Oden Institute REPORT 21-01

January 2021

A comparison of matrix-free isogeometric Galerkin and collocation methods for Karhunen–Loève expansion

by

Michal L. Mika, René R. Hiemstra, Thomas J.R. Hughes, and Dominik Schillinger

Oden Institute for Computational Engineering and Sciences
The University of Texas at Austin
Austin, Texas 78712

Reference: Michal L. Mika, René R. Hiemstra, Thomas J.R. Hughes, and Dominik Schillinger, "A comparison of matrix-free isogeometric Galerkin and collocation methods for Karhunen–Loève expansion ," Oden Institute REPORT 21-01, Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, January 2021.

Noname manuscript No.
(will be inserted by the editor)

A comparison of matrix-free isogeometric Galerkin and collocation methods for Karhunen–Loève expansion

Michal L. Mika · René R. Hiemstra · Thomas J.R. Hughes · Dominik Schillinger

Received: date / Accepted: date

Abstract Numerical computation of the Karhunen–Loève expansion is computationally challenging in terms of both memory requirements and computing time. We compare two state-of-the-art methods that claim to efficiently solve for the K–L expansion: (1) the matrix-free isogeometric Galerkin method using interpolation based quadrature proposed by the authors in [1] and (2) our new matrix-free implementation of the isogeometric collocation method proposed in [2]. Two three-dimensional benchmark problems indicate that the Galerkin method performs significantly better for smooth covariance kernels, while the collocation method performs slightly better for rough covariance kernels.

Keywords Karhunen–Loève expansion · Galerkin · collocation · matrix-free · isogeometric analysis

1 Introduction

It was at the Institute of Mechanics and Computational Mechanics (IBNM) in Hannover, where Peter Wriggers started his academic career. In 1998, he returned to IBNM, directing the institute until 2008. Michal Mika, René Hiemstra and Dominik Schillinger are part of the young generation of researchers in computational engineering and sciences at IBNM. Schillinger arrived in 2019 as a junior faculty member in computational mechanics and scientific computing. His vision for the future is borne by the legacy of excellent research, teaching, mentorship and service that Peter Wriggers and the many colleagues he attracted to Hannover have built and maintained over the past 25 years. Mika is one of the many outstanding graduates of the educational programs in computational methods in engineering initiated by Peter Wriggers in Hannover. It was also Wriggers who established the contact to Tom Hughes at UT Austin, when Mika was looking for an international partner to work with on his M.Sc. thesis. It was there, where the foundations for this work were laid.

The Karhunen–Loève (K–L) expansion decomposes a random field into an infinite linear combination of L^2 orthogonal functions with decreasing energy content. Truncated representations have applications in stochastic finite element analysis (SFEM) [3,4,5], proper orthogonal decomposition (POD) [6,7] and in image processing where the technique is known as principal component analysis (PCA) [8]. All these techniques are closely related and widely used in practice [9].

Numerical approximation of the K–L expansion by means of the Galerkin or collocation method leads to a generalized eigenvalue problem: *Find* $(\mathsf{v}_k^h, \lambda_k^h) \in \mathbb{R}^N \times \mathbb{R}^+$ *such that*

$$\mathsf{A}\mathsf{v}^h = \lambda_k^h \mathsf{Z}\mathsf{v}^h \qquad \text{for} \qquad k = 1, 2, \dots, M. \tag{1}$$

This matrix problem is computationally challenging for the following reasons: (1) the matrix A is dense and thus memory intensive to store explicitly; (2) every iteration

Michal L. Mika
E-mail: mika@ibnm.uni-hannover.de

René R. Hiemstra
E-mail: rene.hiemstra@ibnm.uni-hannover.de

Dominik Schillinger
E-mail: schillinger@ibnm.uni-hannover.de
Institute of Mechanics and Computational Mechanics
Leibniz University Hannover
Appelstr. 9a, 30167 Hannover, Germany

Thomas J.R. Hughes
E-mail: hughes@oden.utexas.edu
Oden Institute for Computational Engineering and Science
The University of Texas at Austin
201 East 24th Street, C0200, Austin, TX 78712-1229 USA

of an iterative eigenvalue solver requires a backsolve of a factorization of Z; and (3) the assembly of A is computationally expensive[1].

In this paper, we investigate and compare two state-of-the-art methods that were recently proposed to efficiently solve for the K–L expansion. The first method is the matrix-free isogeometric Galerkin method proposed by the authors in [1], which uses an advanced quadrature technique to gain high performance that is scalable with polynomial order. The second method is our new matrix-free implementation of the isogeometric collocation method proposed in [2]. As a collocation method it requires far fewer quadrature points than a standard Galerkin method such that the assembly of the collocation equations is simple and efficient.

This paper is structured as follows. In Section 2, we briefly review the basic aspects of the K–L expansion in the context of random field represenations. In Section 3, we concisely present the two matrix-free solution methods and assess their algorithmic complexity. Three-dimensional numerical benchmark problems with comparisons in terms of accuracy and solution time are provided in Section 4. We summarize our conclusions in Section 5 and discuss future work.

2 Karhunen–Loève expansion of random fields

Consider a complete probability space $(\Theta, \Sigma, \mathbb{P})$ where Θ denotes a sample set of random events and $\mathbb{P}$ is a probability measure $\mathbb{P} : \Sigma \to [0,1]$. Let $\alpha(\cdot,\theta) : \Theta \mapsto L^2(\mathcal{D})$ denote a random field on a bounded domain $\mathcal{D} \in \mathbb{R}^d$ with mean $\mu(x) \in L^2(\mathcal{D})$ and covariance function $\Gamma(x,x') \in L^2(\mathcal{D}\times\mathcal{D})$. The K–L expansion of the random field $\alpha(\cdot,\theta)$ requires the solution of an integral eigenvalue problem. Consider the self-adjoint positive semi-definite linear operator $T : L^2(\mathcal{D}) \mapsto L^2(\mathcal{D})$,

$$(T\phi)(x) := \int_{\mathcal{D}} \Gamma(x,x')\phi(x')\,\mathrm{d}x'. \tag{2}$$

The eigenfunctions $\{\phi_i\}_{i\in\mathbb{N}}$ of T are defined by the homogeneous Fredholm integral eigenvalue problem of the second kind,

$$T\phi_i = \lambda_i\phi_i, \quad \phi_i \in L^2(\mathcal{D}) \text{ for } i \in \mathbb{N}. \tag{3}$$

The eigenfunctions ϕ_i are orthonormal in $L^2(\mathcal{D})$ and the corresponding eigenvalues form a non-increasing sequence $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$. The K–L expansion of the random field $\alpha(\cdot,\theta)$ is given as

$$\alpha(x,\theta) = \mu(x) + \sum_{i=1}^{\infty} \sqrt{\lambda_i}\phi_i(x)\xi_i(\theta) \tag{4}$$

where

$$\xi_i(\theta) := \frac{1}{\sqrt{\lambda_i}} \int_{\mathcal{D}} (\alpha(x,\theta) - \mu(x))\,\phi_i(x)\,\mathrm{d}x. \tag{5}$$

Truncating the series in (4) after M terms leads to an approximation of α denoted by α_M. For practical computations in the context of stochastic finite element methods [3,4,5], the truncation order M is typically chosen between 20 and 30 terms [10,4]. Each term in the expansion introduces one stochastic dimension, which is an example for the *curse of dimensionality*.

3 Numerical methods

In this section we briefly review the matrix-free Galerkin method proposed in [1] and introduce our matrix-free implementation of the isogeometric collocation method proposed in [2]. We include an analysis of the algorithmic complexity in terms of the polynomial degree p and number of elements N_e of the d-dimensional spatial domain $\mathcal{D}$.

In both approaches the generalized algebraic eigenvalue problem is first reformulated as a standard algebraic eigenvalue problem using standard linear algebra techniques [11]: *Find* $(\mathsf{v}_k^h, \lambda_k^h) \in \mathbb{R}^N \times \mathbb{R}^+$ *s.t.*

$$\begin{cases} \mathsf{A}'\mathsf{v}'_k = \lambda_k^h \mathsf{v}'_k \\ \mathsf{v}_k^h = \mathsf{C}\mathsf{v}'_k \end{cases} \qquad \text{for} \quad k = 1,2,\ldots,M. \tag{6}$$

Here C is an invertible mapping that depends on Z and A' can be written in terms of A and Z.

3.1 Matrix-free isogeometric Galerkin method

A variational treatment of (3) leads to the following problem: *Find* $(\phi,\lambda) \in L^2(\mathcal{D}) \times \mathbb{R}^+$ *s.t.* $\forall \psi \in L^2(\mathcal{D})$

$$\int_{\mathcal{D}} \left(\int_{\mathcal{D}} \Gamma(x,x')\phi(x')\,\mathrm{d}x' - \lambda\phi(x) \right) \psi(x)\,\mathrm{d}x = 0. \tag{7}$$

From equation (7), the Galerkin method is obtained by replacing $\phi, \psi \in L^2(\mathcal{D})$ by finite dimensional representations $\phi^h, \psi^h \in \mathcal{S}^h \subset L^2(\mathcal{D})$. Being posed in the variational setting, Galerkin methods inherit several advantageous properties such as exact L^2 orthogonality of the numerical eigenvectors and monotonic convergence

[1] Formation and assembly costs for a standard Galerkin method scale $\mathcal{O}(N_e^2 \cdot (p+1)^{3d}))$, where N_e is the number of finite elements, p is the polynomial degree and d is the spatial dimension.

of the numerical eigenvalues [12,13]. Furthermore, powerful tools exist in the variational setting to study the stability and convergence of the method[2].

With a trial space $\mathcal{S}^h := \operatorname{span}\{N_i(x)\}_{i=1,\dots,N}$ the Galerkin method leads to the eigenvalue problem defined in (1) with the system matrices

$$A_{ij} := \int_{\mathcal{D}} N_i(x) \int_{\mathcal{D}} \Gamma(x,x') N_j(x') \,\mathrm{d}x' \,\mathrm{d}x \tag{8a}$$

$$Z_{ij} := \int_{\mathcal{D}} N_i(x) N_j(x) \,\mathrm{d}x \tag{8b}$$

Alternatively, the eigenvalue problem can be solved in the standard form introduced in equation (6) where $\mathsf{A}' := \mathsf{L}^{-1}\mathsf{A}\mathsf{L}^{-\top}$ and $\mathsf{C} := \mathsf{L}^{-\top}$. The matrix L is defined by the lower triangular matrix in the Cholesky decomposition of $\mathsf{Z} = \mathsf{L}\mathsf{L}^{\top}$.

Typically, the space $\mathcal{S}^h$ is spanned by piecewise C^0-continuous polynomial functions on quadrilateral, hexagonal or simplicial elements [13]. Recently, non-uniform rational B-splines (NURBS) have been applied in the context of an isogeometric Galerkin method [14]. These methods commonly evaluate the integrals in (8) using standard numerical quadrature rules. A Gauss–Legendre numerical quadrature rule leads, however, to an algorithmic complexity of $\mathcal{O}(N_e^2 \cdot (p+1)^{3d})$ [1], which becomes excessively expensive with the number of elements N_e, polynomial degree p and spatial dimension d. Furthermore, as mentioned in the introduction, the matrix A is dense and requires $\mathcal{O}(8 \cdot N^2)$ bytes of storage in double precision arithmetic, where N is the number of degrees of freedom in the trial space.

To overcome these limitations, the matrix-free Galerkin method proposed in [1] avoids storing the main system matrix A and achieves computational efficiency by utilizing a non-standard trial space in combination with a specialized quadrature technique, called *interpolation based quadrature*. This approach requires a minimum number of quadrature points and enables application of global sum factorization techniques [15]. We sketch the main ideas of the method and refer to [1] for further details.

Let $\{B_i(\hat{x})\}_{i=1,\dots,N}$ and $\{\tilde{B}_j(\hat{x})\}_{j=1,\dots,\tilde{N}}$ denote two sets of tensor product B-splines of, for simplicity, uniform polynomial degree p. The first set is used in the definition of the trial space, whereas the second set is used in a projection of the kernel $\Gamma(x,x')$ and is a part of the *interpolation based quadrature*. Let $F : \hat{\mathcal{D}} \to \mathcal{D}$ be the geometric mapping from the reference domain to the physical domain. The trial space is defined as

$$\mathcal{S}^h := \operatorname{span}\left\{ B_i(\hat{x}) / \sqrt{\det \mathrm{D}F(\hat{x})} \right\}_{i=1,\dots,N.} \tag{9}$$

[2] In general the stability and convergence analysis are challenging in the context of collocation methods.

The advantage of this particular choice of the trial space is that the mass matrix in (8b) has a *Kronecker* structure and can be factored as $\mathsf{Z} = \mathsf{Z}_d \otimes \cdots \otimes \mathsf{Z}_2 \otimes \mathsf{Z}_1$, where $\{\mathsf{Z}_k\}_{k=1,2,\dots,d}$ are univariate mass matrices. By leveraging this factorization the matrix-vector products of Kronecker matrices can be evaluated in nearly linear time complexity. This also holds for the matrix L in the Cholesky factorization of Z, which is factored as $\mathsf{L} = \mathsf{L}_d \otimes \cdots \otimes \mathsf{L}_2 \otimes \mathsf{L}_1$ from which the respective inverse follows as $\mathsf{L}^{-1} = \mathsf{L}_d^{-1} \otimes \cdots \otimes \mathsf{L}_2^{-1} \otimes \mathsf{L}_1^{-1}$.

The interpolation based quadrature in combination with the choice of the trial space in (9) leads to a factorization of the matrix A as

$$\mathsf{A} = \mathsf{M}^{\top} \tilde{\mathsf{B}}^{-1} \mathsf{J} \Gamma \mathsf{J} \tilde{\mathsf{B}}^{-\top} \mathsf{M}. \tag{10}$$

Here $\Gamma := \Gamma(x_i, x_j) \in \mathbb{R}^{\tilde{N} \times \tilde{N}}$ is the covariance kernel evaluated at the Greville abscissae, $\mathsf{J} \in \mathbb{R}^{\tilde{N} \times \tilde{N}}$ is the square root of a diagonal matrix of determinants of the Jacobian of the mapping at these points and the matrices $\tilde{\mathsf{B}} = \tilde{\mathsf{B}}_d \otimes \cdots \otimes \tilde{\mathsf{B}}_2 \otimes \tilde{\mathsf{B}}_1 \in \mathbb{R}^{\tilde{N} \times \tilde{N}}$ and $\mathsf{M} = \mathsf{M}_d \otimes \cdots \mathsf{M}_2 \otimes \mathsf{M}_1 \in \mathbb{R}^{\tilde{N} \times N}$ are Kronecker product matrices. In fact $\tilde{\mathsf{B}}_k$ and M_k, $k = 1, 2, \dots, d$, are univariate collocation and mass matrices, respectively, which are introduced by the interpolation based quadrature. The computation of the eigenvalues and eigenvectors requires evaluation of matrix-vector products $\mathsf{v}' \mapsto \mathsf{A}'\mathsf{v}'$. This leads to a nine step algorithm presented in [1]. The matrix-vector products with the Kronecker structured matrices $\mathsf{L}^{-\top}$, M, $\mathsf{B}^{-\top}$ and the diagonal matrix J as well as all the respective transpose operations are performed in linear or nearly linear time complexity. The matrix-vector products with the matrix Γ are performed in quadratic time complexity. Hence, our matrix-free algorithm scales quadratically with the dimension of the interpolation space $\tilde{N}$. We note that in this algorithm, the matrix rows of Γ are computed on the fly, which saves memory by not explicitly storing the dense matrix Γ. Memory requirements for the remaining matrices are negligible, since they are either diagonal or Kronecker product matrices. For additional details about the matrix-free method, interpolation based quadrature and Kronecker products, we refer to [1].

3.2 Matrix-free isogeometric collocation method

In contrast to a Galerkin method, a collocation method does not treat the integral equation (3) in a variational manner. Instead, we require the discretized residual

$$r^h(x) := \int_{\mathcal{D}} \Gamma(x,x') \phi^h(x') \,\mathrm{d}x' - \lambda^h \phi^h(x) \tag{11}$$

to vanish at distinct points $x \in \mathcal{D}$. In [2], the geometry and trial spaces are discretized in terms of NURBS basis functions

$$\mathcal{S}^h := \{R_i(x)\}_{i=1,\dots,N} \tag{12}$$

in the sense of the isoparametric approach of isogeometric analysis. In this study, we choose to collocate (11) at the Greville abscissae $\{x_i\}_{i=1,\dots,N}$. The method is expressed concisely in matrix form (1) where the corresponding system matrices are given by

$$A_{ij} := \int_{\mathcal{D}} \Gamma(x_i, x') R_j(x') \,\mathrm{d}x', \qquad Z_{ij} := R_j(x_i). \tag{13}$$

In primal form (6), this means that $\mathsf{A}' = \mathsf{Z}^{-1}\mathsf{A}$ and C is the identity matrix. The matrices A and Z are square and, in general, not symmetric. In contrast to variational methods, where the system matrices are symmetric and positive (semi)-definite by construction, collocation methods do not ensure a real-valued eigensolution for any element size $h > 0$. For an in-depth exposition of the collocation method, we refer the reader to [12], and to [16,17] for details on the isogeometric formulation.

The matrix-free version of the collocation method is derived analogously to the matrix-free Galerkin method described above. Due to the properties of the system matrix Z, instead of the Cholesky decomposition employed in the Galerkin method, we use the *pivoted* LU decomposition, $\mathsf{PZQ} = \mathsf{LU}$, to arrive at the standard matrix form. We observed that without pivoting the matrix-free collocation method suffers from numerical instabilities at polynomial orders $p > 3$. We use the pivoted LU decomposition of Z to apply the inverse of Z to the matrix A and thus obtain A'. The standard algebraic eigenvalue problem is then given by

$$\mathsf{A}'\mathsf{v}' = \lambda \mathsf{v}' \quad \text{where} \quad \mathsf{A}' := \mathsf{Q}\mathsf{U}^{-1}\mathsf{L}^{-1}\mathsf{P}\mathsf{A} \tag{14}$$

Following [1], we choose a row-wise evaluation of the coefficient vector in the standard matrix-vector product $\mathsf{v}' \mapsto \mathsf{A}'\mathsf{v}'$. The optimal evaluation order and further details for each step are given in Algorithm 1.

Algorithm 1 Matrix-free evaluation of the matrix-vector product $\mathsf{v}' \mapsto \mathsf{A}'\mathsf{v}'$ emerging from collocation

Input: $v_j \in \mathbb{R}^N$, $R_{jk} \in \mathbb{R}^{N \times (N_e \cdot N_q)}$, P_{ij}, Q_{ij}, U_{ij}, $L_{ij} \in \mathbb{R}^{N \times N}$, $J_k \in \mathbb{R}^{N_e \cdot N_q}$, $W_k \in \mathbb{R}^{N_e \cdot N_q}$
Output: $v'_i \in \mathbb{R}^N$

1: $y_k \leftarrow R_{jk} v_j$ ▷ Interpolation at quadrature points
2: $y'_k \leftarrow y_k \odot J_k \odot W_k$ ▷ Scaling at quadrature points
3: $z_l \leftarrow \Gamma_{lk} y'_k$ ▷ Kernel evaluation one row at a time
4: $v'_i \leftarrow Q_{it} U^{-1}_{tr} L^{-1}_{rs} P_{sl} z_l$ ▷ Backsolve using LU of Z

3.3 Algorithmic complexity

Matrix-free Galerkin method Under the assumption of $\tilde{N} \propto N$, the formation and assembly costs are negligible compared to the matrix-vector products that scale independently of p as $\mathcal{O}(\tilde{N}^2)$ [1]. The total cost of the method scales as $\mathcal{O}(N_{\text{iter}} \cdot \tilde{N}^2)$, where N_{iter} is the number of iterations of the eigenvalue solver.

Matrix-free collocation method We are interested in the algorithmic complexity of an element-wise assembly procedure for the system matrices that arise from the collocation method. We assume that (1) $\hat{\mathcal{D}}$ has N_e elements; (2) the products on every d-dimensional element $\square^d$ in $\hat{\mathcal{D}}$ are integrated with a quadrature rule $Q(f) := \sum_{k=1}^{N_q} w_k f(x_k)$ with $1 \le N_q \le (p+1)^d$ quadrature points; and (3) the number of collocation points N_c is equal to the number of degrees of freedom N. The leading term in the total cost of formation and assembly arises from the cost of forming the element matrices,

$$\begin{aligned} A^e_{ij} &= \int_{\square^d} \Gamma(\hat{x}_i, \hat{x}') R_j(\hat{x}') \,\mathrm{d}\hat{x}' \\ &\approx \sum_{k=1}^{N_q} w_k \Gamma(\hat{x}_i, \hat{x}'_k) B_j(\hat{x}'_k) = \mathsf{C}_{ik}\mathsf{D}_{kj} \\ &\quad \text{where} \quad C_{ik} = w_k \Gamma(\hat{x}_i, \hat{x}'_k) \quad \text{and} \quad D_{kj} = R_j(\hat{x}'_k) \end{aligned}$$

with $i = 1, \dots, N$ and $j = 1, \dots, (p+1)^d$. The formation cost of C and D is negligible. The matrix-matrix product cost is of $\mathcal{O}\left(N_c N_q (p+1)^d\right)$ and the cost for summation over all N_e is of $\mathcal{O}\left(N_e N_c N_q (p+1)^d\right)$. Now, assuming a Gauss–Legendre quadrature rule with $N_q := (p+1)^d$ quadrature points and the proportionality relationship $N_e \propto N$, a collocation method with $N_c = N$ has a leading cost of $\mathcal{O}\left(N^2 (p+1)^{2d}\right)$.

The algorithmic complexity in the matrix-free formulation is driven by the most expensive steps in Algorithm 1. In a single iteration of the eigenvalue solver, steps 1 and 3 have a complexity $\mathcal{O}(N \cdot N_e \cdot N_q)$. The element-wise multiplication in step 2 scales linearly with the number of quadrature points, $\mathcal{O}(N_e \cdot N_q)$. The last step scales as $\mathcal{O}(N^2)$. Evidently, steps 1 and 3 depend on the number of quadrature points. Since $N_e \cdot N_q \ge N$, they determine the overall cost of the method. Assuming a Gauss-Legendre quadrature rule with $N_q := (p+1)^d$ quadrature points in each element and $N_e \propto N$, the leading cost of a single iteration of the eigenvalue solver is $\mathcal{O}(N^2 (p+1)^d)$. Hence, the total cost of the matrix-free isogeometric collocation method scales as $\mathcal{O}(N_{\text{iter}} \cdot N^2 (p+1)^d)$, where N_{iter} is the number of iterations of the eigenvalue solver.

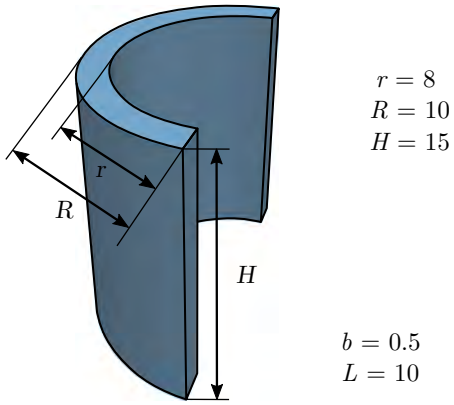

Fig. 1: Benchmark geometry of a half-cylinder. The correlation length $bL = 5$ is used throughout all cases.

Comparison Compared to the matrix-free Galerkin method with interpolation based quadrature, the collocation method scales unfavourably with the polynomial degree. Furthermore, due to the lack of Kronecker structure, it is necessary to compute the pivoted LU decomposition of the full matrix Z. The computational cost of this factorization increases with N as well as p, which is due to an increasing bandwidth of the matrix Z.

Remark 1 If the trial space in the collocation method is based on tensor product B-splines instead of NURBS, then the matrix Z is also a Kronecker product matrix, alleviating the disadvantage at large N and p.

4 Numerical examples

In this section, we compare the accuracy and efficiency of the matrix-free isogeometric Galerkin and collocation methods. In [1], it was shown that the proposed Galerkin method performed especially well in the case of a smooth covariance kernel. For rough kernels, such as the C^0 exponential kernel, the interpolation based quadrature performed suboptimally.

In our study, we benchmark both methods for two kernels of different smoothness and appropriate refinement strategies of the spaces involved: (1) the exponential kernel together with h-refinement and (2) the Gaussian kernel and k-refinement. In both variants, the solution space is equal for the Galerkin and collocation methods. The interpolation space used in the Galerkin method is defined on the same mesh as the solution space, but its continuity is adapted in accordance with the remarks made in [1]. All computations are performed sequentially on a laptop machine with an Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz as well as 2x16 GB of DDR4 2666MHz RAM. Our reference solution is the standard isogeometric Galerkin solution computed on the finest possible mesh with a runtime of roughly 17 hours, tabulated in [1].

4.1 Exponential covariance kernel

In Example 1, we compare the performance with respect to h-refinement assuming an exponential kernel on the half-cylindrical domain shown in Figure 1. The polynomial order in each parametric direction is $p = 2$. We choose a tensor product Gauss–Legendre quadrature rule with $(p+1)^3$ points per element of the domain in the collocation method. In accordance with remarks made in [1] the continuity of the interpolation space of the Galerkin method at the element interfaces is reduced to C^0. Furthermore, at element interfaces where the geometry is C^0, the interpolation space of the Galerkin method is set to C^{-1}.

Our comparative investigation is based on five different resolution cases with respect to the characteristic size h of the solution and interpolation mesh. Our specific choices of mesh size and number of degrees of freedom in the interpolation and solution spaces are summarized in Table 1.

Table 1: Mesh, solution space and interpolation space details in Example 1.

	Case 1	Case 2	Case 3	Case 4	Case 5
h	2.857	1.719	1.556	1.423	1.142
N	1050	2108	2800	3772	5625
$\tilde{N}$	1980	8990	12210	16770	28294

h mesh size in the solution and interpolation mesh
N number of degrees of freedom (dof) in the solution space
$\tilde{N}$ number of dof in the interpolation space (IBQ-Galerkin only)

For Case 1, we visualize the first, second, fourth and sixth eigenfunctions computed by both methods, plotted in Figure 2 on the half-cylinder domain. Figure 3 illustrates that already for the coarsest resolution, both methods produce results that are practically indistinguishable from each other when plotted along a selected cut line.

For a quantitative comparison, let us introduce a relative eigenvalue error ε_i with respect to the reference solution as

$$\varepsilon_i := \varepsilon(\lambda_i^{\mathrm{ref}}, \lambda_i^h) := \frac{|\lambda_i^{\mathrm{ref}} - \lambda_i^h|}{\lambda_i^{\mathrm{ref}}} \tag{15}$$

as well as a mean relative eigenvalue error $\bar{\varepsilon}$ given by

$$\bar{\varepsilon} := \frac{1}{M}\sum_{i=1}^{M} \varepsilon_i = \frac{1}{M}\sum_{i=1}^{M} \frac{|\lambda_i^{\mathrm{ref}} - \lambda_i^h|}{\lambda_i^{\mathrm{ref}}}. \tag{16}$$

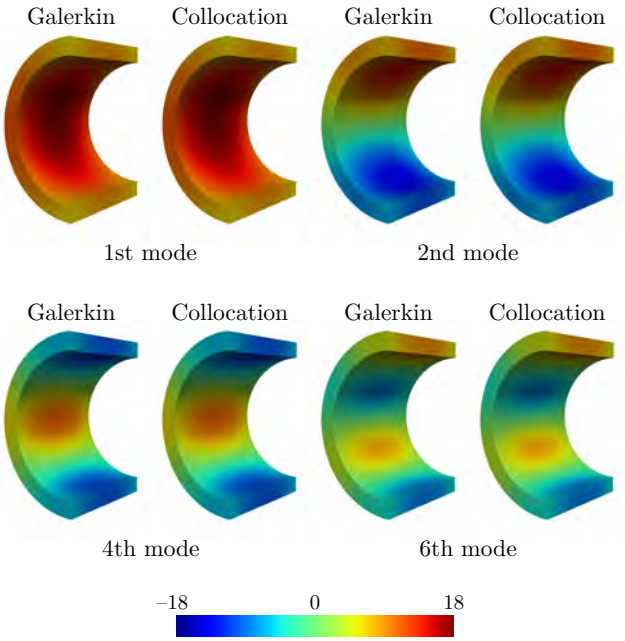

Fig. 2: First, second, fourth and sixth eigenfunctions (Example 1, Case 1).

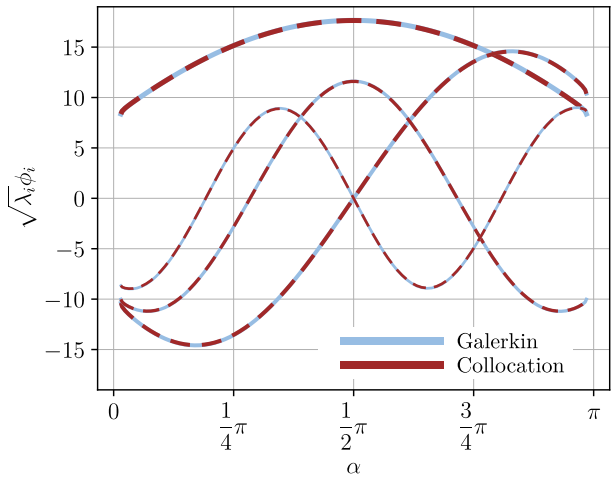

Fig. 3: Line plot in the circumferential direction at the mid-planes of eigenfunctions in Figure 2. Line-width decreases with increasing mode number.

To enable a concise illustration with respect to the five cases defined in Table 1, we define the color coding shown in Table 2. Blue indicates results obtained with the Galerkin method, red indicates results obtained with the collocation method. The change in shading from light to full color indicates the increasing mesh resolution from Case 1 to Case 5.

Figure 4 depicts relative accuracy versus computational time of the iterative eigensolver for the first twenty eigenvalues measured against the reference solution. We observe that the collocation method performs roughly twice as fast at the same level of accuracy.

Table 2: Color-coding to differentiate between five different cases and two different methods.

	Case 1	Case 2	Case 3	Case 4	Case 5
Galerkin					
Collocation					

In Figure 5, we present a detailed assessment of the accuracy of the first five eigenvalues. In addition, we provide an alternative visualization of the timings and the error in the first twenty eigenvalues.

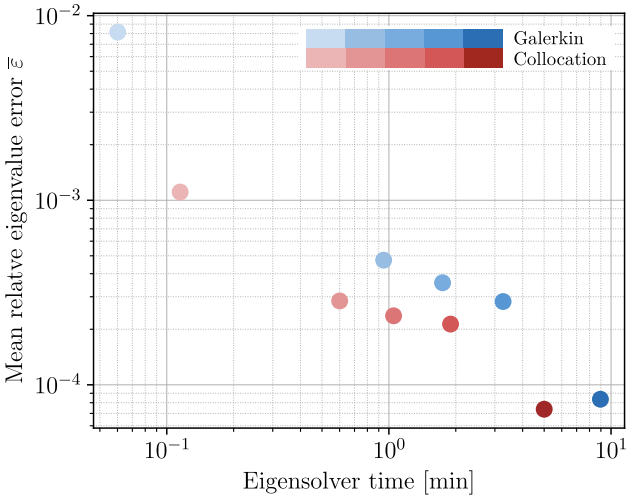

Fig. 4: Mean relative eigenvalue error computed with the first 20 eigenvalues versus the eigensolver time (Example 1, exponential kernel).

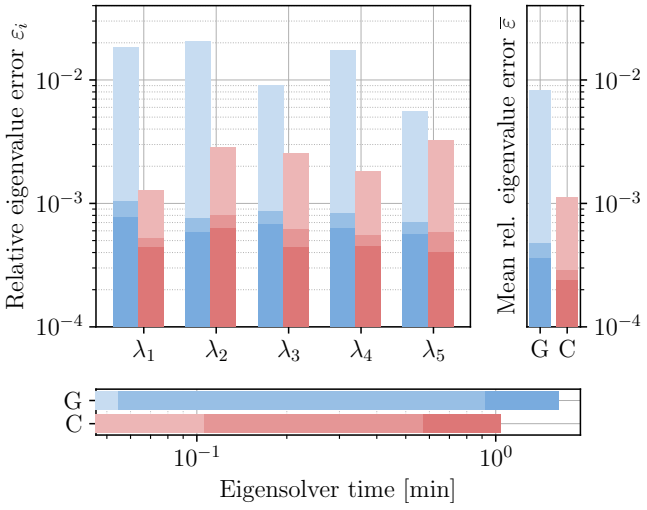

Fig. 5: Error of the first five eigenvalues plotted for Cases 1–3 and corresponding timings and accuracy over the first 20 eigenvalues (Example 1, exponential kernel).

4.2 Gaussian covariance kernel

In Example 2, we compare both methods for a smooth Gaussian covariance kernel. Since the integrand is smooth, we expect that optimally smooth approximation spaces work best. Therefore, we fix the polynomial order p and refine the approximation spaces with C^{p-1} continuity between elements until a target mesh size of 2.857 is reached (k-refinement). The resulting five different cases are summarized in Table 3.

Table 3: Mesh, solution space and interpolation space details in Example 2.

	Case 1	Case 2	Case 3	Case 4	Case 5
p	2	3	4	5	6
N	1050	1628	2340	3198	4214
$\tilde{N}$	1080	1672	2400	3276	4312

p polynomial order of the solution and interpolation space
N number of degrees of freedom (dof) in the solution space
$\tilde{N}$ number of dof in the interpolation space (IBQ-Galerkin only)

Comparing Case 1 in Example 1 with Case 1 in Example 2, we find that the number of degrees of freedom in the interpolation space is smaller. This is due to the increased continuity at element interfaces of the interpolation space of the Galerkin method. This trend is also characteristic for k-refinement and is observable in the remaining Cases 2–5.

We resort again to the color coding of Table 2 to concisely differentiate between the five different resolutions and the two methods. Figure 6 plots the mean relative accuracy of the first twenty eigenvalues versus the eigensolver timings. It is evident that for the smooth Gaussian kernel, the Galerkin method outperforms the collocation method by more than one order of magnitude. Furthermore, in line with the complexity analysis presented in Section 3.3, we observe that the performance gap increases with increasing polynomial order. Following the scheme of Figure 5, we provide a more detailed account of the approximation accuracy of the first five eigenvalues in Figure 7.

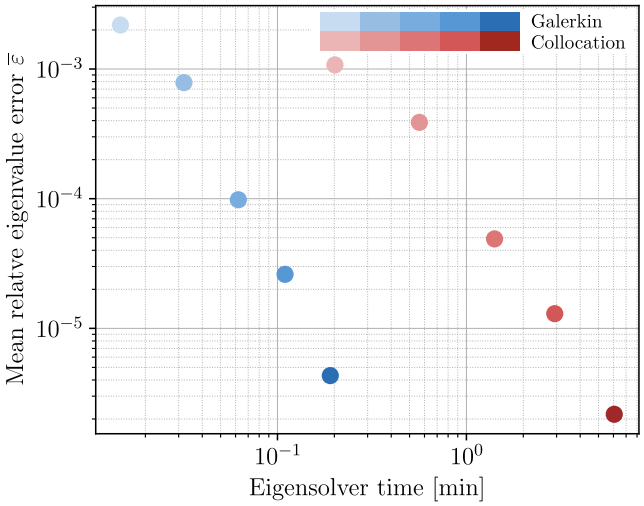

Fig. 6: Mean relative eigenvalue error computed with the first 20 eigenvalues versus the eigensolver time (Example 2, smooth Gaussian kernel).

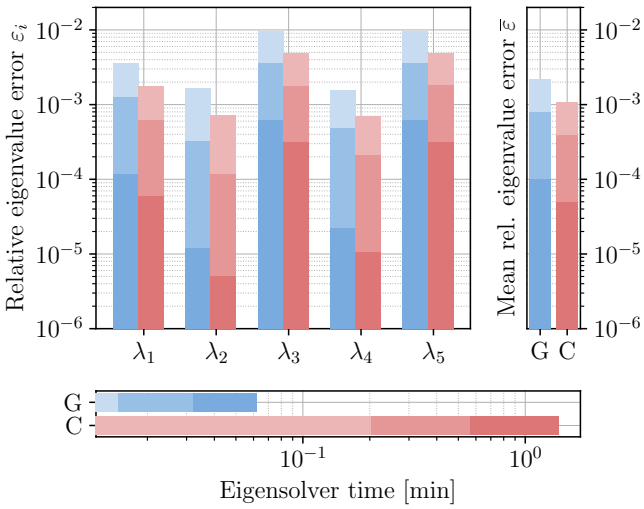

Fig. 7: Error of the first five eigenvalues plotted for Cases 1–3 and corresponding timings and accuracy over the first 20 eigenvalues (Example 1, smooth Gaussian kernel).

5 Conclusions

In this paper, we compared accuracy versus the computational time of two state-of-the-art isogeometric discretization methods for the numerical approximation of the truncated Karhunen–Loève expansion. The first method is the matrix-free isogeometric Galerkin method proposed by the authors in [1]. It achieves its computational efficiency by combining a non-standard trial space with a specialized quadrature technique called *interpolation based quadrature*. This method requires a minimum of quadrature points and relies heavily on global sum factorization. The second method is our new matrix-free version of the isogeometric collocation method proposed in [2]. This method achieves its computational performance by virtue of a low number of point evaluations at collocation points.

On the one hand, our comparative study showed that for a C^0-continuous exponential kernel, the matrix-free collocation method was about twice as fast at the same level of accuracy as the Galerkin method. On

the other hand, our comparative study showed that for a smooth Gaussian kernel, the matrix-free Galerkin method was roughly one order of magnitude faster than the collocation method at the same level of accuracy. Furthermore, the computational advantage of the Galerkin method over the collocation method increases with increasing polynomial degree. These results are not surprising, since it was already shown in [1] that interpolation based quadrature scales virtually independently of the polynomial degree. In our study, we also illustrated via complexity analysis that the matrix-free collocation method scales unfavorably with polynomial order. The suboptimal accuracy of the interpolation based quadrature for rough kernels is also known and was already discussed by the authors in [1]. Besides the aspect of computational performance, we also showed that both methods are highly memory efficient by virtue of their matrix-free formulation.

As for future work, the advantageous properties inherited by the Galerkin method such as symmetric, positive (semi-)definite system matrices, monotonic convergence of the solution and availability of established mathematical framework for stability and convergence deserve a more detailed theoretical discussion with regard to the interpolation based quadrature method. A generalized accuracy study and more numerical benchmarks with existing methods are desirable as well.

Acknowledgements D. Schillinger gratefully acknowledges funding from the German Research Foundation (DFG) through the Emmy Noether Award SCH 1249/2-1.

References

1. M.L. Mika, T.J.R. Hughes, D. Schillinger, P. Wriggers, and R.R. Hiemstra. A matrix-free isogeometric Galerkin method for Karhunen-Loève approximation of random fields using tensor product splines, tensor contraction and interpolation based quadrature. *arXiv:2011.13861 [cs]*, November 2020.
2. R. Jahanbin and S. Rahman. An isogeometric collocation method for efficient random field discretization. *International Journal for Numerical Methods in Engineering*, 117(3):344–369, January 2019.
3. A. Keese. A Review of Recent Developments in the Numerical Solution of Stochastic Partial Differential Equations (Stochastic Finite Elements). *Braunschweig, Institut für Wissenschaftliches Rechnen*, 2003.
4. G. Stefanou. The stochastic finite element method: Past, present and future. *Computer Methods in Applied Mechanics and Engineering*, 198:1031–1051, 2009.
5. B. Sudret and A. Kuyreghian. *Stochastic finite element methods and reliability: a state-of-the-art report.* Berkeley, Department of Civil and Environmental Engineering, University of California, 2000.
6. K. Lu, Y. Jin, Y. Chen, Y. Yang, L. Hou, Z. Zhang, Z. Li, and C. Fu. Review for order reduction based on proper orthogonal decomposition and outlooks of applications in mechanical systems. *Mechanical Systems and Signal Processing*, 123:264–297, May 2019.
7. M. Rathinam and L.R. Petzold. A New Look at Proper Orthogonal Decomposition. *SIAM Journal on Numerical Analysis*, 41(5):1893–1925, January 2003.
8. I.T. Jolliffe and J. Cadima. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A*, 374(2065):20150202, April 2016.
9. Y.C. Liang, H.P. Lee, S.P. Lim, W.Z. Lin, K.H. Lee, and C.G. Wu. Proper orthogonal decomposition and its applications–Part I: Theory. *Journal of Sound and Vibration*, 252(3):527–544, May 2002.
10. M. Eiermann, O.G. Ernst, and E. Ullmann. Computational aspects of the stochastic finite element method. *Computing and Visualization in Science*, 10(1):3–15, February 2007.
11. Y. Saad. *Numerical methods for large eigenvalue problems*. Number 66 in Classics in applied mathematics. Society for Industrial and Applied Mathematics, Philadelphia, rev. ed edition, 2011.
12. K.E. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, 1 edition, June 1997.
13. R.G. Ghanem and P.D. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Springer New York, New York, NY, 1991.
14. S. Rahman. A Galerkin isogeometric method for Karhunen–Loève approximation of random fields. *Computer Methods in Applied Mechanics and Engineering*, 338:533–561, August 2018.
15. A. Bressan and S. Takacs. Sum factorization techniques in Isogeometric Analysis. *Computer Methods in Applied Mechanics and Engineering*, 352:437–460, August 2019.
16. F. Auricchio, L. Beirão Da Veiga, T. J. R. Hughes, A. Reali, and G. Sangalli. Isogeometric collocation methods. *Mathematical Models and Methods in Applied Sciences*, 20(11):2075–2107, November 2010.
17. D. Schillinger, J.A. Evans, A. Reali, M.A. Scott, and T.J.R. Hughes. Isogeometric collocation: Cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations. *Computer Methods in Applied Mechanics and Engineering*, 267:170–232, 2013.