of (5). This approach is also called the Galerkin method.

In convection-dominated regions, the Galerkin method has well-known stability issues for standard (Lagrange-type) FEs, if the element size h does not capture all necessary information about the flow. The Peclet number (sometimes called the mesh Peclet number) [68]

$$\mathrm{Pe}(\mathbf{x}, \boldsymbol{\omega}) = \frac{|\mathbf{a}(\mathbf{x}, \boldsymbol{\omega})|h}{2\kappa(\mathbf{x}, \boldsymbol{\omega})} \tag{15}$$

governs how small the mesh size h should be in order to have a stable solution using basic (Lagrange-type) FE methods.

The error in the FE approximations (14) can be analysed using the Babuška–Osborn theory [4]. We state the error bounds for a simple eigenpair.

Theorem 2 *Let $(\lambda(\boldsymbol{\omega}), u(\boldsymbol{\omega}))$ be an eigenpair of (5) that is simple for all $\boldsymbol{\omega} \in \Omega$, where Ω is a compact domain. Then there exist constants C_λ, C_u, independent of h and $\boldsymbol{\omega}$, such that*

$$|\lambda(\boldsymbol{\omega}) - \lambda_h(\boldsymbol{\omega})| \leq C_\lambda h^2 \tag{16}$$

and $u_h(\boldsymbol{\omega})$ can be normalized such that

$$\|u(\boldsymbol{\omega}) - u_h(\boldsymbol{\omega})\|_V \leq C_u h. \tag{17}$$

Proof See Babuška and Osborn [4] and the appendix, where we show explicitly that the constants are bounded uniformly in $\boldsymbol{\omega}$. □

2.2 Streamline-Upwind Petrov–Galerkin Formulation

A sufficiently small Peclet number (15) guarantees numerical stability of the standard Galerkin method. One can either choose a small overall mesh size h or locally adapt the mesh size to satisfy the stability condition. However, globally reducing the mesh size may lead to a high computational cost, while local adaptations may need to be performed pathwise for each realisation of $\boldsymbol{\omega}$, which in turn leads to complications in the algorithmic design. In this section, we consider using the streamline-upwind Petrov–Galerkin (SUPG) method to improve numerical stability.

The SUPG method was introduced by Brooks and Hughes [10] to stabilize the finite element solution. Since then, the method has been extensively investigated and used in various applications [8, 15, 35, 39, 40, 43]. The SUPG method can be derived in several ways. Here, we introduce its formulation by adding a stabilization term to the bilinear form. An equivalent weak formulation can be obtained by defining a test space with additional test functions in the form $\hat{v}(\mathbf{x}) = v(\mathbf{x}) + p(\mathbf{x})$, where $v(\mathbf{x})$ is a standard test function in the finite element method and $p(\mathbf{x})$ is an additional discontinuous function.

We define the residual operator $\mathcal{R}$ as

$$\mathcal{R}(\boldsymbol{\omega}, \sigma)v = \mathbf{a}(\boldsymbol{\omega}) \cdot \nabla v - \nabla \cdot \kappa(\boldsymbol{\omega})\nabla v - \sigma v, \tag{18}$$

which gives the residual of the convection–diffusion equation (1) for a pair $(\sigma, v) \in \mathbb{C} \times V$. Then, stabilization techniques can be derived from the general formulation

$$\begin{aligned} &\mathcal{A}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) + \mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) \\ &\quad + \sum_{m=1}^{|\mathcal{T}_h|} \int_{D_m} \tau_m(\mathbf{x}, \boldsymbol{\omega}) \left(\mathcal{R}(\boldsymbol{\omega}, \lambda(\boldsymbol{\omega}))u(\mathbf{x}, \boldsymbol{\omega})\right) \left(\mathcal{P}(\boldsymbol{\omega})v(\mathbf{x})\right) \mathrm{d}\mathbf{x} \\ &\quad = \lambda(\boldsymbol{\omega})\langle u(\boldsymbol{\omega}), v\rangle, \end{aligned} \tag{19}$$

where $|\mathcal{T}_h|$ is the number of elements of the mesh $\mathcal{T}_h$, $\mathcal{P}(\boldsymbol{\omega})$ is some stabilization operator and $\tau_m(\boldsymbol{\omega})$ is the stabilization parameter acting in the mth finite element. The stabilization strategy will be determined by $\mathcal{P}(\boldsymbol{\omega})$ and $\tau_m(\boldsymbol{\omega})$.

Various definitions exist for the operator $\mathcal{P}(v, \boldsymbol{\omega})$, such as the Galerkin Least Square method [38], the SUPG method [9, 10, 23], the Unusual Stabilized Finite Element method [5], etc. For the SUPG method, the stablization operator $\mathcal{P}(\boldsymbol{\omega})$ is defined as

$$\mathcal{P}(\boldsymbol{\omega})v = \mathbf{a}(\boldsymbol{\omega}) \cdot \nabla v. \tag{20}$$

Substituting Eqs. (18) and (20) into (19) gives the SUPG weighted residual formulation

$$\begin{aligned}
&\mathcal{A}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) + \mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) + \sum_{m=1}^{|\mathcal{T}_h|} \int_{D_m} \Big(\tau_m(\mathbf{x}, \boldsymbol{\omega}) \big(\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u(\mathbf{x}, \boldsymbol{\omega}) \\
&\quad - \nabla \cdot \kappa(\mathbf{x}, \boldsymbol{\omega}) \nabla u(\mathbf{x}, \boldsymbol{\omega}) - \lambda(\boldsymbol{\omega}) u(\mathbf{x}, \boldsymbol{\omega}) \big) (\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla v(\mathbf{x})) \Big) \mathrm{d}\mathbf{x} \\
&\quad = \lambda(\boldsymbol{\omega}) \langle u(\boldsymbol{\omega}), v \rangle,
\end{aligned}$$

which is equivalent to

$$\begin{aligned}
&\mathcal{A}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) + \mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}), v) + \sum_{m=1}^{|\mathcal{T}_h|} \int_{D_m} \Big(\tau_m(\mathbf{x}, \boldsymbol{\omega}) \big(\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u(\mathbf{x}, \boldsymbol{\omega}) \\
&\quad - \nabla \cdot \kappa(\mathbf{x}, \boldsymbol{\omega}) \nabla u(\mathbf{x}, \boldsymbol{\omega}) \big) (\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla v(\mathbf{x})) \Big) \mathrm{d}\mathbf{x} \\
&\quad = \lambda(\boldsymbol{\omega}) \Big(\langle u(\boldsymbol{\omega}), v \rangle + \sum_{m=1}^{|\mathcal{T}_h|} \int_{D_m} \tau_m(\mathbf{x}, \boldsymbol{\omega}) u(\mathbf{x}, \boldsymbol{\omega}) \mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla v(\mathbf{x}) \mathrm{d}\mathbf{x} \Big).
\end{aligned} \tag{21}$$

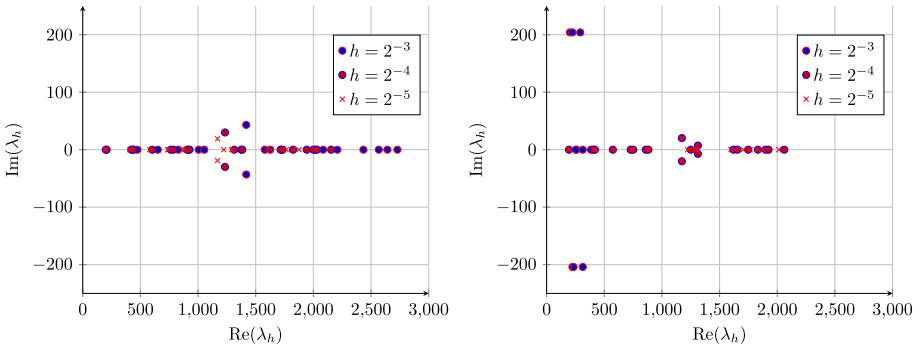
After approximating the weak form (21) by the usual finite-dimensional subspaces, we obtain the discrete variational problem: Find non-trivial (primal) eigenpairs $(\lambda_h(\boldsymbol{\omega}), u_h(\boldsymbol{\omega})) \in \mathbb{C} \times V_h$ such that

$$\begin{aligned}
&\mathcal{A}(\boldsymbol{\omega}; u_h(\boldsymbol{\omega}), v_h) + \mathcal{B}(\boldsymbol{\omega}; u_h(\boldsymbol{\omega}), v_h) + \sum_{m=1}^{|\mathcal{T}_h|} \int_{D_m} \Big(\tau_m(\mathbf{x}, \boldsymbol{\omega}) \big(\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u_h(\mathbf{x}, \boldsymbol{\omega}) \\
&\quad - \nabla \cdot \kappa(\mathbf{x}, \boldsymbol{\omega}) \nabla u_h(\mathbf{x}, \boldsymbol{\omega}) \big) (\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla v_h(\mathbf{x})) \Big) \mathrm{d}\mathbf{x} \\
&\quad = \lambda_h(\boldsymbol{\omega}) \Big(\mathcal{M}(u_h(\boldsymbol{\omega}), v_h) + \sum_{m=1}^{|\mathcal{T}_h|} \int_{D_m} \tau_m(\mathbf{x}, \boldsymbol{\omega}) u_h(\mathbf{x}, \boldsymbol{\omega}) \mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla v_h(\mathbf{x}) \mathrm{d}\mathbf{x} \Big),
\end{aligned} \tag{22}$$

and dual eigenpairs $(\lambda_h^*(\boldsymbol{\omega}), u_h^*(\boldsymbol{\omega})) \in \mathbb{C} \times V_h$ such that

$$\begin{aligned}
&\mathcal{A}(\boldsymbol{\omega}; v_h, u_h^*(\boldsymbol{\omega})) + \mathcal{B}(\boldsymbol{\omega}; v_h, u_h^*(\boldsymbol{\omega})) + \sum_{m=1}^{|\mathcal{T}_h|} \int_{D_m} \Big(\tau_m(\mathbf{x}, \boldsymbol{\omega}) \big(\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla v_h(\mathbf{x}) \\
&\quad - \nabla \cdot \kappa(\mathbf{x}, \boldsymbol{\omega}) \nabla v_h(\mathbf{x}) \big) (\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u_h^*(\mathbf{x}, \boldsymbol{\omega})) \Big) \mathrm{d}\mathbf{x} \\
&\quad = \overline{\lambda_h^*}(\boldsymbol{\omega}) \Big(\mathcal{M}(v_h, u_h^*(\boldsymbol{\omega})) + \sum_{m=1}^{|\mathcal{T}_h|} \int_{D_m} \tau_m(\mathbf{x}, \boldsymbol{\omega}) v_h(\mathbf{x}) \mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) \cdot \nabla u_h^*(\mathbf{x}, \boldsymbol{\omega}) \mathrm{d}\mathbf{x} \Big).
\end{aligned} \tag{23}$$

It follows that the right-hand side matrix is no longer symmetric and is stochastic compared to the mass matrix in the standard Galerkin method.



(a) SUPG eigenvalues. (b) Finite element eigenvalues.

Fig. 1 The first 20 computed eigenvalues of the SUPG (left) and FEM (right) discretizations of the convection–diffusion problem for $\kappa(\mathbf{x}) = 1$ and $\mathbf{a} = [50, 0]^T$ using mesh sizes $h = 2^{-3}, 2^{-4}, 2^{-5}$

In general, finding the optimal stabilization parameter $\tau_m(\mathbf{x}, \boldsymbol{\omega})$ is an open problem, and thus it is defined heuristically [43]. We employ the following stabilization parameter [8, 35]

$$\tau_m(\mathbf{x}, \boldsymbol{\omega}) = \frac{h_m}{2|\mathbf{a}(\mathbf{x}, \boldsymbol{\omega})|} \left(\coth \mathrm{Pe}(\mathbf{x}, \boldsymbol{\omega}) - \frac{1}{\mathrm{Pe}(\mathbf{x}, \boldsymbol{\omega})} \right). \tag{24}$$

However, in practical implementations the following asymptotic expressions of $\tau_m(\mathbf{x}, \boldsymbol{\omega})$ are used

$$\hat{\tau}_m(\boldsymbol{\omega}) = \begin{cases} \max\limits_{\mathbf{x} \in D_m} \dfrac{h_m}{2|\mathbf{a}(\mathbf{x}, \boldsymbol{\omega})|}, & \text{if } \max\limits_{\mathbf{x} \in D_m} \mathrm{Pe}(\mathbf{x}, \boldsymbol{\omega}) \geq 1, \\ \max\limits_{\mathbf{x} \in D_m} \dfrac{h_m^2}{12\kappa(\mathbf{x}, \boldsymbol{\omega})}, & \text{if } \max\limits_{\mathbf{x} \in D_m} \mathrm{Pe}(\mathbf{x}, \boldsymbol{\omega}) < 1. \end{cases} \tag{25}$$

Figure 1 shows the 20 smallest eigenvalues for a single realization of random field $\kappa(\mathbf{x}, \boldsymbol{\omega})$ with velocity $\mathbf{a}(\mathbf{x}, \boldsymbol{\omega}) = [50, 0]^T$ on meshes with size $h = 2^{-3}, 2^{-4}, 2^{-5}$. The standard Galerkin method has non-physical oscillations in the discretized eigenfunction for such a coarse mesh and its two smallest eigenvalues form a complex conjugate pair; this contradicts the fact that the smallest eigenvalue should be real and simple. The SUPG method, on the other hand, has a real smallest eigenvalue, indicating a stable solution.

3 Multilevel Monte Carlo Methods

To compute $\mathbb{E}[\lambda]$, we first approximate the eigenproblem (5) for each $\boldsymbol{\omega} \in \Omega$ and then use a sampling method to estimate the expected value of the approximate eigenvalue. There are two layers of approximation: First the eigenvalue problem is discretized by a numerical method, e.g., FEM or SUPG as in Sect. 2.1, then the resulting discrete eigenproblem is solved by an iterative eigenvalue solver, e.g., the Rayleigh quotient method, such that $\lambda(\boldsymbol{\omega}) \approx \lambda_h(\boldsymbol{\omega}) \approx \lambda_{h,K}(\boldsymbol{\omega})$, where h denotes the meshwidth of the spatial discretization and K denotes the number of iterations used by the eigenvalue solver.

Applying the Monte Carlo method to $\lambda_{h,K}$, the expected eigenvalue can be approximated by the estimator

$$\mathbb{E}[\lambda(\boldsymbol{\omega})] \approx Y_{h,K,N} := \frac{1}{N}\sum_{n=1}^{N} \lambda_{h,K}(\boldsymbol{\omega}_n), \tag{26}$$

where the samples $\{\boldsymbol{\omega}_n\}_{n=1}^N \subset \Omega$ are i.i.d. uniformly on Ω. This introduces a third factor that influences the accuracy of the estimator in (26) in addition to h and K, namely the number of samples N. Note that we assume that the number of iterations K is uniformly bounded in $\boldsymbol{\omega}$.

The standard Monte Carlo estimator in (26) is computationally expensive. To measure its accuracy we use the mean squared error (MSE)

$$\mathrm{MSE}(\mathbb{E}[\lambda(\boldsymbol{\omega})], Y_{h,K,N}) = \mathbb{E}\left[\left|\mathbb{E}[\lambda(\boldsymbol{\omega})] - Y_{h,K,N}\right|^2\right],$$

where the outer expectation is with respect to the samples in the estimator $Y_{h,K,N}$. Under mild conditions, the MSE can be decomposed as

$$\mathrm{MSE}(\mathbb{E}[\lambda], Y_{h,K,N}) = \left|\mathbb{E}[\lambda(\boldsymbol{\omega})] - \mathbb{E}[\lambda_{h,K}(\boldsymbol{\omega})]\right|^2 + \frac{1}{N}\mathrm{var}(\lambda_{h,K}(\boldsymbol{\omega})).$$

In this decomposition, the bias $\left|\mathbb{E}[\lambda(\boldsymbol{\omega})] - \mathbb{E}[\lambda_{h,K}(\boldsymbol{\omega})]\right|$ is controlled by h and K, whereas the variance term decreases linearly with $1/N$. To guarantee that the MSE remains below a threshold ε^2, h and K need to be chosen such that the bias is $O(\varepsilon^2)$, while the sample size needs to satisfy $N = O(\varepsilon^{-2})$. Suppose $K = K(h)$ is sufficiently large so that the bias is solely controlled by h and satisfies $\left|\mathbb{E}[\lambda(\boldsymbol{\omega})] - \mathbb{E}[\lambda_{h,K}(\boldsymbol{\omega})]\right| = O(h^\alpha)$ for some $\alpha > 0$. Suppose further that the computational cost to compute $\lambda_{h,K}(\boldsymbol{\omega})$ for each $\boldsymbol{\omega}$ is $O(h^{-\gamma})$ for some $\gamma > 0$. Then the total computational complexity to achieve an MSE of ε^2 is $O(\varepsilon^{-2-\gamma/\alpha})$. Note that in the best-case scenario, we have $\gamma = d$, i.e., when the computational cost of an eigensolver iteration is linear in the degrees of freedom of the discretization and the number of iterations can be bounded independently of h. Due to the quadratic convergence of algebraic eigensolvers, K is usually controlled very easily.

The multilevel Monte Carlo (MLMC) method offers a natural way to reduce the complexity of the standard Monte Carlo method by spreading the samples over a hierarchy of discretizations. In our setting, we define a sequence of meshes corresponding to mesh sizes $h_0 > h_1 > \cdots > h_L > 0$. This in turn defines a sequence of discretized eigenvalues $\lambda_{h_0,K_0}(\boldsymbol{\omega}), \lambda_{h_1,K_1}(\boldsymbol{\omega}), \ldots, \lambda_{h_L,K_L}(\boldsymbol{\omega})$ that approximate $\lambda(\boldsymbol{\omega})$ with increasing accuracy and increasing computational cost. The MLMC method approximates $\mathbb{E}[\lambda(\boldsymbol{\omega})]$ using the telescoping sum

$$\mathbb{E}[\lambda(\boldsymbol{\omega})] \approx \mathbb{E}[\lambda_L(\boldsymbol{\omega})] = \mathbb{E}[\lambda_0(\boldsymbol{\omega})] + \sum_{\ell=1}^{L} \mathbb{E}[\lambda_\ell(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})], \tag{27}$$

where $\lambda_\ell(\boldsymbol{\omega}) := \lambda_{h_\ell,K_\ell}(\boldsymbol{\omega})$ is the shorthand notation for the discretized eigenvalues. Each expected value of differences in (27) can be estimated by an independent Monte Carlo approximation, leading to the multilevel estimator

$$Y = \sum_{\ell=0}^{L} Y_\ell, \quad Y_\ell = \frac{1}{N_\ell}\sum_{n=1}^{N_\ell} (\lambda_\ell(\boldsymbol{\omega}_{\ell,n}) - \lambda_{\ell-1}(\boldsymbol{\omega}_{\ell,n})). \tag{28}$$

Suppose independent samples are used to compute each Y_ℓ, then

$$\mathbb{E}[Y] = \mathbb{E}[\lambda_L(\boldsymbol{\omega})], \quad \text{var}[Y] = \sum_{\ell=0}^{L} \frac{1}{N_\ell} \text{var}[\lambda_\ell(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})], \tag{29}$$

and the MSE of (28) can also be split into a bias and a variance term, i.e.,

$$\text{MSE}(\mathbb{E}[\lambda(\boldsymbol{\omega})], Y) = |\mathbb{E}[\lambda(\boldsymbol{\omega})] - \mathbb{E}[\lambda_L(\boldsymbol{\omega})]|^2 + \text{var}(Y).$$

Thus, to ensure again a MSE of $O(\varepsilon^2)$, it is sufficient to ensure that the bias, $\left|\mathbb{E}[\lambda(\boldsymbol{\omega})] - \mathbb{E}[\lambda_L(\boldsymbol{\omega})]\right|^2$, and the variance, var$[Y]$, are both less than $\frac{1}{2}\varepsilon^2$. The following theorem from [14] (see also [31]) provides bounds on the computational cost of a general MLMC estimator and applies in particular to (28).

Theorem 3 *Let Q denote a random variable and Q_ℓ its numerical approximation on level ℓ, and suppose C_ℓ is the computational cost of evaluating one realization of the difference $Q_\ell - Q_{\ell-1}$. Consider the multilevel estimator*

$$Y = \sum_{\ell=0}^{L} Y_\ell, \quad Y_\ell = \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} Q_{\ell,n} - Q_{\ell-1,n}, \tag{30}$$

where $Q_{\ell,n}$ is a sample of Q_ℓ and $Q_{-1,n} = 0$, for all n.

If there exist positive constants α, β, γ such that $\alpha \geq \frac{1}{2}\min(\beta, \gamma)$ and

I $|\mathbb{E}[Q_\ell - Q]| = O(h_\ell^\alpha)$ *(convergence of bias),*

II $\text{var}[Y_\ell] = O(h_\ell^\beta)$ *(convergence of variance),*

III $C_\ell = O(h_\ell^{-\gamma})$ *(cost per sample),*

then for any $0 < \varepsilon < e^{-1}$ there exist a constant c, a stopping level L, and sample sizes $\{N_\ell\}_{\ell=0}^{L}$ such that the MSE of Y satisfies $\text{MSE}(\mathbb{E}[Q], Y) \leq \varepsilon^2$ with a total computational complexity, denoted by $C(\varepsilon)$, satisfying

$$C(\varepsilon) \leq \begin{cases} c\varepsilon^{-2}, & \beta > \gamma; \\ c\varepsilon^{-2}(\log \varepsilon)^2, & \beta = \gamma; \\ c\varepsilon^{-2-(\gamma-\beta)/\alpha}, & \beta < \gamma, \end{cases} \tag{31}$$

where the constant c is independent of α, β and γ.

For a given ε, from [14] the maximum level L in Theorem 3 is given by

$$L = \lceil \alpha^{-1} \log_2(\sqrt{2}\, c_I\, \varepsilon^{-1}) \rceil, \tag{32}$$

where c_I is the implicit constant from Assumption *I* (convergence of bias) above. The optimal sample sizes, $\{N_\ell\}$, that minimize the computational cost of the multilevel estimator in Theorem 3 are obtained using a standard Lagrange multipliers argument as in [14] and are given by

$$N_\ell = \left\lceil 2\varepsilon^{-2} \sqrt{\frac{\text{var}[Q_\ell - Q_{\ell-1}]}{C_\ell}} \sum_{i=0}^{L} \sqrt{\text{var}[Q_i - Q_{i-1}] C_i} \right\rceil, \quad \ell = 0, \ldots, L. \tag{33}$$

Since $\beta > 0$, Theorem 3 shows that for all cases in (31), the MLMC complexity is superior to that of Monte Carlo. When $\beta > \gamma$, the variance reduction rate is larger than the rate of

increase of the computational cost, and thus most of the work is spent on the coarsest level. In this case, the multilevel estimator has the best computational complexity. When $\beta < \gamma$ the total computational work of the multilevel estimator may only have a marginal improvement compared to that of the classic Monte Carlo method.

Corollary 1 (Order of convergence) *For $\boldsymbol{\omega} \in \Omega$, let $h > 0$ be sufficiently small and consider two finite element approximations, cf. (12), of the smallest eigenvalue $\lambda(\boldsymbol{\omega})$ of the eigenvalue problem (5) with $h_{\ell-1} = h$ and $h_\ell = h/2$. The expectation of their difference is bounded by*

$$\left|\mathbb{E}[\lambda_\ell(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})]\right| \leq c_1 h_\ell^2, \tag{34}$$

while the variance of the difference is bounded by

$$\mathrm{var}[\lambda_\ell(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})] \leq c_2 h_\ell^4, \tag{35}$$

for two constants c_1, c_2 that are independent of $\boldsymbol{\omega}$, h and ℓ.

Proof Applying Theorem 2, since C_λ is independent of $\boldsymbol{\omega}$ we have

$$\left|\mathbb{E}[\lambda(\boldsymbol{\omega}) - \lambda_\ell(\boldsymbol{\omega})]\right| \leq \mathbb{E}[|\lambda(\boldsymbol{\omega}) - \lambda_\ell(\boldsymbol{\omega})|] \leq C_\lambda \left(\frac{h}{2}\right)^2, \tag{36}$$

and

$$\left|\mathbb{E}[\lambda(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})]\right| \leq \mathbb{E}[|\lambda(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})|] \leq C_\lambda h^2. \tag{37}$$

Therefore, by the triangle inequality, we have

$$\begin{aligned} \left|\mathbb{E}[\lambda_\ell(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})]\right| &= \left|\mathbb{E}[\lambda_\ell(\boldsymbol{\omega}) - \lambda(\boldsymbol{\omega}) + \lambda(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})]\right| \\ &\leq \mathbb{E}[|\lambda(\boldsymbol{\omega}) - \lambda_\ell(\boldsymbol{\omega})|] + \mathbb{E}[|\lambda(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})|] \\ &\leq C_\lambda \left(h^2 + \frac{h^2}{4}\right) = 5 C_\lambda h_\ell^2. \end{aligned} \tag{38}$$

The variance reduction rate comes from the following relation

$$\mathrm{var}[\lambda(\boldsymbol{\omega}) - \lambda_\ell(\boldsymbol{\omega})] \leq \mathbb{E}[(\lambda(\boldsymbol{\omega}) - \lambda_\ell(\boldsymbol{\omega}))^2] \leq C_\lambda^2 \left(\frac{h}{2}\right)^4, \tag{39}$$

and, similarly, by the Cauchy-Schwarz inequality

$$\begin{aligned} \mathrm{var}[\lambda_\ell(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})] &\leq \mathbb{E}[(\lambda_\ell(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega}))^2] \\ &= \mathbb{E}[(\lambda_\ell(\boldsymbol{\omega}) - \lambda(\boldsymbol{\omega}) + \lambda(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega}))^2] \\ &\leq 2\left(\mathbb{E}[(\lambda(\boldsymbol{\omega}) - \lambda_\ell(\boldsymbol{\omega}))^2] + \mathbb{E}[(\lambda(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega}))^2]\right) \\ &\leq 2\left(C_\lambda^2 \left(\frac{h}{2}\right)^4 + C_\lambda^2 h^4\right) = 34 C_\lambda^2 h_\ell^4. \end{aligned}$$

□

Remark 1 In our numerical experiments, we observed that the SUPG approximation of the eigenvalue problem, cf. (22), has similar rates of convergence α and β in MLMC compared to the standard finite element approximation.

An important physical property of the smallest eigenvalue of (5) is that it is real and strictly positive. Clearly, $\mathbb{E}[\lambda] > 0$ as well, and so we would like our multilevel approximation (28) to preserve this property. Below we show that a multilevel approximation based on Galerkin FEM with a geometrically-decreasing sequence of meshwidths is strictly positive provided that h_0 is sufficiently small.

Proposition 2 *Suppose that $h_\ell = h_0 2^{-\ell}$ for $\ell \in \mathbb{N}$ with $h_0 > 0$ sufficiently small and let $\lambda_{h_\ell}(\cdot)$ be the approximation of the smallest eigenvalue using the Galerkin FEM as in (12). Then, for any $L \in \mathbb{N}$, the multilevel approximation of the smallest eigenvalue is strictly positive, i.e.,*

$$\widetilde{Y} := \sum_{\ell=0}^{L} \widetilde{Y}_\ell \;=\; \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \big(\lambda_{h_\ell}(\boldsymbol{\omega}_{\ell,n}) - \lambda_{h_{\ell-1}}(\boldsymbol{\omega}_{\ell,n})\big) \;>\; 0.$$

Proof First, since λ is continuous and strictly positive on $\mathcal{D}$ it can be bounded uniformly from below, i.e., there exists $\breve{\lambda} > 0$ such that

$$\lambda(\boldsymbol{\omega}) \;\geq\; \breve{\lambda} > 0 \quad \text{for all } \boldsymbol{\omega} \in \Omega. \tag{40}$$

For $\ell = 0$, using (16) and (40) we can bound $\lambda_{h_0}(\boldsymbol{\omega})$ uniformly from below by

$$\lambda_{h_0}(\boldsymbol{\omega}) \;=\; \lambda(\boldsymbol{\omega}) - \big(\lambda(\boldsymbol{\omega}) - \lambda_{h_0}(\boldsymbol{\omega})\big) \;\geq\; \breve{\lambda} - C_\lambda h_0^2 \,.$$

Since this bound is independent of $\boldsymbol{\omega}$, it follows that

$$\widetilde{Y}_0 := \frac{1}{N_0} \sum_{n=1}^{N_0} \lambda_{h_0}(\boldsymbol{\omega}_{0,n}) \;\geq\; \frac{1}{N_0} \sum_{n=1}^{N_0} \big(\breve{\lambda} - C_\lambda h_0^2\big) \;=\; \breve{\lambda} - C_\lambda h_0^2. \tag{41}$$

Similarly, for $\ell \geq 1$ using (16) we obtain

$$\begin{aligned} \lambda_{h_\ell}(\boldsymbol{\omega}) - \lambda_{h_{\ell-1}}(\boldsymbol{\omega}) \;&=\; \lambda(\boldsymbol{\omega}) - \lambda_{h_{\ell-1}} - \big(\lambda(\boldsymbol{\omega}) - \lambda_{h_\ell}(\boldsymbol{\omega})\big) \\ &\geq\; -\big|\lambda(\boldsymbol{\omega}) - \lambda_{h_{\ell-1}}\big| - \big|\lambda(\boldsymbol{\omega}) - \lambda_{h_\ell}(\boldsymbol{\omega})\big| \\ &\geq\; -C_\lambda\big(h_{\ell-1}^2 + h_\ell^2\big) \;=\; -9C_\lambda h_0^2\, 2^{-2\ell}\,. \end{aligned}$$

Again, this bound is independent of $\boldsymbol{\omega}$ and so

$$\widetilde{Y}_\ell := \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \big(\lambda_{h_\ell}(\boldsymbol{\omega}_{\ell,n}) - \lambda_{h_{\ell-1}}(\boldsymbol{\omega}_{\ell,n})\big) \;\geq\; -9C_\lambda h_0^2\, 2^{-2\ell}. \tag{42}$$

Finally, we bound the multilevel approximation $\widetilde{Y}$ from below using (41) and (42) as follows,

$$\begin{aligned} \widetilde{Y} \;&=\; \widetilde{Y}_0 + \sum_{\ell=1}^{L} \widetilde{Y}_\ell \;\geq\; \breve{\lambda} - C_\lambda h_0^2 - \sum_{\ell=1}^{L} 9C_\lambda h_0^2\, 2^{-2\ell} \\ &>\; \breve{\lambda} - 9C_\lambda h_0^2 \sum_{\ell=0}^{L} 2^{-2\ell} \;>\; \breve{\lambda} - 9C_\lambda h_0^2 \sum_{\ell=0}^{\infty} 2^{-2\ell} \;=\; \breve{\lambda} - 12C_\lambda h_0^2 \;>\; 0, \end{aligned}$$

where we have used the property that h_0 is sufficiently small, i.e., $h_0 \leq \sqrt{\breve{\lambda}/(12C_\lambda)}$, to ensure $\widetilde{Y} > 0$, as required. □

The result above can be extended beyond the geometric sequence of FE meshwidths to a general sequence of FE meshwidths, provided that $\sum_{\ell=0}^{L} h_\ell^2$ is sufficiently small. Similarly, as in Remark 1, we observe that the MLMC approximations based on SUPG are also strictly positive.

Choosing the number of iterations K_ℓ such that the error of the eigensolver is of the same order as the FE error on each level, i.e., $|\lambda_{h_\ell}(\boldsymbol{\omega}) - \lambda_{h_\ell,K_\ell}(\boldsymbol{\omega})| \lesssim h_\ell^2$ for all $\ell = 0, 1, \ldots, L$

and $\boldsymbol{\omega} \in \Omega$, it can similarly be shown that the multilevel approximation (28) also satisfies $Y > 0$.

To obtain the eigenvalue approximation on level ℓ, choosing a basis for the FE space $V_\ell := V_{h_\ell}$ in (12) leads to a generalized (algebraic) eigenproblem in matrix form for each sample $\boldsymbol{\omega}$, i.e.,

$$\mathbf{A}_\ell(\boldsymbol{\omega})\mathbf{u}_\ell(\boldsymbol{\omega}) = \lambda_\ell(\boldsymbol{\omega})\mathbf{M}_\ell(\boldsymbol{\omega})\mathbf{u}_\ell(\boldsymbol{\omega}), \tag{43}$$

where $\mathbf{u}_\ell(\boldsymbol{\omega})$ is the coefficient vector (with respect to the basis) and $\mathbf{A}_\ell(\boldsymbol{\omega})$, $\mathbf{M}_\ell(\boldsymbol{\omega})$ are the associated FE matrices corresponding to the mesh $\mathcal{T}_\ell := \mathcal{T}_{h_\ell}$. The number of iterations K in the computational cost per sample, as well as the rate of the cost per iteration depend on the choice of the algebraic eigensolver. A variety of solvers can be applied here to solve the generalized eigenvalue problem (43), including power iteration, the QR algorithm, subspace iterations, etc. For our purposes, we only need an eigensolver that is able to compute the smallest eigenvalue, which is real and simple. As such, we consider here two eigenvalue solvers, the *Rayleigh quotient iteration* and the *implicitly restarted Arnoldi method*.

Algorithm 1 The Rayleigh quotient iteration (RQI).

1: Input: $(\mathbf{A}, \mathbf{M}, \boldsymbol{\eta}_0, \boldsymbol{\xi}_0, \lambda_0, \varepsilon, \mathrm{M})$, where $\boldsymbol{\eta}_0, \boldsymbol{\xi}_0, \lambda_0, \varepsilon$ and M are initial left and right eigenvectors, the initial eigenvalue, the error tolerance, and the maximum number of iterations, respectively
2: Set $i \leftarrow 0$
3: **while** $\|\mathbf{A}\boldsymbol{\eta}_i - \lambda\mathbf{M}\boldsymbol{\eta}_i\| > \varepsilon$ and $i \leq M$ **do**
4: Normalize $\boldsymbol{\eta}_i \leftarrow \boldsymbol{\eta}_i \|\boldsymbol{\eta}_i\|_2^{-1}$
5: Normalize $\boldsymbol{\xi}_i \leftarrow \boldsymbol{\xi}_i \|\boldsymbol{\xi}_i\|_2^{-1}$
6: Solve $(\lambda_i\mathbf{M} - \mathbf{A})\boldsymbol{\eta}_{i+1} = \boldsymbol{\eta}_i$
7: Solve $(\lambda_i\mathbf{M} - \mathbf{A})^H\boldsymbol{\xi}_{i+1} = \boldsymbol{\xi}_i$
8: Compute $\lambda_{i+1} \leftarrow (\boldsymbol{\xi}_{i+1}^H\mathbf{A}\boldsymbol{\eta}_{i+1})(\boldsymbol{\xi}_{i+1}^H\mathbf{M}\boldsymbol{\eta}_{i+1})^{-1}$
9: $i \leftarrow i + 1$
10: **end while**
11: Output: $(\boldsymbol{\eta}, \boldsymbol{\xi}, \lambda)$

We first consider the Rayleigh quotient iteration (Algorithm 1), introduced first by Lord Rayleigh in 1894 for a quadratic eigenproblem of oscillations of a mechanical system [57] and then extended in the 1950s and 1960s to non-symmetric generalized eigenproblems [17, 56]. The following lemma, whose proof can be found in Crandall [17] and Ostrowski [56], establishes the error reduction rate of the Rayleigh quotient iteration, which will in turn help to bound the computational cost on each level.

Lemma 1 *Suppose we have an initial guess $\lambda_{\ell,0}(\boldsymbol{\omega})$ to the eigenvalue $\lambda_\ell(\boldsymbol{\omega})$ at the level ℓ and $|\lambda_{\ell,0}(\boldsymbol{\omega}) - \lambda_\ell(\boldsymbol{\omega})|$ is sufficiently small. Then the sequence $\lambda_{\ell,i}(\boldsymbol{\omega})$ converges to $\lambda_\ell(\boldsymbol{\omega})$ quadratically, i.e., there exists a constant $\hat{C}(\boldsymbol{\omega})$ such that*

$$|\lambda_{\ell,i+1}(\boldsymbol{\omega}) - \lambda_\ell(\boldsymbol{\omega})| \leq \hat{C}(\boldsymbol{\omega})|\lambda_{\ell,i}(\boldsymbol{\omega}) - \lambda_\ell(\boldsymbol{\omega})|^2. \tag{44}$$

The computational cost of Rayleigh quotient iteration (RQI) is dominated by the cost of solving two linear systems in each iteration (cf. Lines 6 and 7 of Algorithm 1). For direct solvers, such as LU decomposition, the computational cost depends on the sparsity and bandwidth of the matrices, e.g., for piecewise linear FE applied to (5) and $d = 2$, the cost for solving these linear systems on level ℓ is $O(h_\ell^{-3})$ [26]. However, optimal iterative solvers, such as geometric multigrid methods, are able to achieve the optimal computational

complexity of (or close to) $O(h_\ell^{-d})$. All other steps in Algorithm 1 are linear in the degree of freedoms, and thus $O(h_\ell^{-d})$. Hence, typically the cost per iteration grows with rate $\gamma \geq d$, but it can be as big as $\gamma = 3$ for $d = 2$. The remaining factor in the computational cost is the number of iterations K for the Rayleigh quotient iteration within the MLMC estimator, but this is independent of h_ℓ.

Algorithm 2 Three-grid Rayleigh Quotient iteration (tgRQI).

1: Input: $(\mathbf{A}_\ell, \mathbf{A}_{\ell-1}, \mathbf{A}_0, \mathbf{M}_\ell, \mathbf{M}_{\ell-1}, \mathbf{M}_0, \boldsymbol{\eta}_0, \boldsymbol{\xi}_0, \lambda_0, \ell)$, where $\boldsymbol{\eta}_0', \boldsymbol{\xi}_0', \lambda_0'$ are the initial left and right eigenvectors at level 0, and the initial eigenvalue.
2: $\varepsilon \leftarrow 10^{-10}$, $M \leftarrow 1000$
3: $(\boldsymbol{\eta}_0, \boldsymbol{\xi}_0, \lambda_0) \leftarrow \text{RQI}\,(\mathbf{A}_0, \mathbf{M}_0, \boldsymbol{\eta}_0', \boldsymbol{\xi}_0', \lambda_0', \varepsilon, M)$
4: Interpolate the eigenfunctions from V_0 on $\mathcal{T}_0$ onto $V_{\ell-1}$ on $\mathcal{T}_{\ell-1}$:
$(\boldsymbol{\eta}_{\ell-1}', \boldsymbol{\xi}_{\ell-1}') \leftarrow (\boldsymbol{\eta}_0, \boldsymbol{\xi}_0)$
5: $(\boldsymbol{\eta}_{\ell-1}, \boldsymbol{\xi}_{\ell-1}, \lambda_{\ell-1}) \leftarrow \text{RQI}\,(\mathbf{A}_{\ell-1}, \mathbf{M}_{\ell-1}\boldsymbol{\eta}_{\ell-1}', \boldsymbol{\xi}_{\ell-1}', \lambda_0, \varepsilon, M)$
6: **if** $\ell - 1 = 0$ **then**
7: Output: $\lambda_1 - \lambda_0$
8: **else**
9: Interpolate the eigenfunctions from $V_{\ell-1}$ on $\mathcal{T}_{\ell-1}$ onto V_ℓ on $\mathcal{T}_\ell$: $(\boldsymbol{\eta}_\ell', \boldsymbol{\xi}_\ell') \leftarrow (\boldsymbol{\eta}_{\ell-1}, \boldsymbol{\xi}_{\ell-1})$
10: $(\boldsymbol{\eta}_\ell, \boldsymbol{\xi}_\ell, \lambda_\ell) \leftarrow \text{RQI}\,(\mathbf{A}_\ell, \mathbf{M}_\ell, \boldsymbol{\eta}_\ell', \boldsymbol{\xi}_\ell', \lambda_{\ell-1}, \varepsilon, M)$
11: Output: $\lambda_\ell - \lambda_{\ell-1}$
12: **end if**

Recall the MLMC estimator (28), where at each level ℓ we compute the differences $\lambda_\ell(\boldsymbol{\omega}_n) - \lambda_{\ell-1}(\boldsymbol{\omega}_n)$ for the same sample $\boldsymbol{\omega}_n$. The number of RQI iterations needed for a sufficiently accurate approximation of $\lambda_\ell(\boldsymbol{\omega}_n)$—the more costly level ℓ computation—can be significantly reduced by using the computed approximation of the eigenvalue $\lambda_{\ell-1}(\boldsymbol{\omega}_n)$ on the coarser level as the initial guess, thus also reducing the total computational cost. In fact, we design a three-grid method, similar to the one used in [29] to implement this strategy, which uses the approximate eigenvalue $\lambda_0(\boldsymbol{\omega}_n)$ on level zero with mesh size h_0 as the initial guess for computing eigenvalue $\lambda_{\ell-1}(\boldsymbol{\omega}_n)$ on level $\ell - 1$. Then, $\lambda_{\ell-1}(\boldsymbol{\omega}_n)$ is used as the initial guess for computing $\lambda_\ell(\boldsymbol{\omega}_n)$; see Algorithm 2 for details.

To estimate the computational cost of this three-grid method, we choose again $h_{\ell-1} = h = 2h_\ell$ and denote the exact discrete eigenvalues on level $\ell - 1$ and level ℓ by $\lambda_h(\boldsymbol{\omega}_n)$ and $\lambda_{h/2}(\boldsymbol{\omega}_n)$, respectively. The goal is to control the errors of the eigenvalues $\lambda_{\ell-1}(\boldsymbol{\omega}_n)$ and $\lambda_\ell(\boldsymbol{\omega}_n)$ actually computed using Algorithm 2 to be within the respective discretization errors. Due to the quadratic convergence rate of the RQI (cf. Lemma 1), often only two or three iterations are sufficient to compute a sufficiently accurate approximation $\lambda_0(\boldsymbol{\omega}_n)$ on Level 0 in Line 3 of Algorithm 2. Similarly, in Line 5 of Algorithm 2, two to three iterations of RQI are again sufficient to ensure that the error of the estimated eigenvalue $\lambda_{\ell-1}(\boldsymbol{\omega}_n)$ satisfies

$$|\lambda_{\ell-1}(\boldsymbol{\omega}_n) - \lambda_h(\boldsymbol{\omega}_n)| \leq C_\lambda h_{\ell-1}^2,$$

which is the bound on the discretization error on level $\ell - 1$ in Theorem 2. When $\lambda_{\ell-1}(\boldsymbol{\omega}_n)$ is then used as the initial guess for estimating $\lambda_{h/2}(\boldsymbol{\omega}_n)$, the initial error satisfies

$$|\lambda_{\ell-1}(\boldsymbol{\omega}_n) - \lambda_{h/2}(\boldsymbol{\omega}_n)| \leq |\lambda_{\ell-1}(\boldsymbol{\omega}_n) - \lambda_h(\boldsymbol{\omega}_n)| + |\lambda_h(\boldsymbol{\omega}_n) - \lambda_{h/2}(\boldsymbol{\omega}_n)| \leq \frac{9}{4} C_\lambda h^2,$$

using triangle inequality and Theorem 2 again. Therefore, using Lemma 1 for sufficiently small mesh size h such that $h \leq \frac{2}{9}\big(\hat{C}(\boldsymbol{\omega}_n)C_\lambda\big)^{-1/2}$, one single iteration of RQI on level ℓ suffices such that

$$|\lambda_\ell(\boldsymbol{\omega}_n) - \lambda_{h/2}(\boldsymbol{\omega}_n)| \leq C_\lambda h_\ell^2.$$

In practice, two iterations of RQI are typically used to achieve the target accuracy for $\lambda_\ell(\boldsymbol{\omega}_n)$ in Line 10 of Algorithm 2. These two calls to RQI dominate the computational cost of Algorithm 2 with their four linear solves. Hence, for sparse direct solvers and $d = 2$, the overall computational cost of Algorithm 2 is $O(h_\ell^{-3})$ and $\gamma = 3$ in Theorem 3. The computational complexity of Algorithm 2 can be further reduced using multigrid-based methods to efficiently solve the Rayleigh quotient iterations [11] that potentially offer a rate of $\gamma = d$ (or close to) even in three dimensions. However, it is unclear if the same rate of convergence as for self-adjoint operators can be retained for the convection-dominated problems we are considering here.

Algorithm 3 Multilevel Monte Carlo algorithm.

1: **for** $i = 1 \dots N_0$ **do**
2: Draw a sample $\boldsymbol{\omega}_i$
3: Compute $\lambda_0(\boldsymbol{\omega}_i)$ using either Algorithm 1 or ARPACK
4: **end for**
5: **for** $\ell = 1 \dots L$ **do**
6: **for** $i = 1 \dots N_\ell$ **do**
7: Draw a sample $\boldsymbol{\omega}_i$
8: Compute $\lambda_\ell(\boldsymbol{\omega}_i) - \lambda_{\ell-1}(\boldsymbol{\omega}_i)$ using either Algorithm 2 or ARPACK
9: **end for**
10: **end for**

We also consider the implicitly restarted Arnoldi method [1, 48, 58, 59, 62] and its implementation in the library ARPACK [49] to solve the eigenvalue problem. Compared to the Rayleigh quotient iteration, the Arnoldi method calculates a specified number of eigenpairs that depend on the dimension of the Krylov subspace. The performance of the implicitly restarted Arnoldi method is determined by several factors such as the dimension of the Krylov subspace and the initial vector. To the best of the authors' knowledge, for the eigenvalue problem (12) we are considering here, the convergence rate, and therefore the computational cost, of the implicitly restarted Arnoldi method is not yet known. As such, we numerically estimate the rate variable γ and the computational cost C_ℓ for determining the optimal sample sizes in MLMC. It appears that the number of iterations grows slightly faster than $O(h_\ell^{-1})$ leading to a similar total complexity as RQI for $d = 2$ of $\gamma \approx 3.5$.

4 Extensions of MLMC Method

In this section, we introduce two extensions of the MLMC method for convection–diffusion eigenvalue problems. First, we employ a homotopy method to add stability to the eigensolve for each sample. Second, we replace the Monte Carlo approximation of the expected value on each level in (27) with a quasi-Monte Carlo (QMC) method, which, due to the faster convergence of QMC, allows us to use less samples on each level and improves the overall complexity.

4.1 Homotopy Multilevel Monte Carlo Method

In Carstensen et al. [13], a homotopy method is employed to solve convection–diffusion eigenvalue problems with deterministic coefficients, using the homotopy method to derive adaptation strategies for FE methods. The authors also provided estimates on the convergence rate of the smallest eigenvalue with respect to the homotopy parameter. We aim to investigate the application of this homotopy method in the MLMC method, particularly in designing multilevel models for alleviating numerical instability (due to the high advection velocity) on coarser meshes.

For eigenvalue problems, the homotopy method [50] uses an initial operator $\mathcal{L}_0$—for which the target eigenvalue is easier to compute than that of the original operator $\mathcal{L}$—to form a continuation

$$\mathcal{L}_t = (1 - f(t))\mathcal{L}_0 + f(t)\mathcal{L} \quad \text{for} \quad 0 \le t \le 1, \tag{45}$$

with a function $f : [0; 1] \to [0; 1]$ and $f(0) = 0$, $f(1) = 1$. For the convection–diffusion operator in (1), it is natural to set the diffusion operator as the initial operator. Here we consider a simple linear function $f(t) = t$ to design the sequence of operators used for the homotopy. Given a sequence of homotopy parameters, $0 = t_0 < t_1 < \cdots < t_L = 1$, the homotopy operators with stochastic coefficients define a sequence of eigenvalue problems of the form

$$\begin{aligned} \mathcal{H}(\boldsymbol{\omega}, t_\ell)u(\boldsymbol{\omega}, t_\ell) &= -\nabla \cdot \big(\kappa(\boldsymbol{\omega})\nabla u(\boldsymbol{\omega}, t_\ell)\big) + t_\ell\big(\mathbf{a}(\boldsymbol{\omega}) \cdot \nabla u(\boldsymbol{\omega}, t_\ell)\big) \\ &= \lambda(\boldsymbol{\omega}, t_\ell)u(\boldsymbol{\omega}, t_\ell), \end{aligned} \tag{46}$$

for $\ell = 0, \ldots, L$. The following lemma [13, Lemma 4.1] establishes the homotopy error on the smallest eigenvalue in (46) for fixed $\boldsymbol{\omega}$.

Lemma 2 *Suppose the velocity field* $\mathbf{a}$ *is divergence-free and* $\boldsymbol{\omega}$ *is fixed. The homotopy error—which is defined as the difference between the smallest eigenvalue* $\lambda(\boldsymbol{\omega}, t = 1)$ *of the original operator and that of the homotopy operator in* (46) *satisfies for any* $t \in [0, 1]$

$$|\lambda(\boldsymbol{\omega}, 1) - \lambda(\boldsymbol{\omega}, t)| \le C_{t,\boldsymbol{\omega}}(1 - t), \tag{47}$$

where

$$C_{t,\boldsymbol{\omega}} := \frac{\|\mathbf{a}(\cdot, \boldsymbol{\omega})\|_{L^\infty}\big(\|u(\boldsymbol{\omega}, 1)\|_V + \|u^*(\boldsymbol{\omega}, 1)\|_V\big)}{\langle u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t)\rangle + \langle u(\boldsymbol{\omega}, t), u^*(\boldsymbol{\omega}, 1)\rangle}, \tag{48}$$

and $u^*(\boldsymbol{\omega}, t)$ *is the dual homotopy solution. For* t *sufficiently close to 1 and almost all* $\boldsymbol{\omega} \in \Omega$, $C_{t,\boldsymbol{\omega}} < C_t$ *for some* $C_t < \infty$ *independent of* $\boldsymbol{\omega}$.

Proof First, the primal and dual homotopy eigenvalue problems are

$$\begin{aligned} &\mathcal{A}(\boldsymbol{\omega}; u(\boldsymbol{\omega}, t), v) + t\mathcal{B}(u(\boldsymbol{\omega}, t), v) = \lambda(\boldsymbol{\omega}, t)\langle u(\boldsymbol{\omega}, t), v\rangle && \text{for all } v \in V, \\ &\mathcal{A}(\boldsymbol{\omega}; w, u^*(\boldsymbol{\omega}, t)) + t\mathcal{B}(w, u^*(\boldsymbol{\omega}, t)) = \overline{\lambda^*(\boldsymbol{\omega}, t)}\langle w, u^*(\boldsymbol{\omega}, t)\rangle && \text{for all } w \in V, \end{aligned}$$

where we again normalise the homotopy eigenfunctions such that $\|u(\boldsymbol{\omega}, t)\|_{L^2} = 1 = \|u^*(\boldsymbol{\omega}, t)\|_{L^2}$.

Following the proof of [13, Lemma 4.1], using the homotopy eigenvalue problems we can write the homotopy error as

$$\big[\lambda(\boldsymbol{\omega}, 1) - \lambda(\boldsymbol{\omega}, t)\big]\big[\langle u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t)\rangle + \langle u(\boldsymbol{\omega}, t), u^*(\boldsymbol{\omega}, 1)\rangle\big]$$

$$
\begin{aligned}
&= \lambda(\boldsymbol{\omega}, 1)\langle u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t)\rangle + \overline{\lambda^*(\boldsymbol{\omega}, 1)}\langle u(\boldsymbol{\omega}, t), u^*(\boldsymbol{\omega}, 1)\rangle \\
&\quad - \overline{\lambda^*(\boldsymbol{\omega}, t)}\langle u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t)\rangle - \lambda(\boldsymbol{\omega}, t)\langle u(\boldsymbol{\omega}, t), u^*(\boldsymbol{\omega}, 1)\rangle \\
&= (1-t)\big[\mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t)) + \mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}, t), u^*(\boldsymbol{\omega}, 1))\big], \qquad (49)
\end{aligned}
$$

where we have also used the property $\lambda(\boldsymbol{\omega}, t) = \overline{\lambda^*(\boldsymbol{\omega}, t)}$.

Since $\mathbf{a}(\boldsymbol{\omega})$ is divergence free, we have

$$
\mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}, t), u^*(\boldsymbol{\omega}, 1)) = -\mathcal{B}(\boldsymbol{\omega}; \overline{u^*(\boldsymbol{\omega}, 1)}, \overline{u(\boldsymbol{\omega}, t)}).
$$

Then by the triangle inequality, followed by the Cauchy–Schwarz inequality

$$
\begin{aligned}
&\mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t)) + \mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}, t), u^*(\boldsymbol{\omega}, 1)) \\
&= \mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t)) - \mathcal{B}(\boldsymbol{\omega}; \overline{u^*(\boldsymbol{\omega}, 1)}, \overline{u(\boldsymbol{\omega}, t)}) \\
&\le |\mathcal{B}(\boldsymbol{\omega}; u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t))| + |\mathcal{B}(\boldsymbol{\omega}; \overline{u^*(\boldsymbol{\omega}, 1)}, \overline{u(\boldsymbol{\omega}, t)})| \\
&\le \|\mathbf{a}(\boldsymbol{\omega})\|_{L^\infty} \|\nabla u(\boldsymbol{\omega}, 1)\|_{L^2} \|u^*(\boldsymbol{\omega}, t)\|_{L^2} + \|\mathbf{a}(\boldsymbol{\omega})\|_{L^\infty} \|\nabla u^*(\boldsymbol{\omega}, 1)\|_{L^2} \|u(\boldsymbol{\omega}, t)\|_{L^2} \\
&= \mathbf{a}_{\max}\big(\|u(\boldsymbol{\omega}, 1)\|_V + \|u^*(\boldsymbol{\omega}, 1)\|_V\big), \qquad (50)
\end{aligned}
$$

where we have used the property that the homotopy eigenfunctions are normalized and Assumption 3. Substituting (50) into (49) then rearranging gives the result (47) with $C_{t,\boldsymbol{\omega}}$ as in (48).

Next, we bound $C_{t,\boldsymbol{\omega}}$ independently of $\boldsymbol{\omega}$. Clearly, the numerator is bounded for all t and almost all $\boldsymbol{\omega}$. Next, we show that the denominator is strictly positive. Suppose for a contradiction that $\langle u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t)\rangle = 0$, then this implies that

$$
\langle u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, 1) - u^*(\boldsymbol{\omega}, t)\rangle = \langle u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, 1)\rangle > 0,
$$

since the eigenfunction and dual eigenfunction are not orthogonal if the corresponding eigenvalues satisfy $\lambda(\boldsymbol{\omega}, 1) = \overline{\lambda^*(\boldsymbol{\omega}, 1)}$. However, since $u^*(\boldsymbol{\omega}, t) \to u^*(\boldsymbol{\omega}, 1)$ as $t \to 1$, the left hand side tends to zero whereas the right hand side is strictly positive and independent of t, leading to a contradiction. Hence, for t sufficiently small $\langle u(\boldsymbol{\omega}, 1), u^*(\boldsymbol{\omega}, t)\rangle > 0$ and similarly $\langle u(\boldsymbol{\omega}, t), u^*(\boldsymbol{\omega}, 1)\rangle > 0$. Thus, for t sufficiently small $C_{t,\boldsymbol{\omega}} < \infty$. Since $\mathbf{a}(\boldsymbol{\omega})$ along with the primal and dual eigenfunctions are continuous in $\boldsymbol{\omega}$, it follows that $C_{t,\boldsymbol{\omega}}$ is also continuous in $\boldsymbol{\omega}$ and thus, can be bounded by the maximum over the compact domain Ω,

$$
C_{t,\boldsymbol{\omega}} \le \max_{\boldsymbol{\omega} \in \Omega} C_{t,\boldsymbol{\omega}} =: C_t < \infty.
$$

□

With the homotopy method, the approximation error now comes from three sources: the FE discretization, the iterative eigensolver, and the value of the homotopy parameter. We suppose again that the error due to the eigensolver is bounded from above by the other two sources of error and design multilevel sequences such that the homotopy error and the discretization error are non-increasing with increasing level. Denoting the homotopy parameter and the mesh size at level ℓ by t_ℓ and h_ℓ, respectively, the multilevel sequence

$$
\{(t_0, h_0), (t_1, h_1), \ldots, (t_L, h_L)\},
$$

is designed such that $t_{\ell-1} \le t_\ell$, $h_{\ell-1} \ge h_\ell$, and $t_L = 1$. The multilevel parameters are required to be non-repetitive, i.e., $(t_{\ell-1}, h_{\ell-1}) \ne (t_\ell, h_\ell)$ for all $\ell = 1, \ldots, L$, to ensure an asymptotically decreasing total approximation error in the sequence. However, one of these two parameters is allowed to be the same on two adjacent levels, i.e., either $h_{\ell-1} =$

h_ℓ or $t_{\ell-1} = t_\ell$ is possible. This setting allows for adapting the homotopy parameter to discretisations on different meshes to satisfy the stability condition of the FE approximation.

The resulting MLMC estimator can be derived from the telescoping sum

$$\mathbb{E}[\lambda(\boldsymbol{\omega})] = \mathbb{E}[\lambda_{h_0}(\boldsymbol{\omega}, t_0)] + \sum_{i=1}^{L} \mathbb{E}[\lambda_{h_i}(\boldsymbol{\omega}, t_i) - \lambda_{h_i-1}(\boldsymbol{\omega}, t_{i-1})].$$

Following a similar derivation as that of Corollary 1 and based on the error bound in Lemma 2, we conjecture that the expectation and the variance of the multilevel difference with the homotopy method are bounded by

$$\begin{aligned} |\mathbb{E}[\lambda_{h_\ell}(\boldsymbol{\omega}, t_\ell) - \lambda_{h_{\ell-1}}(\boldsymbol{\omega}, t_{\ell-1})]| &\leq c_1 h_{\ell-1}^2 + c_2(1 - t_{\ell-1}), \\ \mathrm{var}[\lambda_{h_\ell}(\boldsymbol{\omega}, t_\ell) - \lambda_{h_{\ell-1}}(\boldsymbol{\omega}, t_{\ell-1})] &\leq c_3 h_{\ell-1}^4 + c_4(1 - t_{\ell-1})^2, \end{aligned} \tag{51}$$

respectively. This will be used as the guideline for choosing the multilevel sequences in our numerical experiments. We will also demonstrate that the above conjecture is valid in our numerical experiments.

4.2 Multilevel QMC Methods

QMC methods are a class of equal-weight quadrature rules originally designed to approximate high-dimensional integrals on the unit hypercube. A QMC approximation of the expected value of f is given by

$$\mathbb{E}[f] = \int_{[0,1]^s} f(\boldsymbol{\omega})\, \mathrm{d}\boldsymbol{\omega} \approx \frac{1}{N} \sum_{k=1}^{N-1} f(\boldsymbol{\tau}_k), \tag{52}$$

where, in contrast to Monte Carlo methods, the quadrature points $\{\boldsymbol{\tau}_k\}_{k=1}^{N-1} \subset [0,1]^s$ are chosen deterministically to be well-distributed and have good approximation properties in high dimensions. There are several types of QMC methods, including lattice rules, digital nets and randomised rules. The main benefit of QMC methods is that for sufficiently smooth integrands the quadrature error converges at a rate of $\mathcal{O}(N^{-1+\delta})$, $\delta > 0$, or faster, which is better than the Monte Carlo convergence rate of $\mathcal{O}(N^{-1/2})$. For further details see, e.g., [20, 21].

In this paper, we consider randomly shifted lattice rules, which are generated by a single integer vector $\mathbf{z} \in \mathbb{N}^s$ and a single random shift $\boldsymbol{\Delta} \sim \mathrm{Uni}[0,1]^s$. The points are given by

$$\boldsymbol{\tau}_k = \left\{ \frac{k\mathbf{z}}{N} + \boldsymbol{\Delta} \right\} \quad \text{for } k = 0, 1, \ldots, N-1, \tag{53}$$

where $\{\cdot\}$ denotes taking the fractional part of each component. The benefits of random shifting are that the resulting approximation (52) is unbiased and that performing multiple QMC with i.i.d. random shifts provides a practical estimate for the mean-square error using the sample variance of the multiple approximations.

If f is sufficiently smooth (i.e., has square-integrable mixed first derivatives) then a generating vector can be constructed such that the mean-square error (MSE) of a randomly shifted lattice rule approximation satisfies

$$\mathbb{E}\left[\left| \int_{[0,1]^s} f(\boldsymbol{\omega})\, \mathrm{d}\boldsymbol{\omega} - \frac{1}{N} \sum_{k=0}^{N-1} f(\boldsymbol{\tau}_k) \right|^2 \right] \lesssim N^{-1/\eta} \quad \text{for } \eta \in (\tfrac{1}{2}, 1], \tag{54}$$

see, e.g., Theorem 5.10 in [20]. I.e., for $\eta \approx 1/2$ the convergence of the MSE is close to $1/N^2$.

Starting again with the telescoping sum (27), a multilevel QMC (MLQMC) method approximates the expectation of the smallest eigenvalue by using a QMC rule to compute the expectation on each level. MLQMC methods were first introduced in [32] for SDEs, then applied to parametric PDEs in [46, 47] and elliptic eigenvalue problems in [28, 29]. For $L \in \mathbb{N}$ and $\{N_\ell\}_{\ell=0}^L$, the MLQMC approximation is given by

$$Y^{\mathrm{MLQMC}} := \sum_{\ell=0}^{L} Y_\ell^{\mathrm{QMC}}, \quad Y_\ell^{\mathrm{QMC}} := \frac{1}{N_\ell} \sum_{\ell=0}^{N_\ell - 1} \left[\lambda_\ell(\boldsymbol{\tau}_{\ell,k}) - \lambda_{\ell-1}(\boldsymbol{\tau}_{\ell,k}) \right], \tag{55}$$

where we apply a different QMC rule with points $\{\boldsymbol{\tau}_{\ell,k}\}_{k=0}^{N_\ell-1}$ on each level, e.g., an N_ℓ-point randomly shifted lattice rule (53) generated by $\mathbf{z}_\ell$ and an i.i.d. $\boldsymbol{\Delta}_\ell$.

The faster convergence of QMC rules leads to an improved complexity of MLQMC methods compared to MLMC, where in the best case the cost is reduced to close to ε^{-1} for a MSE of ε^2. Following [46], under the same assumptions as in Theorem 3, but with Assumption *II* replaced by

II(b) $\mathrm{MSE}[Y_\ell^{\mathrm{QMC}}] = O(N_\ell^{-1/\eta} h_\ell^\beta)$ with $\eta \in (\frac{1}{2}, 1]$,

the MSE of the MLQMC estimator (55) is bounded above by ε^2 and the cost satisfies

$$C_{\mathrm{MLQMC}}(\varepsilon) \lesssim \begin{cases} \varepsilon^{-2\eta} & \text{if } \beta\eta > \gamma, \\ \varepsilon^{-2\eta} \log_2(\varepsilon^{-1})^{\eta+1} & \text{if } \beta\eta = \gamma, \\ \varepsilon^{-2\eta-(\gamma-\beta\eta)/\alpha} & \text{if } \beta\eta < \gamma. \end{cases}$$

The maximum level L is again given by (32) and $\{N_\ell\}$ are given by

$$N_\ell = \left\lceil N_0 \left(\frac{h_\ell^\beta}{C_\ell} \right)^{\eta/(\eta+1)} C_0 \right\rceil^{1/(\eta+1)} \Bigg)^\eta \Bigg\rceil, \tag{56}$$

where C_ℓ is the cost per sample as in assumption *III* in Theorem 3 and N_0 is chosen as

$$N_0 \simeq \varepsilon^{-2\eta} \left(\sum_{\ell=0}^{L} \left(h_\ell^{\beta\eta} C_\ell \right)^{1/(\eta+1)} \right)^\eta.$$

Verifying Assumption *II(b)* for the convection–diffusion EVP (1) requires performing a technical analysis similar to [28] and in particular, requires bounding the derivatives of the eigenvalue $\lambda(\boldsymbol{\omega})$ and its eigenfunction $u(\boldsymbol{\omega})$ with respect to $\boldsymbol{\omega}$. Such analysis is left for future work. In the numerical results, section we study the convergence of QMC and observe that *II(b)* holds with $\eta \approx 0.61$.

In practice, one should perform multiple, say $R \in \mathbb{N}_0$, QMC approximations corresponding to i.i.d. random shifts, then take the average as the final estimate. In this way, we can also estimate the MSE by the sample variance over the different realisations.

5 Numerical Results

In this section, we present numerical results for three test cases. The quantity of interest in all cases is the smallest eigenvalue of the stochastic convection–diffusion problem (1) in the unit domain $D = [0, 1]^2$. The first two test cases use constant convection velocities at different

magnitudes to benchmark the performance of eigenvalue solvers and finite element discretisation methods in the multilevel setting. In these two test cases, the random conductivity $\kappa(\mathbf{x};\boldsymbol{\omega})$ is modelled as a log-uniform random field constructed through the convolution of s_κ i.i.d. uniform random variables

$$\log\kappa(\mathbf{x};\boldsymbol{\omega}) = \sum_{i=1}^{s_\kappa} \omega_i k(\mathbf{x}-\mathbf{c}_i),$$

with exponential kernels $k(\mathbf{x}-\mathbf{c}_i) = \exp[-\frac{25}{2}\|\mathbf{x}-\mathbf{c}_i\|_2]$, where $\mathbf{c}_i$ are the kernel centers placed uniformly on a 5×5 grid in the domain D. In the third test case, we also make the convection velocity a random field. Specifically, we first construct a log-uniform random field

$$S(\boldsymbol{\omega},\mathbf{x}) = \exp\left[\sum_{i=1}^{s_a} \omega_{i+s_\kappa} k(\mathbf{x}-\mathbf{c}_i)\right], \tag{57}$$

similar to that of the conductivity field using additional s_a i.i.d. uniform random variables. Then, a divergence-free velocity field can be obtained by

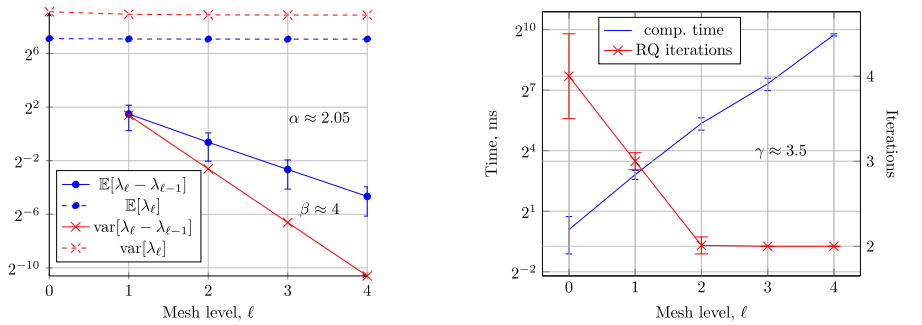
$$\mathbf{a}(\boldsymbol{\omega}) = \left[\frac{\partial S(\boldsymbol{\omega},\mathbf{x})}{\partial x_2}, -\frac{\partial S(\boldsymbol{\omega},\mathbf{x})}{\partial x_1}\right]^\top. \tag{58}$$

We employ the Eigen [34] library for Rayleigh quotient iteration and solve the linear systems using sparse LU decomposition with permutation from the SuiteSparse [18] library. For the implicitly restarted Arnoldi method, we use the ARPACK [49] library with the `SM` mode for finding the smallest eigenvalue. Random variables are generated using the standard C++ library and the pseudo-random seeds are the same across all experiments.

Numerical experiments are organized as follows. For a relatively low convection velocity $\mathbf{a} = [20; 0]^T$, we demonstrate the multilevel Monte Carlo (MLMC) method using the Galerkin FEM discretization. In this case, we also consider applying the homotopy method together with a geometrically refined mesh hierarchy. Then, on a test case with relatively high convection velocity $\mathbf{a} = [50; 0]^T$, we demonstrate the extra efficiency gain offered by the numerically more stable SUPG method, compared with the Galerkin discretization. For the third test case with a random velocity field, we apply SUPG to demonstrate the efficacy and efficiency of our multilevel method. Here we also demonstrate that quasi-Monte Carlo (QMC) samples can be used to replace Monte Carlo samples to further enhance the efficiency of multilevel methods.For all multilevel methods, we consider a sequence of geometrically refined meshes with $h_\ell = h_0 \times 2^{-\ell}$, $\ell = 0, 1, \ldots, 4$, and $h_0 = 2^{-3}$. At the finest level, this gives 16129 degrees of freedom in the discretised linear system. We use 10^4 samples on each level ℓ to compute the estimates of rate variables α, β, γ in the MLMC complexity theorem (cf. Theorem 3).

5.1 Test Case I

In the first experiment, we set $\mathbf{a} = [20; 0]^T$ and use the Galerkin FEM to discretize the convection–diffusion equation. The stopping criteria for the Rayleigh quotient iteration and for the implicitly restarted Arnoldi method are set to be 10^{-12}. In addition, for the implicitly restarted Arnoldi method, the Krylov subspace dimensions (the `ncv` values of ARPACK) are chosen empirically for each mesh size to optimize the number of Arnoldi iterations. They are $m = 20, 40, 70, 70, 100$ for $h = 2^{-3}, 2^{-4}, 2^{-5}, 2^{-6}, 2^{-7}$, respectively.



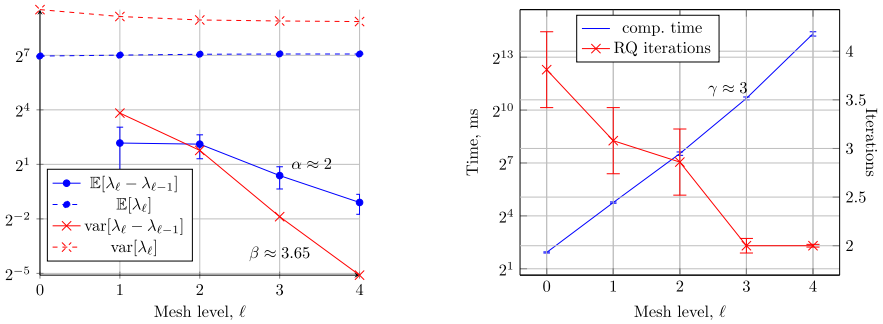
(a) Means and variances of λ_ℓ and $\lambda_\ell - \lambda_{\ell-1}$. (b) Computational time and average RQI.

Fig. 2 MLMC method using tgRQI for Test Case I with $\mathbf{a} = [20; 0]^T$ and Galerkin FEM: **a** mean (blue) and variance (red) of the eigenvalue λ_ℓ (dashed) and of $\lambda_\ell - \lambda_{\ell-1}$ (solid); **b** computational times for one multilevel difference (blue) and average number of Rayleigh quotient iterations (red) on each level. Where shown, the error bars represent $\pm$ one standard deviation (Color figure online)

We demonstrate the efficiency of four variants of the MLMC method: (i) the three-grid Rayleigh quotient iteration (tgRQI) with a model sequence defined by grid refinement; (ii) tgRQI with a model sequence defined by grid refinement and homotopy; (iii) the implicitly restarted Arnoldi method (IRAr) with a model sequence defined by grid refinement; and (iv) IRAr with a model sequence defined by grid refinement and homotopy.

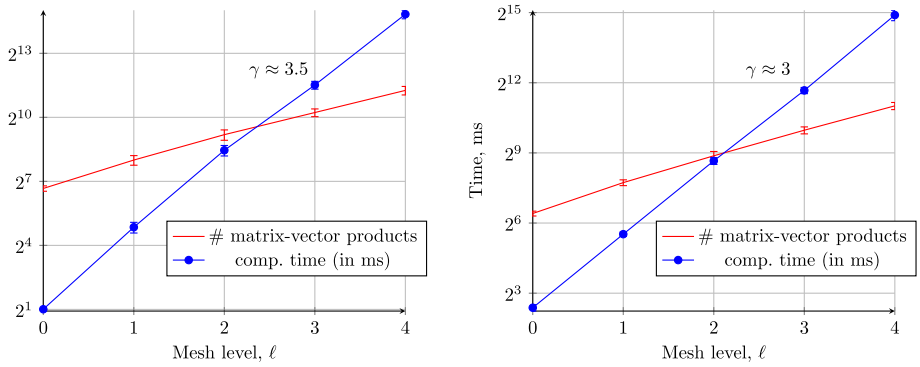
(i) MLMC with tgRQI: Fig. 2 illustrates the mean, the variance and the computational cost of multilevel differences $\lambda_\ell(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})$ of the smallest eigenvalue using tgRQI as the eigenvalue solver (without homotopy). Figure 2a also shows Monte Carlo estimates of the expected mean and variance of the smallest eigenvalue $\lambda_\ell(\boldsymbol{\omega})$ for each of the discretization levels. In addition to the computational cost, Fig. 2b also shows the number of Rayleigh quotient iterations used at each level. We observe that the average number of iterations follows our analysis of the computational cost of tgRQI (cf. Algorithm 2). From these plots, we estimate that the rate variables in the MLMC complexity theorem are $\alpha \approx 2.0$, $\beta \approx 4.0$ and $\gamma \approx 2.41$. Since the variance reduction rate β is larger than the cost increase rate γ, the MLMC estimator is in the best case scenario, with $O(\varepsilon^{-2})$ complexity, as stated in Theorem 3.

(ii) MLMC with homotopy and tgRQI: Next, we consider the homotopy method in the MLMC setting together with tgRQI. We use the conjecture in (51) to set the homotopy parameters such that $1 - t_\ell = O(h_\ell^2)$, $t_0 = 0$ and $t_L = 1$. For $L = 5$, this results in $t_\ell = \{0, 3/4, 15/16, 63/64, 1\}$. With this choice the eigenproblem on the zeroth level contains no convection term and is thus self-adjoint. Figure 3a shows again the means and the variances of the multilevel differences $\lambda_\ell - \lambda_{\ell-1}$ in this setting, together with MC estimates of the expected means and variances of the eigenvalues for each level. The hierarchy of homotopy parameters is chosen to guarantee good variance reduction for MLMC. Indeed, the variance of the multilevel difference decays smoothly with a rate $\beta \approx 3.65$. The expected mean of the difference, on the other hand, stagnates between $\ell = 1$ and $\ell = 2$. However, this initial stagnation is irrelevant for the MLMC complexity theorem; eventually for $\ell \geq 2$, the estimated means of the multilevel differences decrease again with a rate of $\alpha \approx 2$. Figure 3b shows the number of Rayleigh quotient iterations used at each level and the computational cost, which grows with a rate of $\gamma \approx 2.56$ here. This leads to the same asymptotic complexity for MLMC, since the regime is the same, i.e., $\beta > \gamma$, which is the optimal regime in Theorem 3 with a complexity of $O(\varepsilon^{-2})$.



(a) Means and variances of λ_ℓ and $\lambda_\ell - \lambda_{\ell-1}$. (b) Computational times and average RQI.

Fig. 3 MLMC method using homotopy and tgRQI for Test Case I with $\mathbf{a} = [20; 0]^T$ and Galerkin FEM: **a** mean (blue) and variance (red) of the eigenvalue λ_ℓ (dashed) and of $\lambda_\ell - \lambda_{\ell-1}$ (solid); **b** computational times for one multilevel difference (blue) and average number of RQIs (red) on each level. Where shown, the error bars represent ± one standard deviation (Color figure online)



(a) Without homotopy. (b) With homotopy.

Fig. 4 MLMC method using IRAr for Test Case I with $\mathbf{a} = [20; 0]^T$ and Galerkin FEM, both without (**a**) and with (**b**) homotopy: average computational cost (blue) and average number of matrix–vector products (red) per sample of $\lambda_\ell - \lambda_{\ell-1}$. The error bars represent ± one standard deviation (Color figure online)

(iii) MLMC with IRAr: Similar results are obtained by using the implicitly restarted Arnoldi eigenvalue solver (without homotopy). Since the mean and the variance of the multilevel differences in this setting are almost identical to those of the Rayleigh quotient solver, we omit the plots here and only report the computational cost. Figure 4a shows the average number of matrix–vector products and the estimated CPU time for computing each of the multilevel differences, which grows with a rate of $\gamma \approx 3.5$. Here, the increasing dimension of Krylov subspaces with grid refinement likely causes the higher growth rate of computational time compared to the experiment using tgRQI. Nonetheless, the MLMC estimator has again the optimal $O(\varepsilon^{-2})$ complexity.

(iv) MLMC with homotopy and IRAr: Finally, we consider the behaviour of IRAr with homotopy, using the same sequence for the homotopy parameter t_ℓ as in (ii). Again, we only focus on computational cost, showing the average number of matrix–vector products and the CPU time for computing each of the multilevel differences in Fig. 4b. As in (ii), the cost grows at a rate of $\gamma \approx 3$ leading again to the optimal $O(\varepsilon^{-2})$ complexity for MLMC.

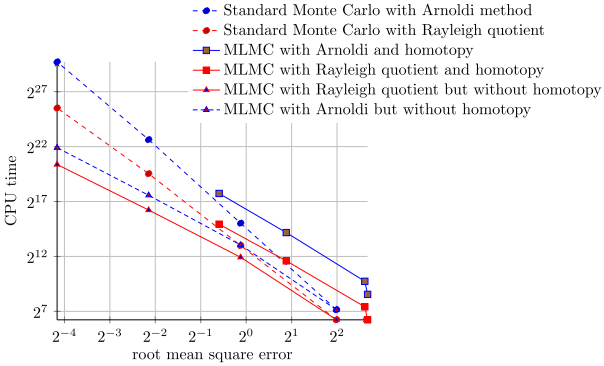


Fig. 5 CPU time versus root mean square error of all estimators in Test Case I

5.1.1 Overall Comparison

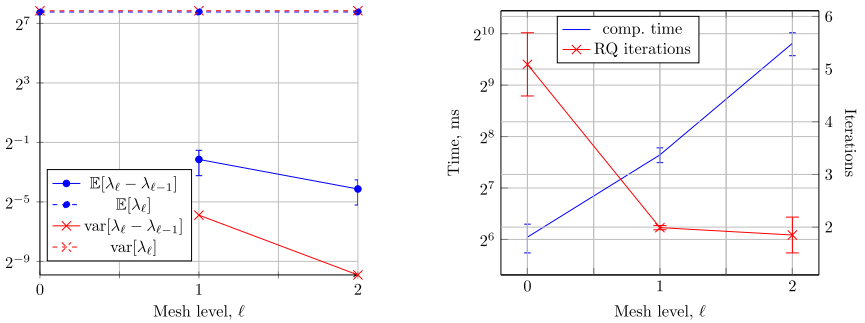
In Fig. 5, we show the CPU time versus the root mean square error for all four presented MLMC estimators together, as well as for standard Monte Carlo estimators using tgRQI (red) and IRAr (blue). The estimated complexity of standard Monte Carlo methods are $O(\varepsilon^{-2.92})$ and $O(\varepsilon^{-3.35})$ for tgRQI and IRAr, respectively. Overall, MLMC using tgRQI (without homotopy) outperforms all other methods, despite that all four MLMC methods achieve the optimal $O(\varepsilon^{-2})$ complexity.

5.2 Test Case II

For the second experiment, we increase the velocity to $\mathbf{a} = [50; 0]^T$ and focus on the comparison between Galerkin and SUPG discretizations. Thus, we only consider the three-grid Rayleigh quotient iteration (tgRQI) with a multilevel sequence based on geometrically refined grids without homotopy. Note that for such a strong convection, five steps in the homotopy approach are insufficient: the eigenvalues for consecutive homotopy parameters are too different to achieve variance reduction in the homotopy-based MLMC method. Its computational complexity is almost the same as the complexity of standard Monte Carlo, namely almost $O(\varepsilon^{-3.5})$. The performance of MLMC with implicitly restarted Arnoldi on the other hand is similar to MLMC with tgRQI.

5.2.1 Galerkin

Due to the higher convection velocity the first two levels are unstable for most of the realizations of $\boldsymbol{\omega}$ as the FEM solution may exhibit non-physical oscillations. Thus, we set the coarsest level for the MLMC method to $h_0 = 2^{-5}$ here. Keeping the same finest grid level $h_L = 2^{-7}$, this means that we only use a total of three levels ($L = 2$) compared to the sequence in Test Case I, which had a total of five levels ($L = 4$). Figure 6a shows the expectation and variance of the multilevel differences. Here, we only have a couple of data points for estimating the rate variables of the MLMC complexity theorem, but the estimates are $\alpha \approx 2$ and $\beta \approx 4$ as expected theoretically. The average number of Rayleigh quotient iterations in Fig. 6b also behaves as in Test Case I with 5 iterations on the coarsest level and 2 iterations on the subsequent levels as expected for the three-grid Rayleigh quotient iteration (Algorithm 2)—recall



(a) Means and variances of λ_ℓ and $\lambda_\ell - \lambda_{\ell-1}$. (b) Computational times and average RQI.

Fig. 6 MLMC method using tgRQI for Test Case II with $\mathbf{a} = [50; 0]^T$ and Galerkin FEM: **a** mean (blue) and variance (red) of the eigenvalue λ_ℓ (dashed) and of $\lambda_\ell - \lambda_{\ell-1}$ (solid); **b** computational time for one multilevel difference (blue) and average number of Rayleigh quotient iterations (red) on each level. Where shown, the error bars represent $\pm$ one standard deviation (Color figure online)

that Levels 1 and 2 here correspond to Levels 3 and 4 in Figs. 2b and 3b. The estimated value for $\gamma \approx 1.88$, and thus the MLMC complexity is still $O(\varepsilon^{-2})$. However, we cannot use as many levels due the numerical stability issues caused by the higher convection velocity, which substantially increases the prefactor in the $O(\varepsilon^{-2})$ cost of the algorithm.

5.2.2 SUPG

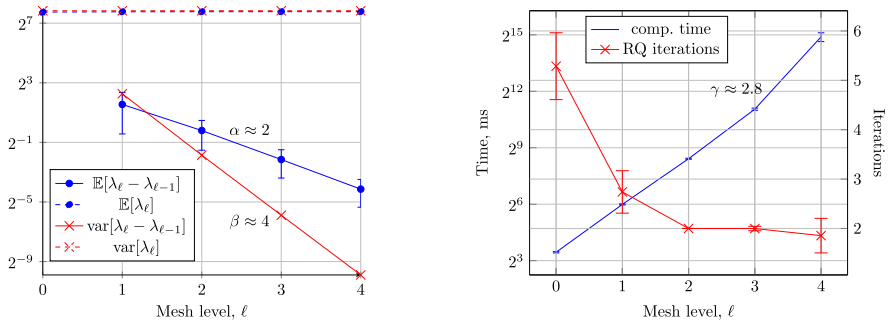
By using the SUPG discretization, we overcome the numerical stability issue and can use all five levels in MLMC, starting with $h_0 = 2^{-3}$. As can be seen in Fig. 7a, the expectation and the variance of the multilevel differences converge with the same rates as for the Galerkin FEM, namely $\alpha \approx 2$ and $\beta \approx 4$ respectively. Also, clearly the use of SUPG leads to stable estimates even on the coarser levels. Figure 7b reports the average number of Rayleigh quotient iterations used at each level and the computational cost. We estimate that the computational cost increases at a rate of $\gamma \approx 2.33$ here. In any case, the use of SUPG in the MLMC also results in the optimal $O(\varepsilon^{-2})$ complexity.

5.2.3 Overall Comparison

Figure 8 shows CPU times versus root mean square errors for the MLMC methods (with tgRQI and without homotopy) using Galerkin FEM and SUPG discretizations. They are compared to a standard Monte Carlo method with Galerkin FEM. Although both MLMC estimates have the optimal $O(\varepsilon^{-2})$ complexity, the stability offered by SUPG enables us to use more, coarser levels, thus leading to a smaller prefactor and a significant computational gain of a factor 10–20 over the Galerkin FEM based method.

5.3 Test Case III

In this experiment, the convection velocity becomes a divergence-free random field generated using (57) and (58). We discretise the eigenvalue problem using SUPG and apply the three-grid Rayleigh quotient iteration (tgRQI) without homotopy to solve multilevel eigenvalue problems. The stopping criteria for tgRQI is set to be 10^{-12}. The same sequence of grid



(a) Means and variances of λ_ℓ and $\lambda_\ell - \lambda_{\ell-1}$. (b) Computational times and average RQI.

Fig. 7 MLMC method using tgRQI for Test Case I with $\mathbf{a} = [20; 0]^T$ and SUPG discretization: **a** mean (blue) and variance (red) of the eigenvalue λ_ℓ (dashed) and of $\lambda_\ell - \lambda_{\ell-1}$ (solid); **b** computational time for one multilevel difference (blue) and average number of Rayleigh quotient iterations (red) on each level. Where shown, the error bars represent $\pm$ one standard deviation (Color figure online)

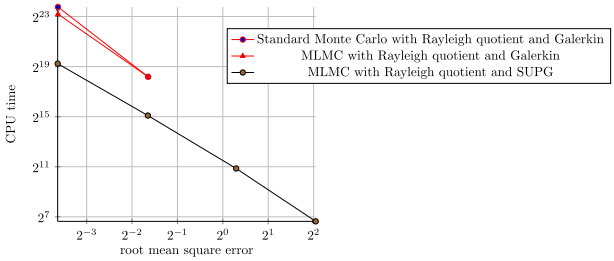
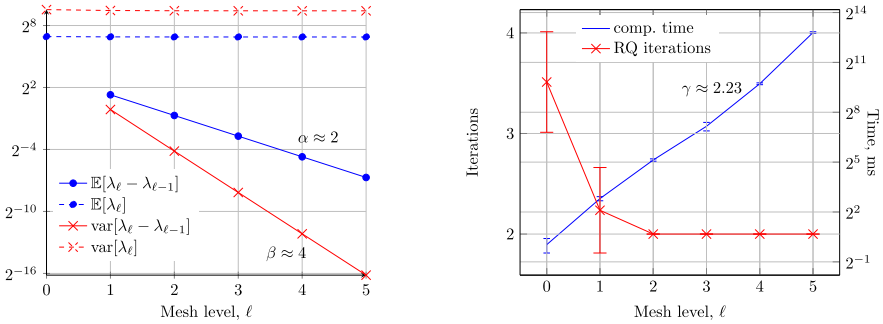


Fig. 8 CPU time versus root mean square error of the estimators in Test Case II

refinements, $h = 2^{-3}, 2^{-4}, 2^{-5}, 2^{-6}, 2^{-7}$, as in previous test cases is used to construct multilevel estimators.

5.3.1 MLMC

Figure 9 illustrates the mean, the variance and the computational cost of multilevel differences $\lambda_\ell(\boldsymbol{\omega}) - \lambda_{\ell-1}(\boldsymbol{\omega})$ of the smallest eigenvalue using tgRQI as the eigenvalue solver. Figure 9a also shows Monte Carlo estimates of the expected mean and variance of the smallest eigenvalue $\lambda_\ell(\boldsymbol{\omega})$ for each of the discretization levels. In addition to the computational cost, Fig. 9b also shows the number of Rayleigh quotient iterations used at each level. We observe that the average number of iterations follows our analysis of the computational cost of tgRQI (cf. Algorithm 2). From these plots, we estimate that the rate variables in the MLMC complexity theorem are $\alpha \approx 2.0$, $\beta \approx 4$ and $\gamma \approx 2.23$. Since the variance reduction rate β is larger than the cost increase rate γ, the MLMC estimator is in the best case scenario, with $O(\varepsilon^{-2})$ complexity, as stated in Theorem 3. In Fig. 11, we compare the computational complexity of MLMC to that of the standard Monte Carlo. Numerically, we observe that the CPU time of MLMC is approximately $O(\varepsilon^{-2.06})$, which is close to the theoretically predicted rate. In comparison, the CPU time of the standard MC is approximately $O(\varepsilon^{-3.2})$ in this test case.



(a) Means and variances of λ_ℓ and $\lambda_\ell - \lambda_{\ell-1}$. (b) Computational times and average RQI.

Fig. 9 MLMC method using tgRQI and SUPG for Test Case III with random velocity and random conductivity: **a** mean (blue) and variance (red) of the eigenvalue λ_ℓ (dashed) and of $\lambda_\ell - \lambda_{\ell-1}$ (solid); **b** computational times for one multilevel difference (blue) and average number of RQIs (red) on each level. Where shown, the error bars represent $\pm$ one standard deviation (Color figure online)

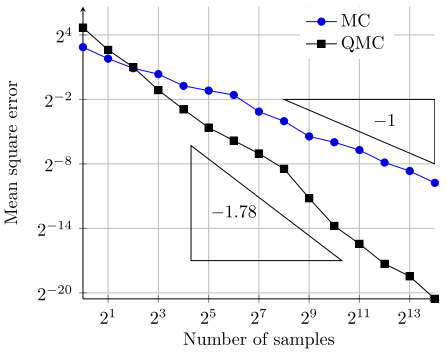
5.3.2 MLQMC

All QMC computations were implemented using Dirk Nuyens' code accompanying [45] and use a randomly shifted embedded lattice rule in base 2, as outlined in [16], with 32 i.i.d. random shifts. In Fig. 10, we plot convergence of the MSE for both MC and QMC for three different cases: for λ_0 in plot (a), for the difference $\lambda_1 - \lambda_0$ in plot (b), and for the difference $\lambda_2 - \lambda_1$ in plot (c). Here the meshwidths are given by $h_0 = 2^{-3}$, $h_1 = 2^{-4}$ and $h_2 = 2^{-5}$. In all cases, QMC outperforms MC, where for λ_0 the MSE for QMC converges at an observed rate of -1.78, whereas MC converges with the rate -1. For the other two cases, which are MSEs of multilevel differences, the QMC converges with an approximate rate of -1.63, which is again clearly faster than the MC convergence rate of -1. This observed MSE convergence for the QMC approximations of the differences implies that *II(b)* holds with $\eta \approx 0.61$. For MLQMC, to choose N_ℓ we use (56) with $\eta \approx 0.61$ and with N_0 scaled such that the overall MSE is less than $\varepsilon^2/\sqrt{2}$ for each tolerance ε. Since we use a base-2 lattice rule, we round up N_ℓ to the next power of 2.

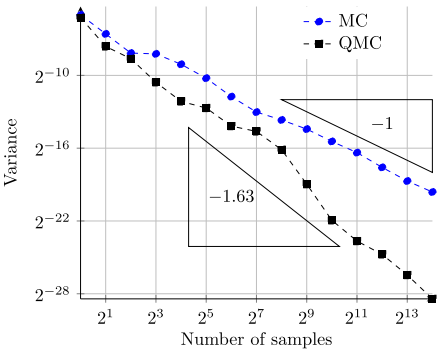
The MLQMC complexity, in terms of CPU time, is plotted in Fig. 11, along with the results for MC and MLMC. Comparing the three methods in Fig. 11, clearly MLQMC provides the best complexity, followed by MLMC then standard MC. In this case, we have the approximate rates $\beta\eta \approx 4 \times 0.61 = 2.44 > \gamma \approx 2.23$, which implies that for MLQMC we are in the optimal regime for the cost with $C_{\mathrm{MLQMC}}(\varepsilon) \lesssim \varepsilon^{-2\eta}$. Numerically, we observe that the rate is given by 1.28, which is very close to the theoretically predicted rate of $2\eta \approx 1.22$.

6 Conclusion

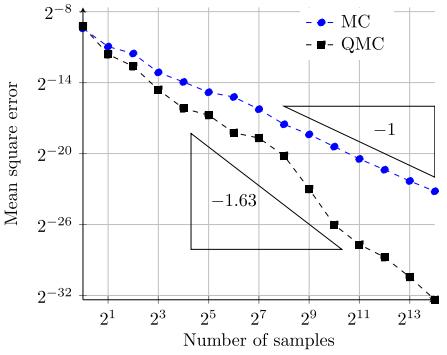
In this paper we have considered and developed various MLMC methods for stochastic convection–diffusion eigenvalue problems in 2D. First, we established certain error bounds on the variational formulation of the eigenvalue problem under assumptions such as eigenvalue gap, boundedness, and other approximation properties. Then we presented the MLMC method based on a hierarchy of geometrically refined meshes with and without homotopy. We also discussed how to improve the computational complexity of MLMC by replacing



(a) λ_0 for $h_0 = 2^{-3}$



(b) $\lambda_1 - \lambda_0$ for $h_0 = 2^{-3}$, $h_1 = 2^{-4}$



(c) $\lambda_2 - \lambda_1$ for $h_1 = 2^{-4}$, $h_2 = 2^{-5}$

Fig. 10 Convergence of QMC and MC methods using tgRQI and SUPG for Test Case III with random velocity and conductivity. Plots **a**–**c** give the MSE of estimators versus sample sizes for grid sizes $h = 2^{-3}, 2^{-4}, 2^{-5}$, respectively. Blue lines with circles and black lines with squares indicate the MSE for MC and QMC, respectively. Dashed lines and solid lines correspond to the MSE of the estimated multilevel differences and the MSE of the estimated eigenvalues, respectively (Color figure online)

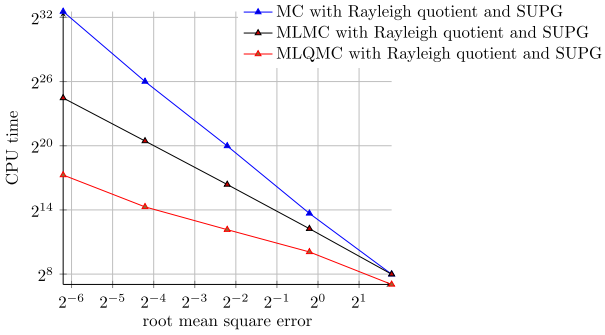


Fig. 11 CPU time versus root mean square error of the estimators in Test Case III

Monte Carlo samples with QMC samples. At last, we provided numerical results for three test cases with different convection velocities.

Test Case I shows that, for low convection velocity, all variants of the MLMC method (based on a Galerkin FEM discretization of the PDE) achieve optimal $O(\varepsilon^{-2})$ complexity, including the one with homotopy. In Test Case II with a high convection velocity, the homotopy-based MLMC does not work anymore—at least without increasing the number of levels—and MLMC based on Galerkin FEM has severe stability restrictions, preventing the use of a large number of levels. This restriction can be circumvented easily by using stable SUPG discretizations. Numerical experiments suggest that MLMC with SUPG achieves the optimal $O(\varepsilon^{-2})$ complexity and is 10–20 times faster than the Galerkin FEM-based versions for the same level of accuracy. In Test Case III, we considered both the conductivity and the convection velocity as random fields and compared the performance of MLMC and MLQMC. In this example, both MLMC and MLQMC deliver computational complexities that are close to the optimal complexities predicted by the theory, while the rate of the computational complexity of MLQMC outperforms that of MLMC.

Appendix: Bounding the Constants in the FE Error

The results in Theorem 2 follow from the Babuška–Osborn theory [4]. In this appendix we show that the constants can be bounded independently of the stochastic parameter.

The Babuška–Osborn theory studies how the continuous solution operators $T_{\boldsymbol{\omega}}, T_{\boldsymbol{\omega}}^* : V \to V$, which for $f, g \in V$ are defined by

$$\begin{aligned} \mathcal{A}(\boldsymbol{\omega}; T_{\boldsymbol{\omega}} f, v) &= \langle f, v\rangle \quad \text{for all } v \in V, \\ \mathcal{A}(\boldsymbol{\omega}; w, T_{\boldsymbol{\omega}}^* g) &= \langle w, g\rangle \quad \text{for all } w \in V, \end{aligned}$$

are approximated by the discrete operators $T_{\boldsymbol{\omega},h}, T_{\boldsymbol{\omega},h}^* : V_h \to V_h$,

$$\begin{aligned} \mathcal{A}(\boldsymbol{\omega}; T_{\boldsymbol{\omega},h} f, v_h) &= \langle f, v_h\rangle \quad \text{for all } v_h \in V_h, \\ \mathcal{A}(\boldsymbol{\omega}; w_h, T_{\boldsymbol{\omega},h}^* g) &= \langle w_h, g\rangle \quad \text{for all } w_h \in V_h. \end{aligned}$$

We summarize the pertinent details here. First, we introduce:

$$\begin{aligned} \eta_h(\lambda(\boldsymbol{\omega})) &:= \sup_{u \in \mathcal{E}(\lambda(\boldsymbol{\omega}))} \inf_{\chi \in V_h} \|u - \chi\|_V, \\ \eta_h^*(\lambda(\boldsymbol{\omega})) &:= \sup_{v \in \mathcal{E}^*(\lambda(\boldsymbol{\omega}))} \inf_{\chi \in V_h} \|v - \chi\|_V, \end{aligned}$$

where the eigenspaces are defined by

$$\begin{aligned} \mathcal{E}(\lambda(\boldsymbol{\omega})) &:= \{u : u \text{ is an eigenfunction of (5) corresponding to } \lambda(\boldsymbol{\omega}),\ \|u\|_{L^2} = 1\}, \\ \mathcal{E}^*(\lambda(\boldsymbol{\omega})) &:= \{u^* : u^* \text{ is an eigenfunction of (8) corresponding to } \lambda(\boldsymbol{\omega}),\ \|u^*\|_{L^2} = 1\}. \end{aligned}$$

The result for the eigenfunction (17) is given by [4, Thm. 8.1], which gives

$$\|u(\boldsymbol{\omega}) - u_h(\boldsymbol{\omega})\|_V \le C_u(\boldsymbol{\omega}) \eta_h(\lambda(\boldsymbol{\omega})), \tag{59}$$

for a constant $C(\boldsymbol{\omega})$ defined below. Since $\lambda(\boldsymbol{\omega})$ is simple, the best approximation property of V_h in $H^2(D)$ followed by Theorem 1 gives

$$\eta_h(\lambda(\boldsymbol{\omega})) \le C_{\mathrm{BAP}} \|u(\cdot, \boldsymbol{\omega})\|_{H^2} h \le C_{\mathrm{BAP}} C_{2,\lambda} |\lambda(\boldsymbol{\omega})| h \le C_{\mathrm{BAP}} C_{2,\lambda} \widehat{\lambda} h, \tag{60}$$

where the best approximation constant C_{BAP} is independent of $\boldsymbol{\omega}$. In the last inequality we have also used that $\lambda(\boldsymbol{\omega})$ is continuous on the compact domain Ω, thus can be bounded uniformly by

$$\widehat{\lambda} := \max_{\boldsymbol{\omega}\in\Omega} |\lambda(\boldsymbol{\omega})| \ < \ \infty. \tag{61}$$

Hence, all that remains is to bound $C_u(\boldsymbol{\omega})$, uniformly in $\boldsymbol{\omega}$. This constant is given by

$$\begin{aligned} C_u(\boldsymbol{\omega}) \ &= \|T_{\boldsymbol{\omega}}\| \, \|u(\boldsymbol{\omega})\|_V \left(1 + \frac{1}{a_{\min}}\right) \frac{\text{length}(\Gamma(\boldsymbol{\omega}))}{\pi} \\ &\quad \times \sup_{\substack{z\in\Gamma \\ h>0}} \|R_z(T_{\boldsymbol{\omega},h})\| \sup_{z\in\Gamma(\boldsymbol{\omega})} \|R_z(T_{\boldsymbol{\omega}})\|, \end{aligned}$$

where $\Gamma(\boldsymbol{\omega})$ is a circle in the complex plane enclosing the eigenvalue $\mu(\boldsymbol{\omega}) = 1/\lambda(\boldsymbol{\omega})$ of $T_{\boldsymbol{\omega}}$, but no other points in the spectrum $\sigma(T_{\boldsymbol{\omega}})$, and for an operator A and $z \in \rho(A) = \mathbb{C} \setminus \sigma(A)$, the *resolvent set* of A, we define the resolvent operator $R_z(A) := (z - A)^{-1}$. Hence, all that remains is to show that $C_u(\boldsymbol{\omega})$ is bounded from above uniformly in $\boldsymbol{\omega}$.

First, by the Lax–Milgram Lemma and the Poincaré inequality $T_{\boldsymbol{\omega}}$ is bounded with $\|T_{\boldsymbol{\omega}}\| \leq C_{\text{Poin}}/a_{\min}$. Also, since $\mathcal{A}$ is coercive (6) and $u(\boldsymbol{\omega})$ satisfies (5), using (61) we have the bound

$$\|u(\boldsymbol{\omega})\|_V \ \leq \sqrt{\frac{\widehat{\lambda}}{a_{\min}}}.$$

Consider next the norm of the resolvent $\|R_z(T_{\boldsymbol{\omega}})\|$ for $\boldsymbol{\omega} \in \Omega$. Note that care must be taken here since the domain for z, namely the resolvent set, changes with $\boldsymbol{\omega}$.

Let $\Gamma(\boldsymbol{\omega}) = \{z \in \mathbb{C} : |z - \mu(\boldsymbol{\omega})| = \gamma/2\}$, where γ is a lower bound on the spectral gap for μ

$$\gamma := \inf_{\boldsymbol{\omega}\in\Omega} \text{dist}(\mu(\boldsymbol{\omega}), \sigma(T_{\boldsymbol{\omega}}) \setminus \{\mu(\boldsymbol{\omega})\}) \ > \ 0.$$

So that for each $\boldsymbol{\omega} \in \Omega$ the circle $\Gamma(\boldsymbol{\omega})$ encloses only $\mu(\boldsymbol{\omega})$ and no other eigenvalues of $T_{\boldsymbol{\omega}}$. Then $z \in \Gamma(\boldsymbol{\omega})$ can be parametrised by both $\boldsymbol{\omega} \in \Omega$ and $\theta \in [0, 2\pi]$,

$$z \ = \ z(\boldsymbol{\omega}, \theta) \ = \ \mu(\boldsymbol{\omega}) + \frac{\gamma}{2} e^{i\theta} \ \in \ \Gamma(\boldsymbol{\omega}).$$

Clearly $z(\cdot, \cdot)$ is continuous in both $\boldsymbol{\omega}$ and θ and belongs to the resolvent set, $z(\boldsymbol{\omega}, \theta) \in \rho(T_{\boldsymbol{\omega}})$, for all $\boldsymbol{\omega} \in \Omega$ and $\theta \in [0, 2\pi]$. Thus, $R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega}})$ is bounded for all $\boldsymbol{\omega} \in \Omega$ and $\theta \in [0, 2\pi]$.

For all $\boldsymbol{\omega} \in \Omega$ we have the bound

$$\sup_{z\in\Gamma(\boldsymbol{\omega})} \|R_z(T_{\boldsymbol{\omega}})\| \ = \ \sup_{\theta\in[0,2\pi]} \|R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega}})\| \ \leq \ \sup_{\substack{\theta\in[0,2\pi] \\ \boldsymbol{\omega}\in\Omega}} \|R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega}})\|.$$

Now, in general the resolvent $R_z(A)$ is continuous in both arguments, z and the (compact) operator A (in fact it is holomorphic, see [42, Theorem IV−3.11]). Since z is continuous in both θ and $\boldsymbol{\omega}$ and $T_{\boldsymbol{\omega}}$ is continuous in $\boldsymbol{\omega}$, it follows that $R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega}})$ is continuous in θ and $\boldsymbol{\omega}$. In turn, the norm $\|R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega}})\|$ is also continuous in θ and $\boldsymbol{\omega}$. Thus, $\|R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega}})\|$ is bounded and continuous on the compact domain $[0, 2\pi] \times \Omega$, and so the maximum is attained for some $(\theta^*, \boldsymbol{\omega}^*) \in [0, 2\pi] \times \Omega$, i.e., for all $\boldsymbol{\omega} \in \Omega$

$$\sup_{z\in\Gamma(\boldsymbol{\omega})} \|R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega}})\| \ \leq \ \max_{\substack{\theta\in[0,2\pi] \\ \boldsymbol{\omega}\in\Omega}} \|R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega}})\| \ = \ \|R_{z(\boldsymbol{\omega}^*,\theta^*)}(T_{\boldsymbol{\omega}^*})\| \ < \ \infty.$$

For h sufficiently small $\|R_z(T_{\boldsymbol{\omega},h})\|$ can be bounded in a similar way.

For $\Gamma(\boldsymbol{\omega})$ deined above length$(\Gamma(\boldsymbol{\omega})) = \pi\gamma$, which is obviously independent of $\boldsymbol{\omega}$. Thus, $C_u(\boldsymbol{\omega}) \leq C_u < \infty$ for all $\boldsymbol{\omega} \in \Omega$, where

$$C_u := \gamma \frac{C_{\text{Poin}}}{a_{\min}} \sqrt{\frac{\widehat{\lambda}}{a_{\min}}} \left(1 + \frac{1}{a_{\min}}\right) \max_{\substack{\theta \in [0,2\pi] \\ \boldsymbol{\omega} \in \Omega}} \|R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega}})\| \sup_{\substack{\theta \in [0,2\pi] \\ \boldsymbol{\omega} \in \Omega \\ h>0}} \|R_{z(\boldsymbol{\omega},\theta)}(T_{\boldsymbol{\omega},h})\|$$

is independent of $\boldsymbol{\omega}$.

For the eigenvalue error (16) we follow the proof of [4, Theorem 8.2]. Since $\lambda(\boldsymbol{\omega})$ is simple, from Theorem 7.2 in [4], the eigenvalue error is bounded by

$$|\lambda(\boldsymbol{\omega}) - \lambda_h(\boldsymbol{\omega})| \leq C_\lambda(\boldsymbol{\omega}) \eta_h(\lambda(\boldsymbol{\omega})) \eta_h^*(\lambda(\boldsymbol{\omega})) \leq C_\lambda(\boldsymbol{\omega}) C_\eta h^2,$$

where in the second inequality we have used (60) and the equivalent bound for the dual eigenvalue, combining the two constants into C_η. By following [4], the constant $C_\lambda(\boldsymbol{\omega})$ can be bounded independently of $\boldsymbol{\omega}$ in a similar way to $C_u(\boldsymbol{\omega})$.

Acknowledgements T. Cui and S. Polishchuk acknowledge support from the Australian Research Council, under grant number CE140100049 (ACEMS). S. Polishchuk acknowledges support from the School of Mathematics at Monash University. T. Cui acknowledges travel support offered by the IWR at Heidelberg University. T. Cui and R. Scheichl further acknowledge support from the Erwin Schrödinger Institute for Mathematics and Physics at the University of Vienna. H. De Sterck acknowledges support from NSERC of Canada.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions

Data Availability Our research does not generate any new data.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Arnoldi, W.E.: The principle of minimized iterations in the solution of the matrix eigenvalue problem. Q. Appl. Math. **9**, 17–29 (1951)
2. Avramova, M.N., Ivanov, K.N.: Verification, validation and uncertainty quantification in multi-physics modeling for nuclear reactor design and safety analysis. Prog. Nucl. Energy **52**, 601–614 (2010)
3. Ayres, D.A.F., Eaton, M.D., Hagues, A.W., Williams, M.M.R.: Uncertainty quantification in neutron transport with generalized polynomial chaos using the method of characteristics. Ann. Nucl. Energy **45**, 14–28 (2012)
4. Babuška, I., Osborn, J.: Eigenvalue problems. In: Ciarlet, P.G., Lions, J.L. (eds.) Handbook of Numerical Analysis, Finite Element Methods (Part 1), vol. 2, pp. 641–787. Elsevier, Amsterdam (1991)
5. Barrenechea, G., Valentin, F.: An unusual stabilized finite element method for a generalized stokes problem. Numer. Math. **92**, 653–677 (2002)
6. Barth, A., Schwab, C., Zollinger, N.: Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. Numer. Math. **119**, 123–161 (2011)

7. Beck, A., Dürrwächter, J., Kuhn, T., Meyer, F., Munz, C.-D., Rohde, C.: *hp*-Multilevel Monte Carlo methods for uncertainty quantification of compressible Navier–Stokes equations. SIAM J. Sci. Comput. **42**(4), B1067–B1091 (2020)
8. Bochev, P.B., Gunzburger, M.D., Shadid, J.N.: Stability of the SUPG finite element method for transient advection–diffusion problems. Comput. Methods Appl. Mech. Eng. **193**(23), 2301–2323 (2004)
9. Broersen, R., Stevenson, R.: A robust Petrov–Galerkin discretisation of convection–diffusion equations. Comput. Math. Appl. **68**(11), 1605–1618 (2014)
10. Brooks, A.N., Hughes, T.J.R.: Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. Comput. Methods Appl. Mech. Eng. **32**, 199–259 (1982)
11. Cai, Z., Mandel, J., McCormick, S.: Multigrid methods for nearly singular linear equations and eigenvalue problems. SIAM J. Numer. Anal. **34**(1), 178–200 (1997)
12. Carnoy, E.G., Geradin, M.: On the practical use of the Lanczos algorithm in finite element applications to vibration and bifurcation problems. In: Kågström, B., Ruhe, A. (eds.) Matrix Pencils, pp. 156–176. Springer, Berlin (1983)
13. Carstensen, C., Gedicke, J., Mehrmann, V., Miedlar, A.: An adaptive homotopy approach for non-self-adjoint eigenvalue problems. Numer. Math. **119**, 557–583, 11 (2011)
14. Cliffe, K.A., Giles, M.B., Scheichl, R., Teckentrup, A.L.: Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. Comput. Vis. Sci. **14**, 3–15 (2011)
15. Cohen, A., Dahmen, W., Welper, G.: Adaptivity and variational stabilization for convection–diffusion equations. Eur. Ser. Appl. Ind. Math. Math. Model. Numer. Anal. **46**, 1247–1273 (2012)
16. Cools, R., Kuo, F.Y., Nuyens, D.: Constructing embedded lattice rules for multivariate integration. SIAM J. Sci. Comput. **28**, 2162–2188 (2006)
17. Crandall, S.H.: Iterative procedures related to relaxation methods for eigenvalue problems. Proc. R. Soc. A Math. Phys. Eng. Sci. **207**, 416–423 (1951)
18. Davis, T.A.: Direct Methods for Sparse Linear Systems. SIAM, Philadelphia (2006)
19. Dick, J., Gantner, R.N., Le Gia, Q.T., Schwab, C.: Higher order Quasi-Monte Carlo integration for Bayesian PDE inversion. Comput. Math. Appl. **77**, 144–172 (2019)
20. Dick, J., Kuo, F.Y., Sloan, I.H.: High dimensional integration: the quasi-Monte Carlo way. Acta Numer. **22**, 133–288 (2013)
21. Dick, J., Pillichshammer, F.: Digital Nets and Sequences: Discrepancy Theory and Quasi-Monte Carlo Integration. Cambridge University Press, New York (2010)
22. Dobson, D., Gopalakrishnan, J., Pasciak, J.: An efficient method for band structure calculations in 3d photonic crystals. J. Comput. Phys. **161**, 668–679 (2000)
23. Donea, J., Huerta, A.: Finite Element Methods for Flow Problems. Wiley, New York (2003)
24. Drummond, I.T., Duane, S., Horgan, R.R.: Scalar diffusion in simulated helical turbulence with molecular diffusivity. J. Fluid Mech. **138**, 75–91 (1984)
25. Duderstadt, J.J., Hamilton, L.J.: Nuclear Reactor Analysis. Wiley, New York (1976)
26. George, A., Ng, E.: On the complexity of sparse QR & LU factorization of finite-element matrices. SIAM J. Sci. Stat. Comput. **9**(5), 849–861 (1988)
27. Giani, S., Graham, I.G.: Adaptive finite element methods for computing band gaps in photonic crystals. Numer. Math. **121**, 31–64 (2012)
28. Gilbert, A.D., Scheichl, R.: Multilevel quasi-Monte Carlo for random elliptic eigenvalue problems I: regularity and error analysis. IMA J. Numer. Anal. (to appear) (2023)
29. Gilbert, A.D., Scheichl, R.: Multilevel quasi-Monte Carlo for random elliptic eigenvalue problems II: efficient algorithms and numerical results. IMA J. Numer. Anal. (to appear) (2023)
30. Giles, M.B.: Multilevel Monte Carlo path simulation. Oper. Res. **56**(3), 607–617 (2008)
31. Giles, M.B.: Multilevel Monte Carlo methods. Acta Numer. **24**, 259–328 (2015)
32. Giles, M.B., Waterhouse, B.: Multilevel quasi-Monte Carlo path simulation. In: Advanced Financial Modelling. Radon Series on Computational and Applied Mathematics, pp. 165–181. De Gruyter, New York (2009)
33. Grisvard, P.: Elliptic Problems in Nonsmooth Domains. SIAM, Philadelphia (2011)
34. Guennebaud, G., Jacob, B. et al.: Eigen v3. http://eigen.tuxfamily.org (2010)
35. Hauke, G.: A simple subgrid scale stabilized method for the advection–diffusion–reaction equation. Comput. Methods Appl. Mech. Eng. **191**, 2925–2947 (2002)
36. Heinrich, S.: Multilevel Monte Carlo methods. In: Margenov, S., et al. (eds.) Large-Scale Scientific Computing, pp. 58–67. Springer, Berlin (2001)
37. Higdon, D.: Space and space-time modeling using process convolutions. In: Anderson, C.W., et al. (eds.) Quantitative Methods for Current Environmental Issues, pp. 37–56. Springer, London (2002)

38. Hughes, T.J.R., Franca, L.P., Hulbert, G.M.: A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective–diffusive equations. Comput. Methods Appl. Mech. Eng. **73**(2), 173–189 (1989)
39. Hughes, T.J.R., Mallet, M.: A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective–diffusive systems. Comput. Methods Appl. Mech. Eng. **58**(3), 305–328 (1986)
40. Hughes, T.J.R., Tezduyar, T.E.: Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations. Comput. Methods Appl. Mech. Eng. **45**(1), 217–284 (1984)
41. Kana, A.A.: Enabling Decision Insight by Applying Monte Carlo Simulations and Eigenvalue Spectral Analysis to the Ship-Centric Markov Decision Process Framework. Ph.D thesis, Univeristy of Michigan, Ann Arbor, Michigan (2016)
42. Kato, T.: Perturbation Theory for Linear Operators. Springer, Berlin (1984)
43. Knobloch, P.: On the definition of the SUPG parameter. Electron. Trans. Numer. Anal. **32**, 76–89 (2008)
44. Kraichnan, R.H.: Diffusion by a random velocity field. Phys. Fluids **13**(1), 22–31 (1970)
45. Kuo, F.Y., Nuyens, D.: Application of quasi-Monte Carlo methods to elliptic PDEs with random diffusion coefficients: a survey of analysis and implementation. Found. Comput. Math. **16**(6), 1631–1696 (2016)
46. Kuo, F.Y., Scheichl, R., Schwab, C., Sloan, I.H., Ullmann, E.: Multilevel quasi-Monte Carlo methods for lognormal diffusion problems. Math. Comp. **86**, 2827–2860 (2017)
47. Kuo, F.Y., Schwab, C., Sloan, I.H.: Multi-level quasi-Monte Carlo finite element methods for a class of elliptic PDEs with random coefficients. Found. Comput. Math. **15**, 411–449 (2015)
48. Lehoucq, R.B.: Analysis and implementation of an implicitly restarted Arnoldi iteration. Ph.D Thesis, Rice University, Houston, Texas (1995)
49. Lehoucq, R.B., Sorensen, D.C., Yang, C.: ARPACK Users' Guide. SIAM, Philadelphia (1998)
50. Lui, S.H., Keller, H.B., Kwok, T.W.C.: Homotopy method for the large, sparse, real non-symmetric eigenvalue problem. SIAM J. Matrix Anal. Appl. **18**(2), 312–333 (1997)
51. McGrail, B.P., Ahmed, S., Schaef, H.T., Owen, A.T., Martin, P.F., Zhu, T.: Gas hydrate property measurements in porous sediments with resonant ultrasonic spectroscopy. J. Geophys. Res. Solid Earth **112**, 1 (2007)
52. Migliori, A.: Resonant ultrasound spectroscopy. Technical Report, Los Alamos National Lab, Los Alamos, NM, USA (2016)
53. Mishra, S., Schwab, C., Sukys, J.: Multi-level Monte Carlo finite volume methods for nonlinear systems of conservation laws in multi-dimensions. J. Comput. Phys. **231**(8), 3365–3388 (2012)
54. Morton, K.W.: Numerical Solution of Convection–Diffusion Problems, vol. 12. CRC Press, Boca Raton (1996)
55. Norton, R.A., Scheichl, R.: Plane wave expansion methods for photonic crystal fibres. Appl. Numer. Math. **63**, 88–104 (2013)
56. Ostrowski, A.M.: On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors. I. Arch. Ration. Mech. Anal. **1**(1), 233–241 (1957)
57. Rayleigh, J.W.S.B.: The Theory of Sound. Macmillan, New York (1894)
58. Saad, Y.: Variations on Arnoldi's method for computing eigen elements of large unsymmetric matrices. Linear Algebra Appl. **34**(C), 269–295 (1980)
59. Saad, Y.: Chebyshev acceleration techniques for solving nonsymmetric eigenvalue problems. Math. Comput. **42**, 567–588 (1984)
60. Scheichl, R., Stuart, A.M., Teckentrup, A.L.: Quasi-Monte Carlo and multilevel Monte Carlo methods for computing posterior expectations in elliptic inverse problems. SIAM/ASA J. Uncertain. Quantif. **5**(1), 493–518 (2017)
61. Schwartz, R.B., Vuorinen, J.F.: Resonant ultrasound spectroscopy: applications, current status and limitations. J. Alloy. Compd. **310**, 243–250 (2000)
62. Scott, J.A.: An Arnoldi code for computing selected eigenvalues of sparse, real, unsymmetric matrices. ACM Trans. Math. Softw. **21**, 432–475 (1995)
63. Stynes, M.: Steady-state convection–diffusion problems. Acta Numer. **14**, 445–508 (2005)
64. Tartakovsky, D.M., Broyda, S.: PDF equations for advective–reactive transport in heterogeneous porous media with uncertain properties. J. Contam. Hydrol. **120–121**, 129–140 (2011)
65. Teckentrup, A.L., Scheichl, R., Giles, M.B., Ullmann, E.: Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients. Numer. Math. **125**, 569–600 (2012)
66. Thomson, W.T.: The Theory of Vibrations with Applications. Prentice-Hall (1981)
67. Zhang, D.: Stochastic Methods for Flow in Porous Media: Coping With Uncertainties. Academic Press, New York (2002)

68. Zienkiewicz, O.C., Taylor, R.L.: Finite Element Method: Fluid Dynamics, vol. 3, 5th edn. Butterworth-Heinemann, Oxford (2000)