

MULTILEVEL QUASI MONTE CARLO METHODS FOR ELLIPTIC PDES WITH RANDOM FIELD COEFFICIENTS VIA FAST WHITE NOISE SAMPLING*

MATTEO CROCI[†], MICHAEL GILES[†], AND PATRICK E. FARRELL[†]

Abstract. When solving partial differential equations (PDEs) with random fields as coefficients, the efficient sampling of random field realizations can be challenging. In this paper we focus on the fast sampling of Gaussian fields using quasi-random points in a finite element and multilevel quasi Monte Carlo (MLQMC) setting. Our method uses the stochastic PDE (SPDE) approach of Lindgren et al. combined with a new fast algorithm for white noise sampling which is tailored to (ML)QMC. We express white noise as a wavelet series expansion that we divide into two parts. The first part is sampled using quasi-random points and contains a finite number of terms in order of decaying importance to ensure good quasi Monte Carlo (QMC) convergence. The second part is a correction term which is sampled using standard pseudo-random numbers. We show how the sampling of both terms can be performed in linear time and memory complexity in the number of mesh cells via a supermesh construction, yielding an overall linear cost. Furthermore, our technique can be used to enforce the MLQMC coupling even in the case of nonnested mesh hierarchies. We demonstrate the efficacy of our method with numerical experiments.

Key words. multilevel quasi Monte Carlo, white noise, nonnested meshes, Matérn Gaussian fields, finite elements, partial differential equations with random coefficients

AMS subject classifications. 65C05, 60G60, 65N30, 60H35, 35R60

DOI. 10.1137/20M1329044

1. Introduction. In this paper we extend the work of [12] to the quasi and multilevel quasi Monte Carlo cases (QMC and MLQMC, respectively). We consider the solution of random elliptic partial differential equations (PDEs) in which Matérn fields, sampled via the stochastic PDE (SPDE) approach [44], appear as coefficients. For instance, a typical problem is as follows: find $\mathbb{E}[P]$, where $P(\omega) = \mathcal{P}(p)$ and \mathcal{P} is a Fréchet differentiable functional of the function p that satisfies

$$(1.1) \quad -\nabla \cdot (F(u(\mathbf{x}, \omega)) \nabla p(\mathbf{x}, \omega)) = f(\mathbf{x}), \quad \mathbf{x} \in G \subset \mathbb{R}^d, \quad \omega \in \Omega,$$

with suitable boundary conditions and smoothness assumptions on the function f and the domain G . $F(x) > 0$ is typically chosen in the literature to be a continuous, locally Lipschitz function (e.g., an exponential) [28, 41, 12]. In this work, we assume that the coefficient $u(\mathbf{x}, \omega)$ is a zero-mean Matérn field approximately sampled by solving the (domain-truncated) Whittle SPDE [55, 44]

$$(1.2) \quad (\mathcal{I} - \kappa^{-2} \Delta)^k u(\mathbf{x}, \omega) = \eta \dot{W}, \quad \mathbf{x} \in D \subset \mathbb{R}^d, \quad \omega \in \Omega, \quad \nu = 2k - d/2 > 0, \quad \kappa = \frac{\sqrt{2\nu}}{\lambda},$$

where \dot{W} is spatial Gaussian white noise in \mathbb{R}^d , defined over a suitable sample space Ω . Here $k > d/4$, $d \leq 3$, and the equality has to hold almost surely (a.s.) and be

*Submitted to the journal's Methods and Algorithms for Scientific Computing section April 2, 2020; accepted for publication (in revised form) June 1, 2021; published electronically August 9, 2021.

<https://doi.org/10.1137/20M1329044>

Funding: This research is supported by EPSRC grant EP/R029423/1 and by the EPSRC Centre for Doctoral Training in Industrially Focused Mathematical Modelling (EP/L015803/1) in collaboration with Simula Research Laboratory.

[†]Mathematical Institute, University of Oxford, Oxford, OX2 6GG, UK (matteo.croci@maths.ox.ac.uk, mike.giles@maths.ox.ac.uk, patrick.farrell@maths.ox.ac.uk).

interpreted in the sense of distributions. The constant $\eta > 0$ is a scaling factor that depends on σ , λ , and ν ; cf. [12]. In what follows we assume that $G \subset\subset D \subset\subset \mathbb{R}^d$, where by $G \subset\subset D$ we indicate that the closure of G is a compact subset of D , and we prescribe homogeneous Dirichlet boundary conditions on ∂D . If the distance between ∂D and ∂G is large enough, then the error introduced by truncating \mathbb{R}^d to D is negligible [50, 40].

A wide range of Gaussian field sampling methods is available in the literature, the most common being the covariance matrix factorization, the Karhunen–Loève expansion of the random field (see section 11.1 in [52]), the hierarchical matrix approximation of the covariance matrix [22, 39, 32, 17], the circulant embedding method [56, 16, 29, 1], and the SPDE approach [55, 44, 12]. We refer the reader to section 2.4 in [10] for a detailed overview and comparison.

In this paper, we only consider the SPDE approach, which consists of sampling a Matérn field by solving (1.2) for a given realization of white noise. Equation (1.2) can be solved after discretization in $O(m)$ or $O(m \log(m))$ cost complexity with an optimal solver such as multigrid. This yields up to an overall $O(m)$ sampling complexity provided that the white noise term can also be sampled in linear cost. In [12], the authors have showed how white noise realizations can be sampled in $O(m)$ complexity within a finite element (FE) and nonnested multilevel Monte Carlo (MLMC) framework.

The new (ML)QMC method we present in this paper is based on the efficient sampling of the white noise term in (1.2) with a hybrid quasi-/pseudo-random sequence. This technique does not require structured grids and is especially designed to work with nonnested mesh hierarchies. While some sampling techniques (e.g., Karhunen–Loève expansion or covariance matrix factorization) are easily extended to the nonnested case, the SPDE approach requires more careful handling [12, 47]. To the best of our knowledge this work is the first in which the SPDE approach is employed in a nonnested MLQMC framework.

For QMC applications it is extremely important for the QMC integrand to have low effective dimensionality and to order the QMC integrand variables in order of decaying importance [7]. For this reason, a common approach in the existing literature about MLQMC methods for elliptic PDEs is the expansion of the random field coefficients as an infinite series of basis functions of $L^2(D)$ that naturally exposes the leading order dimensions in the integrands [41, 42, 28, 15].

When using the SPDE approach and (1.2), the only source of randomness is white noise, and we therefore must expand \dot{W} to achieve the required variable ordering. In this paper, we use a wavelet expansion of \dot{W} . Wavelets in general form a multiresolution orthogonal basis of $L^2(D)$ and are commonly employed within QMC algorithms as their hierarchical structure exposes the leading order dimensions in the integrands while allowing fast $O(m)$ or $O(m \log m)$ complexity operations (depending on the wavelet basis [14]). A classical example of the efficacy of wavelet expansions of white noise (in time) within a QMC method is offered by the Lévy–Ciesielski (or Brownian bridge) construction of Brownian motion. Ubiquitous in mathematical finance, it is commonly used to solve stochastic differential equations with QMC [26, 25]. Inspired by this technique, we choose to expand white noise into a Haar wavelet expansion.¹

In an MLQMC framework, wavelets are used by Kuo, Schwab, and Sloan to sample random fields efficiently, yielding a cost per sample of $O(m \log m)$ using nested

¹Note that the hat functions used in the Lévy–Ciesielski construction are piecewise linear wavelets, their derivatives are Haar wavelets, and white noise in time is the derivative of Brownian motion.

grids [42]. In [34], Herrmann and Schwab use a truncated wavelet expansion of white noise to sample Gaussian fields with the SPDE approach within a nested MLQMC hierarchy. Their work is possibly the closest to ours as they also work with the SPDE approach to Matérn field sampling and use a wavelet expansion of white noise [34]. All these randomized MLQMC methods for elliptic PDEs use randomly shifted lattice rules and derive MLQMC complexity bounds using a pure QMC approach, truncated expansions, and nested hierarchies.

Our work differs from the above-mentioned papers as it specifically focuses on computational aspects. First, we do not derive any MLQMC complexity estimates, but we design our method to work in the general case in which the multilevel hierarchy is nonnested. Second, our objective is to sample white noise exactly in linear (or log-linear) cost complexity within an (ML)QMC framework, similarly as for MLMC in [12]. For this reason, we handle the expansion differently: we do not just truncate it, but we work with the whole infinite expansion of white noise by adding a correction term to the truncation. The truncation term is finite-dimensional, and we sample it with a randomized QMC point sequence; the correction term is infinite-dimensional, and a QMC approach is not feasible. However, the covariance of the correction is known, and we can sample it using pseudo-random numbers with an extension of the technique presented in [12].

The advantage of our hybrid QMC/MC approach is that we can sample white noise exactly, independently from the truncation level and the wavelet degree considered (e.g., while we use Haar wavelets, Herrmann and Schwab in [34] consider higher degree wavelets), without introducing any additional bias into the MLQMC estimate. In contrast, in the aforementioned MLQMC algorithms the expansion must be truncated after enough terms to make the truncation error negligible. Naturally, this advantage comes at a price: since we are using pseudo-random numbers as well, the asymptotic convergence rate of our method with respect to the number of samples N is still the standard Monte Carlo (MC) rate of $O(N^{-1/2})$. Nevertheless, we show that large computational gains can be recovered in practice in a preasymptotic QMC-like regime in which the convergence rate is $O(N^{-\chi})$, $\chi \geq 1/2$, and we derive a partial convergence result (cf. supplementary material in the accompanying file SMM132904.pdf [local/web 327KB] that explains this behavior in the QMC case).

Wavelet spaces are typically nonnested within FE spaces and vice versa, and a transfer operation (i.e., interpolation, projection, or quadrature) is necessarily needed somewhere along the Gaussian field sampling pipeline. While this issue is not mentioned in related work on MLQMC [34], transfer operations are often performed at the end of the sampling process in nonnested MLMC (see, e.g., [30, 48]): the field is first sampled on a usually uniform and structured sampling mesh before transferring it onto the target FE grids. While sampling on a uniform structured grid has its appeal, this approach also has two disadvantages. First, when working with complex geometries and graded meshes, it is desirable for the sampled Matérn field to have the same local accuracy as the solution of the PDE of interest (e.g., (1.1)). Uniform grids must be refined everywhere to satisfy this requirement. Second, the nonnested random field transfer could introduce additional bias, which may be asymptotically larger in some cases. For instance, a piecewise polynomial defined on a uniform grid is not globally differentiable and cannot be accurately integrated or interpolated on a nonnested grid without resorting to more sophisticated techniques.

Our approach solves both issues by directly sampling coupled white noise realizations exactly on each pair of nonnested FE subspaces in the hierarchy so that all the subsequent computations (solution of the SPDE and the PDE of interest, evaluation

of the output functional) can be performed nonintrusively and in parallel without any further interlevel communication. This is done similarly as in our previous work [12], but with the additional feature that the white noise samples are QMC samples obtained via a wavelet expansion. For this sampling operation we adopt the embedded mesh technique by Osborn, Vassilevski, and Villa [47] so that in the MLQMC hierarchy each mesh of G is nested within the corresponding mesh of D and we deal with the nonnestedness of the FE and wavelet spaces via a supermesh construction. The latter is key in our algorithm for drawing exact white noise samples (over the FE spaces). In the independent white noise realization case we construct a two-way supermesh between the graded FE mesh of interest and a uniform “wavelet” mesh. In the MLQMC coupled realization case, we construct a three-way supermesh between the two nonnested FE meshes and the “wavelet” mesh. If all meshes involved are nested, a supermesh construction is not required. Owing to the embedded mesh technique and the drawing of exact white noise samples, we avoid all nonnested transfer operations and any additional bias.

It is worth noting that the QMC and MLQMC methods presented are especially designed to work with nonnested grid hierarchies and that the use of a supermesh construction is merely a requirement of this framework rather than a specific feature of our algorithms. In fact, a supermesh construction might be unavoidable even when the field is sampled via an alternative method such as, e.g., circulant embedding [16], since this is needed to integrate the sampled field exactly on the target nonnested unstructured grid (see [8] for an analysis of quadrature error). For low-smoothness fields ($\nu \leq 1$), Graham et al. have proved that there is no loss in the convergence rate due to nonnested interpolation [30]. However, this operation, albeit faster, still introduces extra bias and harms convergence when smoother fields are used.

We remark that the QMC method presented in this paper has been successfully applied in [13] on an unstructured MRI-derived 3D brain mesh to quantify the uncertainty in models describing brain fluid and solute movement.

This paper is structured as follows: in section 2 we summarize the mathematical background needed to understand the rest of the paper. In section 3 we introduce the Haar wavelet expansion of white noise and its splitting into a truncated term and a correction term. In section 4 we introduce our sampling technique for independent white noise realizations. We extend the white noise sampling method to MLQMC in section 5, where we show how coupled white noise realizations can be sampled efficiently. The algorithms are supported by numerical results, which we present and discuss in section 6. We conclude the paper with a brief summary of the methods and results presented in section 7.

2. Notation and background.

2.1. Notation. In this paper we denote by $L^2(D)$ the space of square-integrable functions over D and by (\cdot, \cdot) the standard $L^2(D)$ inner product. We furthermore indicate with $W^{k,q}(D)$ the standard Sobolev space of integrability order q and differentiability k , with $H^k(D) \equiv W^{k,2}(D)$ and with $H_0^1(D)$ the space of $H^1(D)$ functions that vanish on ∂D in the sense of traces.

Given a sample space Ω , we indicate with $L^2(\Omega, \mathbb{R})$ the space of real-valued *random variables* with finite second moment.

For a given Banach space U over D , we indicate with $L^2(\Omega, U)$ the space of *random fields* $u(\mathbf{x}, \omega)$, $\mathbf{x} \in D$, $\omega \in \Omega$, such that $u(\mathbf{x}, \cdot) \in L^2(\Omega, \mathbb{R})$ for almost every $\mathbf{x} \in D$ and $u(\cdot, \omega) \in U$ a.s. If the $u(\mathbf{x}, \cdot)$ are jointly Gaussian for almost every $\mathbf{x} \in D$, then the random field is a *Gaussian field*, and it is uniquely determined by its mean

$\mu(\mathbf{x})$ and covariance $C(\mathbf{x}, \mathbf{y})$ function. Throughout the paper we will consider only zero-mean fields for simplicity.

A Gaussian field is also a *Matérn field* if its covariance is of the Matérn class, i.e.,

$$(2.1) \quad C(\mathbf{x}, \mathbf{y}) := \frac{\sigma^2}{2^{\nu-1}\Gamma(\nu)} (\kappa r)^\nu \mathcal{K}_\nu(\kappa r), \quad r := \|\mathbf{x} - \mathbf{y}\|_2, \quad \kappa := \frac{\sqrt{2\nu}}{\lambda}, \quad \mathbf{x}, \mathbf{y} \in D,$$

where σ^2 , ν , $\lambda > 0$ are the variance, smoothness parameter, and correlation length of the field, respectively, $\Gamma(x)$ is the Euler Gamma function, and \mathcal{K}_ν is the modified Bessel function of the second kind.

In this paper we will adopt the following definition of a *generalized random field*, first introduced by Itô [37] and extended by Inaba and Tapley [36]. For a given Banach space U , we denote by $\mathcal{L}(U, L^2(\Omega, \mathbb{R}))$ the space of generalized random fields that are continuous linear mappings from U to $L^2(\Omega, \mathbb{R})$. For a given $\xi \in \mathcal{L}(U, L^2(\Omega, \mathbb{R}))$ we indicate the action (or pairing) of ξ onto a function $\phi \in U$ with the notation $\xi(\phi) := \langle \xi, \phi \rangle$.

Possibly the most commonly used generalized random field is Gaussian *white noise*. White noise is defined as follows.

DEFINITION 2.1 (White noise; see Example 1.2 and Lemma 1.10 in [35]). *Let $D \subseteq \mathbb{R}^d$ be an open domain. The white noise $\dot{W} \in \mathcal{L}(L^2(D), L^2(\Omega, \mathbb{R}))$ is a generalized stochastic field such that for any collection of $L^2(D)$ functions $\{\phi_i\}$, if we let $b_i := \langle \dot{W}, \phi_i \rangle$, then $\{b_i\}$ are joint Gaussian random variables with zero mean and covariance given by $\mathbb{E}[b_i b_j] = (\phi_i, \phi_j)$.*

When working with the finite element method (FEM), it is useful to define the L^2 projection of white noise onto an FE space and its properties.

DEFINITION 2.2 (L^2 projection of white noise; Definition 3.2 in [10]). *Let D_h be an FE mesh of an open domain $D \subseteq \mathbb{R}^d$, and let $U_h \subseteq L^2(D)$ be an FE subspace defined over D_h . Let P_h be the L^2 projection onto U_h . The L^2 projection of white noise $P_h \dot{W} \in \mathcal{L}(L^2(D), L^2(\Omega, \mathbb{R}))$ is the generalized random field that satisfies*

$$(2.2) \quad \langle P_h \dot{W}, v \rangle := \dot{W}(P_h v) \equiv \langle \dot{W}, P_h v \rangle \quad \forall v \in L^2(D).$$

LEMMA 2.3. *If $\dim(U_h) < \infty$, then $P_h \dot{W}$ is a “proper” Gaussian field in $L^2(\Omega, U_h) \subseteq L^2(\Omega, L^2(D))$ and $\langle P_h \dot{W}, v \rangle \equiv (P_h \dot{W}, v)$ for all $v \in L^2(D)$. Furthermore,*

$$(2.3) \quad (P_h \dot{W}, v_h) = \langle \dot{W}, v_h \rangle \quad \forall v_h \in U_h, \quad (P_h \dot{W}, v^\perp) = 0 \quad \forall v^\perp \in U_h^\perp;$$

i.e., \dot{W}_h coincides with \dot{W} over U_h and is zero on U_h^\perp .

Proof. Every $v \in L^2(D)$ admits the decomposition $v = v_h + v^\perp$, where $v_h \in U_h$ and $v^\perp \in U_h^\perp$. Given two orthonormal bases $\{\phi_i\}_{i=1}^\infty$ of U_h and $\{\phi_i^\perp\}_{i=1}^\infty$ of U_h^\perp , we can then expand the action of white noise against any $v \in L^2(D)$ as the series

$$(2.4) \quad \langle \dot{W}, v \rangle = \sum_{i=1}^n a_i(\phi_i, v) + \sum_{i=1}^\infty a_i^\perp(\phi_i^\perp, v), \quad \text{where} \quad a_i = \langle \dot{W}, \phi_i \rangle, \quad a_i^\perp = \langle \dot{W}, \phi_i^\perp \rangle.$$

By defining $\dot{W}_h := \sum_{i=1}^n a_i \phi_i \in L^2(\Omega, U_h)$, it is then readily shown that $\dot{W}_h \equiv P_h \dot{W}$ in $L^2(\Omega, L^2(D))$ since $(\dot{W}_h, v^\perp) = 0$ a.s. for all $v^\perp \in U_h^\perp$, and therefore

$$(2.5) \quad \langle P_h \dot{W}, v \rangle := \langle \dot{W}, P_h v \rangle = \sum_{i=1}^n a_i(\phi_i, P_h v) = (\dot{W}_h, v), \quad \text{a.s.,} \quad \forall v \in L^2(D).$$

The remaining part of the thesis readily follows. \square

2.2. Randomized quasi Monte Carlo. Quasi Monte Carlo (QMC) methods retain most of the advantages of standard Monte Carlo (MC) while improving the convergence order with respect to the number of samples. At the heart of QMC for estimating expectations is the reinterpretation/approximation of the expected value as an integral with respect to the uniform distribution over the unit hypercube. This integral is then approximated with an N -point quadrature rule with equal weights:

$$(2.6) \quad \mathbb{E}[P] = \int_{\Omega} P(\omega) d\mathbb{P}(\omega) \approx \int_{[0,1]^s} Y(\mathbf{x}) d\mathbf{x} =: I \approx \frac{1}{N} \sum_{n=1}^N Y(\mathbf{x}_n) =: I_N$$

for some suitable function Y obtained by mapping the integration domain to the unit hypercube (cf. section 2.11 in [15]). Here the first approximation sign indicates that there might be an additional error due to dimension truncation in case the integrand is infinite-dimensional. The second approximation sign simply represents the actual QMC approximation. Here \mathbb{P} is the probability measure of a suitable probability space $(\Omega, \mathcal{A}, \mathbb{P})$, with \mathcal{A} as its σ -algebra. The $\mathbf{x}_n \in [0, 1]^s$ are, unlike in the standard MC case, not chosen at random, but chosen carefully and in a deterministic way so as to minimize the quadrature error. However, these deterministic points often yield error bounds which are difficult to compute. For this reason QMC points are typically randomized to allow a more practical statistical error estimation [43, 49]. In randomized QMC, M independent randomized sequences $\{\{\hat{\mathbf{x}}_{n,m}\}_{n=1}^N\}_{m=1}^M$ are used to obtain the estimator

$$(2.7) \quad \hat{I}_{N,M}(\omega) := \frac{1}{M} \sum_{m=1}^M I_N^m(\omega) = \frac{1}{M} \sum_{m=1}^M \left(\frac{1}{N} \sum_{n=1}^N Y(\hat{\mathbf{x}}_{n,m}(\omega)) \right).$$

Since $I_N^m(\omega)$ is now a random variable, provided that M is large enough a confidence interval can be estimated, and we retain a practical error measure as in the standard MC case. In this work we use $M = 32$ unless otherwise stated.

QMC methods achieve faster-than-MC convergence rates by carefully choosing the point sequence $\{\mathbf{x}_n\}_{n=1}^N$ and its randomization. For a detailed overview of QMC integration theory, we refer the reader to the excellent review article by Dick, Kuo, and Sloan [15].

According to both classical [43, 7] and modern [42, 27, 30, 41, 34] QMC theory, for practical applications of high-dimensional QMC integration it is extremely important to order the integration variables in order of decaying importance and/or possibly reduce the dimensionality of the integrand. This will be a key aspect in the methods we present in this paper.

2.3. Multilevel Monte Carlo methods. The MLMC method was first introduced by Heinrich in [33] for parametric integration and popularized by Giles for stochastic differential equations in [23]. MLQMC, originally presented in [25], combines QMC and MLMC together with the objective of combining their advantages. Assume that it is possible to compute realizations of $P(\omega)$ at different accuracy levels $P_\ell(\omega)$ for $\ell = 1, \dots, L$ of increasing accuracy and computational cost, and that the approximation of P on the finest level P_L is accurate enough. Multilevel methods estimate $\mathbb{E}[P_L]$ through the telescopic sum,

$$(2.8) \quad \mathbb{E}[P] \approx \mathbb{E}[P_L] = \sum_{\ell=1}^L \mathbb{E}[P_\ell - P_{\ell-1}],$$

where $P_0 \equiv 0$. For example, if samples of P are obtained by solving (1.1) with the FEM, the levels of accuracy can be defined by using a hierarchy of meshes (h -refinement) or by increasing the polynomial degree of the FE bases used (p -refinement).

The MLMC and MLQMC estimators are then obtained from (2.8) by approximating each term in the sum with standard MC or randomized QMC, respectively. For MLMC we have

$$(2.9) \quad \mathbb{E}[P_\ell - P_{\ell-1}] \approx \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} (P_\ell - P_{\ell-1})(\omega_\ell^n),$$

in which each $P_\ell - P_{\ell-1}$ sample is coupled in the sense that the samples of P_ℓ and $P_{\ell-1}$ share the same event ω_ℓ^n . Ensuring this coupling is respected is essential for any MLMC algorithm since this coupling is the reason behind the increased efficiency of MLMC with respect to standard MC [23, 24].

Enforcing the same type of coupling is also essential for MLQMC. In the MLQMC case each term in (2.8) is approximated with randomized QMC as follows:

$$(2.10) \quad \mathbb{E}[P_\ell - P_{\ell-1}] \approx \int_{[0,1]^{s_\ell}} Y_\ell(\mathbf{x}) d\mathbf{x} \approx \frac{1}{M} \sum_{m=1}^M \left(\frac{1}{N_\ell} \sum_{n=1}^{N_\ell} Y_\ell(\hat{\mathbf{x}}_{n,m}^\ell) \right) =: \frac{1}{M} \sum_{m=1}^M I_{N_\ell}^{m,\ell}(\omega),$$

where the meaning of each variable is the same as in the QMC case. We now have a hierarchy of integrands $\{Y_\ell\}_{\ell=1}^L$ and of randomized QMC point sequences $\{\{\hat{\mathbf{x}}_{n,m}^\ell\}_{n=1}^{N_\ell}\}_{m=1}^M\}_{\ell=1}^L$ of dimensions $\{s_\ell\}_{\ell=1}^L$. Note that, in the same way as for QMC, MLQMC still requires for good performance that the integration variables on each level be organized in order of decaying importance and possibly that the integrands Y_ℓ have low effective dimensionality.

The theory for MLMC is by now established [23, 24, 9, 53], yielding formulas for the optimal number of samples N_ℓ on each level and for the total MLMC algorithm complexity ($O(\varepsilon^{-2})$ in the best-case scenario; see supplementary material SM1). On the other hand, obtaining a reliable estimate of the QMC convergence rate with respect to N_ℓ for MLQMC is particularly hard, to the extent that theoretical results are only available for a few specific problems and specific QMC point sequences [41, 34]. For this reason, setting up an optimal MLQMC hierarchy with the optimal values of N_ℓ is a challenging task. However, in the best possible case where we get an $O(N^{-\chi})$, $1/2 \leq \chi \leq 1$, QMC rate for each term in the telescoping sum, the benefits of MLMC and QMC can accumulate, yielding a total MLQMC computational cost of $O(\varepsilon^{-1/\chi})$ for a given mean square error tolerance of ε^2 [34]. In this case MLQMC significantly outperforms all other MC methods.

In this paper we employ the original MLQMC algorithm from [25] as it does not require the convergence rate with respect to N to be known a priori. We refer the reader to the supplementary material SM1 for a description of the algorithm.

2.4. Supermeshes. In this section we introduce the concepts of nonnested tessellations/meshes and of a supermesh.

Let T_a and T_b be two tessellations of D . We say that T_a is *nested* within T_b if $\text{vertices}(T_a) \subseteq \text{vertices}(T_b)$ and if for each element $e \in T_a$ there exists a set of elements $E \subseteq T_b$ such that $e = \bigcup_{\hat{e}_i \in E} \hat{e}_i$. We say that T_a and T_b are *nonnested* if T_a is not nested within T_b and vice versa.

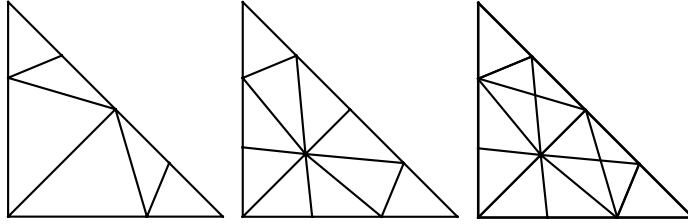


FIG. 2.1. An example of a supermesh construction. The first two meshes on the left are the parent meshes, and the mesh on the right is one of their supermeshes.

To enforce the multilevel (quasi) Monte Carlo coupling across a nonnested mesh hierarchy we use a crucial ingredient: a supermesh construction. Supermeshes are commonly used, e.g., within adaptive discretizations or geometric multigrid algorithms, to transfer discrete fields between nonnested meshes [21]. A supermesh is defined as follows.

DEFINITION 2.4 (Supermesh [20, 21]). *Let $D \subset \mathbb{R}^d$ be an open domain, and let $\mathcal{T}_a, \mathcal{T}_b$ be two tessellations of D . A supermesh S_h of \mathcal{T}_a and \mathcal{T}_b is a common refinement of \mathcal{T}_a and \mathcal{T}_b . More specifically, S_h is a triangulation of D such that*

1. $\text{vertices}(\mathcal{T}_a) \cup \text{vertices}(\mathcal{T}_b) \subseteq \text{vertices}(S_h)$,
2. $\text{measure}(e_S \cap e) \in \{0, \text{measure}(e_S)\}$ for all cells $e_S \in S_h$, $e \in (\mathcal{T}_a \cup \mathcal{T}_b)$.

The first condition means that every parent mesh vertex must also be a vertex of the supermesh, while the second states that every supermesh cell is completely contained within exactly one cell of either parent mesh [21]. The supermesh construction is not unique [21]. We show an example of supermesh construction in Figure 2.1. Efficient algorithms for computing the supermesh are available [46].

It was shown in [11] that if the parent tessellations T_a and T_b are quasi-uniform (cf. Definition 4.4.13 in [6]), then the number of cells of a supermesh constructed via a local supermeshing algorithm (cf. [19]) is linear in the number of cells of T_a and T_b .

2.5. Fast white noise sampling over finite element spaces. When working with MC methods and the SPDE approach, it is essential to draw white noise samples efficiently. In practice, it is typically sufficient to sample realizations of $P_h \dot{W}$ since $P_h \dot{W}$ coincides with \dot{W} over the FE space $U_h = \text{span}\{\phi_1, \dots, \phi_n\}$. In turn, samples of $P_h \dot{W}$ can be obtained by drawing realizations of the Gaussian vector $\mathbf{b} \in L^2(\Omega, \mathbb{R}^n)$ with $b_i := (P_h \dot{W}, \phi_i) = \langle \dot{W}, \phi_i \rangle$, i.e., the action of $P_h \dot{W}$ over the basis functions of U_h . The vector \mathbf{b} has covariance matrix M , where $M_{ij} = (\phi_i, \phi_j)$ (cf. Definition 2.1). Drawing realizations of \mathbf{b} can be expensive since it typically involves the factorization of M (the mass matrix over U_h) that has up to cubic cost complexity. In [12], we presented a linear (optimal) cost complexity sampling algorithm which we use in this paper and we now briefly summarize (see section 4.1 in [12] for further details).

We first define the auxiliary space \tilde{U}_h by restricting the support of the basis functions of U_h onto $\{e_k\}_{k=1}^m$, the cells of D_h . Namely, we define $\tilde{U}_h := \sum_{k=1}^m U_h^b|_{e_k}$, where $U_h^b|_{e_k} = \text{span}\{\phi_1|_{e_k}, \dots, \phi_n|_{e_k}\}$. We indicate with $\{\psi_i\}_{i=1}^{\tilde{n}}$ the basis functions of \tilde{U}_h . Note that by construction $U_h \subseteq \tilde{U}_h \subseteq L^2(D)$. After letting P_h and \tilde{P}_h be the L^2 projections onto U_h and \tilde{U}_h , respectively, the technique in [12] essentially samples $P_h \dot{W}$ as $P_h \dot{W} = P_h \tilde{P}_h \dot{W}$, which is done by first sampling $\tilde{P}_h \dot{W}$ and then projecting it onto U_h . Both operations can be performed in linear cost. Let $\mathbf{a} \in L^2(\Omega, \mathbb{R}^{\tilde{n}})$, $\mathbf{b} \in L^2(\Omega, \mathbb{R}^n)$ be Gaussian vectors such that $a_i = (\tilde{P}_h \dot{W}, \psi_i)$ and $b_i = (P_h \dot{W}, \phi_i)$. The sampling of \mathbf{a} has linear cost complexity in m and is trivially parallelizable since

its covariance matrix is the mass matrix over \dot{U}_h , which is block-diagonal (it has m blocks of small size) and can be factorized efficiently. We can then obtain \mathbf{b} from \mathbf{a} in linear cost since every basis function of U_h can be written as a small sum of the basis functions of \dot{U}_h , and therefore there exists a sparse boolean matrix $L \in \mathbb{R}^{n \times \bar{n}}$ such that $\mathbf{b} = L\mathbf{a}$ a.s. As an example, if we split a $\phi_i \in U_h$ into $\phi_i = \phi_i|_{e_j} + \phi_i|_{e_k}$, then $\phi_i|_{e_j}, \phi_i|_{e_k} \in \dot{U}_h$ and

(2.11)

$$(P_h \dot{W}, \phi_i) = \langle \dot{W}, \phi_i \rangle = \langle \dot{W}, \phi_i|_{e_j} \rangle + \langle \dot{W}, \phi_i|_{e_k} \rangle = (\dot{P}_h \dot{W}, \phi_i|_{e_j}) + (\dot{P}_h \dot{W}, \phi_i|_{e_k}).$$

The matrix L is called an assembly matrix, and its action encodes the assembly procedure performed by any FE code to build global matrices and vectors from local cell contributions [54]. Once a sample of \mathbf{b} has been drawn, the coefficients in the FE expansion of $P_h \dot{W}$ are given by $M^{-1}\mathbf{b}$. Inverting the mass matrix is still a linear cost operation, although it is not needed in the SPDE approach, which only requires samples of \mathbf{b} .

In an MLMC framework, we must couple samples of white noise over two different, possibly nonnested FE spaces V^ℓ and $V^{\ell-1}$. The technique just described can be extended so that coupled white noise samples can also be drawn in linear cost. The idea is to set $U_h = V^\ell + V^{\ell-1}$ and use this technique to sample white noise over U_h . Since $V^\ell, V^{\ell-1} \subseteq U_h$, one can then easily transfer the result over V^ℓ and $V^{\ell-1}$ and obtain coupled samples. The main complication here is that to construct the auxiliary space \dot{U}_h now a supermesh construction is required: by splitting the support of all basis functions of U_h across the supermesh cells, we can ensure that the mass matrix over the space \dot{U}_h still has a block-diagonal structure and can thus be factorized efficiently. We refer the reader to section 4.2 in [12] for further details.

3. Haar wavelet expansion of spatial white noise. For good QMC convergence we need to order the dimensions of the QMC integrand in order of decaying importance so that the largest error components are on the leading dimensions [26, 15]. In what follows we expand white noise into a Haar wavelet series so that the hierarchical structure of Haar wavelets can naturally provide the variable ordering needed for QMC integration.

We start by briefly introducing the Haar wavelet basis; see, e.g., [14]. Let $\mathbb{1}_A(x)$ be the indicator function of a set A , let $\bar{\mathbb{N}} := \{-1\} \cup \mathbb{N}$, and let $x^+ := \max(x, 0)$. The Haar wavelets $H_{l,n}$ for $l \in \bar{\mathbb{N}}$, $n = 0, \dots, 2^{l^+} - 1$ are obtained from the mother wavelet $\Psi(x) := \mathbb{1}_{[0,1/2)}(x) - \mathbb{1}_{[1/2,1)}(x)$ through shifting and rescaling as $H_{l,n} := 2^{l^+/2} \Psi(2^l x - n)$. These wavelets have support of size $|\text{supp}(H_{l,n})| = 2^{-l^+}$ and form an orthonormal basis of $L^2(0,1)$. The Haar system can be generalized to higher dimensions by taking the tensor product of the 1D Haar basis with itself. In the higher dimensional case we then have vector-indexed d -dimensional Haar wavelets $H_{\mathbf{l},\mathbf{n}}(\mathbf{x}) := \bigotimes_{i=1}^d H_{l_i,n_i}(x_i)$, where $\mathbf{l} \in \bar{\mathbb{N}}^d$ and $\mathbf{n} \in \mathbb{N}$. The d -dimensional Haar wavelets have support size $|\text{supp}(H_{\mathbf{l},\mathbf{n}})| = \prod_{i=1}^d 2^{-l_i^+} = 2^{-\|\mathbf{l}^+\|_1}$, and they form an orthonormal basis of $L^2((0,1)^d)$. Generalizing the Haar basis to generic boxed domains is straightforward.

Remark 3.1 (On the nonstandard Haar wavelet basis). The d -dimensional wavelets just introduced are sometimes called the *standard* Haar basis, which leads to log-linear complexity operations (rather than just linear) for $d > 1$. The algorithms we introduce in this paper also work for the *nonstandard* Haar basis, which supports linear complexity operations in all dimensions [14, 3].

Let $z_{\mathbf{l}, \mathbf{n}}(\omega)$ be independent and identically distributed (i.i.d.) standard normal random variables. Furthermore, let $|\mathbf{l}| := \max_i(l_i)$. We can express white noise over $[0, 1]^d$ as a Haar wavelet expansion,

$$(3.1) \quad \dot{W} = \sum_{|\mathbf{l}|=-1}^{\infty} \sum_{\mathbf{n}=0}^{2^{l^+}-1} z_{\mathbf{l}, \mathbf{n}}(\omega) H_{\mathbf{l}, \mathbf{n}}(\mathbf{x}),$$

where $z_{\mathbf{l}, \mathbf{n}} := \langle \dot{W}, H_{\mathbf{l}, \mathbf{n}} \rangle$ for all \mathbf{l}, \mathbf{n} , making the $z_{\mathbf{l}, \mathbf{n}}$ be i.i.d. standard Gaussian random variables. The second summation is to be interpreted as the sum over all \mathbf{n} with components n_i such that $0 \leq n_i \leq 2^{l_i^+} - 1$ for all i . Let $\mathcal{L} \in \mathbb{N}$. We now divide the series into two terms:

$$(3.2) \quad \dot{W} = \dot{W}_{\mathcal{L}} + \dot{W}_R = \sum_{|\mathbf{l}|=-1}^{|\mathcal{L}|} \sum_{\mathbf{n}=0}^{2^{l^+}-1} z_{\mathbf{l}, \mathbf{n}}(\omega) H_{\mathbf{l}, \mathbf{n}}(\mathbf{x}) + \sum_{|\mathbf{l}|=\mathcal{L}+1}^{\infty} \sum_{\mathbf{n}=0}^{2^{l^+}-1} z_{\mathbf{l}, \mathbf{n}}(\omega) H_{\mathbf{l}, \mathbf{n}}(\mathbf{x}).$$

Here the term $\dot{W}_{\mathcal{L}}$ is a “proper” stochastic field in $L^2(\Omega, L^2(D))$, while \dot{W} and \dot{W}_R are generalized stochastic fields with actions defined for all $v \in L^2(D)$ respectively by

$$(3.3) \quad \langle \dot{W}, v \rangle = \sum_{|\mathbf{l}|=-1}^{\infty} \sum_{\mathbf{n}=0}^{(2^{l^+}-1)^+} z_{\mathbf{l}, \mathbf{n}}(H_{\mathbf{l}, \mathbf{n}}, v) \quad \text{and} \quad \langle \dot{W}_R, v \rangle = \sum_{|\mathbf{l}|=\mathcal{L}+1}^{\infty} \sum_{\mathbf{n}=0}^{(2^{l^+}-1)^+} z_{\mathbf{l}, \mathbf{n}}(H_{\mathbf{l}, \mathbf{n}}, v).$$

The idea is then to sample the Gaussian variables in the expression for $\dot{W}_{\mathcal{L}}$ by using a hybrid QMC/MC combination of quasi-random (e.g., Sobol’) and pseudo-random numbers, and to sample \dot{W}_R with pseudo-random numbers only by extending the work in [12].

The reasoning behind this splitting is that it is important to keep the dimensionality of the QMC point sequence relatively low: first, as we will see in the next section, the sampling of \dot{W} expressed this way requires a supermesh construction, and smaller dimensions imply faster \dot{W} samples; second, some QMC point sequences cannot readily or efficiently be sampled in high dimensions;² and third, the approximation properties of some quasi-random sequences deteriorate as the dimensionality grows [26, 15].

4. Sampling independent realizations for QMC. To sample $u(\mathbf{x}, \omega)$, we must solve (1.2). In what follows, we set $\eta = 1$, and we will only consider the $k = 1$ case for simplicity. We refer the reader to [5, 4] for the general $k > d/4$ case. After these simplifications, we obtain

$$(4.1) \quad u - \kappa^{-2} \Delta u = \dot{W}, \quad \mathbf{x} \in D, \quad \text{with} \quad u = 0, \quad \mathbf{x} \in \partial D.$$

From now on we introduce the simplifying assumption that $D = [0, 1]^d$. Relaxing this assumption to general boxed domains is straightforward, but considering more general cases is nontrivial. We remark that the extended domain D is just a tool needed for the SPDE approach and can be chosen arbitrarily. We are therefore free

²For example, the state-of-the-art Sobol’ sequence generator, Broda, can generate the largest dimensional Sobol’ sequences with 65536 dimensions [51]. This might still be too low for an infinite-dimensional PDE setting.

to choose any domain shape for D anyway [44]. Hence working with boxed domains is not really a restriction. It is useful for what comes next to introduce the concept of a Haar mesh and a Haar space.

DEFINITION 4.1 (Haar mesh). Let $D = [0, 1]^d$, and let $\mathcal{L} \in \bar{\mathbb{N}}$. The Haar mesh $D_{\mathcal{L}}$ is the uniform quadrilateral mesh of D whose cells are all regular polyhedra of volume $|\square_H| = 2^{-d(\mathcal{L}+1)}$. Note that for a given \mathcal{L} there are exactly as many cells in $D_{\mathcal{L}}$ as terms in the wavelet expansion (3.2) for $\dot{W}_{\mathcal{L}}$, namely $\mathcal{N}_{\mathcal{L}} = 2^{d(\mathcal{L}+1)}$.

DEFINITION 4.2 (Haar space). Given a Haar mesh $D_{\mathcal{L}}$ with cells $\{\square_k\}_{k=1}^{\mathcal{N}_{\mathcal{L}}}$, we define its Haar space V_H to be the space of piecewise constant functions over it; $V_H := \text{span}(\psi_1, \dots, \psi_{\mathcal{N}_{\mathcal{L}}})$ with $\psi_k := \mathbb{1}_{\square_k}$.

We solve (4.1) with the FEM. Let D_h be a mesh of D , not necessarily nested within the Haar mesh $D_{\mathcal{L}}$. Let $V \subseteq H_0^1(D)$, and let $V_h = \text{span}(\varphi_1, \dots, \varphi_{\bar{m}}) \subseteq V$ be the FE subspace used to solve (4.1) on D_h . In what follows we will refer to D_h as the *FE mesh*, and we assume for simplicity that there are always at most m_e degrees of freedom of V_h on each cell of D_h .

A discrete weak form of (4.1) then reads as follows: find $u_h \in V_h$ such that

$$(4.2) \quad (u_h, v_h) + \kappa^{-2}(\nabla u_h, \nabla v_h) = \langle \dot{W}, v_h \rangle \quad \forall v_h \in V_h.$$

The solution $u_h = \sum_{i=1}^{\bar{m}} u_i \varphi_i$, expressed in terms of the basis functions of V_h , is given by the following linear system for the u_i :

$$(4.3) \quad \mathbf{A} \mathbf{u} = \mathbf{b}, \quad \text{where} \quad A_{ij} := (\varphi_i, \varphi_j) + \kappa^{-2}(\nabla \varphi_i, \nabla \varphi_j), \quad b_i := \langle \dot{W}, \varphi_i \rangle.$$

Now, since $\dot{W} = \dot{W}_{\mathcal{L}} + \dot{W}_R$, the b_i can also be expressed as

$$(4.4) \quad b_i = (\mathbf{b}_{\mathcal{L}})_i + (\mathbf{b}_R)_i, \quad \text{with} \quad (\mathbf{b}_{\mathcal{L}})_i = \langle \dot{W}_{\mathcal{L}}, \varphi_i \rangle, \quad (\mathbf{b}_R)_i = \langle \dot{W}_R, \varphi_i \rangle.$$

Note that we use the $L^2(D)$ inner product notation for $\dot{W}_{\mathcal{L}}$ since $\dot{W}_{\mathcal{L}}$ is in $L^2(\Omega, L^2(D))$.

The task of computing a realization of white noise is therefore equivalent to computing a sample of $\dot{W}_{\mathcal{L}}$ and \dot{W}_R , and consequently of the two vectors $\mathbf{b}_{\mathcal{L}}$ and \mathbf{b}_R . As we will see, the sampling strategies for the two terms are considerably different. Nevertheless, we will explain how both terms can be sampled efficiently in linear or log-linear complexity.

Whenever the FE mesh is nonnested within the Haar mesh, our sampling algorithm requires the construction of a supermesh between the FE and Haar meshes so as to split the support of the wavelets and FE basis functions into smooth portions that can be integrated with high accuracy. Formally, this means that in the nonnested case we construct an auxiliary space $\dot{V}_h \subseteq L^2(D)$ as follows: first, we augment V_h with FE basis functions of the same type with nodal values on ∂D (we take these to be $b \in \mathbb{N}$) to obtain the intermediate space $V_h^b = \text{span}(\varphi_1, \dots, \varphi_{\bar{m}}, \varphi_{\bar{m}+1}, \dots, \varphi_{\bar{m}+b}) \subseteq H^1(D)$. Then, we define \dot{V}_h by restricting the support of the basis functions of V_h^b to the Haar cells so that $\dot{V}_h = \sum_{k=1}^{\mathcal{N}_{\mathcal{L}}} V_h^b|_{\square_k}$, where $V_h^b|_{\square_k} = \text{span}(\varphi_1|_{\square_k}, \dots, \varphi_{\bar{m}+b}|_{\square_k})$. Note that each basis function of \dot{V}_h is entirely contained within a single Haar cell by construction. In this work we assume that constants can be exactly represented by the FE space³ V_h^b (i.e., $c \in V_h^b$ for all $c \in \mathbb{R}$) so that we have $V_h, V_H \subseteq V_h$.

³This is a standard assumption that is satisfied by most FE families; see, e.g., [6].

Remark 4.1. In what follows we will denote by S_h a given supermesh between D_h and $D_{\mathcal{L}}$. Furthermore, we will indicate with $\{\phi_i\}_{i=1}^m$ ($m \in \mathbb{N}$) the basis functions of \dot{V}_h .

We can now proceed with the description of our sampling algorithms.

4.1. Sampling of $\dot{W}_{\mathcal{L}}$. We first consider the efficient sampling of $\dot{W}_{\mathcal{L}}$. Let V_H be the Haar space over the Haar mesh $D_{\mathcal{L}}$. It turns out that $\dot{W}_{\mathcal{L}} \in V_H$ a.s. and that it therefore can be expressed in terms of the basis functions of V_H as $\dot{W}_{\mathcal{L}} = \sum_{k=1}^{\mathcal{N}_{\mathcal{L}}} w_k \psi_k$, where w_k is the value of $\dot{W}_{\mathcal{L}}$ over the Haar cell \square_k . In practice, rather than computing the inner products of each Haar wavelet with the basis functions of \dot{V}_h , it is more straightforward to just compute each entry of $\mathbf{b}_{\mathcal{L}}$ as $(\mathbf{b}_{\mathcal{L}})_i = w_{\kappa(i)} \int_D \phi_i \, d\mathbf{x}$. Here $\kappa(i)$ is the index k of the Haar cell that contains the support of ϕ_i and $w_{\kappa(i)}$ must be computed from each sample of the coefficients in the expansion for $\dot{W}_{\mathcal{L}}$. Before explaining how this is actually done in practice, we prove that $\dot{W}_{\mathcal{L}}$ is the L^2 projection of white noise onto V_H and therefore $\dot{W}_{\mathcal{L}}$ does indeed belong to V_H .

LEMMA 4.3. *Let $V_H := \text{span}(\psi_1, \dots, \psi_{\mathcal{N}_{\mathcal{L}}})$ with $\psi_i := \mathbb{1}_{\square_i}$ be the space of piecewise constant functions over the cells \square_i of the Haar mesh $D_{\mathcal{L}}$. Let P_H be the L^2 projection onto V_H . Then we have that $\dot{W}_{\mathcal{L}} \equiv P_H \dot{W}$ in $L^2(\Omega, L^2(D))$.*

Proof. We note that all the Haar wavelets in the expansion for $\dot{W}_{\mathcal{L}}$ can be represented as a linear combination of basis functions of V_H . Since there are exactly as many wavelets as basis functions of V_H (see Definition 4.1) and since these wavelets are linearly independent, we conclude that the Haar wavelets form a basis of V_H . Therefore, $\dot{W}_{\mathcal{L}} \in L^2(\Omega, V_H)$ and $\langle \dot{W}_R, v \rangle = 0$ for all $v \in V_H$ (cf. (3.3)). Furthermore, for all $v \in L^2(D)$,

$$(4.5) \quad (\dot{W}_{\mathcal{L}}, v) = (\dot{W}_{\mathcal{L}}, P_H v + v^\perp) = (\dot{W}_{\mathcal{L}}, P_H v) = \langle \dot{W}, P_H v \rangle =: (P_H \dot{W}, v)$$

a.s. since $\langle \dot{W}_R, P_H v \rangle = 0$ for all $v \in L^2(D)$ (cf. (3.3)). Here we used the fact that all $v \in L^2(D)$ can be split as $v = P_H v + v^\perp$, where $v^\perp \in V_H^\perp$. \square

Note that the dimension of the space V_H is $\mathcal{N}_{\mathcal{L}} = 2^{d(\mathcal{L}+1)}$, which can easily become a large number. If we were to sample $\dot{W}_{\mathcal{L}}$ with a pure QMC approach, we would therefore need an $\mathcal{N}_{\mathcal{L}}$ -dimensional QMC point sequence, which we might not be able to sample given the restrictions of some modern QMC point generators.⁴ In the interest of reducing the QMC dimension, we reorder the terms in the expression for $\dot{W}_{\mathcal{L}}$ in (3.2) with respect to a total degree ordering rather than a full tensor grid ordering (i.e., we reorder them with respect to the 1-norm of \mathbf{l} rather than the max-norm), and we then only sample the Haar coefficients with $\|\mathbf{l}\|_1 \leq \mathcal{L}$ which are always much less. To fix the notation, for $s \in \bar{\mathbb{N}}$, we define the set

$$(4.6) \quad \mathcal{H}(s) := \{\mathbf{l} \in \bar{\mathbb{N}}^d : \|\mathbf{l}\|_1 = s, |\mathbf{l}| \leq \mathcal{L}\}, \quad \text{with} \quad \mathcal{H}(-1) = \{-\mathbf{1}\}.$$

This is the set of all Haar level vectors in the expansion for $\dot{W}_{\mathcal{L}}$ of a given total degree. After reordering the terms in the expression for $\dot{W}_{\mathcal{L}}$ in (3.2) with respect to

⁴E.g., currently there is no software support for good quality Sobol' sequences of dimension higher than 65536 [51].

the 1-norm of \mathbf{l} , we obtain

$$(4.7) \quad \dot{W}_{\mathcal{L}} = \sum_{s=-1}^{\mathcal{L}} \sum_{\mathbf{l} \in \mathcal{H}(s)} \sum_{\mathbf{n}=0}^{(2^l-1)^+} z_{\mathbf{l},\mathbf{n}}(\omega) H_{\mathbf{l},\mathbf{n}}(\mathbf{x}) + \sum_{s=\mathcal{L}+1}^{d\mathcal{L}} \sum_{\mathbf{l} \in \mathcal{H}(s)} \sum_{\mathbf{n}=0}^{(2^l-1)^+} z_{\mathbf{l},\mathbf{n}}(\omega) H_{\mathbf{l},\mathbf{n}}(\mathbf{x}),$$

where we sample the coefficients in the first sum on the right-hand side with QMC and the remaining coefficients with a standard MC approach. We therefore adopt a hybrid sampling technique for $\dot{W}_{\mathcal{L}}$.

In order to achieve good convergence with respect to the number of QMC samples, we order the QMC dimensions according to s so that the first dimension corresponds to $z_{-1,0}$, the second batch of dimensions corresponds to the terms with $s = 0$, the third batch of dimensions corresponds to the terms with $s = 1$, and so on up until $s = \mathcal{L}$. We map each sampled QMC point (in our case Sobol' with digital shifting [26]) to a Gaussian-distributed sequence by applying the inverse normal CDF. We sample the remaining coefficients independently using a pseudo-random number generator.

Remark 4.2. Note that both orderings (max- and 1-norm) are essential: the white noise expansion (3.1) must be split according to the max-norm so that $\dot{W}_{\mathcal{L}}$ can be interpreted as the projection of white noise onto V_H . In principle, the max-norm could also be used to enforce the ordering required for QMC convergence. However, this would involve sampling for an exceptionally high-dimensional QMC point sequence. Ordering the terms in the expansion for $\dot{W}_{\mathcal{L}}$ according to the 1-norm instead allows us to still enforce a good QMC ordering while reducing the dimension of the QMC point rule used.

We now propose the following algorithm for sampling $\dot{W}_{\mathcal{L}}$.

Algorithm for the sampling of $\dot{W}_{\mathcal{L}}$.

1. Compute the supermesh between the FE mesh and the Haar mesh to obtain $\dot{V}_h = \text{span}(\phi_1, \dots, \phi_m)$. Compute the scalar map $\kappa(i)$ that maps each i to the index k of the Haar cell \square_k that contains the support of ϕ_i , and compute $\int_D \phi_i \, d\mathbf{x}$ for all $i = 1, \dots, m$. This step can be done offline.
2. Sample the vector $\mathbf{z}_{\mathcal{L}} \in \mathbb{R}^{\mathcal{N}_{\mathcal{L}}}$ of the coefficients in the expression (3.2) for $\dot{W}_{\mathcal{L}}$ as $\mathbf{z}_{\mathcal{L}} = [\mathbf{z}_{\text{QMC}}^T, \mathbf{z}_{\text{MC}}^T]^T$, where \mathbf{z}_{QMC} is a randomized QMC sequence point of dimension equal to the number of coefficients with $\|\mathbf{l}\|_1 \leq \mathcal{L}$ and \mathbf{z}_{MC} is sampled with a pseudo-random number generator.
3. Loop over each Haar mesh cell \square_k and sample the values w_k of $\dot{W}_{\mathcal{L}}$ over \square_k as follows. Let $J(\mathbf{l}, \mathbf{n})$ be the index map that given (\mathbf{l}, \mathbf{n}) returns the index j such that $z_{\mathbf{l},\mathbf{n}} = (\mathbf{z}_{\mathcal{L}})_j$ (the two vectors are the same up to reordering), and define $\mathbf{m}_k \in \mathbb{R}^d$ to be the coordinate vector of the midpoint of \square_k . For each $k = 1, \dots, \mathcal{N}_{\mathcal{L}}$ and \mathbf{l} with $|\mathbf{l}| \leq \mathcal{L}$, there is only one wavelet with level vector \mathbf{l} with nonzero support over \square_k . For $i = 1, \dots, d$, its wavelet number is given by $(\bar{\mathbf{n}}_k(\mathbf{l}))_i = \lfloor (\mathbf{m}_k)_i 2^{l_i} \rfloor$ and its sign over \square_k by $\bar{s}_k(\mathbf{l}) := \prod_{i=1}^d s_k(\mathbf{l}_i)$, where the $s_k(\mathbf{l}_i)$ are the signs of the 1D Haar wavelets in the tensor product for $H_{\mathbf{l}, \bar{\mathbf{n}}_k(\mathbf{l})}$, namely

$$(4.8) \quad s_k(\mathbf{l}_i) := 1 - 2(\lfloor (\mathbf{m}_k)_i 2^{l_i+1} \rfloor \pmod{2}).$$

This expression comes from the fact that Haar wavelets are positive on even

Haar cells and negative on odd cells. We set, for all $k = 1, \dots, \mathcal{N}_{\mathcal{L}}$,

$$(4.9) \quad w_k := \sum_{\mathbf{l} \leq \mathcal{L}} \bar{s}_k(\mathbf{l}) \mathbf{z}_{J(\mathbf{l}, \bar{\mathbf{n}}_k(\mathbf{l}))} 2^{||\mathbf{l}^+||_1/2}.$$

4. For all $i = 1, \dots, m$, set $(\mathbf{b}_{\mathcal{L}})_i := w_{\kappa(i)} \int_D \phi_i \, d\mathbf{x}$ (note that here the support of each ϕ_i is entirely contained by the Haar cell $\square_{\kappa(i)}$). Assemble all these contributions into $\mathbf{b}_{\mathcal{L}}$.

Remark 4.3. In points 1 and 3 above we exploit the fact that the Haar mesh is uniform and structured. For instance, we can readily obtain the Haar mesh cell in which any point $\mathbf{p} \in D$ lies: it belongs to the $\lfloor (\mathbf{p})_i 2^{\mathcal{L}+1} \rfloor$ th Haar cell from the origin in the i th coordinate direction. The expressions for $\bar{\mathbf{n}}_k(\mathbf{l})$ and $\bar{s}_k(\mathbf{l})$ in point 3 above also derive from the same considerations.

Remark 4.4 (Complexity of the sampling of $\dot{\mathbf{W}}_{\mathcal{L}}$). Let m be the number of basis functions that span \dot{V}_h , let $\mathcal{N}_{\mathcal{L}} = 2^{d(\mathcal{L}+1)}$ be the number of cells in the Haar mesh, and let $N_{\mathcal{L}} = (\mathcal{L}+2)^d$ be the number of wavelets that are nonzero over a given Haar cell. In general, it is possible to sample $\dot{\mathbf{W}}_{\mathcal{L}}$ in $O(m + N_{\mathcal{L}} \mathcal{N}_{\mathcal{L}})$ complexity, which reduces to $O(m + \mathcal{N}_{\mathcal{L}})$ in the case in which we are using nonstandard Haar wavelets⁵ (cf. Remark 3.1). If D_h is not nested within $D_{\mathcal{L}}$ and a supermesh construction is needed, the cost complexity becomes $O(\mathcal{N}_S m_e + N_{\mathcal{L}} \mathcal{N}_{\mathcal{L}})$, where \mathcal{N}_S is the number of cells in the supermesh between D_h and $D_{\mathcal{L}}$ and m_e is the number of degrees of freedom on each supermesh cell e . Owing to Theorem 1.1 in [11], when D_h is quasi-uniform we have $\mathcal{N}_S = O(\mathcal{N}_h + \mathcal{N}_{\mathcal{L}})$, where \mathcal{N}_h is the number of cells of D_h and $a > 0$. This gives a linear cost complexity in the number of cells of D_h and a log-linear cost complexity in the number of cells of $D_{\mathcal{L}}$ since $N_{\mathcal{L}} = O((\log_2(\mathcal{N}_{\mathcal{L}})/d)^d)$. The log-term can be dropped if we use nonstandard wavelets.

4.2. Sampling of $\dot{\mathbf{W}}_R$. We now consider the efficient sampling of $\dot{\mathbf{W}}_R$. Dealing with an infinite summation is complicated. However, we can circumvent this problem by noting that the covariance of $\dot{\mathbf{W}}_R$ is known since, as $\dot{\mathbf{W}}_{\mathcal{L}}$ is independent from $\dot{\mathbf{W}}_R$ by construction, for all $u, v \in L^2(D)$ we have

$$(4.10) \quad \mathbb{E}[\langle \dot{\mathbf{W}}_R, u \rangle \langle \dot{\mathbf{W}}_R, v \rangle] = \mathbb{E}[\langle \dot{\mathbf{W}}, u \rangle \langle \dot{\mathbf{W}}, v \rangle] - \mathbb{E}[\langle \dot{\mathbf{W}}_{\mathcal{L}}, u \rangle \langle \dot{\mathbf{W}}_{\mathcal{L}}, v \rangle],$$

where the covariance of $\dot{\mathbf{W}}$ is known by Definition 2.1 and the covariance of $\dot{\mathbf{W}}_{\mathcal{L}}$ is given by the following lemma.

LEMMA 4.4. *Let \square_i for $i = 1, \dots, \mathcal{N}_{\mathcal{L}}$ be the i th cell of $D_{\mathcal{L}}$ of volume $|\square_i| = 2^{-d(\mathcal{L}+1)} = |\square_H|$ for all i (see Definition 4.1). Then, for all $u, v \in L^2(D)$,*

$$(4.11) \quad \mathcal{C}_{\mathcal{L}}(u, v) := \mathbb{E}[\langle \dot{\mathbf{W}}_{\mathcal{L}}, u \rangle \langle \dot{\mathbf{W}}_{\mathcal{L}}, v \rangle] = \sum_{i=1}^{\mathcal{N}_{\mathcal{L}}} \frac{1}{|\square_i|} \int_{\square_i} u \, d\mathbf{x} \int_{\square_i} v \, d\mathbf{x}.$$

Proof. Let P_H be the L^2 projection onto V_H ; then for all $u \in L^2(D)$ we have that $P_H u = \sum_{i=1}^{\mathcal{N}_{\mathcal{L}}} u_i \psi_i$ satisfies

$$(4.12) \quad (P_H u, v_H) = (u, v_H) \quad \forall v_H \in V_H.$$

⁵In this case it is possible to use a multidimensional generalization of the Brownian bridge construction (of which $\dot{\mathbf{W}}_{\mathcal{L}}$ is the derivative), which is well known in the computational finance literature [26].

A standard FE calculation gives that the coefficients u_i are given by

$$(4.13) \quad u_i = \frac{1}{|\square_i|} (u, \psi_i) = \frac{1}{|\square_i|} \int_{\square_i} u \, d\mathbf{x}.$$

We conclude by using Lemma 4.3 to show that, for all $u, v \in V$ such that $P_H u = \sum_{i=1}^{\mathcal{N}_{\mathcal{L}}} u_i \psi_i$ and $P_H v = \sum_{i=1}^{\mathcal{N}_{\mathcal{L}}} v_i \psi_i$,

$$(4.14) \quad \begin{aligned} \mathbb{E}[(\dot{W}_{\mathcal{L}}, u)(\dot{W}_{\mathcal{L}}, v)] &= \mathbb{E}[(\dot{W}, P_H u)(\dot{W}, P_H v)] = (P_H u, P_H v) \\ &= \sum_{i,j=1}^{\mathcal{N}_{\mathcal{L}}} u_i v_j (\psi_i, \psi_j) = \sum_{i=1}^{\mathcal{N}_{\mathcal{L}}} |\square_i| u_i v_i = \sum_{i=1}^{\mathcal{N}_{\mathcal{L}}} \frac{1}{|\square_i|} \int_{\square_i} u \, d\mathbf{x} \int_{\square_i} v \, d\mathbf{x}. \quad \square \end{aligned}$$

Remark 4.5. The sampling strategies for $\dot{W}_{\mathcal{L}}$ and \dot{W}_R presented in this work are conceptually different. In the $\dot{W}_{\mathcal{L}}$ case we use the Haar wavelet representation to make sure that the variables in the quasi-random sequence are ordered correctly. Therefore, the use of the Haar representation is crucial in the sampling of $\dot{W}_{\mathcal{L}}$. In the \dot{W}_R case, instead, the ordering is irrelevant as \dot{W}_R is sampled by using pseudo-random numbers. For this reason we can “forget” about the wavelet representation in this case and sample \dot{W}_R as it is done for any standard Gaussian field, i.e., by factorizing its covariance matrix after discretization.

It is then readily shown from Lemma 4.4 and from (4.10) that the covariance of \dot{W}_R is

$$(4.15) \quad \mathcal{C}_R(u, v) := \mathbb{E}[(\dot{W}_R, u)(\dot{W}_R, v)] = (u, v) - \sum_{i=1}^{\mathcal{N}_{\mathcal{L}}} \frac{1}{|\square_i|} \int_{\square_i} u \, d\mathbf{x} \int_{\square_i} v \, d\mathbf{x}$$

for all $u, v \in L^2(D)$. Before proceeding, we show that \mathcal{C}_R is a proper covariance function, i.e., that it is positive semidefinite.

LEMMA 4.5. *The covariance of \dot{W}_R , \mathcal{C}_R , is positive semidefinite.*

Proof. With the same notation as in the proof of Lemma 4.4, we have that, for all $u \in L^2(D)$,

$$(4.16) \quad \begin{aligned} \mathcal{C}_R(u, u) &:= \mathbb{E}[(\dot{W}_R, u)^2] = \mathbb{E}[(\dot{W} - P_H \dot{W}, u)^2] \\ &= \mathbb{E}[(\dot{W}, u - P_H u)^2] = \|u - P_H u\|_{L^2(D)}^2, \end{aligned}$$

since $\dot{W}_R = \dot{W} - \dot{W}_{\mathcal{L}} = \dot{W} - P_H \dot{W}$. Hence $\mathcal{C}_R(u, u)$ is always nonnegative, and it is zero if and only if $u \in V_H$. \square

Remark 4.6. In principle, if $D_{\mathcal{L}}$ is fine enough so that $\dot{W}_{\mathcal{L}} \approx \dot{W}$, then the correction \dot{W}_R is not needed at all. However, Haar wavelets are only piecewise constant, and we might only expect first order convergence of $\dot{W}_{\mathcal{L}}$ to \dot{W} . If so, large QMC dimensions and a very fine Haar mesh would be needed to make the correction term \dot{W}_R negligible, and this translates into very expensive samples of $\dot{W}_{\mathcal{L}}$. If we also compute samples of \dot{W}_R , however, the Haar level can be kept small.

The sampling of \dot{W}_R can be performed independently on each Haar cell. The idea is to first sample the action of \dot{W}_R onto the basis functions of \dot{V}_h . If we focus our attention only on the basis functions $\phi_1, \dots, \phi_{m_k} \in \dot{V}_h$ of support entirely contained

within a given Haar cell \square_k (i.e., the basis functions of $V_h^b|_{\square_k}$), we note that the expression (4.15) for \mathcal{C}_R simplifies to

$$(4.17) \quad \mathcal{C}_R(\phi_i, \phi_j) = (\phi_i, \phi_j) - \frac{1}{|\square_k|} \int_{\square_k} \phi_i \, d\mathbf{x} \int_{\square_k} \phi_j \, d\mathbf{x} \quad \forall i, j \in \{1, \dots, m_k\}.$$

Similarly as in [12], the sampling of \dot{W}_R over \square_k boils down to sampling a zero-mean Gaussian vector \mathbf{b}_R^k with entries $(\mathbf{b}_R^k)_i = \langle \dot{W}_R, \phi_i \rangle$ and covariance matrix C_R^k of entries $(C_R^k)_{ij}$ given by

$$(4.18) \quad \mathbf{b}_R^k \sim \mathcal{N}(0, C_R^k), \quad (C_R^k)_{ij} = \mathcal{C}_R(\phi_i, \phi_j).$$

If we let M_k be the local mass matrix over the space spanned by the $\{\phi_i\}_{i=1}^{m_k}$, with entries $(M_k)_{ij} = (\phi_i, \phi_j)$, and if we let the vector $\mathbf{I}^k \in \mathbb{R}^{m_k}$ be given by

$$(4.19) \quad \mathbf{I}^k = \left[\int_{\square_k} \phi_1 \, d\mathbf{x}, \dots, \int_{\square_k} \phi_{m_k} \, d\mathbf{x} \right]^T,$$

we can write C_R^k as

$$(4.20) \quad C_R^k = M_k - \frac{1}{|\square_k|} \mathbf{I}^k (\mathbf{I}^k)^T.$$

The sampling of a Gaussian vector with this covariance through factorization is expensive, as direct factorization of C_R^k (e.g., Cholesky) has an $O(m_k^3)$ and $O(m_k^2)$ cost and memory complexity, respectively, and it is therefore to be avoided.

We now show how \mathbf{b}_R^k can be sampled efficiently by extending the techniques presented in [12]. The main idea is to first sample a Gaussian vector with covariance M_k in linear complexity and then perform an efficient update to obtain a sample of \mathbf{b}_R^k . We can write the action of \dot{W}_R against each ϕ_i as

$$(4.21) \quad \langle \dot{W}_R, \phi_i \rangle = \langle \dot{W} - \dot{W}_{\mathcal{L}}, \phi_i \rangle = \langle \dot{W}, \phi_i \rangle - \langle \dot{W}_{\mathcal{L}}, \phi_i \rangle = (\mathbf{b}_M^k)_i - w_k (\mathbf{I}^k)_i,$$

where \mathbf{I}^k is given by (4.19), w_k is given by

$$(4.22) \quad w_k = \frac{1}{|\square_k|} \langle \dot{W}, \mathbb{1}_{\square_k} \rangle, \quad w_k \sim \mathcal{N}\left(0, \frac{1}{|\square_k|}\right),$$

and the vector $\mathbf{b}_M^k \in \mathbb{R}^{m_k}$ is given entrywise by

$$(4.23) \quad (\mathbf{b}_M^k)_i = \langle \dot{W}, \phi_i \rangle, \quad i = 1, \dots, m_k.$$

The variables w_k and \mathbf{b}_M^k are by Definition 2.1 all zero-mean joint Gaussian variables with covariance

$$(4.24) \quad \mathbb{E}[w_k w_k] = \frac{1}{|\square_k|}, \quad \mathbb{E}[\mathbf{b}_M^k w_k] = \frac{\mathbf{I}^k}{|\square_k|}, \quad \mathbb{E}[\mathbf{b}_M^k (\mathbf{b}_M^k)^T] = M_k.$$

Thanks to these relations and to (4.21), if we set

$$(4.25) \quad \mathbf{b}_R^k := \mathbf{b}_M^k - w_k \mathbf{I}^k,$$

then the covariance of \mathbf{b}_R^k is correct (cf. (4.20)) since

$$\begin{aligned} \mathbb{E}[\mathbf{b}_R^k(\mathbf{b}_R^k)^T] &= \mathbb{E}[(\mathbf{b}_M^k - w_k \mathbf{I}^k)(\mathbf{b}_M^k - w_k \mathbf{I}^k)^T] \\ &= \mathbb{E}[\mathbf{b}_M^k(\mathbf{b}_M^k)^T] - \mathbb{E}[\mathbf{b}_M^k w_k](\mathbf{I}^k)^T - \mathbf{I}^k \mathbb{E}[\mathbf{b}_M^k w_k]^T + \mathbb{E}[w_k w_k] \mathbf{I}^k (\mathbf{I}^k)^T \\ (4.26) \quad &= M_k - \frac{1}{|\square_k|} \mathbf{I}^k (\mathbf{I}^k)^T. \end{aligned}$$

In what follows, we exploit the fact that constants can be represented exactly by the FE subspace \dot{V}_h , i.e., $c \in \dot{V}_h$ for all $c \in \mathbb{R}$. Let $\phi_k = [\phi_1, \dots, \phi_{m_k}]^T$. This means that for each Haar cell \square_k there exists a vector $\mathbf{c}_k \in \mathbb{R}^{m_k}$ such that $\mathbb{1}_{\square_k} \equiv \mathbf{c}_k \cdot \phi_k$. It is then straightforward to obtain w_k from \mathbf{b}_M^k since

$$(4.27) \quad \mathbf{c}_k \cdot \mathbf{b}_M^k = \sum_{i=1}^{m_k} \langle \dot{W}, (\mathbf{c}_k)_i \phi_i \rangle = \langle \dot{W}, \mathbf{c}_k \cdot \phi_k \rangle = \langle \dot{W}, \mathbb{1}_{\square_k} \rangle = |\square_k| w_k;$$

hence $w_k = |\square_k|^{-1} \mathbf{c}_k \cdot \mathbf{b}_M^k$. Note that \mathbf{c}_k is always known; e.g., for Lagrange basis functions on simplices we have $\mathbf{c}_k = \mathbf{1} \in \mathbb{R}^{m_k}$.

We are now ready to sample \dot{W}_R from its distribution. The idea is to use the strategy presented in [12] and summarized in section 2.5 to sample the L^2 projection of white noise over \dot{V}_h , and then to orthogonalize the result against V_H . This operation can be performed Haar cell-wise as follows.

Algorithm for the efficient sampling of \dot{W}_R .

1. Loop over each Haar cell \square_k .
2. Sample a Gaussian vector $\mathbf{b}_M^k \sim \mathcal{N}(0, M_k)$ in linear cost complexity using the technique presented in section 4.1 of [12] (recall section 2.5). This operation is equivalent to sampling white noise over $V_h^b|_{\square_k}$.
3. Set $w_k := |\square_k|^{-1} \mathbf{c}_k \cdot \mathbf{b}_M^k$ and compute $\mathbf{b}_R^k := \mathbf{b}_M^k - w_k \mathbf{I}^k$. If \square_k is on the boundary of D , remove the entries of \mathbf{b}_R^k corresponding to basis functions of \dot{V}_h which do not vanish on ∂D (recall how we constructed \dot{V}_h).

Remark 4.7. This algorithm has $O(\mathcal{N}_S m_e^3)$ cost and $O(\mathcal{N}_S m_e^2)$ memory complexity, where \mathcal{N}_S is the total number of supermesh cells [12]. As discussed in Remark 4.4, \mathcal{N}_S is of $O(\mathcal{N}_h + \mathcal{N}_{\mathcal{L}})$, where \mathcal{N}_h and $\mathcal{N}_{\mathcal{L}}$ are the number of cells of D_h and of $D_{\mathcal{L}}$, respectively.

5. Sampling coupled realizations for MLQMC. We now generalize the QMC sampling algorithm just presented to the MLQMC case. Compared to standard MC, both MLMC and QMC already bring a significant computational improvement. When the two are combined into MLQMC, it is sometimes possible to obtain the best of both worlds and further improve the computational complexity and speed. However, to do so, we must be able to satisfy the requirements and assumptions underlying both QMC and MLMC: we must order the dimensions of our random input in decaying order of importance as in QMC and introduce an approximation level hierarchy and enforce a good coupling between the levels as in MLMC. We now show how this can be done with white noise sampling.

In what follows we assume we have an MLQMC hierarchy of possibly nonnested FE approximation subspaces $\{V^\ell\}_{\ell=1}^L$ over the meshes $\{D_h^\ell\}_{\ell=1}^L$ and of accuracy increasing with ℓ . Since as in the MLMC case (see [12]) the only stochastic element in (4.1) is white noise, on each MLQMC level we must be able to draw Matérn field samples $u_\ell \in V^\ell$ and $u_{\ell-1} \in V^{\ell-1}$ for $\ell > 1$ that satisfy the following variational

problems coupled by the same white noise sample: for a given $\omega_\ell^n \in \Omega$, find $u_\ell \in V^\ell$ and $u_{\ell-1} \in V^{\ell-1}$ such that

$$(5.1) \quad (u_\ell, v_\ell) + \kappa^{-2}(\nabla u_\ell, \nabla v_\ell) = \langle \dot{W}, v_\ell \rangle (\omega_\ell^n) \quad \forall v_\ell \in V^\ell,$$

$$(5.2) \quad (u_{\ell-1}, v_{\ell-1}) + \kappa^{-2}(\nabla u_{\ell-1}, \nabla v_{\ell-1}) = \langle \dot{W}, v_{\ell-1} \rangle (\omega_\ell^n) \quad \forall v_{\ell-1} \in V^{\ell-1},$$

where the terms on the right-hand side are coupled in the sense that they are centered Gaussian random variables with covariance $\mathbb{E}[\langle \dot{W}, v_t \rangle \langle \dot{W}, v_s \rangle] = (v_t, v_s)$ for $t, s \in \{\ell, \ell-1\}$, as given by Definition 2.1. Again we order the dimensions of white noise by expanding it in the Haar wavelet basis as in (3.1), but this time we allow the Haar level to possibly increase with the MLQMC level, and we split the expansion at the finer Haar level between the two MLQMC levels, \mathcal{L}_ℓ ,

$$(5.3) \quad \dot{W} = \dot{W}_{\mathcal{L}_\ell} + \dot{W}_{R_\ell},$$

where the splitting of the expansion is done in the same way as in (3.2). The choice of Haar levels \mathcal{L}_ℓ on each MLQMC level is problem-dependent. We discuss this further in section 6. From now on we assume that $\mathcal{L}_{\ell-1} \leq \mathcal{L}_\ell$, although extending the methods presented to decreasing Haar level hierarchies is straightforward. Note that the splitting of the expansion at the MLQMC level ℓ is done at Haar level \mathcal{L}_ℓ independently from the value of $\mathcal{L}_{\ell-1}$; i.e., \dot{W} on the coarser grid is always sampled with Haar level \mathcal{L}_ℓ on MLQMC level ℓ and not with Haar level $\mathcal{L}_{\ell-1}$. However, this does not affect the MLQMC telescoping sum since \dot{W} is always sampled without bias on all levels owing to the correction term \dot{W}_{R_ℓ} .

Let $\{\varphi_i^\ell\}_{i=1}^{m_\ell}$ and $\{\varphi_j^{\ell-1}\}_{j=1}^{m_{\ell-1}}$ be the basis functions spanning V^ℓ and $V^{\ell-1}$, respectively. Sampling white noise on both MLQMC levels again means to sample the vectors $\mathbf{b}_{\mathcal{L}}^\ell$, $\mathbf{b}_{\mathcal{L}}^{\ell-1}$, $\mathbf{b}_{R_\ell}^\ell$, and $\mathbf{b}_{R_\ell}^{\ell-1}$, with entries given by

$$(5.4) \quad (\mathbf{b}_{\mathcal{L}}^t)_i := \langle \dot{W}_{\mathcal{L}_\ell}, \varphi_i^t \rangle, \quad (\mathbf{b}_{R_\ell}^t)_i := \langle \dot{W}_{R_\ell}, \varphi_i^t \rangle \quad \text{for } i = 1, \dots, m_t, \quad t \in \{\ell, \ell-1\}.$$

Since we require both a multilevel coupling and a Haar wavelet expansion, this time we need to construct a *three-way* supermesh S_h between $D_{\mathcal{L}_\ell}$, D_h^ℓ , and $D_h^{\ell-1}$ (note that $D_{\mathcal{L}_{\ell-1}}$ is always nested within $D_{\mathcal{L}_\ell}$, so there is no need for a four-way supermesh). Thanks to the supermesh construction, we can split the support of all the basis functions of V^ℓ and $V^{\ell-1}$ to obtain the augmented subspaces \dot{V}^ℓ and $\dot{V}^{\ell-1}$ such that each $\phi_i^\ell \in \dot{V}^\ell$ and $\phi_j^{\ell-1} \in \dot{V}^{\ell-1}$ has support entirely contained within a single Haar cell. This is done in the same way as in the QMC case (cf. section 4) so that $V^t, D_h^t \subseteq \dot{V}^t \subseteq L^2(D)$ for $t \in \{\ell, \ell-1\}$. The sampling of \dot{W} in the MLQMC case is extremely similar to that of the QMC case with only a few differences concerning the sampling of \dot{W}_{R_ℓ} which we will now highlight.

Again, portions of \dot{W}_{R_ℓ} on separate Haar cells of $D_{\mathcal{L}_\ell}$ are independent, and we can therefore sample \dot{W}_{R_ℓ} Haar cell-wise. As in the QMC case, we first sample \dot{W}_{R_ℓ} on the augmented subspaces $\{\dot{V}^\ell\}_{\ell=1}^L$. For each Haar cell \square_k and for $t \in \{\ell, \ell-1\}$, let $\phi_1^t, \dots, \phi_{m_k^t}^t$ be the basis functions of \dot{V}^t with nonzero support over \square_k , and define the Haar cell correction vectors $\mathbf{b}_{R,k}^t$ with entries $(\mathbf{b}_{R,k}^t)_i = \langle \dot{W}, \phi_i^t \rangle$ for $i \in \{1, \dots, m_k^t\}$ and covariances given by

$$(5.5) \quad \mathbb{E}[\mathbf{b}_{R,k}^t (\mathbf{b}_{R,k}^t)^T] = M_k^t - \frac{1}{|\square_k|} \mathbf{I}_t^k (\mathbf{I}_t^k)^T, \quad \mathbb{E}[\mathbf{b}_{R,k}^\ell (\mathbf{b}_{R,k}^{\ell-1})^T] = M_k^{\ell, \ell-1} - \frac{1}{|\square_k|} \mathbf{I}_\ell^k (\mathbf{I}_{\ell-1}^k)^T,$$

where $(M_k^t)_{ij} = (\phi_i^t, \phi_j^t)$, $(M_k^{\ell, \ell-1})_{ij} = (\phi_i^\ell, \phi_j^{\ell-1})$, and $(\mathbf{I}_t^k)_i = \int_D \phi_i^t \, d\mathbf{x}$. If we define w_k as in (4.22), we can again write

$$(5.6) \quad \mathbf{b}_{R,k}^t = \mathbf{b}_{M,k}^t - w_k \mathbf{I}_t^k \quad \text{for } t \in \{\ell, \ell-1\},$$

where $\mathbf{b}_{M,k}^t \sim \mathcal{N}(0, M_k^t)$. Since constants can be represented exactly by both \dot{V}^ℓ and $\dot{V}^{\ell-1}$, i.e., for all $c \in \mathbb{R}$ and for all $t \in \{\ell, \ell-1\}$ we have that $c \in \dot{V}^t$, then there exist two vectors \mathbf{c}_k^ℓ and $\mathbf{c}_k^{\ell-1}$ such that $\mathbb{1}_{\square_k} \equiv \mathbf{c}_k^\ell \cdot \boldsymbol{\phi}_k^\ell \equiv \mathbf{c}_k^{\ell-1} \cdot \boldsymbol{\phi}_k^{\ell-1}$, where $\boldsymbol{\phi}_k^t = [\phi_1^t, \dots, \phi_{m_k^t}^t]^T$ for $t \in \{\ell, \ell-1\}$. The same argument used to derive (4.27) then gives

$$(5.7) \quad w_k = \frac{1}{|\square_k|} \mathbf{c}_k^\ell \cdot \mathbf{b}_{M,k}^\ell = \frac{1}{|\square_k|} \mathbf{c}_k^{\ell-1} \cdot \mathbf{b}_{M,k}^{\ell-1}.$$

We can now proceed with the coupled sampling of \dot{W} for MLQMC as follows.

Algorithm for the efficient sampling of \dot{W} for MLQMC.

1. Compute the three-way supermesh between the FE meshes and the Haar mesh $D_{\mathcal{L}_\ell}$, and construct \dot{V}^ℓ and $\dot{V}^{\ell-1}$. Compute the scalar maps $\kappa^t(i)$ that map each i to the index k of the Haar cell \square_k that contains the support of ϕ_i^t , and compute $\int_D \phi_i^t \, d\mathbf{x}$ for all $i = 1, \dots, m^t$ and for $t \in \{\ell, \ell-1\}$. This step can be done offline.
2. Let $\mathcal{N}_{\mathcal{L}_\ell}$ be the number of cells of $D_{\mathcal{L}_\ell}$. Sample the vector $\mathbf{z}_{\mathcal{L}_\ell} \in \mathbb{R}^{\mathcal{N}_{\mathcal{L}_\ell}}$ of the coefficients in the expression for $\dot{W}_{\mathcal{L}_\ell}$ as $\mathbf{z}_{\mathcal{L}_\ell} = [\mathbf{z}_{\text{QMC}}^T, \mathbf{z}_{\text{MC}}^T]^T$, where \mathbf{z}_{QMC} is a randomized QMC sequence point of dimension equal to the number of coefficients with $\|\mathbf{l}\|_1 \leq \mathcal{L}_\ell$ and \mathbf{z}_{MC} is sampled with a pseudo-random number generator.
3. Compute the Haar cell values \bar{w}_k of $\dot{W}_{\mathcal{L}_\ell}$ over all \square_k for $k = 1, \dots, \mathcal{N}_{\mathcal{L}_\ell}$ in the same way as in the QMC case (this step does not depend on the FE meshes).
4. Use the technique presented in section 4.2 of [12] to work supermesh cell by supermesh cell and sample in linear cost complexity the coupled Gaussian vectors $\mathbf{b}_{M,k}^\ell$ and $\mathbf{b}_{M,k}^{\ell-1}$ with covariance,

$$(5.8) \quad \mathbb{E} \left[\begin{bmatrix} \frac{\mathbf{b}_{M,k}^\ell}{\mathbf{b}_{M,k}^{\ell-1}} \end{bmatrix} \begin{bmatrix} \frac{\mathbf{b}_{M,k}^\ell}{\mathbf{b}_{M,k}^{\ell-1}} \end{bmatrix}^T \right] = \left[\frac{M_k^\ell}{(M_k^{\ell, \ell-1})^T} \mid \frac{M_k^{\ell, \ell-1}}{M_k^{\ell-1}} \right].$$

This operation is equivalent to sampling the L^2 projection of white noise over $V_{S_\ell}|_{\square_k} = \dot{V}^\ell|_{\square_k} + \dot{V}_{\ell-1}|_{\square_k}$ using the technique from section 2.5. See [12] for further details.

5. For all $t \in \{\ell, \ell-1\}$, compute $(\mathbf{b}_{\mathcal{L}_\ell}^t)_i := \bar{w}_{\kappa^t(i)} \int_D \phi_i^t \, d\mathbf{x}$ for all $i = 1, \dots, m^t$; then set $w_k := |\square_k|^{-1} \mathbf{c}_k^{\ell-1} \cdot \mathbf{b}_{M,k}^{\ell-1}$ and compute $\mathbf{b}_{R,k}^t = \mathbf{b}_{M,k}^t - w_k \mathbf{I}_t^k$. If \square_k is on the boundary of D , remove the entries of $\mathbf{b}_{R,k}^t$ corresponding to the basis functions of \dot{V}^t which do not vanish on ∂D (cf. section 4).

Remark 5.1 (Complexity of the sampling of \dot{W} for MLQMC). The overall complexity of this sampling strategy is $O(\mathcal{N}_{S_\ell} m_e + N_{\mathcal{L}_\ell} \mathcal{N}_{\mathcal{L}_\ell})$ in the standard Haar wavelet case and $O(\mathcal{N}_{S_\ell} m_e^\ell + \mathcal{N}_{\mathcal{L}_\ell})$ in the nonstandard case (cf. Remark 3.1), where (cf. Remark 4.4) \mathcal{N}_{S_ℓ} is the number of cells of the three-way supermesh on the MLQMC level ℓ , m_e^ℓ is the maximum number of degrees of freedom of V^ℓ per cell of D_h^ℓ , and $N_{\mathcal{L}_\ell}$ is the number of wavelets that have nonzero support over any of the $\mathcal{N}_{\mathcal{L}_\ell}$ cells of $D_{\mathcal{L}_\ell}$. Since $\mathcal{N}_{S_\ell} = O(\mathcal{N}_h^\ell + \mathcal{N}_{\mathcal{L}_\ell})$ (cf. Theorem 1.1 in [11]), where \mathcal{N}_h^ℓ is the number

of cells of D_h^ℓ , this gives an overall linear cost complexity in \mathcal{N}_h^ℓ and log-linear cost complexity (linear for nonstandard wavelets) in $\mathcal{N}_{\mathcal{L}_\ell}$.

Remark 5.2 (Simpler cases: Nested meshes and p -refinement). When the MLQMC mesh hierarchy is nested and/or the hierarchy is composed by taking a single mesh and increasing the polynomial degree of the FE subspaces, we have $V^{\ell-1} \subseteq V^\ell$. In this case everything we discussed still applies with the following simplifications: only a two-way supermesh between D_h^ℓ and $D_{\mathcal{L}_\ell}$ is needed on each MLQMC level in the h -refinement case. In the p -refinement case we only have one FE mesh D_h , and a single two-way supermesh construction is needed between D_h and the finest Haar mesh $D_{\mathcal{L}_L}$.

Remark 5.3 (Nonnested mesh hierarchies and embedded domains). We assume that we are given a user-provided hierarchy $\{G_h^\ell\}_{\ell=1}^L$ of possibly nonnested FE meshes of the domain G on which we need the Matérn field samples. From this we construct a boxed domain D such that $G \subset\subset D$ and a corresponding hierarchy of Haar meshes $\{D_{\mathcal{L}_\ell}\}_{\ell=1}^L$ and of FE meshes of D , $\{D_h^\ell\}_{\ell=1}^L$. As in [12], it is convenient to construct each D_h^ℓ so that G_h^ℓ is nested within it, so that each Matérn field sample can be transferred exactly and at negligible cost on the mesh on which it is needed (this is the embedded domain strategy proposed in [47]).

Remark 5.4 (Generic wavelets). It should be possible to generalize the presented sampling methods to generic orthogonal, compactly supported wavelets, although it is unclear as to whether this would bring any considerable advantage. This would likely increase the algorithm complexity, and we leave this investigation to future research.

Remark 5.5 (General domains). The same sampling method could possibly be generalized to arbitrary convex domains by introducing “generalized” Haar wavelets and meshes, obtained by partitioning D_h into $\mathcal{N}_{\mathcal{L}}$ subregions, each of which defines a Haar mesh cell. In this case the “Haar cells” would not be boxes, but polygons or polyhedra obtained by aggregating the FEM cells of the subregions, and the “Haar wavelets” would just be piecewise constant over these cells. Establishing any theoretical results in this case would be more complex, but the same algorithm should carry forward after accounting for the fact that the “Haar cells” obtained through aggregation would have variable volume. The advantage of doing this is that no supermesh would then be required in the QMC case (the Haar mesh would be nested within D_h by construction), and only one supermesh construction would be needed (between D_h^ℓ and $D_h^{\ell-1}$) in the nonnested MLQMC case. We leave the nontrivial implementation of this extension to future work.

6. Numerical results. We now test the algorithms presented. We consider problem (1.1) over $G = (-0.5, 0.5)^d$ with $F(x) = \exp(x)$ and forcing term $f = 1$; i.e., we solve

$$(6.1) \quad \begin{aligned} -\nabla \cdot (e^{u(\mathbf{x}, \omega)} \nabla p(\mathbf{x}, \omega)) &= 1, & \mathbf{x} \in G = (-0.5, 0.5)^d, & \omega \in \Omega, \\ p(\mathbf{x}, \omega) &= 0, & \mathbf{x} \in \partial G, & \omega \in \Omega, \end{aligned}$$

where $u(\mathbf{x}, \omega)$ is a Matérn field sampled by solving (4.1) over $D = (-1, 1)^d$ with $\lambda = 0.125$ and mean and standard deviation chosen so that $\mathbb{E}[e^u] = 1$, $\mathbb{V}[e^u] = 0.2$. For simplicity, we take the squared $L^2(G)$ norm of p , $P(\omega) = \|p\|_{L^2(G)}^2(\omega)$, as our output functional of interest.

Remark 6.1. We do not consider functionals of the Matérn field u , and we directly focus on the estimation of $\mathbb{E}[P]$. The reason is that in 2D and 3D the smoothness of u

is low, and we only observe standard MC convergence rates in numerical experiments (not shown).

We solve (4.1) and (6.1) with the FEniCS software package [45]. For simplicity, we consider the h -refinement case, and we discretize the equations using continuous piecewise-linear Lagrange elements. We employ the conjugate gradient routine of PETSc [2] preconditioned by the BoomerAMG algebraic multigrid algorithm from Hypre [18] to solve the linear systems arising from both equations. We declare convergence when the absolute size of the preconditioned residual norm is below a tolerance of 10^{-12} . We employ the libsupermesh software package [46] for the supermesh constructions. We use random digital shifted Sobol' sequences sampled with a custom-built⁶ Python and C wrapper of the Intel Math Kernel Library Sobol' sequence implementation augmented with Joe and Kuo's primitive polynomials and direction numbers [38] (maximum dimension = 21201). All the algorithms presented (as well as the MLMC methods from [12]) are available online within the femlmc software package.⁷

We construct the mesh hierarchies $\{G_h^\ell\}_{\ell=1}^L$ and $\{D_h^\ell\}_{\ell=1}^L$ so that, for all MLQMC levels ℓ , G_h^ℓ is nested within D_h^ℓ , yet $G_h^{\ell-1}$ and $D_h^{\ell-1}$ are not nested within G_h^ℓ and D_h^ℓ , respectively. We take all the meshes in both hierarchies to be simplicial, uniform, and structured with mesh sizes $h_\ell = 2^{-1/2} 2^{-\ell}$ in 2D and $h_\ell = \sqrt{3} 2^{-(\ell+1)}$ in 3D, although we do not exploit this structure in the implementation. The motivation behind this choice of grids is that these simpler test cases make for a cleaner asymptotic behavior, making it easier to match convergence theory with computations.⁸ For an application of the QMC sampling algorithm to more complicated grids and geometries, see [13].

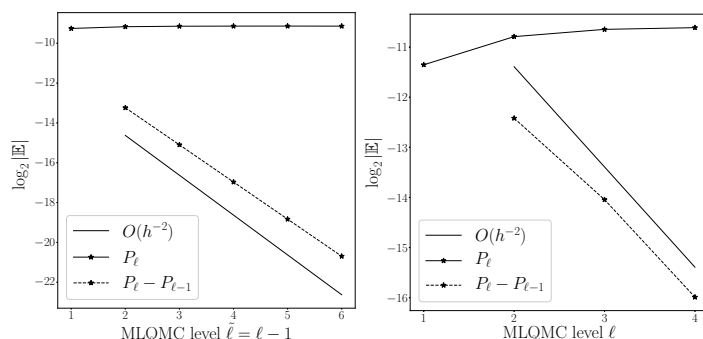


FIG. 6.1. Logarithm of the absolute value of the expected value of P_ℓ and $P_\ell - P_{\ell-1}$ as a function of the MLQMC level ℓ in 2D (left) and 3D (right). We observe a decay rate of $O(h^{-2})$ in both dimensions. These results are independent of the Haar level chosen as we always compute the exact action of white noise independently from the choice of \mathcal{L} . In 2D the coarsest mesh of the hierarchy was dropped by the MLQMC routine as it is too coarse and would not bring any significant performance advantage (the same reasoning as for MLMC; see section 2.6 in [24]).

We first study how the quantities $|\mathbb{E}[P_\ell]|$ and $|\mathbb{E}[P_\ell - P_{\ell-1}]|$ vary as the MLQMC level is increased. Assuming that u can be sampled exactly, we expect the MLMC parameter value α to be $\alpha = \min(\nu + 1, \bar{p} + 1)$, where \bar{p} is the polynomial degree [31]. Numerical results are shown in Figure 6.1, where we observe a decay rate of

⁶Available online at <https://bitbucket.org/croci/mkl.sobol/>.

⁷Available online at <https://bitbucket.org/croci/femlmc/>.

⁸The use of unstructured nonnested grids can make the convergence behavior erratic due to variation in mesh quality across the hierarchy; cf. [12].

$\alpha = 2$ in 2D and 3D. In 3D we might have expected the rate to be 1.5 due to the lack of smoothness of the coefficient u , which is only in $C^{0.5-\epsilon}(\bar{G})$ for any $\epsilon > 0$ [31]. However, at the discrete level, the FEM approximation of u is in $W^{1,\infty}(G)$ a.s. and we might be observing a preasymptotic regime.

As a next step, we analyze the convergence behavior of QMC and MLQMC with respect to the number of samples. In the supplementary material (Theorem SM2.1) we show that in the QMC case we expect an initial faster-than-MC convergence rate followed by a standard MC rate of $O(N^{-1/2})$ and that the higher the Haar level is, the later the transition between the two regimes happens. No results regarding the MLQMC case were derived, but we expect a similar behavior to occur. Furthermore, we would like to determine whether the multilevel technique can improve on QMC by bringing further variance reduction.

We draw inspiration from the original MLQMC paper by Giles and Waterhouse [25], and we study the convergence behavior of both QMC and MLQMC as the MLQMC level is increased. Results are shown in Figure 6.2. We increase the Haar level with the MLQMC level so that the Haar mesh size is always proportional to the FE mesh size, but we consider two different strategies: (1) we choose the Haar mesh size to be comparable to the FE mesh size (Figures 6.2a, 6.2c), and (2) we pick the Haar mesh size to be smaller than the FE mesh size (Figures 6.2b, 6.2d). For both scenarios, we compute the variance $\mathbb{V}_\ell = \mathbb{V}[I_{N_\ell}^{m,\ell}]$ (cf. (2.10)) of the (ML)QMC estimator on MLQMC level ℓ by using $M = 128$ ($M = 64$ in 3D) randomizations of the Sobol' sequence used, and we monitor the quantity $\log_2(N\mathbb{V}_\ell)$ as the number of samples N is increased. Various line styles are used in Figure 6.2 to indicate the different sample sizes. The horizontal lines correspond to QMC and the oblique lines to MLQMC.

For standard MC and MLMC, we have $\mathbb{V}_\ell = O(N^{-1})$, giving a $\log_2(N\mathbb{V}_\ell)$ of $O(1)$. For this reason, if we were observing an MC-like convergence rate, we would see the different lines of Figure 6.2 overlapping. The fact that this does not happen means that we are in fact observing a QMC-like rate which is faster than $O(N^{-1})$ (for the variance). However, it is clear by looking at Figures 6.2a and 6.2c that as N grows the lines get closer to each other, marking a decay to an $O(N^{-1})$ rate of convergence (for the variance) as predicted by Theorem SM2.1 (see the supplementary material in the accompanying file SMM132904.pdf [local/web 327KB]). By comparing the figures on the left-hand side to those on the right-hand side, it is also clear that increasing the Haar level delays the occurrence of this behavior both in the QMC case and in the MLQMC case. Furthermore, it appears that in the MLQMC case the convergence rate decays sooner than in the QMC case. Finally, we note that MLQMC indeed benefits from the combination of QMC and MLMC: the variance of the MLQMC estimator on any level is always smaller than the corresponding QMC estimator for the same number of samples, with large variance reductions on the fine levels.

In Figure 6.3 we take a closer look at one of the 2D examples from Figure 6.2, and we compare the observed (ML)QMC convergence rate with the rate that would be expected in a standard MC regime. We monitor the standard deviation of the (ML)QMC estimators for P_ℓ (left) and $P_\ell - P_{\ell-1}$ (right). Initially, a QMC-like convergence rate is observed for both quantities, which eventually decays to a standard MC rate. This transition occurs later when the Haar level is larger. To see this, compare the different lines (in this test case a higher (ML)QMC level corresponds to a higher Haar level) in Figure 6.3.

We now focus on the 2D case only for simplicity and see how both QMC and

MLQMC perform in practice when applied to (6.1). In Figure 6.4 we study the adaptivity and cost of (ML)QMC as the root mean square error tolerance ε is decreased. We choose Haar meshes with mesh size comparable to the FE mesh size ($|\square_{\mathcal{L}_\ell}|^{1/2} = 2^{-\ell}$), and we fix the number of randomizations to be $M = 32$. In the plot on the left we see how MLQMC automatically selects the number of samples according to the greedy strategy highlighted in the supplementary material SM1 and in [25] so as to satisfy the given error tolerance. As in the MLMC case, more samples are taken on coarse levels, and only a few are taken on the fine levels.

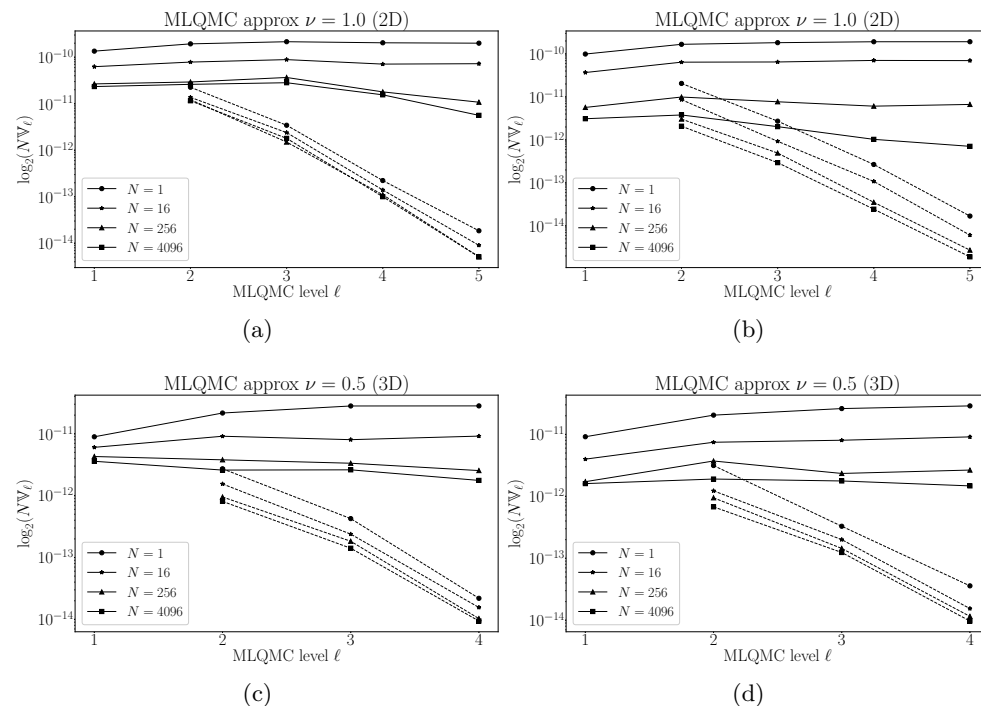


FIG. 6.2. Convergence behavior of (ML)QMC with respect to the number of samples N in 2D (a)–(b) ($M = 128$), and in 3D (c)–(d) ($M = 64$). The plots on the left are obtained by choosing Haar mesh sizes $|\square_{\mathcal{L}_\ell}|^{1/d} = 2^{-\ell}$ in 2D (a) and 3D (c). The plots on the right are obtained by choosing Haar meshes with $|\square_{\mathcal{L}_\ell}|^{1/2} = 2^{-(\ell+2)}$ in 2D (b) and $|\square_{\mathcal{L}_\ell}|^{1/3} = 2^{-(\ell+1)}$ in 3D (d). The (approximately) horizontal and oblique lines correspond to QMC and MLQMC, respectively. Different markers indicate different sample sizes. On the y-axis we monitor (the logarithm of) $N \mathbb{V}_\ell$.

In the plot on the right we show the overall cost of QMC and MLQMC as the root mean square error tolerance ε is reduced. The total cost is computed by taking the cost on each level to be $C_\ell = c_3 2^{\gamma \ell}$, where c_3 and γ are estimated using CPU timings. More specifically, we plot the quantity $\varepsilon^2 C_{\text{tot}}$, where C_{tot} is the total cost. The reason is that the total cost complexity of MLMC for this problem (MLMC parameters: $\beta = 2\alpha = 4$, $\gamma = 2$; cf. supplementary material SM1) is $O(\varepsilon^{-2})$, giving the $\varepsilon^2 C_{\text{tot}}$ factor as $O(1)$ for all ε . The fact that the MLQMC cost line is not horizontal, but decreases as ε is reduced, shows that the total complexity of MLQMC is better than ε^{-2} , i.e., that our MLQMC algorithm has a better-than-MLMC complexity. This improved complexity stems from the fact that we are observing a QMC-like convergence rate with respect to N_ℓ .

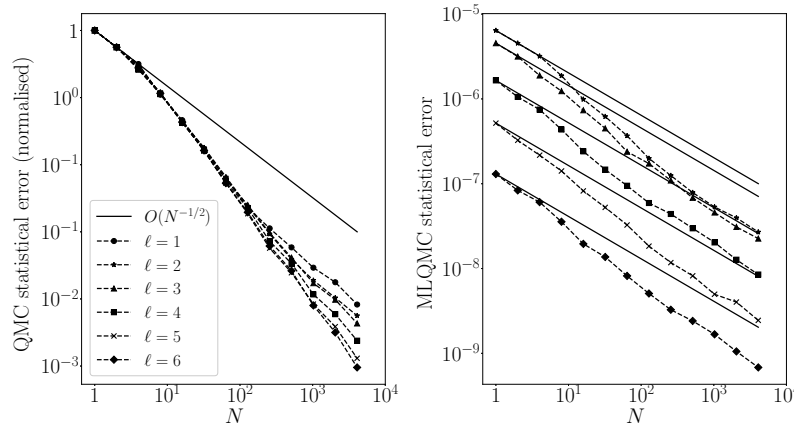


FIG. 6.3. Convergence behavior of (ML)QMC with respect to the number of samples N in 2D. The plots correspond to the same problem and (ML)QMC set-up considered in Figure 6.2b. The continuous lines correspond to the standard MC rate. In the figure on the left, we plot the standard deviation (SD) of the QMC estimator for P_ℓ as a function of the number of samples. In order to make the figures clearer, we normalize this value by dividing by the SD of the estimator when $N = 1$. The actual values can be recovered by looking at the horizontal lines in Figure 6.2b. In the figures on the right, we plot the SD of the MLQMC estimator for $P_\ell - P_{\ell-1}$ as a function of N (dashed lines) and we compare them with the theoretical standard MC convergence behavior (continuous lines). Overall, we observe a preasymptotic QMC rate that eventually decays to a $O(N^{-1/2})$ standard MC rate.

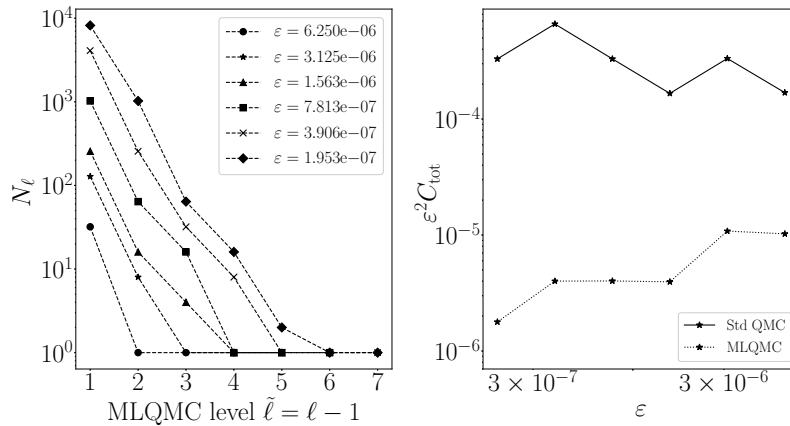


FIG. 6.4. MLQMC convergence for the solution of (6.1). Here $M = 32$ and $\mathcal{L}_\ell = 2 + \ell$. Left plot: We show how the MLQMC algorithm automatically selects the optimal number of samples N_ℓ on each level to achieve a given tolerance ε . Note that we have dropped the first mesh of the hierarchy as it is too coarse and would not bring any significant advantage to the performance of MLQMC (the same reasoning as for MLMC; see section 2.6 in [24]). We observe that on the finest levels only one sample is used, making MLQMC equivalent to plain MLMC on these levels. Right plot: We compare the efficiency of MLQMC with QMC for different tolerances. MLQMC appears to have a better-than- $O(\varepsilon^{-2})$ total cost complexity and significantly outperforms QMC.

As ε is decreased, we expect the cost complexity to decay to an ε^{-2} rate: for extremely fine tolerances, very large sample sizes are required, yielding the asymptotic $O(N^{-1/2})$ standard MC rate and harming the overall cost complexity. However, even in this case, the overall MLQMC cost benefits from the preasymptotic regime, and

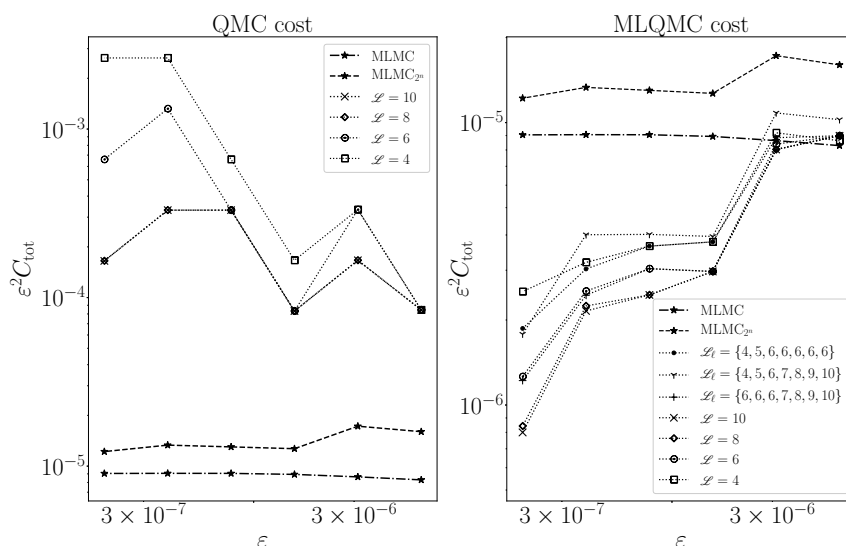


FIG. 6.5. MLMC, QMC, and MLQMC total computational cost needed for the solution of (6.1) with the same FE mesh hierarchy as in Figure 6.4. In the (ML)QMC case, we take $M = 32$ and consider different Haar level hierarchies which correspond to different computational costs. The x -axis and the MLMC lines are the same in both plots. MLQMC outperforms MLMC, which in turn outperforms QMC.

MLQMC still outperforms MLMC (see Figure 6.5). Similarly, QMC initially benefits from a faster convergence rate with respect to N . As the tolerance is decreased, the QMC rate decays to a standard MC rate, and the total cost of QMC starts increasing faster than $O(\varepsilon^{-2})$.

We investigate the performance of (ML)QMC as the Haar levels are varied in Figure 6.5, where we show the total cost of (ML)QMC for different Haar level hierarchies and we compare it with the cost of standard MLMC. The x -axis and the black lines in both plots are the same. We present the costs of two versions of MLMC: the black dash-dotted line corresponds to standard MLMC, while the black dashed line corresponds to an MLMC algorithm in which the number of samples is restricted to be in powers of 2 (this restriction also applies to the MLQMC algorithm we use [25]). MLQMC outperforms MLMC by a factor of approximately 8, depending on the Haar level choice.

Recall the convergence results with respect to the number of samples shown in Figure 6.2: even if the convergence rate decreases as N increases, it is clear from Figure 6.4 (left) that this only happens on the coarse levels where more samples are needed. Since for problem (6.1) and the FEM discretization chosen we are in the “good” case of the MLMC theorem (i.e., $\beta > \gamma$; cf. Theorem SM1.1 in the supplementary material), the multilevel cost is dominated by the sample cost on the coarse levels. We therefore expect to obtain computational gains by increasing the Haar level on the coarse MLQMC levels. At the same time, we do not expect to lose computational efficiency if we decrease the Haar level on the fine levels as these are not dominating the total cost. Note that in the QMC case there is only one level, and the only possible strategy is to keep the Haar level as high as required.

Remark 6.2. The results shown in Figures 6.4 and 6.5 do not account for differences in the cost per sample due to variations in the number of supermesh cells. If the

cost of solving the PDE with random coefficients of interest (e.g., (6.1)) dominates over the cost of sampling white noise realizations, these results are still valid as is. Otherwise, extra care must be taken when using Haar meshes which are much finer than the corresponding FE meshes as this results in a large number of supermesh cells. In the figures this would apply to Haar levels greater than $\mathcal{L}_\ell = \{4, \dots, 10\}$ (gray line in the plot on the right), and there is clearly a trade-off: larger \mathcal{L} means faster decay with respect to N , but larger costs per sample as well.

By looking at Figure 6.5 it is clear that our expectations are met. In the QMC case (plot on the left) we see that a small Haar level results in significant cost increase for small tolerances, while for large Haar levels we retain good convergence with respect to N and a cost complexity which looks just slightly worse than $O(\varepsilon^{-2})$. For this specific problem, the optimal strategy would be to increase the Haar level as the mesh is refined so that the Haar level is increased only when needed. Generally speaking, we believe that it is never advantageous to use Haar meshes much finer than the FE meshes (cf. Remark 6.2).

In the MLQMC case (plot on the right in Figure 6.5), we note that increasing the Haar level on the coarse levels indeed brings computational advantages, and decreasing it on the fine levels does not seem to affect the total cost, as predicted. For this specific problem the optimal strategy therefore consists of increasing the Haar level on the coarse levels and either keeping it constant across the MLQMC hierarchy or possibly even decreasing it. For the 2D MLQMC hierarchy, it seems that a good choice is to fix $\mathcal{L}_\ell = 6$ for all ℓ since a larger Haar level would significantly increase the number of supermesh cells (cf. Remark 6.2). A brief discussion about how the Haar levels should be chosen for more generic problems is given in Remark 6.3.

Overall, our MLQMC strategy outperforms MLMC, which in turn outperforms QMC. Standard MC is always worse than QMC, by up to two orders of magnitude for small ε (not shown).

Remark 6.3. The optimal Haar strategy is likely to change if the problem to be solved belongs to the other two cases of the MLMC theorem (Theorem SM1.1 in the supplementary material), i.e., $\beta = \gamma$ or $\beta < \gamma$. In the first case ($\beta = \gamma$), the total multilevel cost is simultaneously dominated by all levels in the multilevel hierarchy, and we expect in this case that the optimal strategy is to use a Haar mesh hierarchy of mesh sizes comparable to the FE mesh sizes (e.g., as for the gray line in the right plot of Figure 6.5). In the latter case ($\beta < \gamma$), the total multilevel cost is dominated by the fine levels. In this case it might be advantageous to keep the Haar level low on the coarse levels and to increase it on the fine levels.

7. Conclusions. We presented a novel algorithm to efficiently compute the action of white noise and sample Matérn fields within a QMC and MLQMC framework. This algorithm retains the computational efficiency of the MLMC case [12] and still enforces the required multilevel coupling in a nonnested mesh hierarchy. The numerical results show that our technique works well in practice, that the FEM convergence orders observed agree with the theory, and that MLQMC outperforms MLMC and has a better cost complexity in the preasymptotic regime. We remark that while the numerical examples considered here only employ (nonnested) structured grids, our QMC algorithm has also successfully been applied to unstructured MRI-derived brain meshes in [13]. We note that the sampling technique presented extends naturally to any application in which spatial white noise realizations are needed within an FE framework provided that the solution is smooth enough. An open problem

is the derivation of a closed-form expression for the optimal number of samples on each MLQMC level and for the optimal Haar level hierarchy, but we leave this to future research. It would also be interesting to extend the algorithms presented to general higher degree wavelets and domains (cf. Remarks 5.4 and 5.5). The first enhancement (generic wavelets) could possibly improve the convergence rate with respect to the number of samples, while the second (general domains) would reduce the supermeshing complexity and consequently the white noise sample cost.

Acknowledgments. The authors would like to acknowledge useful discussions with Marie Rognes, Robert Scheichl, Endre Süli, Abdul-Lateef Haji-Ali, Alberto Paganini, and Casper Beentjes. The authors would also like to express their thanks to James R. Maddison for his assistance with the implementation of an interface between libsupermesh and FEniCS. Finally, we are grateful to the referees for their useful suggestions and constructive input.

REFERENCES

- [1] M. BACHMAYR, I. G. GRAHAM, V. K. NGUYEN, AND R. SCHEICHL, *Unified Analysis of Periodization-Based Sampling Methods for Matérn Covariances*, preprint, <https://arxiv.org/abs/1905.13522>, 2019.
- [2] S. BALAY, S. ABHYANKAR, M. ADAMS, J. BROWN, P. R. BRUNE, K. BUSCHELMAN, V. EIJKHOUT, W. GROPP, D. KAUSHIK, M. G. KNEPLEY, ET AL., *PETSc Users Manual Revision 3.8*, Tech. Report, Argonne National Laboratory (ANL), Lemont, IL, 2017.
- [3] G. BEYLKIN, R. COIFMAN, AND V. ROKHLIN, *Wavelets in numerical analysis*, in *Wavelets and Their Applications*, Jones & Bartlett, Boston, MA, 1992, pp. 181–210.
- [4] D. BOLIN AND K. KIRCHNER, *The SPDE Approach for Gaussian Random Fields with General Smoothness*, preprint, <https://arxiv.org/abs/1711.04333v1>, 2017.
- [5] D. BOLIN, K. KIRCHNER, AND M. KOVÁCS, *Numerical solution of fractional elliptic stochastic PDEs with spatial white noise*, IMA J. Numer. Anal., 40 (2020), pp. 1051–1073.
- [6] S. BRENNER AND R. L. SCOTT, *The Mathematical Theory of Finite Element Methods*, Texts Appl. Math. 15, Springer, New York, 1994.
- [7] R. E. CAFLISH, W. MOROKOFF, AND A. OWEN, *Valuation of mortgage-backed securities using Brownian bridges to reduce effective dimension*, J. Comput. Finance, 1 (1997), pp. 27–46.
- [8] J. CHARRIER, R. SCHEICHL, AND A. L. TECKENTRUP, *Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods*, SIAM J. Numer. Anal., 51 (2013), pp. 322–352, <https://doi.org/10.1137/110853054>.
- [9] K. A. CLIFFE, M. B. GILES, R. SCHEICHL, AND A. L. TECKENTRUP, *Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients*, Comput. Vis. Sci., 14 (2011), pp. 3–15.
- [10] M. CROCI, *Multilevel Monte Carlo Methods for Uncertainty Quantification in Brain Simulations*, Ph.D. thesis, University of Oxford, Oxford, UK, 2019.
- [11] M. CROCI AND P. E. FARRELL, *Complexity bounds on supermesh construction for quasi-uniform meshes*, J. Comput. Phys., 414 (2020), pp. 1–7.
- [12] M. CROCI, M. B. GILES, M. E. ROGNES, AND P. E. FARRELL, *Efficient white noise sampling and coupling for multilevel Monte Carlo with nonnested meshes*, SIAM/ASA J. Uncertain. Quantif., 6 (2018), pp. 1630–1655, <https://doi.org/10.1137/18M1175239>.
- [13] M. CROCI, V. VINJE, AND M. E. ROGNES, *Fast uncertainty quantification of tracer distribution in the brain interstitial fluid with multilevel and quasi Monte Carlo*, Internat. J. Numer. Methods Biomed. Engrg., 37 (2021), e3412, <https://doi.org/10.1002/cnm.3412>.
- [14] I. DAUBECHIES, *Orthonormal bases of compactly supported wavelets*, Comm. Pure Appl. Math., 41 (1988), pp. 909–996.
- [15] J. DICK, F. Y. KUO, AND I. H. SLOAN, *High-dimensional integration: The quasi-Monte Carlo way*, Acta Numer., 22 (2013), pp. 133–288.
- [16] C. R. DIETRICH AND G. N. NEWSAM, *Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix*, SIAM J. Sci. Comput., 18 (1997), pp. 1088–1107, <https://doi.org/10.1137/S1064827592240555>.
- [17] J. DÖLZ, H. HARBRECHT, AND C. SCHWAB, *Covariance regularity and \mathcal{H} -matrix approximation for rough random fields*, Numer. Math., 135 (2017), pp. 1045–1071.
- [18] R. D. FALGOUT AND U. M. YANG, *hypre: A library of high performance preconditioners*, in

- International Conference on Computational Science, Springer, New York, 2002, pp. 632–641.
- [19] P. E. FARRELL, *Galerkin Projection of Discrete Fields via Supermesh Construction*, Ph.D. thesis, Imperial College London, London, 2009.
 - [20] P. E. FARRELL AND J. R. MADDISON, *Conservative interpolation between volume meshes by local Galerkin projection*, Comput. Methods Appl. Mech. Engrg., 200 (2011), pp. 89–100.
 - [21] P. E. FARRELL, M. D. PIGGOTT, C. C. PAIN, G. J. GORMAN, AND C. R. WILSON, *Conservative interpolation between unstructured meshes via supermesh construction*, Comput. Methods Appl. Mech. Engrg., 198 (2009), pp. 2632–2642.
 - [22] M. FEISCHL, F. Y. KUO, AND I. H. SLOAN, *Fast random field generation with H-matrices*, Numer. Math., 140 (2018), pp. 639–676.
 - [23] M. B. GILES, *Multilevel Monte Carlo path simulation*, Oper. Res., 56 (2008), pp. 607–617.
 - [24] M. B. GILES, *Multilevel Monte Carlo methods*, Acta Numer., 24 (2015), pp. 259–328.
 - [25] M. B. GILES AND B. J. WATERHOUSE, *Multilevel quasi-Monte Carlo path simulation*, in Advanced Financial Modelling, Radon Ser. Comput. Appl. Math. 8, De Gruyter, Berlin, 2009, pp. 165–181.
 - [26] P. GLASSERMAN, *Monte Carlo Methods in Financial Engineering*, Springer, New York, 2004.
 - [27] I. GRAHAM, F. KUO, D. NUYENS, R. SCHEICHL, AND I. SLOAN, *Quasi-Monte Carlo methods for elliptic PDEs with random coefficients and applications*, J. Comput. Phys., 230 (2011), pp. 3668–3694.
 - [28] I. G. GRAHAM, F. Y. KUO, J. A. NICHOLS, R. SCHEICHL, C. SCHWAB, AND I. H. SLOAN, *Quasi-Monte Carlo finite element methods for elliptic PDEs with log-normal random coefficients*, Numer. Math., 131 (2013), pp. 329–368.
 - [29] I. G. GRAHAM, F. Y. KUO, D. NUYENS, R. SCHEICHL, AND I. H. SLOAN, *Analysis of circulant embedding methods for sampling stationary random fields*, SIAM J. Numer. Anal., 56 (2018), pp. 1871–1895, <https://doi.org/10.1137/17M1149730>.
 - [30] I. G. GRAHAM, F. Y. KUO, D. NUYENS, R. SCHEICHL, AND I. H. SLOAN, *Circulant embedding with QMC: Analysis for elliptic PDE with lognormal coefficients*, Numer. Math., 140 (2018), pp. 479–511, <https://doi.org/10.1007/s00211-018-0968-0>.
 - [31] W. HACKBUSCH, *Elliptic Differential Equations: Theory and Numerical Treatment*, Springer-Verlag, New York, 1992.
 - [32] W. HACKBUSCH, *Hierarchical Matrices: Algorithms and Analysis*, Springer, New York, 2015.
 - [33] S. HEINRICH, *Multilevel Monte Carlo methods*, in Large-Scale Scientific Computing, Springer, Berlin, Heidelberg, 2001, pp. 58–67.
 - [34] L. HERRMANN AND C. SCHWAB, *Multilevel quasi-Monte Carlo integration with product weights for elliptic PDEs with lognormal coefficients*, ESAIM Math. Model. Numer. Anal., 53 (2019), pp. 1507–1552.
 - [35] T. HIDA, H.-H. KUO, J. POTTHOFF, AND L. STREIT, *White Noise: An Infinite Dimensional Calculus*, Springer, Dordrecht, The Netherlands, 1993.
 - [36] H. INABA AND B. D. TAPLEY, *Generalized random processes: A theory and the white Gaussian process*, SIAM J. Control, 13 (1975), pp. 719–735, <https://doi.org/10.1137/0313041>.
 - [37] K. ITÔ, *Stationary random distributions*, Mem. College Sci. Univ. Kyoto Ser. A Math., 28 (1954), pp. 209–223.
 - [38] S. JOE AND F. Y. KUO, *Constructing Sobol' sequences with better two-dimensional projections*, SIAM J. Sci. Comput., 30 (2008), pp. 2635–2654, <https://doi.org/10.1137/070709359>.
 - [39] B. N. KHOROMSKIJ, A. LITVINENKO, AND H. G. MATTHIES, *Application of hierarchical matrices for computing the Karhunen-Loève expansion*, Computing, 84 (2009), pp. 49–67.
 - [40] U. KHRISTENKO, L. SCARABOSIO, P. SWIERCZYNSKI, E. ULLMANN, AND B. WOHLMUTH, *Analysis of boundary effects on PDE-based sampling of Whittle-Matérn random fields*, SIAM/ASA J. Uncertain. Quantif., 7 (2019), pp. 948–974, <https://doi.org/10.1137/18M1215700>.
 - [41] F. Y. KUO, R. SCHEICHL, C. SCHWAB, I. H. SLOAN, AND E. ULLMANN, *Multilevel quasi-Monte Carlo methods for lognormal diffusion problems*, Math. Comp., 86 (2017), pp. 2827–2860.
 - [42] F. Y. KUO, C. SCHWAB, AND I. H. SLOAN, *Multi-level quasi-Monte Carlo finite element methods for a class of elliptic PDEs with random coefficients*, Found. Comput. Math., 15 (2015), pp. 411–449.
 - [43] C. LEMIEUX, *Monte Carlo and Quasi-Monte Carlo Sampling*, Springer Ser. Statist., Springer-Verlag, New York, 2009.
 - [44] F. LINDGREN, H. RUE, AND J. LINDSTRÖM, *An explicit link between Gaussian fields and Gaussian Markov random fields: The stochastic partial differential equation approach*, J. R. Stat. Soc. Ser. B Stat. Methodol., 73 (2011), pp. 423–498.
 - [45] A. LOGG, K.-A. MARDAL, AND G. WELLS, *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*, Lect. Notes Comput. Sci. Engrg. 84, Springer,

- Berlin, 2012.
- [46] J. R. MADDISON, P. E. FARRELL, AND I. PANOURGIAS, *Parallel Supermeshing for Multimesh Modelling*, Tech. Report eCSE03-08, ARCHER, 2013.
 - [47] S. OSBORN, P. S. VASSILEVSKI, AND U. VILLA, *A Multilevel, Hierarchical Sampling Technique for Spatially Correlated Random Fields*, preprint, <https://arxiv.org/abs/1703.08498>, 2017.
 - [48] S. OSBORN, P. ZULIAN, T. BENSON, U. VILLA, R. KRAUSE, AND P. S. VASSILEVSKI, *Scalable hierarchical PDE sampler for generating spatially correlated random fields using non-matching meshes*, Numer. Linear Algebra Appl., 25 (2018), e2146, <https://doi.org/10.1002/nla.2146>.
 - [49] A. B. OWEN, *Variance and discrepancy with alternative scramblings*, ACM Trans. Model. Comput. Simul., 13 (2003), pp. 363–378.
 - [50] R. POTSEPAEV AND C. L. FARMER, *Application of stochastic partial differential equations to reservoir property modelling*, in Proceedings of the ECMOR XII-12th European Conference on the Mathematics of Oil Recovery, Vol. 2, 2014.
 - [51] I. M. SOBOLOV, D. ASOTSKY, A. KREININ, AND S. KUCHERENKO, *Construction and comparison of high-dimensional Sobol' generators*, Wilmott, 56 (2011), pp. 64–79.
 - [52] T. J. SULLIVAN, *Introduction to Uncertainty Quantification*, Texts Appl. Math. 63, Springer, Cham, 2015.
 - [53] A. L. TECKENTRUP, R. SCHEICHL, M. B. GILES, AND E. ULLMANN, *Further analysis of multi-level Monte Carlo methods for elliptic PDEs with random coefficients*, Numer. Math., 125 (2013), pp. 569–600.
 - [54] A. J. WATHEN, *Realistic eigenvalue bounds for the Galerkin mass matrix*, IMA J. Numer. Anal., 7 (1987), pp. 449–457.
 - [55] P. WHITTLE, *On stationary processes in the plane*, Biometrika, 41 (1954), pp. 434–449.
 - [56] A. T. A. WOOD AND G. CHAN, *Simulation of stationary Gaussian processes in $[0, 1]^d$* , J. Comput. Graph. Statist., 3 (1994), pp. 409–432.