

# Effects of Interpolation Error and Bias on the Random Mesh Finite Element Method for Inverse Problems

Anne Poot  
Iuri Rocha  
Pierre Kerfriden  
Frans van der Meer

*Delft University of Technology, Delft, The Netherlands*  
*Delft University of Technology, Delft, The Netherlands*  
*Mines Paris–PSL, Évry, France*  
*Delft University of Technology, Delft, The Netherlands*

## Abstract

Bayesian inverse problems are an important application for probabilistic solvers of partial differential equations: when fully resolving numerical error is computationally infeasible, probabilistic solvers can be used to consistently model the error and propagate it to the posterior. In this work, the performance of the random mesh finite element method (RM-FEM) is investigated in a Bayesian inverse setting. We show how interpolation error negatively affects the RM-FEM posterior, and how these negative effects can be diminished. In scenarios where FEM is biased for a quantity of interest, we find that RM-FEM struggles to accurately model this bias.

Proceedings of the 1<sup>st</sup> International Conference on Probabilistic Numerics (ProbNum) 2025,  
Sophia Antipolis, France. PMLR Volume 271. Copyright 2025 with the author(s)

## 1 Introduction

Within the field of probabilistic numerics, one key area of interest is the probabilistic modeling of partial differential equation (PDE) solver error. In the past decade, many approaches have been proposed, which can be roughly divided into a Bayesian and a frequentist category. In the Bayesian category, a (usually Gaussian process) prior distribution is assumed over the solution space, which is conditioned on the PDE’s strong-form residuals [1, 2, 3]. Many classical solvers of PDEs can be reformulated in a Bayesian manner [4], including the method of lines [5], the finite element method (FEM) [6] and the finite volume method [7]. Methods in the frequentist category, on the other hand, typically rely on perturbation of the model in order to induce a distribution over model predictions to capture the model error. Although these solver-perturbing methods are often focused on solving ordinary differential equations [8, 9], they can be extended to PDEs as well [10].

In this work, we focus on a particular perturbation-based PDE solver presented by Abdulle and Garegnani [11], which aims to model finite element discretization error by randomizing the node locations of the finite element mesh. The method they present is appealing both from a theoretical and a practical point of view. Theoretically, they are able to derive two error estimators from a second moment of the distribution over the gradients of the solution, by proving equivalence to the classic error estimator from [12]. The practical appeal of the method lies in its ease of implementation: because the distribution of  $u(x)$  only depends on randomized node locations, samples can easily be computed with off-the-shelf finite element software.

One driving motivation behind the development of probabilistic approaches to FEM is their potential application to (Bayesian) inverse problems. The typical proce-

cedure to infer a set of parameters of the PDE from a set of observations involves Monte Carlo sampling, with each sample requiring an FEM solve. To keep this approach computationally feasible, it might be necessary to resort to a coarse finite element discretization. However, this will reduce the accuracy of both the FEM solution and the resulting parameter estimates. The aim of taking a probabilistic approach to FEM is not necessarily to improve the accuracy of the posterior mean, but rather to incorporate the uncertainty due to discretization error in the posterior variance. We say that a probabilistic formulation of FEM is performing well in an inverse setting if 1) its posterior mean is at least as close to the ground truth as the posterior obtained with traditional FEM, and 2) its posterior variance accurately reflects the mismatch between the posterior mean and ground truth due to discretization error. The aim of this work is to investigate the performance of RM-FEM in the context of a Bayesian inverse problem, focussing specifically on the effects of interpolation error and solver bias.

The paper is structured as follows: in [Section 2](#), we present the base model used throughout the paper and compare the results of FEM and RM-FEM in an inverse setting. The influence of interpolation error and bias on the RM-FEM posterior are investigated in [Sections 3](#) and [4](#), respectively. Finally, we present our conclusions in [Section 5](#).

## 2 Base model

The analysis in this work is largely built on the work of Abdulle and Garegnani [11, Section 4.2.1]. Before expanding on it, we first present their setup as a base model, and reproduce their results for both FEM and RM-FEM.

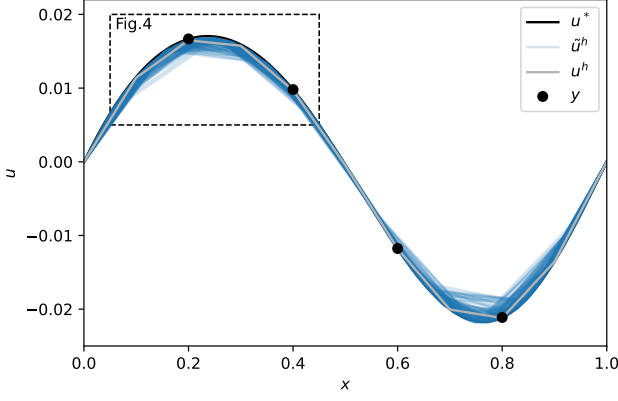


Figure 1: Comparison of the FEM solution  $u^h(x)$  and 100 samples from RM-FEM  $\tilde{u}^h(x)$ . The reference solution  $u^*(x)$  is shown in black, as well as the 4 observations  $y$  that will be used in the inverse problem. All fields have been obtained using the reference parameters  $\xi^*$ .

## 2.1 Forward model

We consider the following 1D diffusion equation:

$$\begin{aligned} -\nabla \cdot (\kappa(x) \nabla u(x)) &= f(x) \quad \forall x \in \Omega \\ u(x) &= 0 \quad \forall x \in \partial\Omega \end{aligned} \quad (1)$$

on the domain  $\Omega = (0, 1)$ . The diffusion coefficient  $\kappa(x)$  is parametrized by:

$$\kappa(x) = \exp \left\{ \sum_{k=1}^4 \frac{\xi_k}{\sqrt{\lambda_k}} \phi_k(x) \right\} \quad (2)$$

where  $\{\lambda_k, \phi_k(x)\}$  is the  $k$ th eigenpair of the differential operator  $-\frac{d}{dx^2}$ , i.e.,  $\lambda_k = k^2\pi^2$  and  $\phi_k = \sqrt{2} \sin(k\pi x)$ . We take the forcing term to be  $f(x) = \sin(2\pi x)$  and set the parameter vector of our reference solution to  $\xi = [\xi_1 \ \xi_2 \ \xi_3 \ \xi_4]^T = [1 \ 1 \ 1/4 \ 1/4]^T$ . The corresponding reference solution  $u^*(x)$  is shown in black in Fig. 1.

In order to solve Eq. (1), we discretize the domain using  $n$  elements with linear shape functions. A uniform mesh is used, so the element size  $h$  is given simply by  $h = 1/n$ . For the assembly of the stiffness matrix and force vector, a fourth-order Gauss-Legendre integration scheme is used. In Fig. 1, the finite element solution using  $n = 10$  elements is shown in gray.

## 2.2 Inverse model

We now formulate the inverse problem, aiming to infer  $\kappa(x)$  from a set of observations  $y$ . Our observation model is defined by:

$$y = \mathcal{Y}[u(x)] + \beta \quad \beta \sim \mathcal{N}(0, \sigma_e^2 \mathbf{I}) \quad (3)$$

Here, the observation operator  $\mathcal{Y}$  takes a solution field  $u(x)$  as input, and returns the evaluation of that field at 4 equally spaced locations  $\mathbf{X} = \{i/5\}_{i=1}^4$ . The observations  $y$  are generated by applying this operator to the reference solution  $u^*(x)$  and corrupting the result with i.i.d. Gaussian noise ( $\sigma_e = 10^{-5}$ ). Compared to [11], the observation noise is reduced to make the effect of discretization error more visible: for  $\sigma_e = 10^{-4}$ ,

the effect of discretization error on the posterior is too small to be noticeable compared to the uncertainty due to observation noise. We use 4 instead of 9 observation locations in order to better show the effect of excluding observation node locations from perturbation in Section 3.1.

For our Bayesian inverse model, we assume a prior distribution over  $\kappa(x)$  via the parametrization in Eq. (2) by putting a prior on its parameters:

$$p(\xi) = \mathcal{N}(0, \mathbf{I}) \quad (4)$$

The likelihood is defined according to Eq. (3), with  $\sigma_e = 10^{-5}$ . Since  $u(x)$  is fully determined by  $\xi$ , we have:

$$p(y|\xi) = p(y|u(x)) = \mathcal{N}(\mathcal{Y}[u(x)], \sigma_e^2 \mathbf{I}) \quad (5)$$

Note that the model used to fit the data is almost exactly the same model as the one used to generate the data, with the notable difference that the data generation model uses  $n^* = 1000$  elements, whereas the fitted model uses a much coarser mesh with  $n = \{10, 20, 40\}$ . Although such similarity between the data generation model and Bayesian inversion model can be considered bad practice [13], in our case it is a deliberate choice that helps distinguish the influence of discretization from other forms of model misspecification.

## 2.3 Markov chain Monte Carlo

The posterior distribution is given by Bayes' theorem:

$$p(\xi|y) = \frac{p(\xi)p(y|\xi)}{p(y)} \quad (6)$$

In order to sample from  $p(\xi|y)$  we employ a random walk Metropolis algorithm [14]. The algorithm is started at  $\xi_0 = 0$ , and the initial proposal distribution  $q(\xi)$  is set to be equal to the prior distribution  $p(\xi)$ . We employ a burn-in period of 10 000 samples, during which the covariance of the proposal distribution is scaled isotropically to obtain a good acceptance ratio. After the burn-in period, the covariance of the proposal distribution is fixed, the burn-in samples are discarded, and another  $N = 10\,000$  samples are drawn. All results presented in this paper relate only to this second set of samples.

## 2.4 RM-FEM

For our random mesh finite element implementation, we perturb the node locations according to [11, Equation 2.4]:

$$\tilde{X}_i = X_i + h^p \alpha_i \quad (7)$$

where  $h = 1/n$ , and  $\alpha_i \sim \mathcal{U}(-1/2, 1/2)$ . We set  $p = 1$  in light of [11, Theorem 2.9]. This distribution over node locations induces a distribution over FEM solutions  $\tilde{u}(x)$ , of which 100 samples are shown in blue in Fig. 1.

In the random mesh setting,  $u(x)$  is now no longer fully determined by  $\xi$ . Therefore, we need to marginalize

over all random meshes  $\tilde{\mathbf{X}}$  to compute the likelihood:

$$p(\mathbf{y}|\boldsymbol{\xi}) = \int_{\tilde{\mathbf{X}}} p(\mathbf{y}|\boldsymbol{\xi}, \tilde{\mathbf{X}}) d\tilde{\mathbf{X}} \quad (8)$$

There are two approaches to perform this marginalization in an inverse setting, laid out in [15]. The first approach, Monte Carlo within Metropolis [16], replaces the integral in Eq. (8) by a sample approximation:

$$p(\mathbf{y}|\boldsymbol{\xi}) \approx \frac{1}{M} \sum_{j=1}^M p(\mathbf{y}|\boldsymbol{\xi}, \tilde{\mathbf{X}}_j) \quad (9)$$

In [17], the resulting approximate posterior distribution is shown to converge asymptotically to the true posterior  $p(\boldsymbol{\xi}|\mathbf{y})$  as  $M$  grows large. The second approach, Metropolis within Monte Carlo, performs the marginalization by approximating the posterior instead:

$$p(\boldsymbol{\xi}|\mathbf{y}) \approx \frac{1}{M} \sum_{j=1}^M p(\boldsymbol{\xi}|\mathbf{y}, \tilde{\mathbf{X}}_j) \quad (10)$$

This involves a full random walk Metropolis run on each perturbed mesh, and averaging the resulting  $M$  sets of samples obtain an ensemble approximation of  $p(\boldsymbol{\xi}|\mathbf{y})$ .

Although the latter approach is easier to implement than the former, it tends to produce a rough multimodal approximation of  $p(\boldsymbol{\xi}|\mathbf{y})$  for the problem we are interested in here. For this reason, we opt for the first approach, and approximate the likelihood  $p(\mathbf{y}|\boldsymbol{\xi})$  according to Eq. (9), with  $M = 10$ .

## 2.5 Results

In Fig. 2, the resulting posterior distributions  $p(\kappa(x)|\mathbf{y})$  are shown, using both FEM and RM-FEM and for  $h = \{1/10, 1/20, 1/40\}$ . To get a better insight in the fitted parameters, a pairgrid plot of  $p(\boldsymbol{\xi}|\mathbf{y})$  based on the same data is shown in Fig. 3. The mean, standard deviation, as well as the error between the mean and ground truth are given in Table 1. Looking at the FEM results in Fig. 3a, we can see that the width of the posterior distributions is not affected by the number of elements in the mesh  $n$ . This is not unexpected, since the shape of the posterior is determined only by the prior and the likelihood, neither of which accounts for the model misspecification due to discretization error. This results in overconfidence of the model for  $h = 1/10$  and  $h = 1/20$ , where the posterior probability density at the ground truth is essentially zero. For  $h = 1/40$ , the model no longer appears overconfident, since at this point the uncertainty due to discretization error has become sufficiently small compared to the uncertainty due to observation noise. In Table 1, we see that for  $h = 1/40$  the error in the posterior mean is less than 1 standard deviation for each  $\xi_i$ .

Turning to the RM-FEM results in Fig. 3b, we can see that the number of elements used in the model is affecting the width of the posterior distribution that is obtained. Indeed, as  $n$  increases, the posterior mean moves in the direction of the ground truth, and the width of the posterior decreases accordingly. However,

comparing the width of the RM-FEM posterior in Fig. 3b to the FEM posterior error in Fig. 3a, the model now appears to be underconfident. Taking  $p(\xi_2|\mathbf{y})$  as an example with  $h = 1/40$ , we see in Table 1 that RM-FEM yields a 4-fold increase in the standard deviation compared to FEM (from 0.0055 to 0.0234), despite the original FEM error of 0.035 being less than 1 standard deviation. This increase in variance despite convergence having been reached can be seen as a form of underconfidence of the model.

More worryingly, the posterior mean is further away from the ground truth for the RM-FEM posterior than for the FEM posterior. For  $h = 1/10$ , the RM-FEM posterior mean falls outside the domain of the pairgrid plots for all  $\xi_i$ . Looking at  $p(\xi_1|\mathbf{y})$  with  $h = 1/20$ , the absolute error of the posterior mean has increased from 0.0038 for FEM to 0.0563 for RM-FEM. Even for  $h = 1/40$ , which is sufficient for an accurate FEM posterior, RM-FEM introduces an additional error to the posterior mean.

## 3 Interpolation error

The explanation for the underconfidence of RM-FEM and the deterioration of its posterior mean has a local and a global component. In this section, we will focus on the local component, interpolation error, by considering its relative contribution to the total discretization error. Since our observations  $\mathbf{y}$  are point evaluations of the solution field  $u(x)$ , we will consider the  $L_2$  norm of the displacement error:

$$\|e(x)\| = \left( \int_{\Omega} (u^*(x) - u^h(x))^2 d\Omega \right)^{\frac{1}{2}} \quad (11)$$

We will use a weighted norm for the nodal error of the finite element solution:

$$\|\mathbf{e}\| = \left( \sum_{i=1}^m h_i (u_i^* - u_i^h)^2 \right)^{\frac{1}{2}} \quad (12)$$

and define the relative nodal error  $\zeta$  and the relative interpolation error  $\eta$  as:

$$\zeta = \frac{\|\mathbf{e}\|}{\|e(x)\|} \quad \eta = 1 - \zeta \quad (13)$$

The nodal discretization error  $\mathbf{e}$  is relatively small for FEM solutions to smooth elliptic 1D differential equations. In fact, for a constant  $\kappa(x)$ , the finite element solution is exact in the nodes and the relative nodal error  $\zeta$  and interpolation error  $\eta$  would be exactly 0 and 1, respectively. Although for the problem at hand, the nodal FEM solution is not exact, the relative interpolation error is still close to 1.

In Fig. 4, the reference solution  $u^*(x)$  is shown in black as well as the FEM solution  $u^h(x)$  of the unperturbed mesh with  $h = 1/10$  and that of a single perturbed mesh  $\tilde{u}^h(x)$  in gray. For both FEM solves, the ground truth parameters  $\boldsymbol{\xi}^*$  have been used. Since the observation locations coincide with the unperturbed node locations, the error of the unperturbed FEM prediction in the

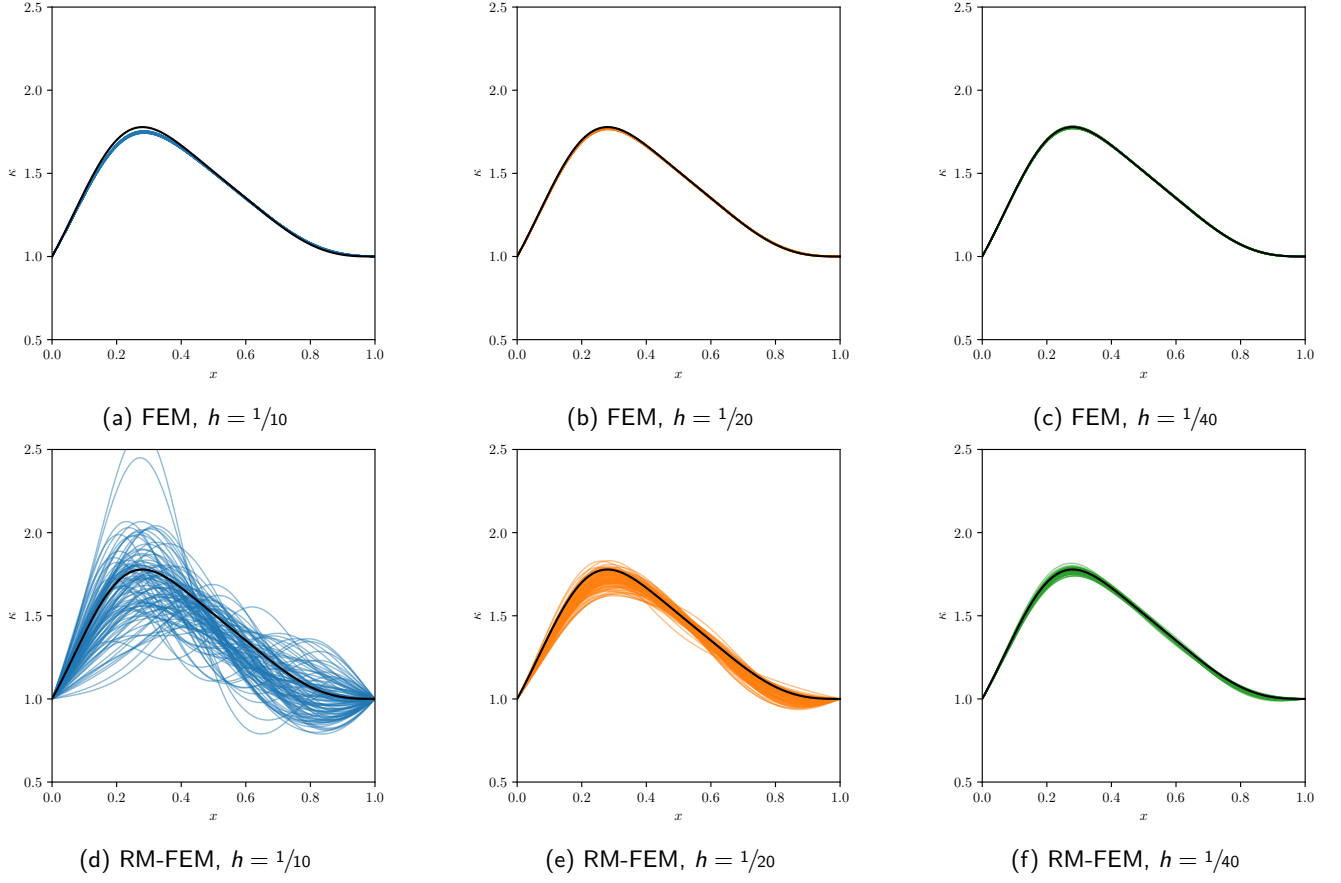


Figure 2: Reconstruction of Figure 10 from [11].

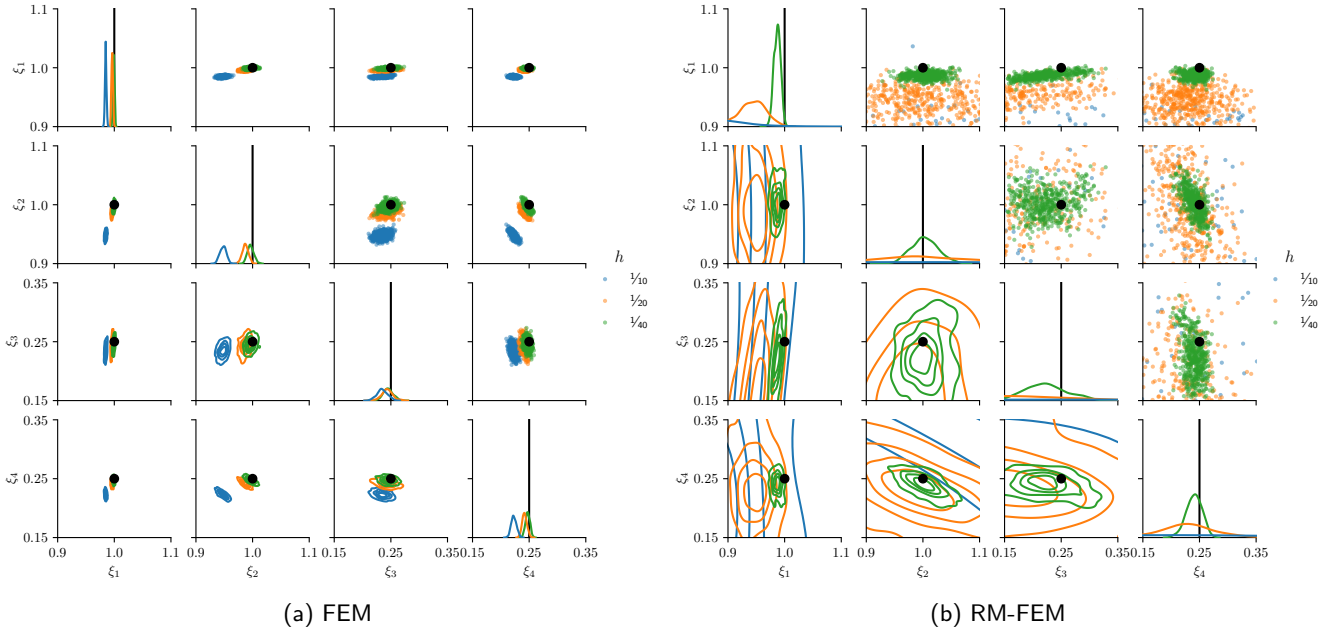


Figure 3: Pairgrid plot of the posterior  $p(\xi|y)$  for both FEM and RM-FEM, using  $h = \{1/10, 1/20, 1/40\}$ . The true values of each parameter is indicated by a black line or dot.

Table 1: Summary of the mean and standard deviation of the posterior  $p(\xi|y)$  as well as the error of the posterior mean  $|\xi^* - \mathbb{E}[p(\xi|y)]|$ , for all FEM approaches investigated in this paper.

	$h$	1D/2D FEM			1D RM-FEM			1D RM-FEM, fix obs			2D RM-FEM		
		mean	std	error	mean	std	error	mean	std	error	mean	std	error
$\xi_1$	$1/10$	.9848	.0013	.0152	.8207	.0900	.1793	.9805	.0052	.0195	.9064	.0428	.0936
	$1/20$	.9962	.0013	.0038	.9437	.0231	.0563	.9944	.0021	.0056	.9686	.0114	.0314
	$1/40$	.9991	.0013	.0009	.9869	.0061	.0131	.9986	.0014	.0014	.9926	.0033	.0074
$\xi_2$	$1/10$	.9479	.0057	.0521	.9820	.4145	.0180	.9329	.0179	.0671	1.0039	.1877	.0039
	$1/20$	.9871	.0054	.0129	.9947	.0932	.0053	.9807	.0066	.0193	.9887	.0457	.0113
	$1/40$	.9965	.0055	.0035	1.0035	.0234	.0035	.9946	.0057	.0054	.9983	.0123	.0017
$\xi_3$	$1/10$	.2349	.0089	.0151	.1913	.6465	.0587	.2264	.0243	.0236	.0845	.2961	.1655
	$1/20$	.2433	.0096	.0067	.1133	.1239	.1367	.2408	.0118	.0092	.1834	.0679	.0666
	$1/40$	.2453	.0085	.0047	.2221	.0367	.0279	.2432	.0096	.0068	.2352	.0207	.0148
$\xi_4$	$1/10$	.2230	.0046	.0270	.1433	.3002	.1067	.2175	.0213	.0325	.1713	.1247	.0787
	$1/20$	.2418	.0041	.0082	.2287	.0540	.0213	.2387	.0075	.0113	.2360	.0278	.0140
	$1/40$	.2474	.0040	.0026	.2405	.0137	.0095	.2470	.0042	.0030	.2448	.0077	.0052

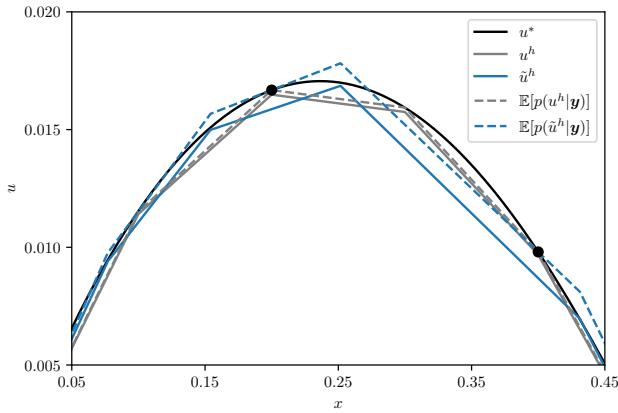


Figure 4: The FEM solution to Eq. (1) on the reference mesh and a single perturbed mesh (in gray and blue, respectively). The posterior mean that is obtained when solving the Bayesian inference problem on either mesh is shown with a dashed line. The reference solution  $u^*(x)$  and observations  $y$  are shown in black.

observation locations is significantly smaller than the error of the perturbed FEM prediction.

We now solve the inverse problem on both meshes using the procedure described in Section 2.3, and plot the resulting posterior mean in blue in Fig. 4. The smaller the distance between the FEM solution given the reference parameters  $u^h(x)|\xi^*$  and the posterior (mean) FEM solution  $\mathbb{E}[p(u^h(x)|y)]$ , the smaller the error in the likelihood will be and therefore the smaller the effect of discretization error on the posterior distribution  $p(\xi|y)$ . For this 1D problem, the relative nodal error  $\zeta$  is close to 0, and as a result the FEM solution is quite accurate, even for  $h = 1/10$ . This explains why the inverse solution using FEM shown in Figs. 2a and 3a is very accurate, even for  $h = 1/10$ .

For the perturbed mesh, we see that interpolation error significantly affects the accuracy of the prediction at the observation locations. As a result, large mismatch arises between the perturbed FEM solution given the reference parameters  $\tilde{u}^h(x)|\xi^*$  and the posterior mean of the perturbed mesh  $\mathbb{E}[p(\tilde{u}^h(x)|y)]$ . Note that this effect will occur for any perturbed mesh: the nodal loca-

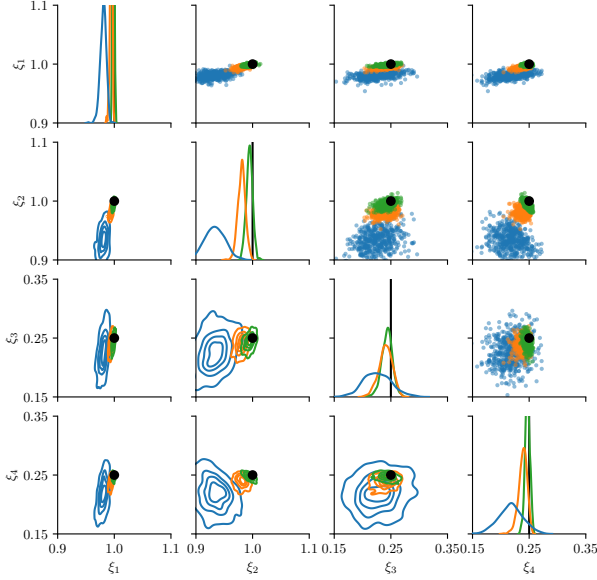
tions of the unperturbed mesh are optimal in the sense that they minimize the FEM error at the observation locations, and consequently every perturbed mesh will be suboptimal in this regard. This in turn causes a larger error in the likelihood, which translates to both the underconfidence and the deterioration of the posterior mean we observed in Fig. 3b.

### 3.1 Fixed observation nodes

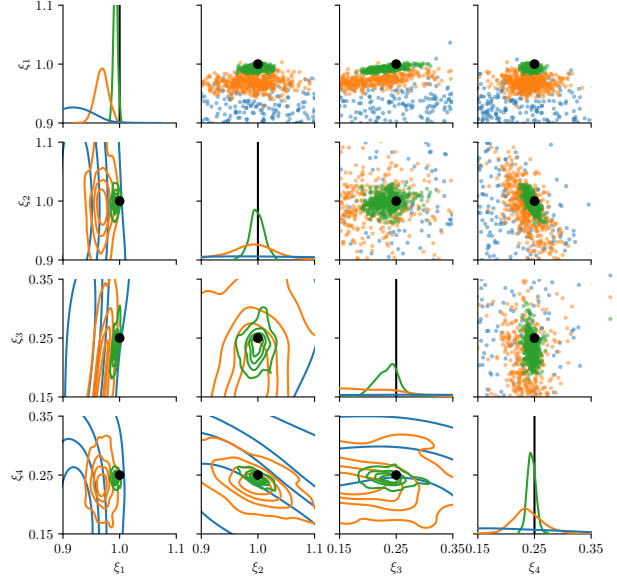
Although worrisome, this issue is not necessarily insurmountable. One potential approach to resolving it is to perturb only the nodes in the mesh that are not at one of the observation locations. Although this would be problematic for  $h = 1/5$ , in which case all nodes are observed and so none are perturbed, this approach could be feasible for finer meshes. The smaller  $h$  is, the smaller the effect of fixing the 4 observation nodes is expected to be.

The resulting posterior  $p(\xi|y)$  is shown in Fig. 5a, and its moments are summarized in Table 1 under “1D RM-FEM, fix obs”. We observe that fixing the observed nodes reduces the width of the RM-FEM posterior for all mesh densities  $h$  and parameters  $\xi_i$ , while maintaining the desired increase in confidence with mesh refinement. Additionally, although a slight increase in the error is still visible compared to the FEM posterior mean, the severe deterioration of the RM-FEM posterior mean has been mitigated by fixing the observation nodes. Furthermore, looking again at  $p(\xi_2|y)$  with  $h = 1/40$ , we see hardly any difference in the standard deviation of the RM-FEM posterior with fixed observation nodes, compared to the FEM posterior. This is a desirable property, since for this mesh density, the FEM error is small compared to the uncertainty due to observation noise. We conclude that fixing the observed nodes helps reduce both the underconfidence of RM-FEM and the deterioration of its posterior mean.





(a) RM-FEM, fixed observation nodes



(b) RM-FEM, 2D

Figure 5: Pairgrid plot of the posterior  $p(\xi|y)$  for 1D RM-FEM with fixed observation nodes, and 2D RM-FEM, using  $h = \{1/10, 1/20, 1/40\}$ . The true parameter values are indicated by a black line or dot.

### 3.2 Generalizability

We note that the conclusion of the previous section is quite specific to the 1D case with nodal observations. If observations are made over a small region rather than a single point, the effect of interpolation error would be dampened. Similarly, this relatively large interpolation error  $\eta$  is typical for FEM solutions to elliptic PDEs in 1D, but not in 2D or 3D. For more realistic problems, most of the discretization error is due to nodal error  $\zeta$  rather than interpolation error  $\eta$ . Finally, higher-dimensional problems require a different perturbation scheme to guarantee a valid finite element discretization. For example, in the case of quadrilateral elements, the maximal perturbation that still guarantees mesh validity is a disk centered on the nodes with a radius  $r = 1/4\sqrt{2}h$ . Considering only the  $x$ -component, we would obtain a semicircular distribution on the support  $(-1/4\sqrt{2}h, 1/4\sqrt{2}h)$  rather than a uniform distribution on the support  $(-1/2h, 1/2h)$ .

Although all three of these factors are expected to reduce the relative effect of interpolation error in a more realistic setting, we will single out the final point (i.e., a different perturbation scheme) to put it to the test. We change to a 2D domain  $\Omega = (0, 1) \times (0, 1/10)$ , with a Dirichlet boundary on the left and right  $\Gamma_d = \{0, 1\} \times (0, 1/10)$  and a Neumann boundary on the top and bottom  $\Gamma_n = (0, 1) \times \{0, 1/10\}$ , such that  $\Gamma_d \cup \Gamma_n = \partial\Omega$  and  $\Gamma_d \cap \Gamma_n = \emptyset$ . We adapt the PDE such that  $\kappa(\mathbf{x})$ ,  $f(\mathbf{x})$  and  $u(\mathbf{x})$  are all constant with respect to the vertical coordinate  $y$ , and the reference solution  $u^*$  as a function of  $x$  is exactly the same as the solution to Eq. (1):

$$\begin{aligned} -\nabla \cdot (\kappa(\mathbf{x}) \nabla u(\mathbf{x})) &= f(\mathbf{x}) & \forall \mathbf{x} \in \Omega \\ u(\mathbf{x}) &= 0 & \forall \mathbf{x} \in \Gamma_d \\ \frac{\partial u}{\partial y}(\mathbf{x}) &= 0 & \forall \mathbf{x} \in \Gamma_n \end{aligned} \quad (14)$$

The coarsest mesh consists of 10 square elements with

side length  $h = 1/10$  and linear shape functions. Each finer mesh is obtained by splitting each element into 4 smaller square elements. For our RM-FEM implementation, we perturb each node by choosing from a uniform distribution on a disk with radius  $r = 1/4\sqrt{2}h$ , centered on the node location on the reference mesh. The boundary nodes are perturbed in the same manner, but projected back onto the boundary  $\partial\Omega$  to ensure that the geometry of the domain is identical for each mesh. We define the observation locations for this 2D problem to be  $\mathbf{X} = \{(i/5, 1/20)\}_{i=1}^4$ , and use the same observation values  $\mathbf{y}$  as in the 1D case. When solving the inverse problem with FEM, we obtain exactly the same posterior distribution as in the 1D case, shown in Fig. 3a.

However, when using RM-FEM for this 2D case, we obtain the posterior shown in Fig. 5b, which does differ from the 1D RM-FEM posterior shown in Fig. 3b. Similarly to 1D RM-FEM with fixed observation nodes, we see in Table 1 that for this 2D problem, the width of the RM-FEM posterior has reduced significantly compared to 1D RM-FEM, and that the increase of confidence with mesh refinement is maintained. Furthermore, the error of the 2D RM-FEM posterior mean is smaller than that of 1D RM-FEM for all parameters  $\xi_i$ . Despite the difference between 1D and 2D RM-FEM not being as great as the difference between 1D RM-FEM with and without fixed observation nodes, we can conclude that the 2D perturbation scheme reduces the severity of the deterioration of the posterior mean. We expect that the deterioration of the posterior mean would vanish altogether for problems with regional observations (as opposed to point observations) and problems where the relative nodal error  $\zeta$  dwarfs the relative interpolation error  $\eta$ .

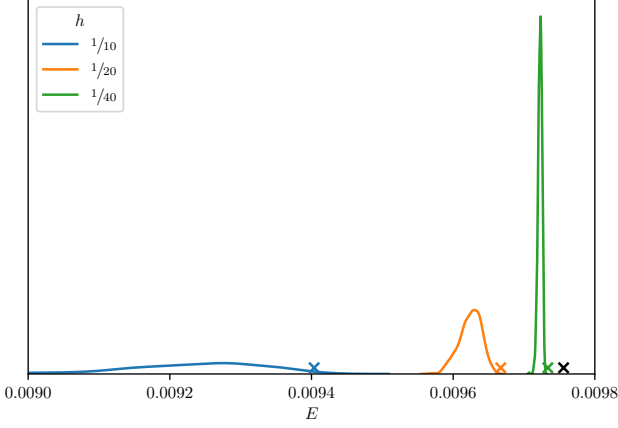


Figure 6: RM-FEM distributions of the total energy  $\tilde{E}^h$  for  $h = \{1/10, 1/20, 1/40\}$ . The total energy of the unperturbed meshes  $E^h$  are indicated by colored crosses. The reference total energy  $E^*$  is indicated by a black cross.

## 4 Bias

Although we have seen in the previous section that the adverse effects of interpolation error on the performance of RM-FEM can be dampened to some extent, some points remain unexplained. In particular, it is still unclear why RM-FEM does not necessarily improve the accuracy of the parameter estimate provided by the posterior mean compared to FEM. For instance, throughout all examples shown in Figs. 3 and 5,  $\xi_1$  can be seen to be consistently underestimated. As we will see, the reason for this lies not in some local effect like the one we explored in the previous section, but rather lies in a global bias in the FEM estimate of the solution.

To explain this global effect, we turn back to the forward problem presented in Section 2.1 and consider the total energy  $E^*$  in the system as well as its FEM approximation  $E^h$ :

$$E^* = \int_{\Omega} u(x)f(x)dx \quad E^h = \mathbf{u}^T \mathbf{f} \quad (15)$$

FEM always underestimates this quantity, as a direct result of the fact that FEM searches the optimal solution in the subspace spanned by the shape functions. This implicitly neglects the energy in the orthogonal space, and thus produces an estimate of the total energy  $E^*$  that is too low.

We now compute  $E^h$  using both FEM and RM-FEM for  $h = \{1/10, 1/20, 1/40\}$ , and show the results in Fig. 6. As expected, the total energy  $E^*$  is underestimated by FEM for  $h = 1/10$ , and approaches the reference value as the mesh is refined. We see this same underestimation occur when looking at the distributions produced by RM-FEM: none of the perturbed meshes could ever produce a total energy estimate  $E^h$  that is higher than or equal to the true total energy  $E^*$ . This underestimation of the total energy typically results in FEM solutions that are biased toward 0, and this is compensated in the inverse setting by underestimating  $\xi_1$ .

An additional observation to make about Fig. 6 is that the energy estimate provided by FEM is close to the

optimal energy of the RM-FEM distribution. This is in line with common knowledge that smooth 1D differential equations like the one studied here are best discretized by a uniform mesh. More broadly, the aim of discretization strategies is to choose a mesh such that  $E^h$  is as close to  $E^*$  as possible for a given number of nodes or elements. A fundamental limitation of RM-FEM is that it pays little heed to the care that is typically put into mesh selection. If the original FEM mesh was chosen arbitrarily, then the RM-FEM mean energy  $\mathbb{E}[\tilde{E}^h]$  would on average coincide with the FEM estimate  $E^h$ . However, since the FEM mesh is chosen close to optimally, we see a deterioration of the RM-FEM mean energy  $\mathbb{E}[\tilde{E}^h]$  when comparing against the FEM estimate  $E^h$ .

We emphasize that these results do not contradict any of the conclusions drawn in [11]. The error estimators derived from the distribution over random meshes, as well as their application to adaptive mesh refinement are not implicated by any part of our study. In fact, it may well be possible to derive an error estimator for  $E^*$  from the RM-FEM distribution given in Fig. 6, since the test for a valid error estimator is whether it shares the same order of convergence as the true error. From this point of view, whether  $E^*$  falls inside or outside the 95% confidence interval of the distribution is irrelevant. However, when attempting to derive confidence bounds for a quantity of interest, or when applying RM-FEM in an inverse setting, caution is required.

We can think of FEM as a point estimator of the solution  $u^*(x)$  and of RM-FEM as a bootstrapping method<sup>1</sup>, whose moments provide useful information about the uncertainty of our point estimate. The bootstrap can be a useful tool to estimate the variance of an estimator, and reduce the variance through bootstrap aggregation, but is often ill-equipped to estimate bias. This carries over to RM-FEM, which is useful as an error estimator, but which produces unhelpful distributions for quantities of interest for which FEM is known to be biased. The energy norm used in this section is a clear example of this and the bias we observed for  $\xi_1$  falls in the same category. It is not obvious how one should correct such biases in RM-FEM.

## 5 Conclusion

In this work, we investigated the effects of interpolation error and bias on the performance of RM-FEM in an inverse setting. In 1D, interpolation error can deteriorate the method’s performance, yielding a posterior mean that is further away from the ground truth and an overly large variance, when compared to the posterior mean that is obtained using traditional FEM. This effect can be dampened by excluding the nodes at the observation locations from the perturbation scheme, and

<sup>1</sup>The term “bootstrap” is sometimes used specifically for case resampling. We are using the term here in a broader sense, i.e., turning one dataset into  $D$  datasets, fitting  $D$  models, and leveraging their joint performance. This includes case resampling, but also encompasses other approaches (e.g., the parametric bootstrap, the Bayesian bootstrap and residual resampling).

is generally less pronounced in more practical settings. A more pernicious drawback of RM-FEM is its quantification of discretization error for quantities of interest for which FEM is known to be biased, in which case the RM-FEM distribution is biased as well, possibly even more so. Although RM-FEM makes a useful tool for error estimation and adaptive mesh refinement, we recommend caution when employing it in an inverse setting, particularly in cases where FEM produces biased estimates of the observed quantity of interest.

## References

- [1] Maziar Raissi and George Em Karniadakis. Machine Learning of Linear Differential Equations using Gaussian Processes. *Journal of Computational Physics*, 348:683–693, November 2017. arXiv:1701.02440 [cs, math, stat].
- [2] Jon Cockayne, Chris J. Oates, Tim J. Sullivan, and Mark Girolami. Probabilistic Numerical Methods for Partial Differential Equations and Bayesian Inverse Problems, July 2017. arXiv:1605.07811 [cs, math, stat].
- [3] Liu Yang, Xuhui Meng, and George Em Karniadakis. B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. *Journal of Computational Physics*, 425:109913, January 2021.
- [4] Marvin Pförtner, Ingo Steinwart, Philipp Hennig, and Jonathan Wenger. Physics-Informed Gaussian Process Regression Generalizes Linear PDE Solvers, December 2023. arXiv:2212.12474 [cs, math, stat].
- [5] Nicholas Krämer, Jonathan Schmidt, and Philipp Hennig. Probabilistic Numerical Method of Lines for Time-Dependent Partial Differential Equations. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, pages 625–639. PMLR, May 2022. ISSN: 2640-3498.
- [6] Anne Poot, Pierre Kerfriden, Iuri Rocha, and Frans van der Meer. A Bayesian approach to modeling finite element discretization error. *Statistics and Computing*, 34(5):167, August 2024.
- [7] Tim Weiland, Marvin Pförtner, and Philipp Hennig. Scaling up Probabilistic PDE Simulators with Structured Volumetric Information, June 2024. arXiv:2406.05020 [cs].
- [8] Han Cheng Lie, Andrew M. Stuart, and Tim J. Sullivan. Strong convergence rates of probabilistic integrators for ordinary differential equations. *Statistics and Computing*, 29(6):1265–1283, November 2019.
- [9] Assyr Abdulle and Giacomo Garegnani. Random time step probabilistic methods for uncertainty quantification in chaotic and geometric numerical integration. *Statistics and Computing*, 30(4):907–932, July 2020.
- [10] Patrick R. Conrad, Mark Girolami, Simo Särkkä, Andrew M. Stuart, and Konstantinos Zygalakis. Statistical analysis of differential equations: introducing probability measures on numerical solutions. *Statistics and Computing*, 27(4):1065–1082, July 2017.
- [11] Assyr Abdulle and Giacomo Garegnani. A probabilistic finite element method based on random meshes: A posteriori error estimators and Bayesian inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 384:113961, October 2021.
- [12] Ivo Babuška and Werner C. Rheinboldt. Analysis of Optimal Finite-Element Meshes in  $\mathbb{R}^1$ . *Mathematics of Computation*, 33(146):435–463, 1979. Publisher: American Mathematical Society.
- [13] Armand Wirgin. The inverse crime, January 2004. arXiv:math-ph/0401050.
- [14] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092, June 1953.
- [15] Giacomo Garegnani. Sampling Methods for Bayesian Inference Involving Convergent Noisy Approximations of Forward Maps, November 2021. arXiv:2111.03491 [math].
- [16] Mark A Beaumont. Estimation of Population Growth or Decline in Genetically Monitored Populations. *Genetics*, 164(3):1139–1160, July 2003.
- [17] Christophe Andrieu and Gareth O. Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725, April 2009. Publisher: Institute of Mathematical Statistics.