A Bayesian approach to modeling finite element discretization error

Poot, Anne; Kerfriden, Pierre; Rocha, Iuri; van der Meer, Frans

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

ORIGINAL PAPER

A Bayesian approach to modeling finite element discretization error

Anne Poot[1] · Pierre Kerfriden[2] · Iuri Rocha[1] · Frans van der Meer[1]

Abstract

In this work, the uncertainty associated with the finite element discretization error is modeled following the Bayesian paradigm. First, a continuous formulation is derived, where a Gaussian process prior over the solution space is updated based on observations from a finite element discretization. To avoid the computation of intractable integrals, a second, finer, discretization is introduced that is assumed sufficiently dense to represent the true solution field. A prior distribution is assumed over the fine discretization, which is then updated based on observations from the coarse discretization. This yields a posterior distribution with a mean that serves as an estimate of the solution, and a covariance that models the uncertainty associated with this estimate. Two particular choices of prior are investigated: a prior defined implicitly by assigning a white noise distribution to the right-hand side term, and a prior whose covariance function is equal to the Green's function of the partial differential equation. The former yields a posterior distribution with a mean close to the reference solution, but a covariance that contains little information regarding the finite element discretization error. The latter, on the other hand, yields posterior distribution with a mean equal to the coarse finite element solution, and a covariance with a close connection to the discretization error. For both choices of prior a contradiction arises, since the discretization error depends on the right-hand side term, but the posterior covariance does not. We demonstrate how, by rescaling the eigenvalues of the posterior covariance, this independence can be avoided.

Keywords Finite element method · Probabilistic numerics · Uncertainty quantification · Error estimation · Bayesian inference

1 Introduction

In recent years, the Bayesian paradigm has become a popular framework to perform uncertainty quantification. It has found its application in global optimization (Mockus 1989), inverse modeling (Stuart 2010) and data assimilation (Law et al. 2015) contexts, among others. Commonly, given some numerical model, a prior distribution is assumed over its parameters, and the Bayesian paradigm provides a consistent framework to estimate these parameters and to quantify and propagate their associated uncertainty. It should be noted, however, that even if complete certainty could be obtained over the model parameters, there would still be a remaining uncertainty to the solution due to approximations made in the numerical model. This key observation is what underpins the current trend towards probabilistic numerics.

At the core of probabilistic numerics, the estimation of an unknown field is recast as a statistical inference problem, which allows for the estimation of the field with some uncertainty measure (Larkin 1972; Diaconis 1988). Early examples of the application of Bayesian probabilistic numerics include computing integrals (O'Hagan 1991) and solving ordinary differential equations (Skilling 1992). More recently, following a "call to arms" from Hennig et al. (2015), a large push has been made to apply this framework to a wide range of problems, ranging from solving linear systems (Hennig 2015; Cockayne et al. 2019; Wenger et al. 2020) to quadrature (Karvonen and Särkkä 2017; Briol et al. 2017) to solving ordinary differential equations (Schober

✉ Anne Poot
a.poot-1@tudelft.nl

Pierre Kerfriden
pierre.kerfriden@minesparis.psl.eu

Iuri Rocha
i.rocha@tudelft.nl

Frans van der Meer
f.p.vandermeer@tudelft.nl

1 Faculty of Civil Engineering and Geosciences, Delft University of Technology, Stevinweg 1, 2628 CN Delft, The Netherlands

2 Centres des Matériaux, Mines Paris – PSL, 63-65 Rue Henri Auguste Desbruères, 91100 Évry, France

et al. 2014; Hennig et al. 2014; Teymur et al. 2016). For a general overview of the current state-of-the-art of probabilistic numerics, the reader is referred to Hennig et al. (2022). Most relevant for the work presented in this paper are the probabilistic numerical methods that have been developed for the solving of partial differential equations, which can be roughly divided into two categories: meshfree probabilistic solvers, and solver-perturbing error estimators.

The first category (Chkrebtii et al. 2016; Cockayne et al. 2017; Raissi et al. 2018; Wang et al. 2021) can be seen as a way to find solutions to partial differential equations directly from the strong form in a Bayesian manner. A prior is assumed over the solution field, which is updated by evaluating its derivatives on a grid of collocation points, allowing for a solution to be obtained without needing to apply a finite element discretization over the domain. This approach to solving partial differential equations shares some similarities with Bayesian physics-informed neural networks (Raissi et al. 2019; Yang et al. 2021), the main difference lying in the function that is being fitted at the collocation points. The way in which these meshfree solvers relate to traditional collocation methods is similar to the way in which Bayesian physics-informed neural networks relate to their deterministic counterparts.

The second category (Conrad et al. 2017; Kersting and Hennig 2018; Lie et al. 2019) is focused on estimating the discretization error of traditional solvers for differential equations. For ordinary differential equations, the usual time integration step is taken, after which the solution is perturbed by adding Gaussian noise, representing the uncertainty in the time integration result. Similarly, for partial differential equations, the traditional spatial discretization is perturbed using small support Gaussian random fields, which reflect the uncertainty introduced by the mesh. In Abdulle and Garegnani (2020, 2021), a similar approach is taken, but rather than adding noise to the solution, an uncertainty is introduced by perturbing the time step size or finite element discretization. A more formal mathematical basis for probabilistic numerical methods can be found in Cockayne et al. (2019), where a more rigorous definition of the term is outlined and a common framework underpinning these two seemingly separate categories is established.

It is worth noting that these probabilistic numerical methods are a deviation from traditional error estimators (Babuška and Rheinboldt 1979; Babuška and Miller 1987; Zienkiewicz and Zhu 1987), as they embed the model error into the method itself, rather than estimate it a posteriori. This inherently affects the model output, which depending on the context can be a desirable or undesirable property. In Rouse et al. (2021), a method is presented to obtain full-field error estimates by assuming a Gaussian process prior over the discretization error, and updating it based on a set of traditional estimators of error in quantities of interest. This way, a distribution representing the finite element discretization error can be obtained in a non-intrusive manner.

The shared goal of these methods is to accurately describe the errors made due to limitations of our numerical models, though their method of modeling error differs. At the core, the meshfree probabilistic solvers model error as the result of using a finite number of observations to obtain a solution to an infinite-dimensional problem. The solver-perturbing error estimators, on the other hand, take an existing discretization, like the one used in the finite element method, and assign some uncertainty measure to the existing solver. This begs the question: what happens if the methodology from the meshfree probabilistic solvers is applied to existing mesh-based solvers of partial differential equations? Little research has thus far been conducted to answer this question, though two particular works are worth pointing out.

A brief remark is made in Bilionis (2016) describing a Bayesian probabilistic numerical method whose posterior mean is equivalent to the finite element solution. However, this idea is then discarded due to infinite variances arising in the posterior distribution. In Pförtner et al. (2023), the probabilistic meshfree solvers from Cockayne et al. (2017) are generalized to methods of weighted residuals, which includes the finite element method. Of particular relevance to our work is their construction of prior distributions whose posterior mean is guaranteed to be equivalent to the usual finite element solution. Doing this would allow one to replace the traditional finite element solver with the probabilistic one, in order to quantify the finite element discretization error. However, their experimental results are limited to one-dimensional test cases, possibly because the application of their formulation to unstructured triangular or quadrilateral meshes would result in integrals in the information operator that are computationally too expensive.

In this work, we propose a probabilistic numerical method for the modeling of finite element discretization error. The solution is endowed with a Gaussian process prior, which is then updated based on observations of the right-hand side from a finite element discretization. This allows for the approximation of the true solution while including the uncertainty resulting from the finite discretization that is applied. Rather than work directly with the Gaussian process distribution over the exact solution space, we introduce a second discretization over the domain that is fine enough to represent the exact solution. This second discretization helps to avoid the infinite variances brought up in Bilionis (2016) as well as the computationally expensive integrals from Pförtner et al. (2023). We present a class of priors that naturally accounts for the smoothness of the partial differential equation at hand, and show how the assembly of large full covariance matrices can be avoided. A particular focus of this work is on the relationship between the posterior covariance of our formulation and the finite element discretization error. The relationship

between these two quantities is often left to intuition, reasoning along the lines that since the posterior covariance contains remaining model uncertainty, it must reflect the discretization error. We challenge this assumption and investigate more thoroughly which conditions need to be met before the posterior covariance can reasonably be said to capture the finite element discretization error.

The underlying goal of the development of a Bayesian model for the finite element discretization error is to enable the propagation of discretization error to quantities of interest through the computational pipelines that arise in multiscale modeling, inverse modeling and data assimilation settings. This consistent treatment of discretization error in turn allows for more informed decisions to be made about its impact on the model output. To give a concrete example, in Girolami et al. (2021), a Bayesian framework for the assimilation of measurement data and finite element models is presented. Within this framework, a model misspecification component is defined, which is endowed with a squared-exponential Gaussian process prior. The Bayesian formulation of the finite element method that we derive in this work would allow for a more informative choice of prior distribution over the model misspecification component, for example by separating out the discretization error from the error associated with other modeling assumptions.

In the context of Bayesian inverse modeling, our proposed method could prove particularly useful. For the Metropolis-Hastings sampling strategies that are commonly employed, a finite element solve is necessary for each sample that is drawn, which typically needs to be done tens or hundreds of thousands of times. The goals of having a negligible discretization error and a computational cost that is not prohibitive can therefore be in conflict. Rather than attempt to fully resolve the discretization error, it can be more practical to use a coarse mesh and account for the associated error in the likelihood of the Bayesian inverse model. To do this, a probability density that is reflective of the discretization error of the coarse solve is needed, which is what our Bayesian formulation of the finite element method aims to provide.

The outline of this paper is as follows: in Sect. 2, we derive our Bayesian formulation of the finite element method. This is followed by a discussion on the choice of prior covariance in Sect. 3, where two different choices of prior distribution are investigated. Two examples, a one-dimensional tapered bar and a two-dimensional perforated plate, are showcased throughout this section to validate the conclusions drawn from theory. Finally, in Sect. 4, the conclusions of this paper are drawn and discussed.

2 Bayesian finite element method

In this section, the proposed Bayesian version of the finite element method is derived. Although the method is applicable to a broad range of linear elliptic partial differential equations, for the purposes of demonstration, we will consider Poisson's equation:

$$\begin{aligned} -\Delta u(\mathbf{x}) &= f(\mathbf{x}) \quad &&\text{in } \Omega \\ u(\mathbf{x}) &= 0 \quad &&\text{on } \partial\Omega \end{aligned} \tag{1}$$

Here, Ω and $\partial\Omega$ are the domain and its boundary, respectively. $u(\mathbf{x})$ and $f(\mathbf{x})$ are the solution and forcing term, which are linked through the Laplace operator Δ.

2.1 Continuous formulation

We will start with the derivation of a continuous posterior distribution over the solution space conditioned on the finite element force vector, largely following Bilionis (2016). As usual, the problem is restated in its weak formulation:

$$\int_\Omega \nabla u(\mathbf{x}) \cdot \nabla v(\mathbf{x}) \, d\mathbf{x} = \int_\Omega f(\mathbf{x}) v(\mathbf{x}) \, d\mathbf{x} \quad \forall v(\mathbf{x}) \in \mathcal{V} \tag{2}$$

We search $u(\mathbf{x}) \in \mathcal{V}$, where $\mathcal{V} = H_0^1$ is a Sobolev space of functions over Ω that are weakly once-differentiable and vanish at the boundary $\partial\Omega$. This space is equipped with an inner product and thus also forms a Hilbert space. Now, a discretization is defined over the domain using a set of locally supported shape functions $\{\psi_i(\mathbf{x})\}_{i=1}^m$, which span a finite-dimensional space $\mathcal{W}^h \subset \mathcal{V}$. The test function $v^h(\mathbf{x})$ can be defined in terms of these shape functions:

$$v^h(\mathbf{x}) = \sum_{i=1}^m v_i \psi_i(\mathbf{x}) \quad \text{with } \psi_i(\mathbf{x}) \in \mathcal{W}^h \tag{3}$$

Since Eq. (2) has to hold for all $v^h(\mathbf{x}) \in \mathcal{W}^h$, the weights v_i can be chosen at will. Substituting Eq. (3) into Eq. (2), a finite set of m equations in constructed by choosing $v_i = \delta_{ij}$ for the jth equation, where δ_{ij} is the Kronecker delta function. This yields the entries of the finite element force vector $\mathbf{g}$:

$$g_i = \int_\Omega f(\mathbf{x}) \psi_i(\mathbf{x}) \, d\mathbf{x} \tag{4}$$

We can relate the solution $u(\mathbf{x})$ to the force vector $\mathbf{g}$ via the linear operator $\mathcal{L}$:

$$\mathcal{L}[u(\mathbf{x})] = \mathbf{g} \qquad (5)$$

where $\mathcal{L}[u(\mathbf{x})] = \left[\mathcal{L}_1[u(\mathbf{x})]\, \mathcal{L}_2[u(\mathbf{x})] \ldots \mathcal{L}_m[u(\mathbf{x})] \right]^T$ is given by:

$$\mathcal{L}_i[u(\mathbf{x})] = \int_\Omega \nabla u(\mathbf{x}) \cdot \nabla \psi_i(\mathbf{x})\, d\mathbf{x} \qquad (6)$$

A centered Gaussian process with a positive definite covariance function $k(\mathbf{x}, \mathbf{x}')$ is now assumed over the solution $u(\mathbf{x})$:

$$u(\mathbf{x}) \sim \mathcal{GP}\left(0, k(\mathbf{x}, \mathbf{x}')\right) \qquad (7)$$

Because we have a linear map $\mathcal{L}$ from $u(\mathbf{x})$ to $\mathbf{g}$, conditioning $u(\mathbf{x})$ on $\mathbf{g}$ yields another Gaussian process distribution (Pförtner et al. 2023):

$$u(\mathbf{x})|\mathbf{g} \sim \mathcal{GP}\left(m^*(\mathbf{x}), k^*(\mathbf{x}, \mathbf{x}')\right) \qquad (8)$$

Here, the posterior mean function $m^*(\mathbf{x})$ and covariance function $k^*(\mathbf{x}, \mathbf{x}')$ are given by[1]:

$$
\begin{aligned}
m^*(\mathbf{x}) &= \mathcal{L}'\left[k(\mathbf{x}, \mathbf{z}')\right] \mathbf{L}^{-1}\mathbf{g} \\
k^*(\mathbf{x}, \mathbf{x}') &= k(\mathbf{x}, \mathbf{x}') - \mathcal{L}'\left[k(\mathbf{x}, \mathbf{z}')\right] \mathbf{L}^{-1}\mathcal{L}\left[k(\mathbf{z}, \mathbf{x}')\right]
\end{aligned}
\qquad (9)
$$

where $\mathbf{L} = \mathcal{L}\left[\mathcal{L}'\left[k(\mathbf{z}, \mathbf{z}')\right]\right]$ is the Gram matrix. The posterior mean function $m^*(\mathbf{x})$ provides a full-field estimate of the solution $u(\mathbf{x})$. The posterior covariance function $k^*(\mathbf{x}, \mathbf{x}')$ indicates the uncertainty associated with this estimate due to the fact that it was obtained using only a finite set of shape functions. Since the finite discretization is the only source of uncertainty in our model, we can intuit some association between this posterior covariance and the finite element discretization error.

The formulation presented thus far can be contextualized in the method of weighted residuals framework presented in Pförtner et al. (2023). Specifically, our continuous formulation is equivalent to choosing the information operator $\mathcal{I}[u(\mathbf{x})] = \left[\mathcal{I}_1[u(\mathbf{x})]\, \mathcal{I}_2[u(\mathbf{x})] \ldots \mathcal{I}_m[u(\mathbf{x})] \right]^T$ in their framework to be given by:

$$\mathcal{I}_i[u(\mathbf{x})] = \int_\Omega \nabla u(\mathbf{x}) \cdot \nabla \psi_i(\mathbf{x})\, d\mathbf{x} - \int_\Omega f(\mathbf{x}) \psi_i(\mathbf{x})\, d\mathbf{x} \qquad (10)$$

---

[1] To avoid confusion when applying $\mathcal{L}$ to the covariance function $k(\mathbf{z}, \mathbf{z}')$, we use $\mathcal{L}$ and $\mathcal{L}'$ to denote that gradients and integrals are computed with respect to $\mathbf{z}$ and $\mathbf{z}'$, respectively.

Unfortunately, the integrals that arise in the expressions for the posterior mean and covariance functions in Eq. (9) are generally intractable. For some arbitrary covariance function $k(\mathbf{x}, \mathbf{x}')$, the integration over the shape functions $\psi_i(\mathbf{x})$ and $\psi_j(\mathbf{x}')$ cannot be performed without putting severe restrictions on which shape functions are permitted. This in turn puts severe constraints on the domain shape, which undercuts the core strength of the finite element method, namely its ability to solve partial differential equations on complicated domains. On the other hand, we can design the covariance function such that these integrals do become tractable, for example by following Bilionis (2016) and setting $k(\mathbf{x}, \mathbf{x}') = G(\mathbf{x}, \mathbf{x}')$, or following Owhadi (2015) and setting $k(\mathbf{x}, \mathbf{x}') = \int_\Omega \int_\Omega G(\mathbf{x}, \mathbf{z}) G(\mathbf{x}', \mathbf{z}') \delta(\mathbf{z} - \mathbf{z}')\, d\mathbf{z}\, d\mathbf{z}'$, where $\delta(\mathbf{x})$ is a Dirac delta function. However, in both of these expressions, the Green's function $G(\mathbf{x}, \mathbf{x}')$ associated with the operator $-\Delta$ is required, which is generally not available for a given partial differential equation. Since our aim is to develop a general Bayesian framework for modeling finite element discretization error, a new approach is needed that does not impose restrictions on the choice of shape functions or require access to the Green's function.

2.2 Discretized formulation

This motivates us to approximate $u(\mathbf{x})$ in the finite-dimensional space $\mathcal{V}^h$ spanned by a second set of locally supported shape functions $\{\phi_j(\mathbf{x})\}_{j=1}^n$. This defines the trial function $u^h(\mathbf{x})$:

$$u(\mathbf{x}) \approx u^h(\mathbf{x}) = \sum_{j=1}^n u_j \phi_j(\mathbf{x}) \quad \text{with } \phi_j(\mathbf{x}) \in \mathcal{V}^h \qquad (11)$$

Note that this is not the same set of shape functions as the one used to define the force vector in Eq. (4). In fact, since our aim is to model the discretization error that arises by choosing $v(\mathbf{x}) \in \mathcal{W}^h$ rather than and $v(\mathbf{x}) \in \mathcal{V}$, it is important that the error associated with the projection of an arbitrary function $w(\mathbf{x}) \in \mathcal{V}$ onto $\mathcal{V}^h$ is small compared to the error associated with its projection onto $\mathcal{W}^h$. Loosely speaking, we assume that $\mathcal{V}^h$ is sufficiently expressive to serve as a stand-in for $\mathcal{V}$.

Substituting Eqs. (3) and (11) into Eq. (2) yields the matrix formulation of the problem:

$$\mathbf{H}\mathbf{u} = \mathbf{g} \qquad (12)$$

The elements of the stiffness matrix $\mathbf{H}$ are given by:

$$H_{ij} = \int_\Omega \nabla \psi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x})\, d\mathbf{x} \qquad (13)$$

The assumption that $\mathcal{V}^h$ is more expressive than $\mathcal{W}^h$ implies that $\mathbf{u}$ will have a larger dimensionality than $\mathbf{g}$ and thus that $\mathbf{H}$

is a rectangular matrix and that Eq. (12) describes an under-determined system. However, the fact that this system of equations has an infinite set of solutions need not pose a problem, due to the regularizing effect of the prior assumed over $u(\mathbf{x})$.

Since the solution field $u(\mathbf{x})$ has been reduced from the infinite-dimensional space $\mathcal{V}$ to the finite-dimensional $\mathcal{V}^h$, the distribution assumed over the solution in Eq. (7) needs to be reduced accordingly. Instead of an infinite-dimensional Gaussian process, we obtain a finite-dimensional zero-mean normal distribution with a positive definite covariance matrix $\boldsymbol{\Sigma}$:

$$\mathbf{u} \sim \mathcal{N}(0, \boldsymbol{\Sigma}) \tag{14}$$

The joint distribution of $\mathbf{u}$ and $\mathbf{g}$ is now given by:

$$\begin{bmatrix} \mathbf{g} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} \mathbf{Hu} \\ \mathbf{u} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{H\Sigma H}^T & \mathbf{H\Sigma} \\ \mathbf{\Sigma H}^T & \boldsymbol{\Sigma} \end{bmatrix}\right) \tag{15}$$

Conditioning $\mathbf{u}$ on $\mathbf{g}$ yields the following posterior distribution:

$$\mathbf{u}|\mathbf{g} \sim \mathcal{N}\left(\mathbf{m}^*, \boldsymbol{\Sigma}^*\right) \tag{16}$$

Here, the posterior mean vector $\mathbf{m}^*$ and covariance matrix $\boldsymbol{\Sigma}^*$ are given by:

$$\begin{aligned} \mathbf{m}^* &= \boldsymbol{\Sigma}\mathbf{H}^T \left(\mathbf{H\Sigma H}^T\right)^{-1} \mathbf{g} \\ \boldsymbol{\Sigma}^* &= \boldsymbol{\Sigma} - \boldsymbol{\Sigma}\mathbf{H}^T \left(\mathbf{H\Sigma H}^T\right)^{-1} \mathbf{H\Sigma} \end{aligned} \tag{17}$$

Similar to the continuous formulation presented in Sect. 2.1, $\mathbf{m}^*$ can be interpreted as providing an estimate of the solution $u(\mathbf{x})$ in the fine space $\mathcal{V}^h$, while observing the right-hand side $f(\mathbf{x})$ only in the coarse space $\mathcal{W}^h$. The posterior covariance matrix $\boldsymbol{\Sigma}^*$ then provides an indication of the uncertainty associated with this estimate due to the fact that only observations from the coarse mesh are used to obtain this estimate. Note that if the test and trial spaces are chosen to be the same (i.e. $\mathcal{W}^h = \mathcal{V}^h$), $\boldsymbol{\Sigma}^*$ reduces to a null matrix, reflecting the fact that there no longer exists a discretization error between $\mathcal{V}^h$ and $\mathcal{W}^h$.

2.3 Hierarchical shape functions

Thus far, the only requirement that has been put on the choice of $\mathcal{V}^h$ and $\mathcal{W}^h$ is that the error between $\mathcal{V}$ and $\mathcal{V}^h$ is small compared to the error between $\mathcal{V}$ and $\mathcal{W}^h$. We now add a second restriction, namely that $\mathcal{W}^h \subset \mathcal{V}^h$. This defines a hierarchy between these two spaces, and implies that any function defined in $\mathcal{W}^h$ can be expressed in $\mathcal{V}^h$. One way to ensure this hierarchy in practice is to first define a coarse

mesh corresponding to $\mathcal{W}^h$, and then refine it hierarchically to obtain a fine mesh corresponding to $\mathcal{V}^h$. Alternatively, it is possible to use only a single mesh, and use linear and quadratic shape functions over the same finite elements to define $\mathcal{W}^h$ and $\mathcal{V}^h$, respectively.

From the hierarchy between $\mathcal{V}^h$ and $\mathcal{W}^h$, it follows that the basis functions that span the coarse space $\mathcal{W}^h$ can be written as a linear combination of the basis functions that span the fine space $\mathcal{V}^h$. In other words, there exists a matrix[2] $\boldsymbol{\Phi}^T$ that maps a vector of fine shape functions $\boldsymbol{\phi}(\mathbf{x}) = \begin{bmatrix} \phi_1(\mathbf{x}) & \phi_2(\mathbf{x}) & \ldots & \phi_n(\mathbf{x}) \end{bmatrix}^T$ to a vector of coarse shape functions $\boldsymbol{\psi}(\mathbf{x}) = \begin{bmatrix} \psi_1(\mathbf{x}) & \psi_2(\mathbf{x}) & \ldots & \psi_m(\mathbf{x}) \end{bmatrix}^T$:

$$\boldsymbol{\psi}(\mathbf{x}) = \boldsymbol{\Phi}^T \boldsymbol{\phi}(\mathbf{x}) \tag{18}$$

This allows Eq. (13) to be rewritten as:

$$\begin{aligned} H_{ij} &= \int_\Omega \nabla \sum_{k=1}^n \Phi_{ki}\phi_k(\mathbf{x}) \cdot \nabla\phi_j(\mathbf{x}) \, d\mathbf{x} \\ &= \sum_{k=1}^n \Phi_{ki} \int_\Omega \nabla\phi_k(\mathbf{x}) \cdot \nabla\phi_j(\mathbf{x}) \, d\mathbf{x} \end{aligned} \tag{19}$$

As a result, $\mathbf{H}$ can be expressed as:

$$\mathbf{H} = \boldsymbol{\Phi}^T \mathbf{K} \tag{20}$$

where $\mathbf{K}$ is the fine-scale (square and symmetric) stiffness matrix that would follow if both trial and test functions came from the fine space $\mathcal{V}^h$:

$$K_{ij} = \int_\Omega \nabla\phi_i(\mathbf{x}) \cdot \nabla\phi_j(\mathbf{x}) \, d\mathbf{x} \tag{21}$$

Following a similar line of reasoning, the coarse stiffness matrix $\mathbf{K_c}$, that would be found if both trial and test functions came from the coarse space $\mathcal{W}^h$, can be written in terms of $\boldsymbol{\Phi}$ and $\mathbf{K}$:

$$\mathbf{K_c} = \boldsymbol{\Phi}^T \mathbf{K} \boldsymbol{\Phi} \tag{22}$$

Similarly to Eq. (19), we can rewrite Eq. (4) as:

$$\begin{aligned} g_i &= \int_\Omega f(\mathbf{x}) \sum_{k=1}^n \Phi_{ki}\phi_k(\mathbf{x}) \, d\mathbf{x} \\ &= \sum_{k=1}^n \Phi_{ki} \int_\Omega f(\mathbf{x})\phi_k(\mathbf{x}) \, d\mathbf{x} \end{aligned} \tag{23}$$

---

[2] Note that $\boldsymbol{\Phi}$ has been defined in terms of its transpose in order to make expressions in later sections consistent with common notation for least squares, proper orthogonal decomposition, and so on.

And so, **g** can be expressed as:

$$\mathbf{g} = \mathbf{\Phi}^T \mathbf{f} \tag{24}$$

where **f** is the fine-scale force vector that arises by integrating the forcing term over the fine-scale test functions:

$$f_i = \int_\Omega f(\mathbf{x}) \phi_i(\mathbf{x}) \, d\mathbf{x} \tag{25}$$

Finally, we define the reference solution $\hat{\mathbf{u}}$ as the solution to the fine-scale system of equations that is obtained by choosing both the test and trial spaces to be the fine space $\mathcal{V}^h$:

$$\mathbf{K}\hat{\mathbf{u}} = \mathbf{f} \tag{26}$$

In the remainder of this work, discretization error is defined with respect to $\hat{\mathbf{u}}$. Specifically, the finite element discretization error **e** is defined as the difference between the fine-scale reference solution $\hat{\mathbf{u}}$ and the coarse-scale solution projected to the fine space:

$$\mathbf{e} = \mathbf{K}^{-1}\mathbf{f} - \mathbf{\Phi}\mathbf{K_c}^{-1}\mathbf{g} \tag{27}$$

2.4 Boundary conditions

It is worth considering how the application of boundary conditions in the fine space translates to the shape functions in the coarse space. To do this, $\boldsymbol{\phi}(\mathbf{x})$ is split into $\boldsymbol{\phi_i}(\mathbf{x})$ and $\boldsymbol{\phi_d}(\mathbf{x})$, where the subscript **d** refers to the nodes on the part of the boundary where Dirichlet conditions are applied, and the subscript **i** refers to all other nodes (i.e. both internal nodes and non-Dirichlet boundary nodes). This could be considered abuse of notation, since $\mathcal{V}^h \subset \mathcal{V}$, which is already constrained by the Dirichlet boundary conditions, so from this point of view, $\boldsymbol{\phi_d}(\mathbf{x})$ should not exist. However, in most practical finite element implementations, shape functions are assigned to the boundary nodes as well in order to facilitate the inclusion of inhomogeneous boundary conditions in the model.

The boundary conditions in the coarse space follow from $\boldsymbol{\phi_d}(\mathbf{x})$ and $\mathbf{\Phi}$, since $\boldsymbol{\psi_d}(\mathbf{x})$ is defined as the elements of $\boldsymbol{\psi}(\mathbf{x})$ where the rows of $\mathbf{\Phi}$ belonging to $\boldsymbol{\phi_d}(\mathbf{x})$ have non-zero entries. As a result, Eq. (18) can be split as follows:

$$\begin{bmatrix} \boldsymbol{\psi_i}(\mathbf{x}) \\ \boldsymbol{\psi_d}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi_{ii}}^T & \mathbf{0} \\ \mathbf{\Phi_{id}}^T & \mathbf{\Phi_{dd}}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi_i}(\mathbf{x}) \\ \boldsymbol{\phi_d}(\mathbf{x}) \end{bmatrix} \tag{28}$$

Note that the fact that $\mathbf{\Phi_{di}} = \mathbf{0}$ does not introduce any loss of generality: any non-zero element of $\mathbf{\Phi_{di}}$ would by definition of $\boldsymbol{\psi_d}(\mathbf{x})$ be an element of $\mathbf{\Phi_{dd}}$, not $\mathbf{\Phi_{di}}$. From Eqs. (20) and

(28), it follows that:

$$\mathbf{H_{ii}} = \mathbf{\Phi_{ii}}^T \mathbf{K_{ii}} \tag{29}$$

Similarly, from Eqs. (24) and (28), it follows that:

$$\mathbf{g_i} = \mathbf{\Phi_{ii}}^T \mathbf{f_i} \tag{30}$$

Commonly, Dirichlet boundary conditions are enforced by eliminating the corresponding degrees of freedom, and solving the system that remains. Due to the simple relation that $\mathbf{\Phi_{ii}}$ provides between $\mathbf{H_{ii}}$ and $\mathbf{K_{ii}}$ (Eq. 29) as well as $\mathbf{g_i}$ and $\mathbf{f_i}$ (Eq. 30), all relationships described in Sects. 2.2 and 2.3 still hold when applied only to the internal nodes of the system. From this point onward, we will therefore only consider the internal nodes of the system. This also means that only the part of the covariance matrix related to the internal nodes $\mathbf{\Sigma_{ii}}$ needs to be considered, and so the requirement of positive definiteness of $\mathbf{\Sigma}$ can be relaxed to a requirement of positive definiteness of only $\mathbf{\Sigma_{ii}}$. The subscripts **i** (for vectors) and **ii** (for matrices) will be left implied in order to declutter the notation.

In the remainder of this paper, we will limit ourselves to partial differential equations with homogeneous boundary conditions. However, the method can easily be extended to inhomogeneous Dirichlet and Neumann boundary conditions. Details on how inhomogeneous boundary conditions can be enforced are given in Appendix A.

3 Choice of prior covariance

Thus far, the prior covariance matrix $\mathbf{\Sigma}$ has not been specified. The choice of $\mathbf{\Sigma}$ is subject to two main requirements. The first requirement is that $\mathbf{\Sigma}$ needs to have a sparse representation. Since $\mathbf{\Sigma}$ is a $n \times n$ matrix, where n is the number of degrees of freedom of the fine discretization, explicitly computing, storing and applying operations on the full matrix would quickly become prohibitively expensive. As a result, the traditional approach of using a kernel to directly compute all entries of $\mathbf{\Sigma}$ would be infeasible. Instead, the prior is defined implicitly by assigning a sparse covariance matrix to the fine-scale force vector **f**, which implicitly defines the covariance matrix of the solution vector $\mathbf{\Sigma}$, but does not require us to explicitly compute it. For certain kernel-based priors, an equivalent stochastic partial differential equation can be shown to exist, which allows for a similar sparse representation (see for example Roininen et al. (2014)).

The second requirement is that the choice of prior distribution needs to be appropriate for the partial differential equation at hand. For instance, if the infinitely differentiable squared exponential prior were assumed on the solution field $u(x)$, this would imply C^∞ continuity on the right-hand side

field $f(x)$. From a modeling point of view, this would be an undesirable assumption to make, since it is very restrictive concerning what forcing terms are permitted. On the other hand, if the prior is not smooth enough, samples from the prior would exhibit unphysical discontinuities in $u(x)$ or its gradient fields. In short, the prior needs to respect the smoothness of the partial differential equation to which it is applied.

In this section, a particular class of priors that meets both of these requirements is presented by means of two test cases. The first test case, presented in Fig. 1, concerns a one-dimensional mechanics problem described by the following ordinary differential equation with homogeneous boundary conditions:

$$-\frac{\mathrm{d}}{\mathrm{d}x}\left(EA(x)\frac{\mathrm{d}u}{\mathrm{d}x}\right) = f(x) \quad \text{in } \Omega = (0, 1)$$
$$u(x) = 0 \qquad \text{on } \partial\Omega = \{0, 1\} \tag{31}$$

Here, the distributed load $f(x) = 1$, Young's modulus $E = 1$ and the cross-sectional area $A(x) = 0.1 - 0.099x$. This setup describes a tapered bar with a constant load, where both the left and right end are clamped, as shown in Fig. 1. The fine-scale discretization consists of a uniform mesh with 64 elements ($n = 64$) and linear shape functions. Three different levels of uniform coarse discretization are used: $m = 4$, $m = 16$ and $m = 64$. Note that in all cases, since n is a multiple of m, the shape functions are defined hierarchically in accordance with Sect. 2.3.

The second case is shown in Fig. 2 and concerns a two-dimensional mechanics problem. A plate ($L = 4$, $H = 2$) with a hole ($R = 0.8$) is clamped on its left edge and loaded by a constant horizontal body load $f_x = 1$ The plate has unit thickness, Young's modulus $E = 3$ and Poisson's ratio $\nu = 0.2$. The problem is meshed non-uniformly, as shown in Fig. 2a. For the coarse mesh, a characteristic length $h = 0.5$ is used at the left and right edge, but around the hole a refinement is applied. The refinements below and above the hole have a characteristic length of $h = 0.2$ and $h = 0.05$, respectively. The fine mesh is generated by dividing each coarse element into 4 smaller triangular elements. In Fig. 2b, it can be seen how this difference in mesh density on different sides of the hole results in a larger discretization error below the hole than above it. For reference, the fine-scale and coarse-scale solution are shown in Figs. 2c and d, respectively.

3.1 A sparse right-hand side prior

Following the approach taken in Cockayne et al. (2017), rather than assuming a prior measure directly on the displacement field $u(\mathbf{x})$, we assume a centered Gaussian process prior with covariance function $k_\mathrm{f}(\mathbf{x}, \mathbf{x}')$ over the forcing term $f(\mathbf{x})$:
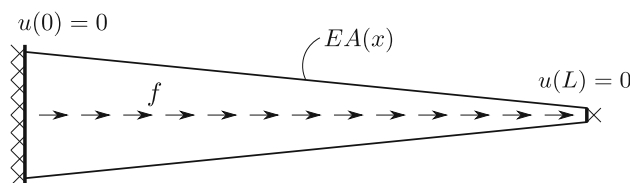

$$u(0) = 0 \qquad\qquad EA(x) \qquad\qquad u(L) = 0$$

$$f$$

Fig. 1 Schematic overview of the tapered bar test case

$$f(\mathbf{x}) \sim \mathcal{GP}\left(0, k_\mathrm{f}(\mathbf{x}, \mathbf{x}')\right) \tag{32}$$

This implicitly defines an equivalent prior on $u(\mathbf{x})$:

$$u(\mathbf{x}) \sim \mathcal{GP}\left(0, k_\mathrm{nat}(\mathbf{x}, \mathbf{x}')\right) \tag{33}$$

Here, the covariance function k_nat can be expressed in terms of $k_\mathrm{f}(\mathbf{x}, \mathbf{x}')$ and the Green's function $G(\mathbf{x}, \mathbf{x}')$ associated with the operator of the partial differential equation:

$$k_\mathrm{nat}(\mathbf{x}, \mathbf{x}') = \int_\Omega \int_\Omega G(\mathbf{x}, \mathbf{z}) G(\mathbf{x}', \mathbf{z}') k_\mathrm{f}(\mathbf{z}, \mathbf{z}') \, \mathrm{d}\mathbf{z} \, \mathrm{d}\mathbf{z}' \tag{34}$$

In Cockayne et al. (2017), this kernel is described as "natural" in the sense that the operator $-\Delta$ (see Eq. 1) uniquely maps from the Hilbert space associated with the forcing term covariance function $k_f(\mathbf{x}, \mathbf{x}')$ to the one associated with $k_\mathrm{nat}(\mathbf{x}, \mathbf{x}')$. Each sample from $u(\mathbf{x})$ drawn from this natural kernel has an equivalent sample from $f(\mathbf{x})$ and vice versa. Unfortunately, since the Green's function is generally not available for a given partial differential equation, Cockayne et al. (2017) discards this natural kernel is then discarded in favor of a Matern or Wendland kernel with the appropriate level of smoothness.

However, because we avoid this problem by introducing the fine-scale discretization, there is no need here to step away from the natural prior approach. Instead, it can be approximated by applying the fine-scale finite element discretization first, and only then finding the natural covariance matrix for the solution vector $\mathbf{u}$. Given the prior distribution over $f(\mathbf{x})$ in Eq. (32) and the definition of the force vector in Eq. (21), it follows that:

$$\mathbf{f} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_\mathbf{f}\right) \tag{35}$$

where the force vector covariance matrix $\boldsymbol{\Sigma}_\mathbf{f}$ is given by:

$$\boldsymbol{\Sigma}_\mathbf{f} = \int_\Omega \int_\Omega k_\mathrm{f}(\mathbf{x}, \mathbf{x}') \boldsymbol{\phi}(\mathbf{x}) \boldsymbol{\phi}(\mathbf{x}')^T \, \mathrm{d}\mathbf{x}' \, \mathrm{d}\mathbf{x} \tag{36}$$

The resulting prior distribution over $\mathbf{u}$ then becomes:

$$\mathbf{u} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{K}^{-1}\boldsymbol{\Sigma}_\mathbf{f}\mathbf{K}^{-1}\right) \tag{37}$$

(a) Schematic overview

(b) Discretization error $\mathbf{e}$

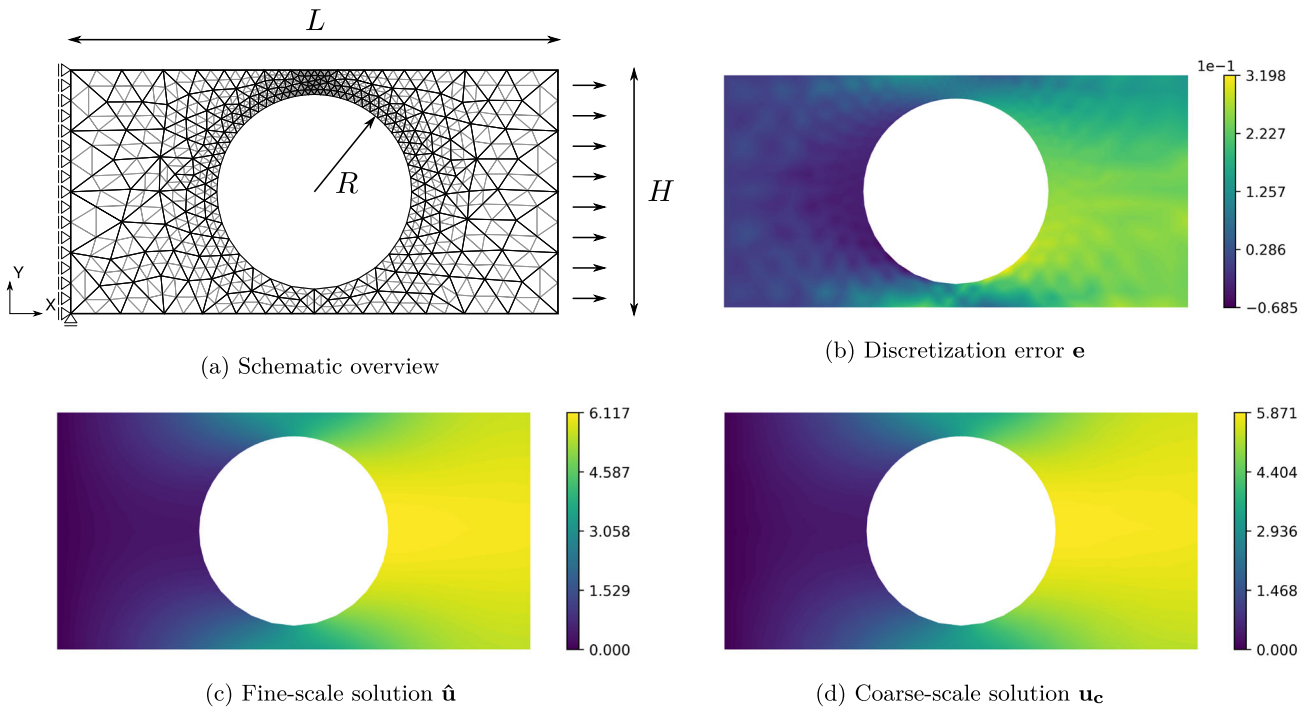(c) Fine-scale solution $\hat{\mathbf{u}}$

(d) Coarse-scale solution $\mathbf{u_c}$

Fig. 2 Overview of the perforated plate bar test case

Note the similarity to the natural kernel in Eq. (34), with $\mathbf{K}^{-1}$ and $\boldsymbol{\Sigma}_{\mathbf{f}}$ taking a similar role as $G(\mathbf{x}, \mathbf{x}')$ and $k_{\mathrm{f}}(\mathbf{x}-\mathbf{x}')$, respectively (Peker 2023). Also similarly, each sample from $\mathbf{u}$ has an equivalent sample from $\mathbf{f}$ and vice versa. Conceptually, our choice of prior is the same as Cockayne et al. (2017), except that we are working in the finite-dimensional space of the discretized system, rather than the infinite-dimensional space of the original partial differential equation. The advantage of working in the finite-dimensional space is that $\mathbf{K}^{-1}$ is computable, and as a result the natural prior can still be used.

Given this choice of prior and using Eq. (20), the posterior distribution of the displacement field is given by:

$$\mathbf{u}|\mathbf{g} \sim \mathcal{N}\left(\mathbf{m}^*, \boldsymbol{\Sigma}^*\right) \tag{38}$$

with the following posterior mean $\mathbf{m}^*$ and posterior covariance $\boldsymbol{\Sigma}^*$:

$$\mathbf{m}^* = \mathbf{K}^{-1}\boldsymbol{\Sigma}_{\mathbf{f}}\boldsymbol{\Phi}\left(\boldsymbol{\Phi}^T\boldsymbol{\Sigma}_{\mathbf{f}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\mathbf{f}$$
$$\boldsymbol{\Sigma}^* = \mathbf{K}^{-1}\left(\mathbf{I} - \boldsymbol{\Sigma}_{\mathbf{f}}\boldsymbol{\Phi}\left(\boldsymbol{\Phi}^T\boldsymbol{\Sigma}_{\mathbf{f}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\right)\boldsymbol{\Sigma}_{\mathbf{f}}\mathbf{K}^{-1} \tag{39}$$

The presence of $\mathbf{K}^{-1}$ in Eq. (39) might appear in conflict with our previously stated requirement of sparsity in the covariance matrices, since the inverse of $\mathbf{K}$ is typically full. However, using an ensemble, the prior and posterior distributions can be approximated and sampled without needing to explicitly compute this matrix inverse; only fine-scale linear solves are necessary. The details on this ensemble approximation can be found in Appendix B.

Naturally, the need for fine-scale linear solves makes the computational cost of the proposed method on par with obtaining the fine-scale finite element solution, rather than that of the coarse-scale solve as one might hope. Although acceleration of the method falls beyond the scope of this paper, we do want to highlight two potential strategies to alleviate the computational cost. The first is to employ Langevin dynamics–based sampling schemes similar to Akyildiz et al. (2021), which relies on $\boldsymbol{\Sigma}^{-1}$ rather than $\boldsymbol{\Sigma}$ to sample the posterior. A second potential approach is to approximate the posterior using a finite number of conjugate gradient iterations. In Wenger et al. (2023), an approach is presented to acccount for the error this introduces in a consistent Bayesian manner with little additional computation cost.

3.2 White noise prior

Within the natural prior framework, the main choice that remains is what right-hand side covariance function $k_{\mathrm{f}}(\mathbf{x}, \mathbf{x}')$ to assume. For now, we will follow the choice of Cockayne et al. (2017) to use the prior from Owhadi (2015), and assume $k_{\mathrm{f}}(\mathbf{x}, \mathbf{x}')$ to be a Dirac delta function $\delta(\mathbf{x})$, scaled by a single hyperparameter α:

$$k_{\mathrm{f}}(\mathbf{x}, \mathbf{x}') = \alpha^2\delta(\mathbf{x} - \mathbf{x}') \tag{40}$$

This defines a white noise field over $f(\mathbf{x})$ with a standard deviation that is equal to α. The covariance matrices $\mathbf{\Sigma_f}$ and $\mathbf{\Sigma}$ then follow directly from Eqs. (36) and (37):

$$\mathbf{\Sigma_f} = \alpha^2 \mathbf{M} \quad \mathbf{\Sigma} = \alpha^2 \mathbf{K}^{-1} \mathbf{M} \mathbf{K}^{-1} \tag{41}$$

where $\mathbf{M}$ is the fine-scale (square and symmetric) mass matrix, given by:

$$M_{ij} = \int_\Omega \phi_i(\mathbf{x}) \phi_j(\mathbf{x}) \, \mathrm{d}\mathbf{x} \tag{42}$$

Note that under this choice of prior covariance, the sparsity requirement that was put on $\mathbf{\Sigma}$ has been met. The resulting posterior mean vector and covariance matrix are then given by:

$$\begin{aligned} \mathbf{m}^* &= \mathbf{K}^{-1} \mathbf{M} \mathbf{\Phi} \left(\mathbf{\Phi}^T \mathbf{M} \mathbf{\Phi} \right)^{-1} \mathbf{\Phi}^T \mathbf{f} \\ \mathbf{\Sigma}^* &= \alpha^2 \mathbf{K}^{-1} \left(\mathbf{I} - \mathbf{M} \mathbf{\Phi} \left(\mathbf{\Phi}^T \mathbf{M} \mathbf{\Phi} \right)^{-1} \mathbf{\Phi}^T \right) \mathbf{M} \mathbf{K}^{-1} \end{aligned} \tag{43}$$

It can be seen that for this choice of prior, the hyperparameter α does not affect the posterior mean, and only serves as a scaling factor of the posterior covariance. Given this hyperparameter-independence, we choose to simply set $\alpha = 1$ for the remainder of this work. A small observation noise ($\sigma_e^2 = 10^{-12}$) is added to the term $\mathbf{\Phi}^T \mathbf{M} \mathbf{\Phi}$ in Eq. (43), to ensure that this matrix is invertible.

In Fig. 3a, the resulting prior and posterior distributions for the tapered bar problem are shown for the number of coarse elements m equal to 4. Several pieces of information about the problem, in absence of knowledge of the right-hand side term, can be found encoded in the prior. We can see how the enforcement of boundary conditions described in Fig. 2.4 indeed results in a distribution whose samples respect the boundary conditions imposed at $x = 0$ and $x = 1$. Furthermore, a larger prior standard deviation is found in the region where the bar is thinner, reflecting the fact that a small perturbation in the right-hand side in this region would have a more pronounced effect on the displacement field. Considering the posterior distribution, we see that its mean falls between the coarse- and fine-scale reference solutions. Lastly, it can be seen that the region where the posterior standard deviation is largest corresponds to the region where the discretization error is largest.

In Figs. 3b and d, we have increased the number of degrees of freedom of the coarse mesh m to 16 and 64 respectively, to study its effect on the posterior distribution. As the coarse-scale solution approaches the fine-scale solution, the posterior mean approaches the fine-scale solution accordingly. Additionally, the posterior standard deviation shrinks along with the discretization error until the coarse mesh density meets the fine one at $m = n = 64$. At this point, only a

small posterior standard deviation remains due to the small observation noise that was included in the model.

In Fig. 4, the posterior moments are plotted when the same prior distribution is applied to the two-dimensional perforated plate problem. We find that the results for this two-dimensional test case are quite different from those for the previous one-dimensional case. It can be observed in Fig. 4a that the posterior mean almost exactly matches the fine-scale solution shown in Fig. 2c. An explanation for this can be found by considering Eq. (39) and noting that the posterior mean $\mathbf{m}^*$ is equivalent to the reference solution $\hat{\mathbf{u}}$, except that the force vector $\mathbf{f}$ has been replaced by $\hat{\mathbf{f}}$, a weighted projection of $\mathbf{f}$ onto the column space of $\mathbf{\Phi}$:

$$\hat{\mathbf{f}} = \mathbf{P} \mathbf{f} = \mathbf{\Sigma_f} \mathbf{\Phi} \left(\mathbf{\Phi}^T \mathbf{\Sigma_f} \mathbf{\Phi} \right)^{-1} \mathbf{\Phi}^T \mathbf{f} \tag{44}$$

In other words, $\mathbf{f}$ is mapped to the coarse space, scaled, mapped back to the fine space and rescaled to obtain $\hat{\mathbf{f}}$. The quality of this projection depends on the weights given by $\mathbf{\Sigma_f}$, and there exists a sense in which the choice of $\mathbf{\Sigma_f} = \mathbf{M}$ is optimal: it minimizes the projection error of the forcing term $f(\mathbf{x})$ to the coarse space $\mathcal{W}^h$ in the L^2-norm (Larson and Bengzon (2013), Theorem 1.1):

$$\arg\min_{f^h(\mathbf{x}) \in \mathcal{W}^h} \| f(\mathbf{x}) - f^h(\mathbf{x}) \|_{L^2(\Omega)}^2 = \boldsymbol{\psi}(\mathbf{x})^T \left(\mathbf{\Phi}^T \mathbf{M} \mathbf{\Phi} \right)^{-1} \mathbf{\Phi}^T \mathbf{f} \tag{45}$$

This optimality helps explain the close correspondence between the fine-scale solution and posterior mean given in Figs. 2c and 4a, respectively.

Though this might appear to be a desirable property, for the purposes of modeling discretization error, it is actually detrimental. To understand this, let us consider the following equality:

$$\mathbf{\Sigma}^* \mathbf{\Sigma}^{-1} \hat{\mathbf{u}} = \hat{\mathbf{u}} - \mathbf{m}^* \tag{46}$$

This expression can be easily verified by substituting the expressions for $\hat{\mathbf{u}}$, $\mathbf{\Sigma}$, $\mathbf{m}^*$ and $\mathbf{\Sigma}^*$ found in Eqs. 39 to 26. The left-hand side of Eq. (46) can be understood as quantifier of the amount of "contraction" of the prior distribution due to the observed data. In the extreme case where there is no contraction of the covariance, we find that the posterior covariance matrix $\mathbf{\Sigma}^*$ is equal to the prior covariance matrix $\mathbf{\Sigma}$, and consequently $\mathbf{\Sigma}^* \mathbf{\Sigma}^{-1} = \mathbf{I}$ and $\mathbf{m}^* = \mathbf{0}$. At the other extreme, where the posterior covariance is given by $\mathbf{\Sigma}^* = \epsilon \mathbf{I}$ and we let $\epsilon \to 0$, we find that the left-hand side approaches the null vector and as a result $\mathbf{m}^* \to \hat{\mathbf{u}}$. As more observations are included, the posterior distribution moves from the former extreme to the latter.
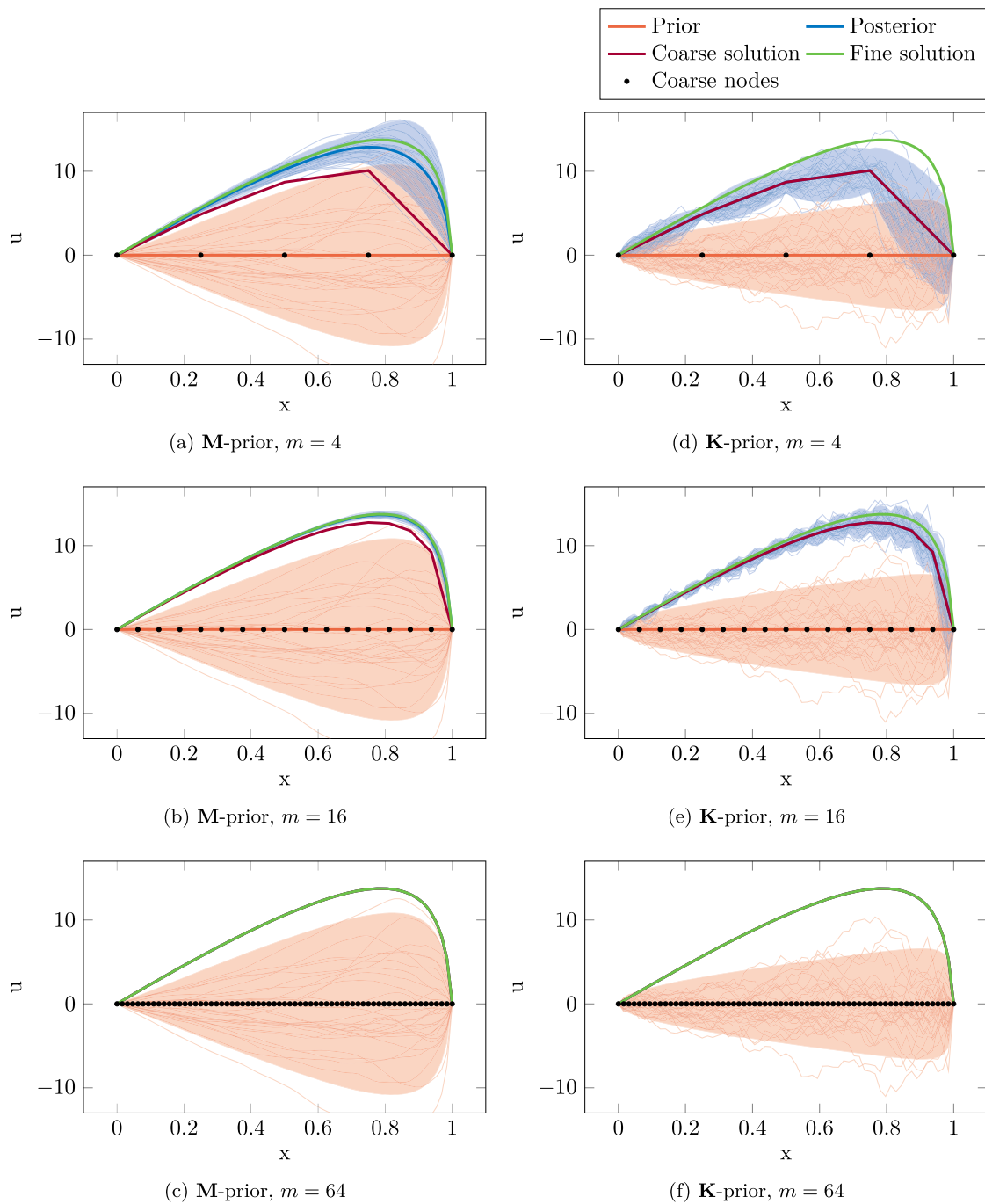
Fig. 3 Prior and posterior distributions of the 1D tapered bar problem. For comparison, the fine-scale and coarse-scale reference solutions have been included. From each distribution, 30 samples have been plotted. The shaded regions correspond to the 95% credible intervals of the distributions

It becomes clear that the posterior mean vector $\mathbf{m}^*$ and posterior covariance matrix $\mathbf{\Sigma}^*$ are inextricably linked. This property is not necessarily a problematic one. In fact, from the typical probabilistic numerics point of view, where the solving procedure is interpreted as an inherently probabilistic process (Hennig et al. 2022), the fact that the posterior covariance tends to zero as the posterior mean approaches the true solution is the desired kind of behavior. However, if

our goal is to have the discretization error reflected in the posterior covariance, then this connection to the posterior mean does pose a problem: it is not possible to simultaneously obtain a posterior mean that approaches the true solution and a posterior covariance that is indicative of the coarse-scale discretization error. And indeed, when comparing the posterior standard deviation σ^* in Fig. 4b to the discretization error
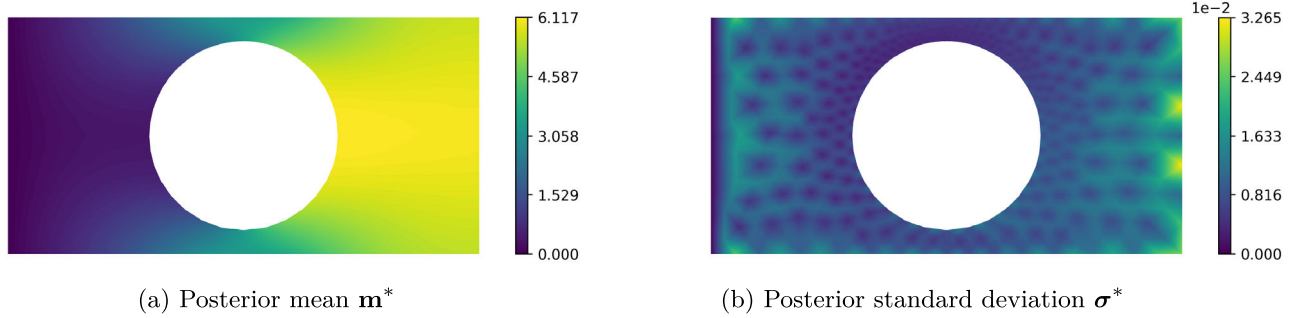
(a) Posterior mean $\mathbf{m}^*$

(b) Posterior standard deviation $\boldsymbol{\sigma}^*$

Fig. 4 Posterior moments of the perforated plate test case with $\boldsymbol{\Sigma_f} = \mathbf{M}$

e in Fig. 2b, we see that the regions of largest discretization error are not reflected in the posterior standard deviation.

3.3 Green's function prior

This crucial observation motivates us to reevaluate our initial choice of prior. Given how Eq. (46) relates the posterior covariance matrix $\boldsymbol{\Sigma}^*$ to the difference between the reference solution $\hat{\mathbf{u}}$ and the posterior mean vector $\mathbf{m}^*$, it makes sense to choose a prior that will yield a posterior mean equal to the coarse-scale solution $\mathbf{u_c}$. Additionally, from a discretization error modeling point of view, it is more sensible to have a posterior mean that is equal to the coarse-scale solution $\mathbf{u_c}$ than to have a posterior mean that improves on it. After all, the aim from the outset has been to interpret the finite element discretization error as a source of uncertainty surrounding the coarse-scale finite element solve.

In Pförtner et al. (2023), a method is presented to construct a prior whose posterior mean matches exactly the coarse-scale finite element solution $\mathbf{u_c}$ from an initial prior an arbitrary mean function $m(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$. However, we will opt instead for the method presented in Bilionis (2016), which is to set the prior covariance function equal to the Green's function $G(\mathbf{x}, \mathbf{x}')$ of the partial differential equation at hand. For Poisson's equation, this choice of prior yields the following right-hand side covariance function $k_\mathrm{f}(\mathbf{x}, \mathbf{x}')$:

$$k_\mathrm{f}(\mathbf{x}, \mathbf{x}') = -\Delta \delta(\mathbf{x} - \mathbf{x}') \tag{47}$$

Substitution of this expression into Eq. (36), applying integration by parts and subsequent substitution into Eq. (37) yields the following expressions for $\boldsymbol{\Sigma_f}$ and $\boldsymbol{\Sigma}$:

$$\boldsymbol{\Sigma_f} = \mathbf{K} \quad \boldsymbol{\Sigma} = \mathbf{K}^{-1} \tag{48}$$

Intuitively, we again find $\mathbf{K}^{-1}$ as the finite-dimensional counterpart of $G(\mathbf{x}, \mathbf{x}')$. The advantages of introducing the fine-scale discretization as a stand-in for the infinite-dimensional partial differential equation once again become apparent:

the fact that Green's function is generally unavailable does not pose a problem anymore. Furthermore, the objection raised in Bilionis (2016) that for Poisson's equation in two or three dimensions, the Green's function $G(\mathbf{x}, \mathbf{x}')$ is infinite at $\mathbf{x} = \mathbf{x}'$ and can therefore not be a useful indicator of model uncertainty does not apply in our case: for any valid finite element discretization the finite-dimensional inverse stiffness matrix $\mathbf{K}^{-1}$ only has finite-valued entries. In the phrasing of Alberts and Bilionis (2023), the introduction of the fine-scale discretization offers a way to truncate the integration over functions at the smallest scales.

This choice of prior in turn results in the following posterior mean vector and covariance matrix:

$$\mathbf{m}^* = \boldsymbol{\Phi}\left(\boldsymbol{\Phi}^T \mathbf{K} \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^T \mathbf{f}$$
$$\boldsymbol{\Sigma}^* = \mathbf{K}^{-1} - \boldsymbol{\Phi}\left(\boldsymbol{\Phi}^T \mathbf{K} \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^T \tag{49}$$

Note that according to Eqs. (22) and (24), $\boldsymbol{\Phi}^T \mathbf{K} \boldsymbol{\Phi}$ and $\boldsymbol{\Phi}^T \mathbf{f}$ are equal to the coarse stiffness matrix $\mathbf{K_c}$ and coarse force vector $\mathbf{g}$, respectively. As a result, we find that indeed the posterior mean vector $\mathbf{m}^*$ is exactly equal to the solution of the coarse system $\mathbf{u_c}$, projected to the fine space. Returning now to Eq. (46), we find that for this choice of prior, this expression simplifies to a surprisingly simple relationship between the posterior covariance matrix $\boldsymbol{\Sigma}^*$ and the discretization error **e** as defined in Eq. (27):

$$\boldsymbol{\Sigma}^* \mathbf{f} = \mathbf{e} \tag{50}$$

Since this relation holds for any fine-scale force vector $\mathbf{f}$, and $\boldsymbol{\Sigma}^*$ is independent of $\mathbf{f}$, this posterior covariance matrix $\boldsymbol{\Sigma}^*$ can be used to determine the discretization error **e** for an arbitrary forcing term. In this sense, $\boldsymbol{\Sigma}^*$ can be said to fully encode the discretization error associated with the geometry and discretization of the problem at hand.

In Fig. 3d–f, the prior and posterior distributions that follow when applying this prior to the tapered bar problem are shown. Again, the number of coarse elements m is equal to 4, 16 and 64, respectively. As expected, the posterior mean

$\mathbf{m}^*$ can be seen to equal the coarse-scale solution $\mathbf{u_c}$ in these figures. Similar to Fig. 3a to 3c, the largest posterior standard deviation σ^* is found in the region where the bar is thinnest. However, for this prior there is a notable reduction of the posterior standard deviation around the coarse-scale nodes. This reduction in the standard deviation is reflective of the fact that at these nodes, the coarse solution is more accurate than in the regions between the coarse-scale nodes, where the solution is interpolated via the coarse-scale shape functions.

Another notable difference between the two priors in Fig. 3 is the smoothness of the samples. We see in Fig. 3a–c that the samples from the white noise prior presented in Fig. 3.2 have a visible smoothness to them. In contrast, the samples from the Green's function prior shown in Fig. 3d–f appear jagged and rough. In fact, for this one-dimensional Poisson problem, each sample $\tilde{u}(x)$ drawn from the Green's function prior $k(x, x') = G(x, x')$ can be shown to be continuous, but nowhere differentiable. This is the result of the fact that at $x = x'$, the Green's function is continuous (i.e. $\lim_{\delta \to 0} G(x - \delta, x) = \lim_{\delta \to 0} G(x + \delta, x)$), but at that same point its derivative is discontinuous (i.e. $\lim_{\delta \to 0} G'(x - \delta, x) \neq \lim_{\delta \to 0} G'(x + \delta, x)$) (Bayin 2006). The samples of a Gaussian process are mean-square continuous if $k(\mathbf{x}, \mathbf{x}')$ is continuous at $\mathbf{x} = \mathbf{x}'$ and are k times mean-square differentiable if $k(\mathbf{x}, \mathbf{x}')$ is $2k$ times differentiable at $\mathbf{x} = \mathbf{x}'$ (Rasmussen and Williams 2005). From the fact that the Green's function is not differentiable at $x = x'$, it thus follows that the samples drawn from this process are everywhere continuous but nowhere differentiable. Note that this only applies to the infinite-dimensional solution space $\mathcal{V}$. The finite-dimensional space $\mathcal{V}^h$ spanned by the fine-scale shape functions $\boldsymbol{\phi}(x)$ is still weakly once-differentiable for both priors.

We now turn to the perforated plate example, for which the results are shown in Fig. 5. The posterior mean $\mathbf{m}^*$ in Fig. 5a can be seen to exactly match the coarse-scale finite element solution $\mathbf{u_c}$ in Fig. 2d for this problem as well. Unfortunately, the posterior covariance σ^* shown in Fig. 5b appears again to bear little resemblance to the discretization error $\mathbf{e}$ from Fig. 2d. This might seem surprising, given the direct relationship between posterior covariance $\boldsymbol{\Sigma}^*$ and discretization error $\mathbf{e}$ given in Eq. (50). Indeed, we can multiply the posterior covariance $\boldsymbol{\Sigma}^*$ by the fine-scale force vector $\mathbf{f}$ to recover the discretization error exactly (see Fig. 5c), but this does not translate to a posterior standard deviation σ^* that can be interpreted directly. This is a consequence of the fact that the posterior covariance $\boldsymbol{\Sigma}^*$ depends only on the material stiffness (via $\mathbf{K}$) and node locations (via $\boldsymbol{\Phi}$), but not on the magnitude of the force vector $\mathbf{f}$ at those locations. One benefit that results from this independence is that given the posterior covariance matrix $\boldsymbol{\Sigma}^*$ from one load case, it is possible to compute the discretization error for any other load case

virtually for free.[3] However, the drawback of this independence is that, since the discretization error $\mathbf{e}$ does depend on the load applied to the structure, a load-independent posterior standard deviation σ^* cannot adequately represent the discretization error for any specific load case. Paradoxically, because the posterior covariance matrix $\boldsymbol{\Sigma}^*$ encodes the discretization error $\mathbf{e}$ for all load cases simultaneously, it fails to represent the discretization error for any one load case in particular. This paradox is not unique to our Bayesian formulation of the finite element method, and arises in many Gaussian process–based probabilistic solver of differential equations, including meshfree probabilistic solvers (Bilionis 2016; Cockayne et al. 2017) and probabilistic methods of weighted residuals (Pförtner et al. 2023). In all these cases, the error between the posterior mean function $m^*(\mathbf{x})$ and exact solution $u(\mathbf{x})$ is dependent on the right-hand side term, but the posterior covariance function $k^*(\mathbf{x}, \mathbf{x}')$ meant to represent this error is not.

3.4 Incorporating force term information

This raises the question whether it is possible to break this independence of the posterior covariance matrix $\boldsymbol{\Sigma}^*$ and the fine-scale force vector $\mathbf{f}$. Doing so appears to be necessary to capture the load-dependent discretization error $\mathbf{e}$ in the posterior standard deviation σ^*. Returning to Eq. (50), we can understand the multiplication of $\boldsymbol{\Sigma}^*$ by $\mathbf{f}$ through the eigendecomposition of $\boldsymbol{\Sigma}^*$:

$$\boldsymbol{\Sigma}^* = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{-1} \tag{51}$$

Here the columns of $\mathbf{Q}$ are the eigenvectors of $\boldsymbol{\Sigma}^*$ and $\boldsymbol{\Lambda}$ is a diagonal matrix whose entries are its eigenvalues in descending order. Since $\boldsymbol{\Sigma}^*$ is real positive definite, its eigenvalues are all positive real numbers, and $\mathbf{Q}$ is an orthogonal matrix, which implies that $\mathbf{Q}^{-1} = \mathbf{Q}^T$.

The decomposition in Eq. (51) allows for a straightforward interpretation of the multiplication of $\boldsymbol{\Sigma}^*$ by $\mathbf{f}$. First, $\mathbf{Q}^{-1}$ performs a change of basis $\tilde{\mathbf{f}} = \mathbf{Q}^{-1}\mathbf{f}$, expressing $\mathbf{f}$ in terms of the basis spanned by the eigenvectors instead of the standard basis. In this basis, $\tilde{\mathbf{f}}$ is rescaled by the eigenvalues $\boldsymbol{\Lambda}$ to obtain the discretization error $\tilde{\mathbf{e}}$ expressed in terms of the eigenbasis. Finally, $\mathbf{Q}$ performs a change of basis on $\tilde{\mathbf{e}}$ back to the standard basis $\mathbf{e} = \mathbf{Q}\tilde{\mathbf{e}}$. Since $\boldsymbol{\Lambda}$ is a diagonal matrix, the operation $\tilde{\mathbf{e}} = \boldsymbol{\Lambda}\tilde{\mathbf{f}}$ comes down to a simple element-wise multiplication:

$$\tilde{e}_i = \lambda_i \tilde{f}_i \tag{52}$$

---

[3] All that is needed is a matrix–vector multiplication of the posterior covariance matrix $\boldsymbol{\Sigma}^*$ and the fine-scale force vector $\mathbf{f}$ of the new load case.

(a) Posterior mean $\mathbf{m}^*$

(b) Posterior standard deviation $\boldsymbol{\sigma}^*$

(c) Error recovery $\boldsymbol{\Sigma}^*\mathbf{f}$

(d) Rescaled posterior standard deviation $\hat{\boldsymbol{\sigma}}^*$
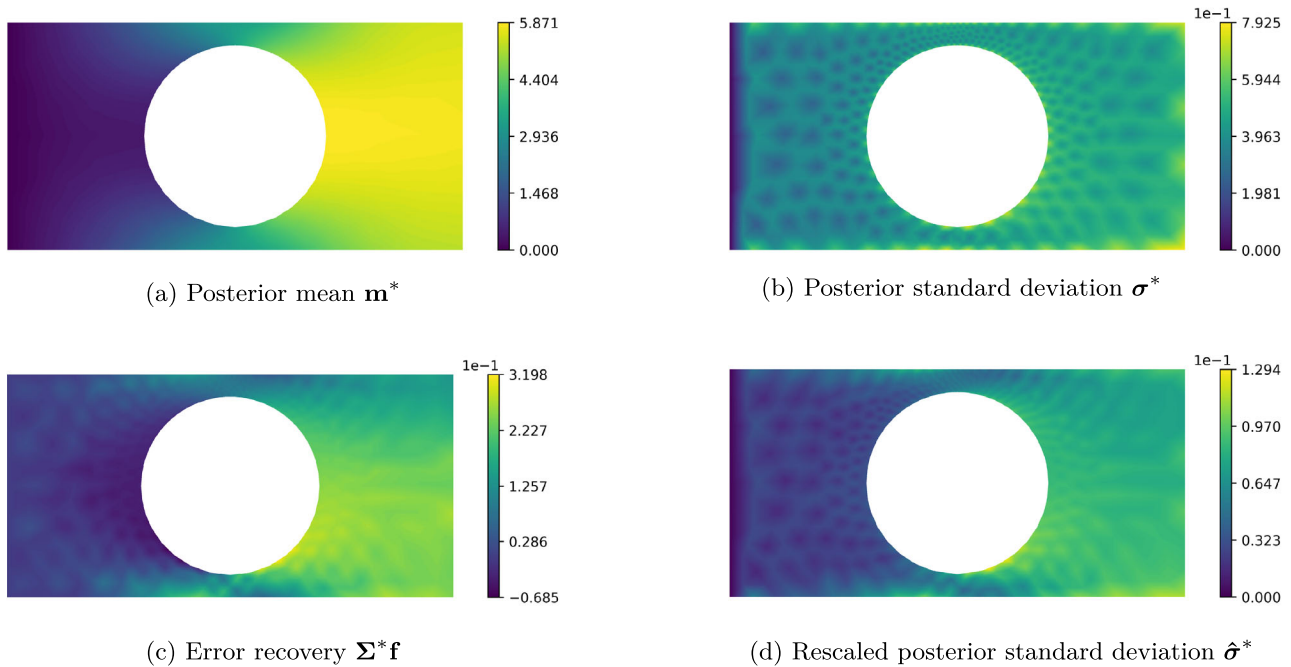
Fig. 5 Posterior moments of the perforated plate test case with $\boldsymbol{\Sigma_f} = \mathbf{K}$

Rather than interpreting Eq. (52) as a rescaling of each element of the force vector $\tilde{f}_i$ by its corresponding eigenvalue λ_i, one could argue equally well that it is the eigenvalue λ_i that is rescaled by $\tilde{f}_i$ instead. In order to break the independence of the posterior covariance matrix $\boldsymbol{\Sigma}^*$ and force vector $\mathbf{f}$, we replace the original eigenvalues λ_i with ones that are rescaled by $\tilde{f}_i$ Thus, $\boldsymbol{\Lambda}$ is replaced by a diagonal matrix $\mathbf{E}$, whose diagonal entries are given by $|\tilde{e}_i|$, yielding a new covariance matrix $\hat{\boldsymbol{\Sigma}}^*$:

$$\hat{\boldsymbol{\Sigma}}^* = \mathbf{QEQ}^{-1} \tag{53}$$

Since all entries of $\mathbf{E}$ are nonnegative, this rescaled covariance matrix $\hat{\boldsymbol{\Sigma}}^*$ is positive semi-definite, and thus a valid covariance matrix. In Fig. 5d, the standard deviation $\hat{\boldsymbol{\sigma}}^*$ of this rescaled covariance matrix is shown. Comparing to the discretization error $\mathbf{e}$ in Fig. 2b, we see a clear similarity between these two fields. At last, we appear to have arrived at a distribution with a covariance matrix that can meaningfully capture the discretization error.

One shortcoming of this ad hoc approach to incorporating forcing term information in our posterior distribution, is that it is a deviation from the Bayesian paradigm used thus far, since there is no guarantee that there exists an equivalent prior distribution that would yield this rescaled posterior covariance matrix. Additionally, if there does exist an equivalent prior, it is unclear what posterior mean this equivalent prior would produce. Our motivation for presenting this approach nonetheless is to demonstrate not only that it is impossible to

obtain an interpretable posterior standard deviation $\boldsymbol{\sigma}^*$ if the posterior covariance matrix $\boldsymbol{\Sigma}^*$ is independent of the forcing term $\mathbf{f}$, but also that it is possible to obtain an interpretable standard deviation by incorporating forcing term information.

4 Conclusions

In this work, we presented a Bayesian approach to the modeling of finite element discretization error. A Gaussian process prior is assumed over the solution space, which is conditioned on the force vector from a finite element discretization. To avoid the computation of intractable integrals, a second, finer mesh is introduced, which is assumed to be sufficiently fine to represent the true solution. The two meshes are constructed in a hierarchical manner, such that the coarse-scale shape functions can be fully expressed in terms of fine-scale shape functions. The Gaussian process prior on the solution space yields a normal distribution prior on the fine-scale solution vector. For linear partial differential equations, conditioning this prior on the coarse-scale force vector produces a normally distributed posterior on the solution vector.

Two different prior covariance functions have been investigated: a white noise prior covariance on the forcing term, and a Green's function prior covariance on the solution term. The white noise prior covariance is shown to produce a posterior mean vector that is close to the fine-scale reference solution. However, an undesirable consequence of this prop-

erty is that the corresponding posterior covariance matrix becomes less informative of the discretization error between the coarse-scale and fine-scale solutions. The Green's function prior, on the other hand, can be shown to produce exactly the coarse-scale solution as its posterior mean. Additionally, the discretization error can be recovered exactly from the posterior covariance matrix by multiplying it by the fine-scale force vector. Because the posterior covariance matrix does not depend on the values of the forcing term, it can be multiplied by any arbitrary forcing term to reproduce exactly the discretization error for that forcing term. The drawback of this independence, however, is that by itself, a force-independent posterior covariance matrix cannot be informative of the force-dependent discretization error. We have shown how by rescaling the eigenvalues of the posterior covariance matrix based on the fine-scale force vector, a distribution can be obtained whose standard deviation corresponds to the discretization error.

One major drawback of the proposed method, as is the case for many probabilistic numerical methods is its computational cost, since it relies on fine-scale solves to sample from the posterior. Although several potential approaches to approximate or circumvent these fine-scale solves have been identified, these ideas still need to be put into practice in future work. Furthermore, the formulation in this work has assumed linearity on the partial differential equations and Gaussianity on the prior distribution. Extensions of the method beyond these assumptions are not trivial. Finally, the underlying reason for the development of a Bayesian model for finite element discretization error is to allow for the consistent treatment of discretization error through computational pipelines. In this work, the focus has been on the forward problem, and the fundamentals of our Bayesian formulation of the finite element method. The demonstration of the method in an inverse modeling or data assimilation context has been left for future work.

Appendix A: Inhomogeneous boundary conditions

In Sect. 2.4, only homogeneous Dirichlet boundary conditions were considered. Here, we expand on this and demonstrate how both homogeneous and inhomogeneous Dirichlet and Neumann boundary conditions can be included in the model. This is accomplished by assigning a statistically independent normal distribution to the displacement along the Dirichlet boundary $\mathbf{u_d}$, as well as the force along the Neumann boundary $\mathbf{f_n}$:

$$\begin{aligned} \mathbf{u_d} &\sim \mathcal{N}\left(\mathbf{m_d}, \beta^2 \boldsymbol{\Sigma_d}\right) \\ \mathbf{f_n} &\sim \mathcal{N}\left(\mathbf{m_n}, \gamma^2 \boldsymbol{\Sigma_n}\right) \end{aligned} \tag{A1}$$

The assignment of these prior distributions to the Dirichlet and Neumann boundary conditions produces the following prior mean and covariance of the forcing term:

$$\begin{aligned} \tilde{\mathbf{m}}_{\mathbf{f}} &= \mathbf{K_{id}} \mathbf{m_d} + \mathbf{m_n} \\ \tilde{\boldsymbol{\Sigma}}_{\mathbf{f}} &= \boldsymbol{\Sigma_f} + \beta^2 \mathbf{K_{id}} \boldsymbol{\Sigma_d} \mathbf{K_{id}}^T + \gamma^2 \boldsymbol{\Sigma_n} \end{aligned} \tag{A2}$$

By adjusting β and γ, the effects of particular loads can be emphasized or de-emphasized.

This generalization allows for a modeling choice when enforcing inhomogeneous Dirichlet boundary conditions. These can be strongly enforced in the prior, by setting $\boldsymbol{\Sigma_d} = \mathbf{0}$ and making $\mathbf{m_d}$ equal to the true displacement value at the boundary. Alternatively, they can be weakly enforced by setting $\mathbf{m_d} = \mathbf{0}$, and instead assigning a non-zero covariance $\boldsymbol{\Sigma_d}$. In this case, the Dirichlet boundaries are enforced in a weak sense, because their enforcement is only due to the right-hand side modifications being included in the observations. Naturally, a combination of these two approaches, where both $\mathbf{m_d}$ and $\boldsymbol{\Sigma_d}$ are non-zero is also valid. For homogeneous Dirichlet boundary conditions, setting $\mathbf{m_d} = \mathbf{0}$ and $\boldsymbol{\Sigma_d} = \mathbf{0}$ already strongly enforces the boundary conditions, but this strong enforcement can be weakened by introducing a non-zero $\boldsymbol{\Sigma_d}$. For Neumann boundary conditions, the same modeling choice between strong and weak enforcement of the boundary conditions applies.

A final point to address is which covariance structure should be applied to the Dirichlet and Neumann covariances $\boldsymbol{\Sigma_d}$ and $\boldsymbol{\Sigma_n}$. For single point loads and single point constraints, the answer to this question is trivial, namely a null matrix, except for a unit diagonal entry associated with the point load or constraint degree of freedom. If the problem contains multiple independent point loads or constraints, the covariance structure of $\boldsymbol{\Sigma_d}$ and $\boldsymbol{\Sigma_n}$ is still relatively straightforward: in this case, the off-diagonal terms of $\boldsymbol{\Sigma_d}$ and $\boldsymbol{\Sigma_n}$ can simply be set to 0. However, if for example an inhomogeneous Dirichlet or Neumann boundary condition is applied along an edge, this assumption of independence does not hold, and a full covariance structure needs to be obtained for $\boldsymbol{\Sigma_d}$ and $\boldsymbol{\Sigma_n}$.

Appendix B: Sampling the prior and posterior

The main computational bottleneck of the method lies in the handling of large covariance matrices. Since both the prior and posterior covariance matrix are full $n \times n$ matrices, their explicit computation, storage and handling quickly becomes infeasible as n increases. Additionally, $\mathbf{K}^{-1}$ appears in the expressions for both of these matrices, which suffers from similar problems when computed explicitly. In this section, methods of sampling exactly from the prior and posterior

distributions are discussed. From these samples, the mean vectors and covariance matrices can be approximated.

B.1: Ensemble approximation

Instead of a mean vector and covariance matrix, an ensemble $\mathbf{X}$ is used to represent the prior distribution. $\mathbf{X}$ is an $n \times N$ matrix containing N samples from the prior distribution. The prior mean and covariance can be approximated by computing the sample mean $\hat{\mathbf{m}}$ and sample covariance $\hat{\boldsymbol{\Sigma}}$ of the ensemble. The accuracy of this approximation is depends on the size of the ensemble: as N increases, the sample mean vector and covariance matrix converge to their exact counterparts, but the computational cost increases accordingly. By the central limit theorem, the sample mean and covariance will converge to the true sample mean and covariance at a rate of $\frac{1}{\sqrt{N}}$ (Berry 1941).

The entries of the sample mean vector and covariance matrix are given by:

$$
\begin{aligned}
\hat{\mathbf{m}}_i &= \frac{1}{N} \sum_{j=1}^{N} \mathbf{X}_{ij} \\
\hat{\boldsymbol{\Sigma}}_{ij} &= \frac{1}{N-1} \sum_{k=1}^{N} \left(\mathbf{X}_{ik} - \hat{\mathbf{m}}_i\right) \left(\mathbf{X}_{jk} - \hat{\mathbf{m}}_j\right)
\end{aligned}
\tag{B3}
$$

The sample covariance matrix can also be expressed as the following decomposition:

$$
\hat{\boldsymbol{\Sigma}} = \frac{1}{N-1} \hat{\mathbf{F}} \hat{\mathbf{F}}^T
\tag{B4}
$$

Here, $\hat{\mathbf{F}}$ is the sample residual matrix, given by:

$$
\hat{\mathbf{F}} = \mathbf{X} - \hat{\mathbf{m}} \mathbf{1}_N^T
\tag{B5}
$$

In the expression above, $\mathbf{1}_N$ is a vector of ones of size N.

B.2: Prior sampling

One key observation to make about the prior covariance in Eq. (41) is that the $\mathbf{K}^{-1}$ term only appears as a premultiplier at the front and as a postmultiplier at the end of the expression. As a result, instead of obtaining a sample $\tilde{\mathbf{u}}$ from $\mathbf{u} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{K}^{-1}\boldsymbol{\Sigma}_{\mathbf{f}}\mathbf{K}^{-1}\right)$, a sample $\tilde{\mathbf{f}}$ can be obtained from $\mathbf{f} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{f}}\right)$. Since $\boldsymbol{\Sigma}_{\mathbf{f}}$ is a sparse matrix, the Cholesky decomposition of $\boldsymbol{\Sigma}_{\mathbf{f}}$ that is needed to sample $\tilde{\mathbf{f}}$ is relatively cheap compared to that of a full matrix. If $\boldsymbol{\Sigma}_{\mathbf{f}}$ is approximated by diagonalizing it, its Cholesky decomposition is trivial. After obtaining $\tilde{\mathbf{f}}$, we can compute $\tilde{\mathbf{u}}$ by solving $\mathbf{K}\tilde{\mathbf{u}} = \tilde{\mathbf{f}}$.

Since multiple solves are needed of the same system, but with a changing right-hand side vector, a direct solver approach is a natural choice. Here, we use the CHOLMOD library (Davis 2013), which solves sparse linear systems by finding a sparse Cholesky factorization of $\mathbf{K}$, and then solving for $\tilde{\mathbf{u}}$ through backward substitution. Notice that only the cheap backward substitutions need to be repeated for each sample, and the main computational bottleneck of obtaining the factorization needs to be performed only once. Additionally, for both the white-noise prior from Sect. 3.2 as well as the Green's function prior from Sect. 3.3, $\boldsymbol{\Sigma}_{\mathbf{f}}$ and will have the same sparsity structure as $\mathbf{K}$. This means that the permutation that minimizes the fill-ins in the Cholesky decomposition of $\mathbf{K}$ can be reused for that of $\boldsymbol{\Sigma}_{\mathbf{f}}$. One drawback of using a direct solver, however, is that it tends to scale poorly as the number of degrees of freedom in the system increases. For systems with a large number of degrees of freedom, it might therefore be necessary to find alternative sampling methods, though this falls beyond the scope of this paper.

B.3: Posterior sampling from u via f

To obtain posterior samples with the mean and covariance from Eq. (43), a similar approach can be applied. Instead of computing a posterior sample $\tilde{\mathbf{u}}^*$ directly from $\mathbf{u}|\mathbf{g} \sim \mathcal{N}\left(\mathbf{m}^*, \boldsymbol{\Sigma}^*\right)$, a sample of the force vector posterior $\tilde{\mathbf{f}}^*$ is obtained, from the following distribution

$$
\mathbf{f}|\mathbf{g} \sim \mathcal{N}\left(\mathbf{m_f}^*, \boldsymbol{\Sigma}_{\mathbf{f}}^*\right)
\tag{B6}
$$

where the force posterior mean vector $\mathbf{m_f}^*$ and force posterior covariance matrix $\boldsymbol{\Sigma}_{\mathbf{f}}^*$ are given by:

$$
\mathbf{m_f}^* = \boldsymbol{\Sigma}_{\mathbf{f}} \boldsymbol{\Phi} \left(\boldsymbol{\Phi}^T \boldsymbol{\Sigma}_{\mathbf{f}} \boldsymbol{\Phi} + \sigma_e^2 \mathbf{I}\right)^{-1} \mathbf{g} = \mathbf{G_f} \mathbf{g}
\tag{B7}
$$

and

$$
\begin{aligned}
\boldsymbol{\Sigma}_{\mathbf{f}}^* &= \boldsymbol{\Sigma}_{\mathbf{f}} - \boldsymbol{\Sigma}_{\mathbf{f}} \boldsymbol{\Phi} \left(\boldsymbol{\Phi}^T \boldsymbol{\Sigma}_{\mathbf{f}} \boldsymbol{\Phi} + \sigma_e^2 \mathbf{I}\right)^{-1} \boldsymbol{\Phi}^T \boldsymbol{\Sigma}_{\mathbf{f}} \\
&= \left(\mathbf{I} - \mathbf{G_f} \boldsymbol{\Phi}^T\right) \boldsymbol{\Sigma}_{\mathbf{f}}
\end{aligned}
\tag{B8}
$$

Here, $\mathbf{G_f}$ is the force vector Kalman gain matrix, which is given by:

$$
\mathbf{G_f} = \boldsymbol{\Sigma}_{\mathbf{f}} \boldsymbol{\Phi} \left(\boldsymbol{\Phi}^T \boldsymbol{\Sigma}_{\mathbf{f}} \boldsymbol{\Phi} + \sigma_e^2 \mathbf{I}\right)^{-1}
\tag{B9}
$$

If the force posterior covariance matrix is written in its so-called Joseph form, it becomes clear how posterior force vector samples can be computed from prior force vector samples:

$$
\boldsymbol{\Sigma}_{\mathbf{f}}^* = \left(\mathbf{I} - \mathbf{G_f} \boldsymbol{\Phi}^T\right) \boldsymbol{\Sigma}_{\mathbf{f}} \left(\mathbf{I} - \mathbf{G_f} \boldsymbol{\Phi}^T\right)^T + \sigma_e^2 \mathbf{G_f} \mathbf{G_f}^T
\tag{B10}
$$

A prior force vector sample $\tilde{\mathbf{f}}$, as well as a sample $\tilde{\mathbf{e}}$ from the observation noise $\mathbf{e} \sim \mathcal{N}\left(\mathbf{0}, \sigma_e^2 \mathbf{I}\right)$ are then drawn independently. The sample from the force vector posterior is then given by:

$$
\begin{aligned}
\tilde{\mathbf{f}}^* &= \mathbf{m_f}^* + \left(\mathbf{I} - \mathbf{G_f}\boldsymbol{\Phi}^T\right)\tilde{\mathbf{f}} + \mathbf{G_f}\tilde{\mathbf{e}} \\
&= \tilde{\mathbf{f}} + \mathbf{G_f}\left(\mathbf{g} - \boldsymbol{\Phi}^T\tilde{\mathbf{f}} + \tilde{\mathbf{e}}\right)
\end{aligned}
\tag{B11}
$$

At this point the sample from the posterior distribution of the displacement field can be obtained by solving $\mathbf{K}\tilde{\mathbf{u}}^* = \tilde{\mathbf{f}}^*$.

B.4: Posterior sampling from u directly

This approach of computing posterior samples by updating prior samples based on perturbed observations of the data $\mathbf{g} - \tilde{\mathbf{e}}$ can also be applied to $\mathbf{u}$ directly. Starting from Eq. (B11), $\tilde{\mathbf{f}}$ and $\boldsymbol{\Phi}^T$ can be replaced by $\tilde{\mathbf{u}}$ and $\mathbf{H}$, respectively. The force vector Kalman gain matrix given in Eq. (B9) can be replaced by the Kalman gain matrix $\mathbf{G}$ associated with the displacement field:

$$
\begin{aligned}
\mathbf{G} &= \mathbf{K}^{-1}\boldsymbol{\Sigma_f}\boldsymbol{\Phi}\left(\boldsymbol{\Phi}^T\boldsymbol{\Sigma_f}\boldsymbol{\Phi} + \sigma_e^2\mathbf{I}\right)^{-1} \\
&= \boldsymbol{\Sigma}\mathbf{H}^T\left(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^T + \sigma_e^2\mathbf{I}\right)^{-1}
\end{aligned}
\tag{B12}
$$

A prior sample $\tilde{\mathbf{u}}$ is computed as described in Appendix B.2, and an independent observation noise sample $\tilde{\mathbf{e}}$ is obtained as before from $\mathbf{e} \sim \mathcal{N}\left(\mathbf{0}, \sigma_e^2\mathbf{I}\right)$. The posterior sample $\tilde{\mathbf{u}}^*$ is then given by:

$$
\begin{aligned}
\tilde{\mathbf{u}}^* &= \mathbf{m}^* + \left(\mathbf{I} - \mathbf{GH}\right)\tilde{\mathbf{u}} + \mathbf{G}\tilde{\mathbf{e}} \\
&= \tilde{\mathbf{u}} + \mathbf{G}\left(\mathbf{g} - \mathbf{H}\tilde{\mathbf{u}} + \tilde{\mathbf{e}}\right)
\end{aligned}
\tag{B13}
$$

Note that this approach requires two $\mathbf{K}\tilde{\mathbf{u}} = \tilde{\mathbf{f}}$ solves to obtain a single sample from the posterior, as opposed to the approach presented in Appendix B.3, where only a single solve is needed. One reason to still prefer the method of sampling directly from $\mathbf{u}$ is that the final solve of $\mathbf{K}\tilde{\mathbf{u}}^* = \tilde{\mathbf{f}}^*$ does not extend well to non-linear problems. Although non-linear partial differential equations fall beyond the scope of this paper, the option to sample directly from $\mathbf{u}$ keeps the door open to this class of problems.

Author contributions P.K. derived the core conceptualization of the Bayesian finite element method. A.P. expanded on this framework under guidance of the other authors. F.M., I.R. and A.P. created the finite element code in Python, and A.P. implemented the Bayesian finite element framework. A.P. generated all results and prepared all figures. I.R. and F.M. handled funding acquisition. A.P. wrote the main manuscript text, and all authors reviewed the manuscript.

References

Alberts, A., Bilionis, I.: Physics-informed information field theory for modeling physical systems with uncertainty quantification. J. Comput. Phys. **486**, 112100 (2023). https://doi.org/10.1016/j.jcp.2023.112100

Akyildiz, Ö.D., Duffin, C., Sabanis, S., Girolami, M.: Statistical finite elements via Langevin dynamics. arXiv:2110.11131 [cs, math, stat] (2021)

Abdulle, A., Garegnani, G.: Random time step probabilistic methods for uncertainty quantification in chaotic and geometric numerical integration. Stat. Comput. **30**(4), 907–932 (2020). https://doi.org/10.1007/s11222-020-09926-w

Abdulle, A., Garegnani, G.: A probabilistic finite element method based on random meshes: a posteriori error estimators and Bayesian inverse problems. Comput. Methods Appl. Mech. Eng. **384**, 113961 (2021). https://doi.org/10.1016/j.cma.2021.113961

Bayin, S.: Mathematical Methods in Science and Engineering, 2nd edn. John Wiley & Sons, Hoboken (2006)

Berry, A.C.: The accuracy of the Gaussian approximation to the sum of independent variates. Trans. Am. Math. Soc. **49**(1), 122–136 (1941). https://doi.org/10.2307/1990053

Bilionis, I.: Probabilistic solvers for partial differential equations. arXiv:1607.03526 [math] (2016)

Babuška, I., Miller, A.: A feedback finite element method with a posteriori error estimation: part I. The finite element method and some basic properties of the a posteriori error estimator. Comput. Methods Appl. Mech. Eng. **61**(1), 1–40 (1987). https://doi.org/10.1016/0045-7825(87)90114-9

Briol, F.-X., Oates, C.J., Girolami, M., Osborne, M.A., Sejdinovic, D.: Probabilistic integration: a role in statistical computation? Stat. Sci. **34**(1), 1–22 (2017). https://doi.org/10.1214/18-STS660

Babuška, I., Rheinboldt, W.C.: Analysis of optimal finite-element meshes in $\mathbb{R}^1$. Math. Comput. **33**(146), 435–463 (1979). https://doi.org/10.2307/2006290

Chkrebtii, O.A., Campbell, D.A., Calderhead, B., Girolami, M.: Bayesian solution uncertainty quantification for differential equations. Bayesian Anal. **11**(4), 1239–1267 (2016). https://doi.org/10.1214/16-BA1017

Conrad, P.R., Girolami, M., Särkkä, S., Stuart, A.M., Zygalakis, K.: Statistical analysis of differential equations: introducing probability measures on numerical solutions. Stat. Comput. **27**(4), 1065–1082 (2017). https://doi.org/10.1007/s11222-016-9671-0

Cockayne, J., Oates, C.J., Ipsen, I.C.F., Girolami, M.: A Bayesian conjugate gradient method (with discussion). Bayesian Anal. **14**(3), 937–1012 (2019). https://doi.org/10.1214/19-BA1145

Cockayne, J., Oates, C.J., Sullivan, T.J., Girolami, M.: Probabilistic numerical methods for partial differential equations and Bayesian inverse problems. arXiv:1605.07811 [cs, math, stat] (2017)

Cockayne, J., Oates, C.J., Sullivan, T.J., Girolami, M.: Bayesian probabilistic numerical methods. SIAM Rev. **61**(4), 756–789 (2019). https://doi.org/10.1137/17M1139357

Davis, T.A.: User Guide for CHOLMOD: a sparse Cholesky factorization and modification package. Technical Report (2013)

Diaconis, P.: Bayesian numerical analysis. In: Berger, J.O., Gupta, S.S. (eds.) Statistical Decision Theory and Related Topics IV, pp. 163–175. Springer, New York (1988)

Girolami, M., Febrianto, E., Yin, G., Cirak, F.: The statistical finite element method (statFEM) for coherent synthesis of observation data and model predictions. Comput. Methods Appl. Mech. Eng. **375**, 113533 (2021). https://doi.org/10.1016/j.cma.2020.113533

Hennig, P.: Probabilistic interpretation of linear solvers. SIAM J. Optim. **25**(1), 234–260 (2015). https://doi.org/10.1137/140955501

Hennig, P., Hauberg, S.: Probabilistic solutions to differential equations and their application to Riemannian statistics. In: Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, pp. 347–355. PMLR, Reykjavik (2014). https://proceedings.mlr.press/v33/hennig14.html

Hennig, P., Osborne, M.A., Girolami, M.: Probabilistic numerics and uncertainty in computations. Proc. R. Soc. A Math. Phys. Eng. Sci. **471**(2179), 20150142 (2015). https://doi.org/10.1098/rspa.2015.0142

Hennig, P., Osborne, M.A., Kersting, H.P.: Probabilistic Numerics: Computation as Machine Learning. Cambridge University Press, Cambridge (2022)

Kersting, H.P., Hennig, P.: Active uncertainty calibration in Bayesian ODE solvers. arXiv:1605.03364 [cs, math, stat] (2018)

Karvonen, T., Särkkä, S.: Classical quadrature rules via Gaussian processes. In: 2017 IEEE 27th International Workshop on Machine Learning for Signal Processing (MLSP), pp. 1–6 (2017). https://doi.org/10.1109/MLSP.2017.8168195

Larkin, F.M.: Gaussian measure in Hilbert space and applications in numerical analysis. Rocky Mt. J. Math. **2**(3), 379–421 (1972). https://doi.org/10.1216/RMJ-1972-2-3-379

Larson, M.G., Bengzon, F.: The Finite Element Method: Theory, Implementation, and Applications. Texts in Computational Science and Engineering, Springer, Berlin, Heidelberg (2013)

Lie, H.C., Stuart, A.M., Sullivan, T.J.: Strong convergence rates of probabilistic integrators for ordinary differential equations. Stat. Comput. **29**(6), 1265–1283 (2019). https://doi.org/10.1007/s11222-019-09898-6

Law, K., Stuart, A.M., Zygalakis, K.: Data Assimilation: A Mathematical Introduction. Texts in Applied Mathematics, Springer, Cham (2015)

Mockus, J.: Bayesian Approach to Global Optimization: Theory and Applications. Mathematics and its Applications, Springer, Dordrecht (1989)

O'Hagan, A.: Bayes-Hermite quadrature. J. Stat. Plan. Inference **29**(3), 245–260 (1991). https://doi.org/10.1016/0378-3758(91)90002-V

Owhadi, H.: Bayesian numerical homogenization. Multiscale Model. Simul. **13**(3), 812–828 (2015). https://doi.org/10.1137/140974596

Peker, U.: Analyzing the influence of prior covariances on a Bayesian finite element method. Master's Thesis, TU Delft (2023). http://resolver.tudelft.nl/uuid:880758ca-6a09-4fd8-b95d-8cfb3283cca6

Pförtner, M., Steinwart, I., Hennig, P., Wenger, J.: Physics-informed gaussian process regression generalizes linear PDE solvers. arXiv:2212.12474 [cs, math, stat] (2023)

Roininen, L., Huttunen, J.M.J., Lasanen, S.: Whittle-Matérn priors for Bayesian statistical inversion with applications in electrical impedance tomography. Inverse Probl. Imaging **8**(2), 561–586 (2014). https://doi.org/10.3934/ipi.2014.8.561

Rouse, J.P., Kerfriden, P., Hamadi, M.: A probabilistic hierarchical submodelling approach through a posteriori Bayesian state estimation of finite element error fields (2021). https://hal.archives-ouvertes.fr/hal-03462530

Raissi, M., Perdikaris, P., Karniadakis, G.E.: Numerical Gaussian processes for time-dependent and nonlinear partial differential equations. SIAM J. Sci. Comput. **40**(1), 172–198 (2018). https://doi.org/10.1137/17M1120762

Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J. Comput. Phys. **378**, 686–707 (2019). https://doi.org/10.1016/j.jcp.2018.10.045

Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. MIT Press, Cambridge (2005)

Schober, M., Duvenaud, D.K., Hennig, P.: Probabilistic ODE solvers with Runge–Kutta means. In: Advances in Neural Information Processing Systems, pp. 739–747. Curran Associates Inc, New York (2014)

Skilling, J.: Bayesian solution of ordinary differential equations. In: Maximum Entropy and Bayesian Methods, pp. 23–37. Springer, Dordrecht (1992)

Stuart, A.M.: Inverse problems: a Bayesian perspective. Acta Numer. **19**, 451–559 (2010). https://doi.org/10.1017/S0962492910000061

Teymur, O., Zygalakis, K., Calderhead, B.: Probabilistic linear multistep methods. In: Advances in Neural Information Processing Systems. Curran Associates Inc, New York (2016)

Wang, J., Cockayne, J., Chkrebtii, O.A., Sullivan, T.J., Oates, C.J.: Bayesian numerical methods for nonlinear partial differential equations. Stat. Comput. **31**(5), 1–20 (2021). https://doi.org/10.1007/s11222-021-10030-w

Wenger, J., Hennig, P.: Probabilistic linear solvers for machine learning. In: Advances in Neural Information Processing Systems, pp. 6731–6742. Curran Associates Inc, New York (2020)

Wenger, J., Pleiss, G., Pförtner, M., Hennig, P., Cunningham, J.P.: Posterior and computational uncertainty in Gaussian processes. arXiv:2205.15449 [cs, math, stat] (2023)

Yang, L., Meng, X., Karniadakis, G.E.: B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. J. Comput. Phys. **425**, 109913 (2021). https://doi.org/10.1016/j.jcp.2020.109913

Zienkiewicz, O.C., Zhu, J.Z.: A simple error estimator and adaptive procedure for practical engineering analysis. Int. J. Numer. Meth. Eng. **24**(2), 337–357 (1987). https://doi.org/10.1002/nme.1620240206