Applied and Computational Mechanics 20 (2026) 67–90

Applied and Computational Mechanics

A scalable solver for sampling-free uncertainty quantification of time-dependent partial differential equations

S. Sharma[a,*]

[a] *Department of Civil and Environmental Engineering, Carleton University, Ottawa, Canada*

Received 23 February 2026; accepted 19 June 2026

Abstract

This research addresses the uncertainty quantification of time-dependent partial differential equations (PDEs) with random parameters. The stochastic Galerkin method, a sampling-free intrusive approach, is employed instead of sampling- or quadrature-based methods to overcome the slow convergence and high computational cost associated with high-resolution models. An acoustic wave propagation problem with a log-normal random field approximation for wave speed is illustrated. The stochastic partial differential equation with the inputs and outputs expanded using polynomial chaos expansion (PCE) is transformed into a set of coupled deterministic PDEs and discretized to yield a system of linear equations. To handle the increased memory requirements with increasing mesh size, time step and number of random parameters, domain decomposition-based (DD-based) solvers are utilized. A conjugate gradient iterative solver with a two-level Neumann-Neumann preconditioner is applied to the symmetric positive definite system matrix showing their efficient scalability. This combination of the stochastic Galerkin method and DD-based solvers enables large-scale real-time applications involving acoustic waves.

Keywords: acoustic wave propagation, uncertainty quantification, parallel solvers, sampling-free methods, domain decomposition methods

1. Introduction

Uncertainty quantification (UQ) can be regarded as the process of identifying, propagating and quantifying the uncertainty in computational models [13, 31]. Uncertainty in these models can be associated with the assumptions used in modelling, inherent variabilities in parameters, initial and/or boundary conditions, the associated data for calibration, etc. Inclusion of these uncertainties into the computational modelling framework provides a response with quantified uncertainty which is essential for high consequence systems such as aerospace, military applications, medicine, epidemiology, economics, etc.

This article focuses on the parametric uncertainty (uncertainty in model parameters including initial or boundary conditions) and their quantification. A functional dependence of uncertainty from input to output is not explicitly available for most physical models. Moreover, this dependence is nonlinear even for a linear system with Gaussian randomness. For example, consider the case of a linear spring with Gaussian variability in stiffness. The uncertainty in the output can be non-Gaussian because of the nonlinear dependence of the model parameters on the output.

Monte Carlo simulations (MCS) provide an estimate of the uncertainty in output by propagating identically distributed independent samples of input through the model. The mean

*Corresponding author. Tel.: +1 613 520 26 00, e-mail: sudhipv@cmail.carleton.ca.
https://doi.org/10.24132/acm.2026.1094

estimate converges to the truth with increasing number of samples. However, the slow rate of convergence necessitates a large number of samples which is prohibitive for high-resolution PDE models. PCE-based surrogates are widely used to model parametric uncertainties in PDEs replacing MCS [5, 9, 25, 34, 35]. Random quantities are expanded in this method using orthogonal polynomials of random variables forming a subspace. Projections on to this subspace minimizing the error provide the deterministic coefficients in the expansion which are used for computing the response statistics. Non-intrusive spectral projection (NISP) or collocation methods approximate the computation of these deterministic coefficients using sampling or quadrature utilizing the deterministic code as a black box [9, 10, 20]. Even though this allows the modeller to reuse an existing deterministic model, an increasing number of stochastic parameters leads to an exponential increase in the number of samples or quadrature points necessary for accuracy. This becomes prohibitive for high-resolution PDE models.

Uncertainty for high-resolution PDE models can also be tackled using intrusive/sampling-free approaches [13]. Unlike sampling-based approaches, which require a large number of realizations to achieve convergence, intrusive methods directly propagate input uncertainty through the governing model. This is achieved by altering the original PDE by a stochastic PDE. This extra effort in one-time conversion can be compensated by an accurate solution compared to NISP and a reduction in execution time for the same set of expansion parameters [10]. The spectral stochastic finite element method (SSFEM), also known as the stochastic Galerkin method, was originally developed by Ghanem and Spanos [13] and is a popular intrusive approach for solving stochastic PDEs with random parameters. In the following text, the term stochastic Galerkin method is used for consistency. Stochastic Galerkin method essentially transforms the original stochastic PDE into a set of coupled deterministic PDEs by a Galerkin projection. Finite element discretizations of these deterministic PDEs lead to a large set of linear or nonlinear equations. This stochastic system contains a coupled set of $(P + 1) \times N$ equations, where P is the number of terms in the spectral expansion of the output and N is the number of deterministic degrees of freedom in the finite element discretization. Solving this ensuing large system of equations becomes a formidable task considering the coupled structure and computational requirements for inversion.

In this article, a DD-based algorithm is utilized to tackle this problem by distributing the memory and computations to many processes. A non-overlapping DD-based Neumann-Neumann preconditioner is used along with the conjugate gradient iterative solver which accelerates its convergence. Increasing the number of random variables or order of expansion decreases the sparsity and increases the condition number of the stochastic system which can be handled using these two-level preconditioners. Thus, application of DD-based methods provides a natural way to decompose the overwhelmingly large system of equations from the intrusive method and allows the construction of preconditioners to iterative solvers. Wave propagation problems are solved using collocation and stochastic Galerkin methods having random coefficients in [4, 6, 14, 23, 24, 36]. However, they do not utilize domain decomposition methods or a large number of random variables which this article focuses on.

On the other hand, "embarrassingly parallel" algorithms can be easily applied for sampling-based approaches by solving a single or many samples in individual cores. However, for high-resolution PDE models a single core may not be able to handle the memory requirement even for the deterministic system. Application of DD-based methods to reduce the memory requirement can also be applied here by decomposing the deterministic problem into subdomains before sampling. However, for each sample evaluation, the communication cost between processes can significantly impact the overall time to solution and reduce the efficiency of the method.

Even though the deterministic time-dependent problem has an improved condition number compared to the static case with addition of mass and damping matrices, significant challenges still exist in terms of stochastic problems with respect to parallel scaling.

1. The random input parameters of the model are nonlinearly transformed to the output and thus small changes to the inputs can have drastic effects on the output for a stochastic time-dependent problem.
2. A wave propagation problem deals with values in the range sufficiently high as the amplitude of forcing or initial condition and significantly small in the range of zero. These varying data ranges and the Courant-Friedrichs-Lewy (CFL) condition necessitate a sufficiently small time step and spatial discretization. It is also important to note that the quantification of uncertainty can become erroneous without a sufficiently discretized deterministic model.
3. The uncertainty for a time-dependent problem varies drastically with time for the wave propagation problem. The probability density functions (PDFs) at different time steps of the time-dependent problem can be non-Gaussian and with multiple peaks [27]. To capture these uncertainties, we need higher order expansions which will increase the dimensionality of the problem.

From the above considerations, the main contributions of the article are listed as

1. development of the mathematical framework for a DD-based two-level parallel scalable solver for the time-dependent problem in a deterministic and stochastic setting;
2. application of the stochastic Galerkin method for a time-dependent acoustic wave propagation problem with a random field representation of wave speed;
3. demonstrating the scalability of the solver with respect to random parameters.

This article is organised as follows: The deterministic acoustic wave propagation and the deterministic two-level Neumann-Neumann preconditioner formulations are explained first. The stochastic wave propagation with a random process representation of wave speed is formulated using the stochastic Galerkin method along with the probabilistic version of the preconditioner. The numerical section illustrates the strong and weak scalability of the solver for the stochastic problem. Many related numerical investigations are presented in this section: A stochastic one-dimensional axial bar vibration problem is considered to compare the different uncertainty quantification methods for a time-dependent problem. The stochastic Galerkin and non-intrusive methods are compared against Monte Carlo solutions. The verifications of solutions for deterministic and stochastic problems in two dimensions are also presented.

2. Acoustic wave propagation in deterministic media

Acoustic wave equation models the propagation of pressure waves through a fluid medium [28]. The initial disturbance moves fluid particles to transfer momentum to adjacent particles, sustaining wave propagation across the medium. The speed of particles in motion is a characteristic of the medium called characteristic speed of propagation or wave speed. This characteristic speed is a function of the temperature, pressure and other characteristics of the fluid medium. In the deterministic setting, this wave speed is assumed to be a constant, whereas in some cases it might be necessary to take the spatio-temporal variation of the wave speed into consideration for accurately predicting the wave mechanics. The amplitude of oscillation for each particle is measured as the sound pressure whose strength depends on the initial disturbance. The initial-boundary value problem for a wave propagating through a two-dimensional domain D with a

constant wave speed c can be written as [28, 37]

$$\frac{\partial^2 u(\mathbf{x},t)}{\partial t^2} + \eta \frac{\partial u(\mathbf{x},t)}{\partial t} - c^2 \, \nabla^2 u(\mathbf{x},t) = f(\mathbf{x},t) \quad \text{in } D \times (0,T), \tag{1}$$

$$u(\mathbf{x},t) = \Phi(\mathbf{x},t) \quad \text{on } \Gamma_D \times (0,T), \tag{2}$$

$$\nabla u(\mathbf{x},t) \cdot \hat{\mathbf{n}} = \Psi(\mathbf{x},t) \quad \text{on } \Gamma_N \times (0,T), \tag{3}$$

$$u(\mathbf{x},t) = u_0(\mathbf{x}) \quad \text{in } D, \tag{4}$$

$$\frac{\partial u(\mathbf{x},t)}{\partial t} = v_0(\mathbf{x}) \quad \text{in } D, \tag{5}$$

where u is the acoustic pressure, $\Phi(\mathbf{x},t)$ and $\Psi(\mathbf{x},t)$ are the functional forms of Dirichlet and Neumann boundary conditions, $u_0(\mathbf{x})$ and $v_0(\mathbf{x})$ represent the initial pressure and the corresponding initial rate of change of the acoustic pressure, and $f(\mathbf{x},t)$ is the external forcing. $\Gamma_D \cup \Gamma_N$ is the complete boundary formed by the union of Dirichlet and Neumann parts and $\hat{\mathbf{n}}$ represents the outward normal to the boundary at $\mathbf{x}$. The weak form for the deterministic acoustic wave equation is given by [37]

$$\left(\frac{\partial^2 u}{\partial t^2}, v\right) + \eta \left(\frac{\partial u}{\partial t}, v\right) + a(u,v) = (f,v) \quad \forall v \in V, \tag{6}$$

where

$$a(u,v) = c^2 \int_D \nabla u \cdot \nabla v \; \mathrm{d}\mathbf{x}, \qquad (f,v) = \int_D f v \; \mathrm{d}\mathbf{x},$$

and the test function $v(\mathbf{x})$ lies on $V = \{v \in H^1(\Omega) : v = 0 \, \text{on} \, \Gamma_D\}$. The above weak form is semi-discretized using the finite element method leading to [26, 37]

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f}, \tag{7}$$

where $\mathbf{M}, \mathbf{C}, \mathbf{K}$ are the mass, damping and stiffness matrices, respectively, $\mathbf{f}$ is the force vector, and $\mathbf{u}$ is the acoustic pressure. The damping matrix $\mathbf{C}$ is assumed to be the Rayleigh damping generated by combining mass and stiffness matrices with appropriate coefficients [16, 27]. Depending on the wavelength or frequency, spatial mesh resolution Δh and the time integration step Δt should be selected carefully (i.e., high-frequency waves having short wavelengths require small Δh and Δt). The system of ordinary differential equations (ODEs) in (7) can be solved numerically using an appropriate time integration scheme. Explicit time integration schemes provide the solution of the current time step in terms of the previous time step. However, these schemes are conditionally stable and result in smaller time step requirements following the CFL condition for a one-dimensional problem as [11, 19, 26]

$$\frac{c \; \Delta t}{\Delta h} = C_{\text{CFL}} \leq C_{\text{max}}, \tag{8}$$

where c is the maximum wave speed in the domain, C_{CFL} is the CFL number, and C_{max} is the maximum allowable value. This condition becomes more restrictive for two-dimensional and three-dimensional problems. Implicit time integration schemes provide the solution for the current time step by solving a system of equations involving both the current and previous time steps. The implicit scheme is computationally more expensive than the explicit method but has unconditional stability which allows larger time steps and thus reduces total time consumption

[18, 26]. The Newmark-beta implicit scheme, which has a second-order convergence and an unconditional stability property, is selected in this study for the time integration [1, 15]. The acoustic pressure and its higher order time-derivatives (such as velocity v and acceleration a) at a given time step for this scheme can be written as [15]

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t\, \mathbf{v}_n + \frac{1}{2}(1-2\zeta)\Delta t^2\, \mathbf{a}_n + \zeta \Delta t^2\, \mathbf{a}_{n+1}, \tag{9}$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + (1-\gamma)\Delta t\, \mathbf{a}_n + \gamma \Delta t\, \mathbf{a}_{n+1}, \tag{10}$$

where $\gamma = 1/2$ and $\zeta = 1/4$ are constants. Consequently, the time-discretized ODEs (7) lead to the following linear systems:

$$\tilde{\mathbf{K}}\mathbf{u}_{n+1} = \tilde{\mathbf{f}}_{n+1}, \tag{11}$$

where $\tilde{\mathbf{K}}$ is called the transient operator or dynamic stiffness matrix expressed as [11]

$$\tilde{\mathbf{K}} = \mathbf{M}\frac{1}{\zeta \Delta t^2} + \mathbf{C}\frac{\gamma}{\zeta \Delta t} + \mathbf{K} \tag{12}$$

with the forcing $\tilde{\mathbf{f}}_{n+1}$ computed as

$$\tilde{\mathbf{f}}_{n+1} = \mathbf{f}_{n+1} + \mathbf{f}_m + \mathbf{f}_c, \tag{13}$$

where

$$\mathbf{f}_m = \mathbf{M}\tilde{\mathbf{u}}_m, \qquad \mathbf{f}_c = \mathbf{C}\tilde{\mathbf{u}}_c, \tag{14}$$

and

$$\tilde{\mathbf{u}}_m = \frac{\mathbf{u}_n + \Delta t\, \mathbf{v}_n}{\zeta \Delta t^2} + \frac{(1-2\zeta)\, \mathbf{a}_n}{2\zeta}, \qquad \tilde{\mathbf{u}}_c = \gamma \Delta t\, \tilde{\mathbf{u}}_m - \mathbf{v}_n - (1-\gamma)\Delta t\, \mathbf{a}_n. \tag{15}$$

Note that the equation (11) at each time step involves the solution of only a static system with varying forcing terms. The transient stiffness operator in (12) does not depend on time and can be pre-assembled and reused reducing computational time and storage requirements. The coefficient matrix in (11) is symmetric and positive-definite which can be solved using a conjugate-gradient-based iterative solver with a non-overlapping DD-based preconditioner as described in the next section.

3. Deterministic two-level Neumann-Neumann preconditioner

Domain decomposition methods rely on the idea of dividing the original domain into overlapping or non-overlapping subdomains with smaller sub-problems which can be solved in parallel using high-performance computing systems. All the neighbouring subdomains share a common boundary with a set of nodes called interface nodes, as shown in Fig. 1. The interface nodes can also be decomposed into corner and remaining nodes as shown which will be used for constructing the coarse grid as explained later. The remaining nodes are interior to each subdomain and are called interior nodes.

Eliminating the interior unknowns from each of these subdomains generates a dense linear system generally referred to as the interface Schur complement system [2, 22, 33]. The Schur complement system has a better condition number than that of the original system and thus takes less computational cost to solve [2, 32]. The direct solution to the Schur complement system and back substitution to find the interior unknowns is generally known as the direct substructuring method [2]. For large Schur complement systems, an iterative solver with an

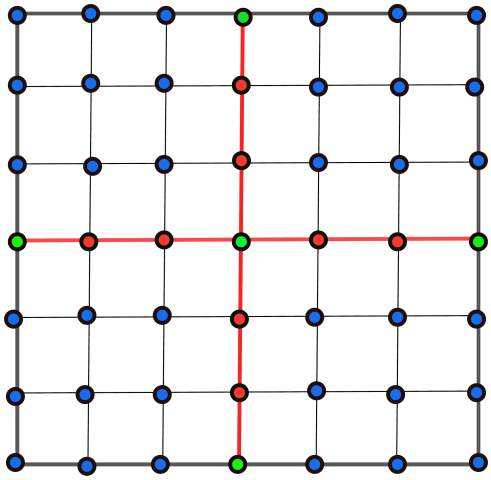


Fig. 1. A unit square domain with four subdomains showing interior (*blue*), interface (*red* + *green*), corner (*green*) and remaining (*red*) nodes

efficient preconditioner is used. Several variants of non-overlapping DD-based preconditioners are developed which scale with respect to mesh size and number of subdomains. This article utilizes a two-level Neumann-Neumann preconditioner which is analogous to the balancing domain decomposition by constraints (BDDC) method [3, 21].

At each time step, a linear system as in (11) is solved using the DD-based preconditioner as

$$\tilde{\mathbf{K}}\mathbf{u} = \tilde{\mathbf{f}}. \tag{16}$$

The nodes inside each subdomain are decomposed as interior $\mathbf{u}_I^s$ and interface $\mathbf{u}_\Gamma^s$ nodes as [32]

$$\mathbf{u}^s = \begin{Bmatrix} \mathbf{u}_I^s \\ \mathbf{u}_\Gamma^s \end{Bmatrix}, \tag{17}$$

where the subscripts I and Γ denote the interior and the interface, respectively. A restriction operator $\mathbf{R}_s$, consisting of zeros and ones mapping the global interface nodes to the subdomain interface nodes, is constructed as [32]

$$\mathbf{u}_\Gamma^s = \mathbf{R}_s \mathbf{u}_\Gamma. \tag{18}$$

The equilibrium equations for each of the subdomains can be written in terms of interior and interface nodes as [32]

$$\begin{bmatrix} \tilde{\mathbf{K}}_{II}^s & \tilde{\mathbf{K}}_{I\Gamma}^s \\ \tilde{\mathbf{K}}_{\Gamma I}^s & \tilde{\mathbf{K}}_{\Gamma\Gamma}^s \end{bmatrix} \begin{Bmatrix} \mathbf{u}_I^s \\ \mathbf{u}_\Gamma^s \end{Bmatrix} = \begin{Bmatrix} \tilde{\mathbf{f}}_I^s \\ \tilde{\mathbf{f}}_\Gamma^s \end{Bmatrix}, \tag{19}$$

where $\tilde{\mathbf{f}}_I^s$ and $\tilde{\mathbf{f}}_\Gamma^s$ correspond to the interior and interface forces, respectively, which contain contributions from mass and damping terms of the previous time step, as shown in (14). The decomposed force vector with these contributions is written as

$$\begin{Bmatrix} \tilde{\mathbf{f}}_{mI}^s \\ \tilde{\mathbf{f}}_{m\Gamma}^s \end{Bmatrix} = \begin{bmatrix} \mathbf{M}_{II}^s & \mathbf{M}_{I\Gamma}^s \\ \mathbf{M}_{\Gamma I}^s & \mathbf{M}_{\Gamma\Gamma}^s \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}}_{mI}^s \\ \tilde{\mathbf{u}}_{m\Gamma}^s \end{Bmatrix} \tag{20}$$

and

$$\begin{Bmatrix} \tilde{\mathbf{f}}^s_{cI} \\ \tilde{\mathbf{f}}^s_{c\Gamma} \end{Bmatrix} = \begin{bmatrix} \mathbf{C}^s_{II} & \mathbf{C}^s_{I\Gamma} \\ \mathbf{C}^s_{\Gamma I} & \mathbf{C}^s_{\Gamma\Gamma} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{u}}^s_{cI} \\ \tilde{\mathbf{u}}^s_{c\Gamma} \end{Bmatrix}, \tag{21}$$

where $\tilde{\mathbf{u}}^s_{mI}$, $\tilde{\mathbf{u}}^s_{m\Gamma}$, $\tilde{\mathbf{u}}^s_{cI}$ and $\tilde{\mathbf{u}}^s_{c\Gamma}$ for corresponding interior and interface nodes can be calculated as in (15). The global system combining all subdomains is written as

$$\begin{bmatrix} \tilde{\mathbf{K}}^1_{II} & \cdots & 0 & \tilde{\mathbf{K}}^1_{I\Gamma}\mathbf{R}_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \tilde{\mathbf{K}}^{n_s}_{II} & \tilde{\mathbf{K}}^{n_s}_{I\Gamma}\mathbf{R}_{n_s} \\ \mathbf{R}_1^T\tilde{\mathbf{K}}^1_{\Gamma I} & \cdots & \mathbf{R}^T_{n_s}\tilde{\mathbf{K}}^{n_s}_{\Gamma I} & \displaystyle\sum_{s=1}^{n_s}\mathbf{R}_s^T\tilde{\mathbf{K}}^s_{\Gamma\Gamma}\mathbf{R}_s \end{bmatrix} \begin{Bmatrix} \mathbf{u}^1_I \\ \vdots \\ \mathbf{u}^{n_s}_I \\ \mathbf{u}_\Gamma \end{Bmatrix} = \begin{Bmatrix} \tilde{\mathbf{f}}^1_I \\ \vdots \\ \tilde{\mathbf{f}}^{n_s}_I \\ \displaystyle\sum_{s=1}^{n_s}\mathbf{R}_s^T\tilde{\mathbf{f}}^s_\Gamma \end{Bmatrix}. \tag{22}$$

After applying Gaussian block elimination, the above equation can be recast as [29]

$$\begin{bmatrix} \tilde{\mathbf{K}}^1_{II} & \cdots & 0 & \tilde{\mathbf{K}}^1_{I\Gamma}\mathbf{R}_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \tilde{\mathbf{K}}^{n_s}_{II} & \tilde{\mathbf{K}}^{n_s}_{I\Gamma}\mathbf{R}_{n_s} \\ 0 & \cdots & 0 & \displaystyle\sum_{s=1}^{n_s}\mathbf{R}_s^T\mathbf{S}_s\mathbf{R}_s \end{bmatrix} \begin{Bmatrix} \mathbf{u}^1_I \\ \vdots \\ \mathbf{u}^{n_s}_I \\ \mathbf{u}_\Gamma \end{Bmatrix} = \begin{Bmatrix} \tilde{\mathbf{f}}^1_I \\ \vdots \\ \tilde{\mathbf{f}}^{n_s}_I \\ \displaystyle\sum_{s=1}^{n_s}\mathbf{R}_s^T\mathbf{g}^s_\Gamma \end{Bmatrix}, \tag{23}$$

where $\mathbf{S}_s$ and $\mathbf{g}^s_\Gamma$ are the subdomain level Schur complement and corresponding force vector defined as

$$\mathbf{S}_s = \tilde{\mathbf{K}}^s_{\Gamma\Gamma} - \tilde{\mathbf{K}}^s_{\Gamma I}[\tilde{\mathbf{K}}^s_{II}]^{-1}\tilde{\mathbf{K}}^s_{I\Gamma}, \qquad \mathbf{g}^s_\Gamma = \tilde{\mathbf{f}}^s_\Gamma - \tilde{\mathbf{K}}^s_{\Gamma I}[\tilde{\mathbf{K}}^s_{II}]^{-1}\tilde{\mathbf{f}}^s_I. \tag{24}$$

It is possible to combine the subdomain level interface problems into a global system of equations as [32]

$$\mathbf{S}\mathbf{u}_\Gamma = \mathbf{g}_\Gamma, \tag{25}$$

where

$$\mathbf{S} = \sum_{s=1}^{n_s}\mathbf{R}_s^T\mathbf{S}_s\mathbf{R}_s, \qquad \mathbf{g}_\Gamma = \sum_{s=1}^{n_s}\mathbf{R}_s^T\mathbf{g}^s_\Gamma. \tag{26}$$

Note that by eliminating the interior unknowns from the global system of equations, a smaller dense interface problem is generated (to be solved before interior unknowns can be computed). A two-level Neumann-Neumann preconditioner can be constructed for the interface problem by decomposing the interface into corner and remaining nodes as [32]

$$\begin{Bmatrix} \mathbf{u}^s_r \\ \mathbf{u}^s_c \end{Bmatrix} = \begin{Bmatrix} \mathbf{R}^r_s \\ \mathbf{R}^c_s \end{Bmatrix} \mathbf{u}^s_\Gamma, \tag{27}$$

where $\mathbf{u}^s_c$ are the corner nodes (also known as cross points shown in Fig. 1), and $\mathbf{u}^s_r$ are the remaining nodes. The corner nodes are identified as the nodes on the interface that are shared among more than two subdomains and the end nodes of interface edges. The remaining interface nodes are on the interface shared between only two subdomains. A local Dirichlet problem on these subdomains can be solved by assuming the values of the interface unknowns. It is thus possible to calculate the residual force developed on the subdomain boundaries owing to the

erroneous initial guess of interface values. This can be written in terms of the Schur complement of each subdomain as [32]

$$\mathbf{r}_\Gamma^s = \mathbf{S}_s \mathbf{u}_\Gamma^s - \mathbf{g}_\Gamma^s. \tag{28}$$

The contributions from all the subdomains are aggregated to obtain the global residual force vector from which the incremental solution for the next iteration (j) is computed as [32]

$$\mathbf{r}_{\Gamma_j} = \sum_{s=1}^{n_s} \mathbf{R}_s^T \mathbf{r}_\Gamma^s. \tag{29}$$

The residual force vector for each subdomain can then be computed as [32]

$$\mathbf{r}_\Gamma^s = \mathbf{D}_s \mathbf{R}_s \mathbf{r}_{\Gamma_j}, \tag{30}$$

where the partition of the unity matrix $\mathbf{D}_s$ is computed to distribute the subdomain contributions without redundancy as [22]

$$\sum_{s=1}^{n_s} \mathbf{R}_s^T \, \mathbf{D}_s \, \mathbf{R}_s = \mathbf{I}. \tag{31}$$

The increment in solution at the boundary due to the residual vector can be computed from the subdomain Schur complement problem as defined earlier. Generally, it is not necessary to assemble the subdomain level Schur complement explicitly and compute its inverse. Only an effect of this matrix is calculated by solving a Neumann problem as [32]

$$\begin{bmatrix} \tilde{\mathbf{K}}_{ii}^s & \tilde{\mathbf{K}}_{ir}^s & \tilde{\mathbf{K}}_{ic}^s \\ \tilde{\mathbf{K}}_{ri}^s & \tilde{\mathbf{K}}_{rr}^s & \tilde{\mathbf{K}}_{rc}^s \\ \tilde{\mathbf{K}}_{ci}^s & \tilde{\mathbf{K}}_{cr}^s & \tilde{\mathbf{K}}_{cc}^s \end{bmatrix} \begin{Bmatrix} \mathbf{x}_i^s \\ \mathbf{u}_r^s \\ \mathbf{u}_c^s \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \tilde{\mathbf{f}}_r^s \\ \tilde{\mathbf{f}}_c^s \end{Bmatrix}, \tag{32}$$

where

$$\begin{Bmatrix} \tilde{\mathbf{f}}_r^s \\ \tilde{\mathbf{f}}_c^s \end{Bmatrix} = \begin{Bmatrix} \mathbf{R}_s^r \\ \mathbf{R}_s^c \end{Bmatrix} \mathbf{r}_\Gamma^s. \tag{33}$$

Eliminating the variable $\mathbf{x}_i^s$ by solving a Dirichlet problem, one obtains [32]

$$\begin{bmatrix} \mathbf{S}_{rr}^s & \mathbf{S}_{rc}^s \mathbf{B}_c^s \\ \sum_{s=1}^{n_s} \mathbf{B}_c^{sT} \mathbf{S}_{cr}^s & \sum_{s=1}^{n_s} \mathbf{B}_c^{sT} \mathbf{S}_{cc}^s \mathbf{B}_c^{sT} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_r^s \\ \mathbf{u}_c^s \end{Bmatrix} = \begin{Bmatrix} \tilde{\mathbf{f}}_r^s \\ \sum_{s=1}^{n_s} \mathbf{B}_c^{sT} \tilde{\mathbf{f}}_c^s \end{Bmatrix}, \tag{34}$$

where each Schur complement matrix can be defined for the corresponding corner and remaining nodes as

$$\mathbf{S}_{\alpha\beta}^s = \mathbf{A}_{\alpha\beta}^s - \mathbf{A}_{\alpha i}^s [\mathbf{A}_{ii}^s]^{-1} \mathbf{A}_{i\beta}^s \,, \tag{35}$$

where α and β are dummy variables with values representing r and c, respectively, and $\mathbf{B}_c^s$ is the Boolean restriction operator mapping global corner nodes to local corner nodes as [32]

$$\mathbf{u}_c^s = \mathbf{B}_c^s \mathbf{u}_c. \tag{36}$$

Elimination of the remaining nodes from (34) leads to the coarse problem of our preconditioner as [32]

$$\mathbf{F}_{cc} \mathbf{u}_c = \mathbf{d}_c, \tag{37}$$

where the operator $\mathbf{F}_{cc}$ and the corresponding force vector $\mathbf{d}_c$ for corner nodes are defined as

$$\mathbf{F}_{cc} = \sum_{s=1}^{n_s} \mathbf{B}_c^{sT} \left(\mathbf{S}_{cc}^s - \mathbf{S}_{cr}^s [\mathbf{S}_{rr}^s]^{-1} \mathbf{S}_{rc}^s \right) \mathbf{B}_c^s \,, \tag{38}$$

$$\mathbf{d}_c = \sum_{s=1}^{n_s} \mathbf{B}_c^{sT} \left(\tilde{\mathbf{f}}_c^s - \mathbf{S}_{cr}^s [\mathbf{S}_{rr}^s]^{-1} \tilde{\mathbf{f}}_r^s \right). \tag{39}$$

The remaining interface unknown $\mathbf{u}_r^s$ can be computed in parallel as follows:

$$\mathbf{S}_{rr}^s \mathbf{u}_r^s = \tilde{\mathbf{f}}_r^s - \mathbf{S}_{rc}^s \mathbf{B}_c^s \mathbf{u}_c \,. \tag{40}$$

Further, the global interface solution can be computed as

$$\mathbf{u}_\Gamma = \sum_{s=1}^{n_s} \mathbf{R}_s^T \mathbf{D}_s \left(\mathbf{R}_s^{rT} \mathbf{u}_r^s + \mathbf{R}_s^{cT} \mathbf{u}_c^s \right). \tag{41}$$

Finally, the two-level Neumann-Neumann preconditioner can be expressed as [32]

$$\mathbf{M}_{\text{NNC}}^{-1} = \sum_{s=1}^{n_s} \mathbf{R}_s^T \mathbf{D}_s \left(\mathbf{R}_s^{rT} [\mathbf{S}_{rr}^s]^{-1} \mathbf{R}_s^r \right) \mathbf{D}_s \mathbf{R}_s + \mathbf{R}_0^T [\mathbf{F}_{cc}]^{-1} \mathbf{R}_0 \,, \tag{42}$$

where $\mathbf{R}_0$ is defined as

$$\mathbf{R}_0 = \sum_{s=1}^{n_s} \mathbf{B}_c^{sT} \left(\mathbf{R}_s^c - \mathbf{S}_{cr}^s [\mathbf{S}_{rr}^s]^{-1} \mathbf{R}_s^r \right) \mathbf{D}_s \mathbf{R}_s \,. \tag{43}$$

A conjugate gradient iterative solver equipped with the above two-level Neumann-Neumann preconditioner is used to solve the system (11). A pseudo-code for acoustic wave propagation using a two-level Neumann-Neumann preconditioner is outlined in Algorithm 1 in the following section. More details on the implementation and algorithms for the preconditioned conjugate gradient iterative solver with the above preconditioner can be found in [7, 8, 32].

4. Acoustic wave propagation in a random medium

The speed of propagation of acoustic waves through a medium is dependent on the specific characteristics of the medium such as pressure, temperature, density, etc., which can vary significantly. Thus, considering the wave speed as a constant parameter, as described in previous sections, is not appropriate for highly consequential applications such as medical science, military, aerospace, etc., where quantified variability in predictions is important. In this section, the uncertain nature of wave speed is considered as a random field. The stochastic PDE for acoustic waves considering the wave speed as a random field can be written as [24]

$$\frac{\partial^2 u(\mathbf{x},t,\theta)}{\partial t^2} + \eta \frac{\partial u(\mathbf{x},t,\theta)}{\partial t} - \nabla \cdot (c_s(\mathbf{x},\theta) \nabla u(\mathbf{x},t,\theta)) = f(\mathbf{x},t) \qquad \text{in } D \times (0,T) \times \Omega, \tag{44}$$

$$u(\mathbf{x},t,\theta) = \Phi(\mathbf{x},t,\theta) \quad \text{on } \Gamma_D \times (0,T) \times \Omega, \tag{45}$$

$$\nabla u(\mathbf{x},t,\theta) \cdot \hat{\mathbf{n}} = \Psi(\mathbf{x},t,\theta) \quad \text{on } \Gamma_N \times (0,T) \times \Omega, \tag{46}$$

$$u(\mathbf{x},t,\theta) = u_0(\mathbf{x},\theta) \qquad \text{in } D \times \Omega, \tag{47}$$

$$\frac{\partial u(\mathbf{x},t,\theta)}{\partial t} = v_0(\mathbf{x},\theta) \qquad \text{in } D \times \Omega, \tag{48}$$

Algorithm 1. Pseudo code for acoustic wave propagation in a deterministic medium

Initial condition: $n = 0$, u_0, v_0, a_0
for each process, $s = 0, 1, 2, \cdots, n_s$ **do**
    Assemble K^s, stiffness matrix
    Assemble M^s, mass matrix
    Assemble C^s, damping matrix
    Assemble f^s, force vector
    Decompose u_0, v_0, a_0 as $u^s_{\Gamma,0}$, $v^s_{\Gamma,0}$, $a^s_{\Gamma,0}$, $u^s_{I,0}$, $v^s_{I,0}$, $a^s_{I,0}$
    Compute transient stiffness matrix: $\tilde{K}^s = M^s \frac{1}{\zeta \Delta t^2} + C^s \frac{\gamma}{\zeta \Delta t} + K^s$, see (12)
end for
while $n < T_N$ **do**
    $t = (n+1)\Delta t$
    for $s = 1 : n_s$ **do**
        Compute mass component of force: $f^s_{mI,n}$, $f^s_{m\Gamma,n}$, see (20)
        Compute damping component of force: $f^s_{cI,n}$, $f^s_{c\Gamma,n}$, see (21)
        $\tilde{f}^s_{\Gamma,n+1} = f^s_{\Gamma,n+1} + f^s_{m\Gamma,n} + f^s_{c\Gamma,n}$
        $\tilde{f}^s_{I,n+1} = f^s_{I,n+1} + f^s_{mI,n} + f^s_{cI,n}$
        $g_\Gamma = g_\Gamma + R_s^T \left(\tilde{f}^s_\Gamma - \tilde{K}^s_{\Gamma I} \left[\tilde{K}^s_{II} \right]^{-1} \tilde{f}^s_I \right) R_s$
    end for
    Apply conjugate gradient iterative solver with two-level Neumann-Neumann preconditioner: $\tilde{K} u_{n+1} = \tilde{F}_{n+1}$, see (42)
    Compute $u_{\Gamma,n+1}$, $u^s_{I,n+1}$
    Update initial solution with new solution as $u^s_{\Gamma,0} = R_s u_{\Gamma,n+1}$, $u^s_{I,0} = u^s_{I,n+1}$
    $n = n + 1$
end while
Output: $U(ndof, ndof, T_N)$

where u is the random acoustic pressure field and $c(\mathbf{x}, \theta)$ is the square of the wave speed represented as a log-normal random field. Ω represents the set of all possible outcomes associated with the probability space (Ω, σ, P) and θ is the random aspect of the problem [14, 24]. $\Phi(\mathbf{x}, t, \theta)$, $\Psi(\mathbf{x}, t, \theta)$ are the random Dirichlet and Neumann boundary conditions and $u_0(\mathbf{x}, \theta)$ and $v_0(\mathbf{x}, \theta)$ are the random initial pressure and velocity of the particle. $\Gamma_D \cup \Gamma_N$ is the complete boundary formed by the union of Dirichlet and Neumann parts and $\hat{\mathbf{n}}$ represents the outward unit normal to the boundary at $\mathbf{x}$. For the current model, initial and boundary conditions are considered deterministic (same as in Section 2) even though they can also be considered random.

Spatial discretization of the weak form leads to a system of stochastic ODEs as

$$\mathbf{M}(\theta)\, \ddot{\mathbf{u}}(\theta) + \mathbf{C}(\theta)\dot{\mathbf{u}}(\theta) + \mathbf{K}(\theta)\mathbf{u}(\theta) \approx \mathbf{f}, \tag{49}$$

where $\mathbf{M}$, $\mathbf{C}$, $\mathbf{K}$ are the random mass, damping and stiffness matrices (although in this special case, $\mathbf{M}$ is considered deterministic). The square of wave speed, c_s in the above problem, is expanded as a log-normal random field (enforcing positivity) computed as the exponential of a Gaussian field with a known covariance function. The Karhunen-Loève expansion (KLE) provides a means to expand any input random process having a known covariance function

in terms of uncorrelated orthogonal random variables. An exponential covariance kernel is assumed for the spectral expansion of the underlying Gaussian process in two dimensions as [13]

$$\mathcal{H}(x_1, x_2; y_1, y_2) = \sigma^2 \exp\left[-\left(\frac{|x_1 - x_2|}{b_x} + \frac{|y_1 - y_2|}{b_y}\right)\right], \tag{50}$$

where σ^2 is the variance of the process, and b_x and b_y are the correlation lengths in the x- and y-directions. The decomposition of this covariance kernel in terms of eigenvalues and eigenfunctions can be written as [13]

$$g(\mathbf{x}, \theta) \approx g_0(\mathbf{x}) + \sum_{n=1}^{L} \xi_n(\theta) \sqrt{\lambda_n}\, \phi_n(\mathbf{x}), \tag{51}$$

where $g_0(\mathbf{x})$ is the mean field, λ_n and ϕ_n are the eigenvalues and eigenfunctions of the covariance kernel, and $\xi_1, \xi_2, \ldots, \xi_L$ are the uncorrelated Gaussian random variables having zero mean and unit variance. The log-normal field for the square of wave speed $c_s(\mathbf{x}, \theta)$ is computed by taking the exponential of the expansion in (51) and then approximated using PCE as [12]

$$c_s(\mathbf{x}, \theta) \approx \sum_{i=0}^{M} c_i(\mathbf{x}) \frac{\langle \Psi_i(\boldsymbol{\eta}) \rangle}{\langle \Psi_i^2(\boldsymbol{\eta}) \rangle}\, \Psi_i(\boldsymbol{\xi}), \tag{52}$$

where

$$c_0(\mathbf{x}) \approx \exp\left[g_0(\mathbf{x}) + \frac{1}{2} \sum_{i=1}^{L} g_i^2(\mathbf{x})\right] \tag{53}$$

is the mean of the log-normal expansion and each $\langle \Psi_i(\boldsymbol{\eta}) \rangle$ is the expectation of functionals centered around $g_i(\mathbf{x})$, see [7, 17, 32]. Each $\langle \Psi_i(\boldsymbol{\xi}) \rangle$ is the orthogonal polynomial chaos basis satisfying [13]

$$\langle \Psi_i(\boldsymbol{\xi}), \Psi_j(\boldsymbol{\xi}) \rangle \;=\; \langle \Psi^2{}_i(\boldsymbol{\xi}) \rangle\, \delta_{ij}\,. \tag{54}$$

The random field for the acoustic pressure $u(\mathbf{x}, t, \theta)$ can also be expanded in terms of PCE as

$$u(\mathbf{x}, t, \theta) \approx \sum_{j=0}^{N} u_j(\mathbf{x}, t)\, \Psi_j(\boldsymbol{\xi}), \tag{55}$$

where u_j are the deterministic polynomial chaos coefficients to be computed using the stochastic Galerkin method. Applying the PCE for input and output terms in (49) provides

$$\sum_{l=0}^{L} \mathbf{M}_l \Psi_l(\boldsymbol{\xi}) \sum_{j=0}^{N} \ddot{\mathbf{u}}_j \Psi_j(\boldsymbol{\xi}) + \sum_{q=0}^{Q} \mathbf{C}_q \Psi_q(\boldsymbol{\xi}) \sum_{j=0}^{N} \dot{\mathbf{u}}_j \Psi_j(\boldsymbol{\xi}) + \sum_{i=0}^{M} \mathbf{K}_i \Psi_i(\boldsymbol{\xi}) \sum_{j=0}^{N} \mathbf{u}_j \Psi_j(\boldsymbol{\xi}) \approx \mathbf{f}, \tag{56}$$

where $\mathbf{M}_i, \mathbf{C}_i, \mathbf{K}_i$ are the deterministic coefficient matrices. The Galerkin projection of the above stochastic system of equations onto a PCE basis leads to

$$\sum_{j=0}^{N} \sum_{l=0}^{L} \langle \Psi_l \Psi_j \Psi_k \rangle \mathbf{M}_l \ddot{\mathbf{u}}_j + \sum_{j=0}^{N} \sum_{q=0}^{Q} \langle \Psi_q \Psi_j \Psi_k \rangle \mathbf{C}_q \dot{\mathbf{u}}_j + \sum_{j=0}^{N} \sum_{i=0}^{M} \langle \Psi_i \Psi_j \Psi_k \rangle \mathbf{K}_i \mathbf{u}_j = \langle \mathbf{f} \Psi_k \rangle, \tag{57}$$

$k = 0, 1, 2, \ldots, N$. It results in a large deterministic system of ODEs as

$$\boldsymbol{\mathcal{M}} \ddot{\mathcal{U}} + \boldsymbol{\mathcal{C}} \dot{\mathcal{U}} + \boldsymbol{\mathcal{K}} \mathcal{U} = \mathcal{F}, \tag{58}$$

where $\boldsymbol{\mathcal{M}}$, $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{K}}$ are the block matrices with each block defined as

$$[\boldsymbol{\mathcal{M}}]_{jk} = \sum_{l=0}^{L} G_{ljk}\,\mathbf{M}_l, \qquad [\boldsymbol{\mathcal{C}}]_{jk} = \sum_{q=0}^{Q} G_{qjk}\,\mathbf{C}_q, \qquad [\boldsymbol{\mathcal{K}}]_{jk} = \sum_{i=0}^{M} G_{ijk}\,\mathbf{K}_i, \tag{59}$$

where

$$G_{ijk} = \langle \Psi_i \Psi_j \Psi_k \rangle \tag{60}$$

and

$$[\mathcal{F}]_k = \langle \mathbf{f} \Psi_{\mathrm{k}} \rangle. \tag{61}$$

The solution vector and its higher order derivatives are represented as $\mathcal{U} = [u_0, u_1, u_2, \ldots, u_N]$. Note that each u_j contains the solution for all degrees of freedom in the grid corresponding to the PCE coefficient. For the current model, a deterministic mass matrix is used which then reduces $(59)_1$ to a block diagonal matrix as

$$[\boldsymbol{\mathcal{M}}]_{jk} = \langle \Psi_j \Psi_k \rangle\, \delta_{jk}\, \mathbf{M}, \tag{62}$$

where δ_{jk} is the Kronecker delta function. The damping matrix coefficients for Rayleigh damping using the mean mass and random stiffness matrix can be computed as (see [16, 27] for details)

$$\mathbf{C}_i = \alpha_0 \mathbf{M} + \alpha_1 \mathbf{K}_i, \tag{63}$$

where α_0 and α_1 are the coefficients computed as in [16, 27]. Time discretization of this system can be carried out using the Newmark-beta scheme. From the subdomain level mass, damping and stiffness matrices, we can build the transient stiffness matrix as

$$\tilde{\boldsymbol{\mathcal{K}}} = \boldsymbol{\mathcal{M}} \frac{1}{\zeta \Delta \mathrm{t}^2} + \boldsymbol{\mathcal{C}} \frac{\gamma}{\zeta \Delta \mathrm{t}} + \boldsymbol{\mathcal{K}}. \tag{64}$$

Now, the DD-based assembly of these stochastic matrices and the construction of a two-level Neumann-Neumann preconditioner is presented.

5. Probabilistic two-level Neumann-Neumann preconditioner

The formulation of the probabilistic two-level Neumann-Neumann solver is explained in this section corresponding to the acoustic wave propagation in random media explained in Section 4. The spatial discretization of the stochastic problem (44) can be assembled into random mass, damping and stiffness matrices as

$$\mathbf{M}(\theta)\, \ddot{\mathbf{u}}(\theta) + \mathbf{C}(\theta) \dot{\mathbf{u}}(\theta) + \mathbf{K}(\theta) \mathbf{u}(\theta) \approx \mathbf{f}, \tag{65}$$

where each of these random matrices admits a PCE in the form

$$\mathbf{K}(\theta) = \sum_{i=0}^{M} \mathbf{K}_i \Psi_i(\boldsymbol{\xi}). \tag{66}$$

The spatial domain of the problem can be decomposed into several non-overlapping domains with interior and interface nodes, as shown in Fig. 1. This decomposition allows the stiffness

matrix to be assembled[1] at the subdomain level as

$$\mathbf{K}_i = \begin{bmatrix} \mathbf{K}^{1}_{II,i} & \cdots & 0 & \mathbf{K}^{1}_{I\Gamma,i}\mathbf{R}_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \mathbf{K}^{n_s}_{II,i} & \mathbf{K}^{n_s}_{I\Gamma,i}\mathbf{R}_{n_s} \\ \mathbf{R}_1^T\mathbf{K}^{1}_{\Gamma I,i} & \cdots & \mathbf{R}^T_{n_s}\mathbf{K}^{n_s}_{\Gamma I,i} & \displaystyle\sum_{s=1}^{n_s}\mathbf{R}_s^T\mathbf{K}^{s}_{\Gamma\Gamma,i}\mathbf{R}_s \end{bmatrix}. \tag{67}$$

After the Galerkin projection of the system (57), each of the stochastic subdomain level matrices can be assembled from (67) as

$$[\boldsymbol{\mathcal{K}}_{\alpha\beta}]^s_{jk} = \sum_{i=0}^{M} G_{ijk}\mathbf{K}^s_{\alpha\beta,i}, \tag{68}$$

where α and β correspond to the interior and interface nodes, respectively, and s represent the subdomain. Thus, the transient stiffness matrix can be assembled using stochastic subdomain blocks of mass, damping and stiffness as

$$[\tilde{\boldsymbol{\mathcal{K}}}_{\alpha\beta}]^s = [\boldsymbol{\mathcal{M}}_{\alpha\beta}]^s\frac{1}{\zeta\Delta \mathrm{t}^2} + [\boldsymbol{\mathcal{C}}_{\alpha\beta}]^s\frac{\gamma}{\zeta\Delta \mathrm{t}} + [\boldsymbol{\mathcal{K}}_{\alpha\beta}]^s. \tag{69}$$

Notice the difference between the construction of stochastic matrices in (68) and $(59)_3$, as well as the difference between (69) and (64).

Performing global assembly of these subdomain level transient stiffness matrices, one obtains

$$\begin{bmatrix} \tilde{\boldsymbol{\mathcal{K}}}^{1}_{II} & \cdots & 0 & \tilde{\boldsymbol{\mathcal{K}}}^{1}_{I\Gamma}\boldsymbol{\mathcal{R}}_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \tilde{\boldsymbol{\mathcal{K}}}^{n_s}_{II} & \tilde{\boldsymbol{\mathcal{K}}}^{n_s}_{I\Gamma}\boldsymbol{\mathcal{R}}_{n_s} \\ \boldsymbol{\mathcal{R}}_1^T\tilde{\boldsymbol{\mathcal{K}}}^{1}_{\Gamma I} & \cdots & \boldsymbol{\mathcal{R}}^T_{n_s}\tilde{\boldsymbol{\mathcal{K}}}^{n_s}_{\Gamma I} & \displaystyle\sum_{s=1}^{n_s}\boldsymbol{\mathcal{R}}_s^T\tilde{\boldsymbol{\mathcal{K}}}^{s}_{\Gamma\Gamma}\boldsymbol{\mathcal{R}}_s \end{bmatrix} \begin{Bmatrix} \mathcal{U}^{1}_I \\ \vdots \\ \mathcal{U}^{n_s}_I \\ \mathcal{U}_\Gamma \end{Bmatrix} = \begin{Bmatrix} \tilde{\mathcal{F}}^{1}_I \\ \vdots \\ \tilde{\mathcal{F}}^{n_s}_I \\ \displaystyle\sum_{s=1}^{n_s}\boldsymbol{\mathcal{R}}_s^T\tilde{\mathcal{F}}^{s}_\Gamma \end{Bmatrix}, \tag{70}$$

where

$$\mathcal{U}^s_I = [\mathbf{u}^s_{I,0}, \ldots, \mathbf{u}^s_{I,j}]^T, \qquad \mathcal{U}_\Gamma = [\mathbf{u}_{\Gamma,0}, \ldots, \mathbf{u}_{\Gamma,j}]^T, \tag{71}$$

$$\boldsymbol{\mathcal{R}}_s = \mathrm{blockdiag}\left(\mathbf{R}_{s,0}, \ldots, \mathbf{R}_{s,j}\right). \tag{72}$$

Similar to the deterministic case, applying a Gaussian block elimination allows the construction of the Schur complement as

$$\boldsymbol{\mathcal{S}}\mathcal{U}_\Gamma = \mathcal{G}_\Gamma, \tag{73}$$

where

$$\boldsymbol{\mathcal{S}} = \sum_{s=1}^{n_s}\boldsymbol{\mathcal{R}}_s^T\left(\tilde{\boldsymbol{\mathcal{K}}}^{s}_{\Gamma\Gamma} - \tilde{\boldsymbol{\mathcal{K}}}^{s}_{\Gamma I}[\tilde{\boldsymbol{\mathcal{K}}}^{s}_{II}]^{-1}\tilde{\boldsymbol{\mathcal{K}}}^{s}_{I\Gamma}\right)\boldsymbol{\mathcal{R}}_s = \sum_{s=1}^{n_s}\boldsymbol{\mathcal{R}}_s^T\boldsymbol{\mathcal{S}}_s\boldsymbol{\mathcal{R}}_s, \tag{74}$$

$$\mathcal{G}_\Gamma = \sum_{s=1}^{n_s}\boldsymbol{\mathcal{R}}_s^T\left(\tilde{\mathcal{F}}^{s}_\Gamma - \tilde{\boldsymbol{\mathcal{K}}}^{s}_{\Gamma I}[\tilde{\boldsymbol{\mathcal{K}}}^{s}_{II}]^{-1}\tilde{\mathcal{F}}^{s}_I\right) = \sum_{s=1}^{n_s}\boldsymbol{\mathcal{R}}_s^T\mathcal{G}_{\Gamma s}. \tag{75}$$

[1]Note that this decomposition is applied to mass and damping matrices as well but not shown explicitly for brevity.

To construct the coarse grid, the solution vector is decomposed again to create the global coarse problem on corner nodes as (similar to the deterministic setting in (27))

$$\begin{Bmatrix} \mathcal{U}_r^s \\ \mathcal{U}_c^s \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{\mathcal{R}}_s^r \\ \boldsymbol{\mathcal{R}}_s^c \end{Bmatrix} \mathcal{U}_\Gamma^s, \tag{76}$$

$$\boldsymbol{\mathcal{F}}_{cc}\, \mathcal{U}_c = \mathcal{Q}_c, \tag{77}$$

where the operator $\boldsymbol{\mathcal{F}}_{cc}$ and the corresponding force vector $\mathcal{Q}_c$ for corner nodes is defined as

$$\boldsymbol{\mathcal{F}}_{cc} = \sum_{s=1}^{n_s} \boldsymbol{\mathcal{B}}_c^{sT} \left(\boldsymbol{\mathcal{S}}_{cc}^s - \boldsymbol{\mathcal{S}}_{cr}^s [\boldsymbol{\mathcal{S}}_{rr}^s]^{-1} \boldsymbol{\mathcal{S}}_{rc}^s \right) \boldsymbol{\mathcal{B}}_c^s, \tag{78}$$

$$\mathcal{Q}_c = \sum_{s=1}^{n_s} \boldsymbol{\mathcal{B}}_c^{sT} \left(\boldsymbol{\mathcal{F}}_c^s - \boldsymbol{\mathcal{S}}_{cr}^s [\boldsymbol{\mathcal{S}}_{rr}^s]^{-1} \boldsymbol{\mathcal{F}}_r^s \right), \tag{79}$$

where

$$\boldsymbol{\mathcal{B}}_c^s = \text{blockdiag}(\mathbf{B}_{c,0}^s, \ldots, \mathbf{B}_{c,j}^s) \tag{80}$$

and $\boldsymbol{\mathcal{S}}_{\alpha\beta}^s = \tilde{\boldsymbol{\mathcal{K}}}_{\alpha\beta}^s - \tilde{\boldsymbol{\mathcal{K}}}_{\alpha I}^s [\tilde{\boldsymbol{\mathcal{K}}}_{II}^s]^{-1} \tilde{\boldsymbol{\mathcal{K}}}_{I\beta}^s$ is the Schur complement for corner and remaining nodes. Finally, the stochastic two-level Neumann-Neumann preconditioner can be written as

$$\boldsymbol{\mathcal{M}}_{\text{NNC}}^{-1} = \sum_{s=1}^{n_s} \boldsymbol{\mathcal{R}}_s^T \boldsymbol{\mathcal{D}}_s \left(\boldsymbol{\mathcal{R}}_s^{rT} [\boldsymbol{\mathcal{S}}_{rr}^s]^{-1} \boldsymbol{\mathcal{R}}_s^r \right) \boldsymbol{\mathcal{D}}_s \boldsymbol{\mathcal{R}}_s + \boldsymbol{\mathcal{R}}_0^T [\boldsymbol{\mathcal{F}}_{cc}]^{-1} \boldsymbol{\mathcal{R}}_0, \tag{81}$$

where $\boldsymbol{\mathcal{R}}_0$ is defined as

$$\boldsymbol{\mathcal{R}}_0 = \sum_{s=1}^{n_s} \boldsymbol{\mathcal{B}}_c^{sT} \left(\boldsymbol{\mathcal{R}}_s^c - \boldsymbol{\mathcal{S}}_{cr}^s [\boldsymbol{\mathcal{S}}_{rr}^s]^{-1} \boldsymbol{\mathcal{R}}_s^r \right) \boldsymbol{\mathcal{D}}_s \boldsymbol{\mathcal{R}}_s. \tag{82}$$

Note that $\boldsymbol{\mathcal{D}}_s$ is the subdomain level stochastic counterpart of the scaling matrix which can be computed using the stochastic restriction matrix in (72) similar to the deterministic case in (31). Even though the construction of the two-level Neumann-Neumann preconditioner in the stochastic setting is similar to the deterministic setting, each component of the stochastic preconditioner now encompasses the information pertaining to all PCE coefficients in them. More details on the implementations and algorithms for the probabilistic two-level Neumann-Neumann preconditioner can be found in [7, 8, 32].

6. Numerical results

This section discusses the numerical experiments conducted for acoustic wave propagation on a two-dimensional random medium. The spatial domain is a unit square with Dirichlet boundary conditions on all four sides (fixed, $u = 0$). The CFL number for all experiments uses a value of 0.65 according to (8). All the scalability studies are conducted for a duration of 0.5 s and the time step size is adjusted accordingly with changing mesh sizes. The number of iterations for all experiments reported is the average number of iterations over all time steps and the computational time reported is the complete duration of the simulation. Note that the acoustic pressure has units of Pa (N/m^2 or dB), but it is not explicitly mentioned in numerical results for simplicity. The verification of the stochastic Galerkin method is illustrated by comparing the results to MCS. Further, the numerical and parallel scalability of the probabilistic

two-level Neumann-Neumann DD-based solver are presented. The scalability with respect to the stochastic parameters (increasing number of random variables and order of expansion) are also illustrated. For all numerical experiments concerning the stochastic case in this section, an exponential covariance kernel as in (50) is used for representing the underlying Gaussian process with zero mean and standard deviation of 0.1. The PCE for the input has three random variables with second order while the output PCE uses third order unless explicitly specified. The PCE basis functions are normalized with respect to their standard deviation in all numerical experiments.

6.1. Comparison with one-level preconditioners

This section compares the performance of different preconditioners (both one-level and two-level) to illustrate the need for a two-level method for acoustic wave propagation problems in the deterministic medium. Two one-level preconditioners, the lumped preconditioner and the Neumann-Neumann preconditioner (see [32] for details on these preconditioners), are chosen to test against the performance of the two-level Neumann-Neumann preconditioner. One-level preconditioners are shown to lose scalability with an increasing number of subdomains because of the lack of information exchange across several subdomains [22, 30, 33]. This is rectified in two-level methods (such as the two-level Neumann-Neumann) by using a coarse grid which couples all the subdomains. A square domain with 1 563 vertices and a time step of 0.01 is chosen for the study.

Fig. 2 reports the average iteration count over all the time steps for the different preconditioners with an increasing number of subdomains. As the number of subdomains increases, the iteration counts of the lumped preconditioner increase rapidly. Even though the one-level Neumann-Neumann preconditioner has lower iteration counts than the lumped one, it also shows an increase in the number of iterations. The two-level Neumann-Neumann preconditioner in contrast has the lowest iteration counts with a modest increase for an increasing number

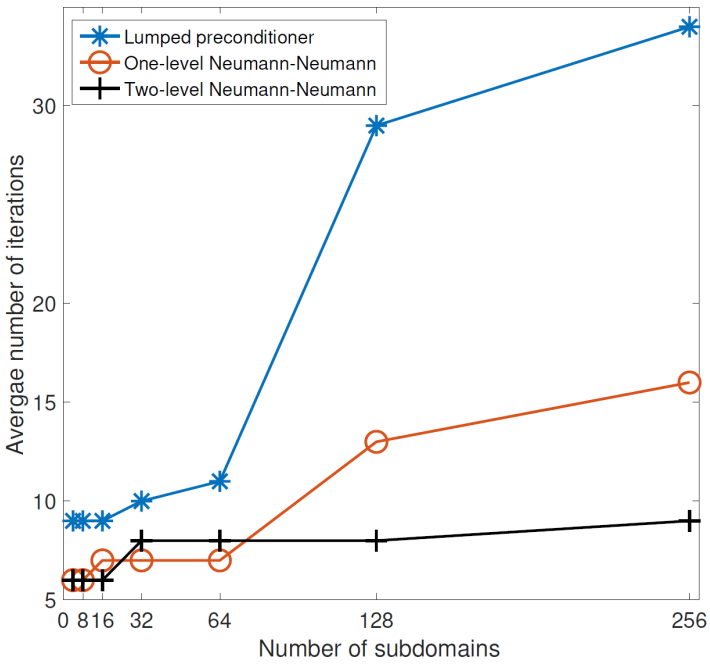


Fig. 2. Comparison of DD-based preconditioners applied to a deterministic acoustic wave propagation problem on a square domain featuring 1 563 vertices with a time step of $\Delta t = 0.01$

of subdomains. An interesting feature of the current transient formulation is that the addition of mass and damping terms to the stiffness operator removes the singularity issues associated with floating subdomains, i.e., subdomains whose local stiffness matrices are singular due to insufficient Dirichlet constraints [32].

6.2. Verification for acoustic wave in a stochastic medium

The stochastic Galerkin method is compared against MCS for acoustic wave propagation with a random wave speed in a two-dimensional domain, as represented by (44). The initial condition is a Gaussian pulse at $x_0 = 0.7$, $y_0 = 0.7$ in the following form:

$$u_0(x_0 = 0.7, y_0 = 0.7) = \beta \exp\left[-\frac{(x - x_0)^2 + (y - y_0)^2}{\alpha}\right], \tag{83}$$

where $\beta = 1$ and $\alpha = 0.01$. A time step size of 6.5×10^{-3} is used with a finite element mesh of 13 472 vertices. The Rayleigh damping coefficients are computed as $\alpha_0 = 0.5445$ and $\alpha_1 = 0.0174$. The mean and standard deviation of the pressure for the stochastic Galerkin method and MCS at three different points in the domain are shown in Fig. 3. The results show good agreement between both approaches for all three points, verifying the stochastic Galerkin method. The standard deviation of the pressure in time shows large variations in contrast to the mean pressure which is captured well by the stochastic Galerkin method. The contour plots of the pressure field at four different time steps are shown in Fig. 4. The complex spatial features in standard deviation with the reflection of waves from boundaries can be observed in the right column. This underlines the complexity and importance of uncertainty quantification for wave propagation problems.

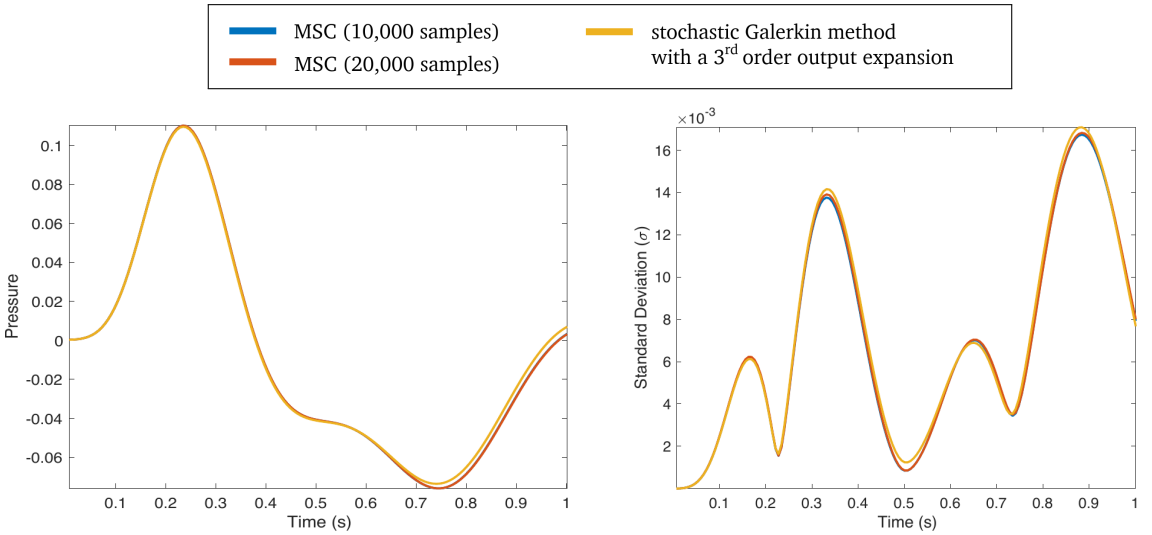
6.3. Scalability with mesh size

The strong and weak scalability of the two-level Neumann-Neumann solver for wave propagation on a random medium are discussed in this section. The strong numerical and parallel scalability of the solver for a 7-random-variable case (120 PCE terms) with a fixed mesh size of 13 472 vertices (leading to a total problem size of 1.6 million) is shown in Fig. 5. The average number of iterations is constant with an increasing number of subdomains (Fig. 5a), by contrast, the time to solution decreases (Fig. 5b). A significant reduction in memory consumption per core with an increasing number of subdomains for a 7-random-variable PCE is shown in Fig. 5c. The memory requirement per core reaches 15 GB for the 32-core configuration. The need for using DD-based solvers for stochastic problems is evident.

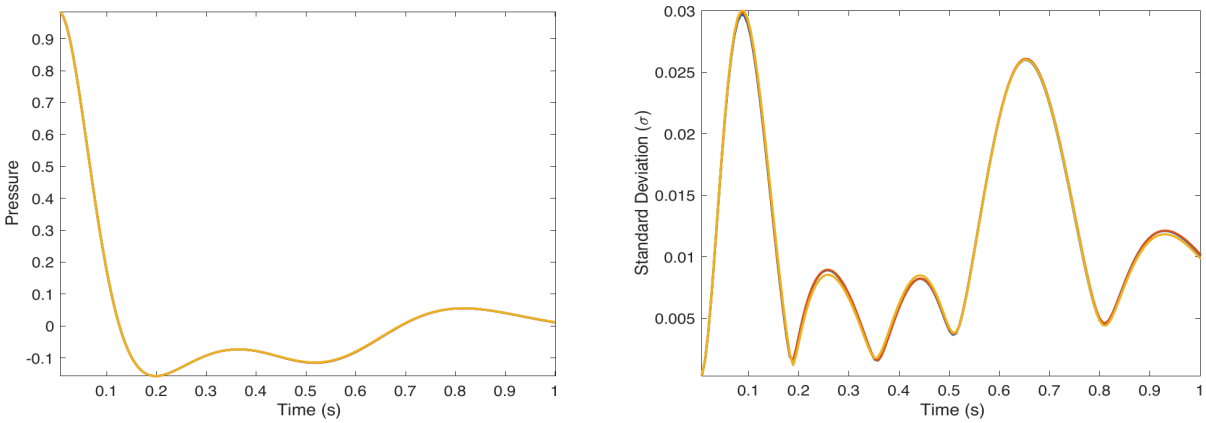
The weak numerical and parallel scalability of the solver is illustrated in Fig. 6. The global problem size and the number of subdomains are increased simultaneously with the problem size per core fixed ($\approx$ 3 400 degrees of freedom). A 3-random-variable case is used here with a number of processes increasing from 80 to 720 (leading to a maximum problem size of 2.4 million). The average iteration counts remain constant with an increasing number of processes showing excellent weak scaling. However, the computational time increases significantly despite the nearly constant iteration counts. This is likely due to the combined parallel overhead for all the time steps.

6.4. Scalability with stochastic parameters

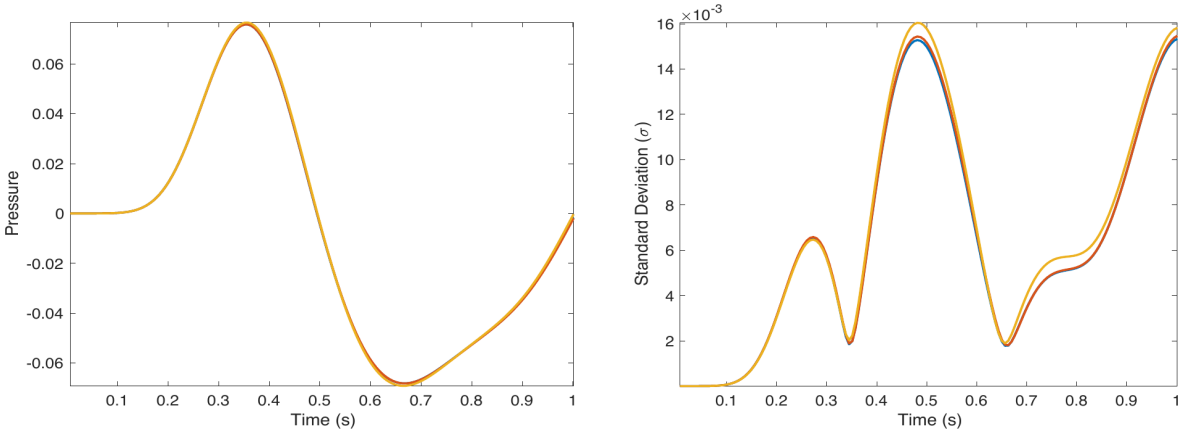
This section discusses the scalability of the probabilistic two-level Neumann-Neumann solver with an increasing number of random variables and order of expansions. In this case, the mesh



(a) $x = 0.5, y = 0.5$

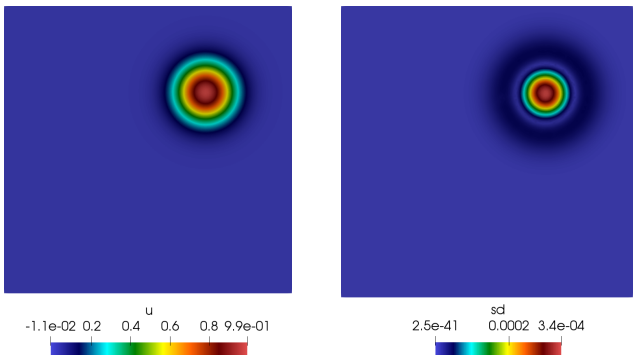


(b) $x = 0.7, y = 0.7$

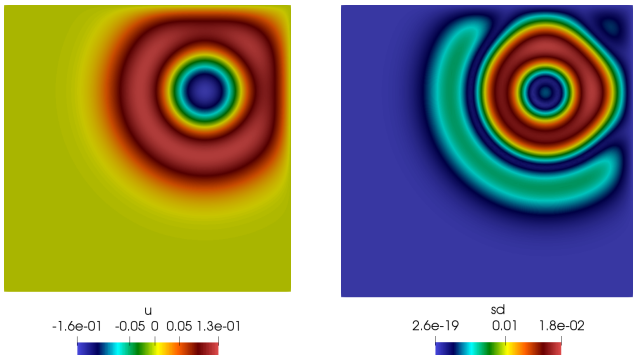


(c) $x = 0.2, y = 0.7$

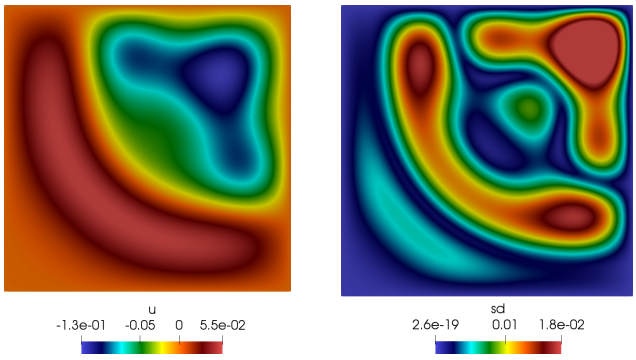
Fig. 3. Verification of the stochastic Galerkin method against Monte Carlo simulations (MCS) for acoustic wave propagation. The left and right columns denote the mean acoustic pressure field and the standard deviation of the pressure field, respectively. The responses are evaluated over time (t in seconds) at three pre-specified spatial locations (a)–(c) with given coordinates



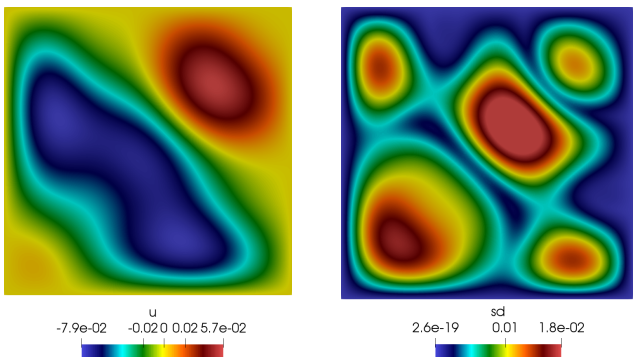
(a) $t = 0.0065\,\mathrm{s}$



(b) $t = 0.25\,\mathrm{s}$

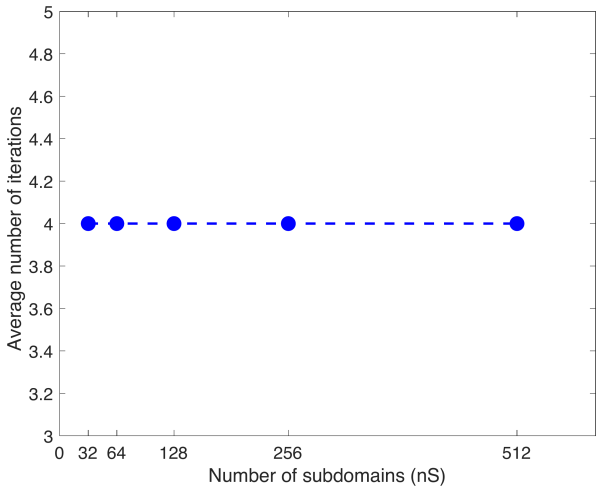


(c) $t = 0.5\,\mathrm{s}$

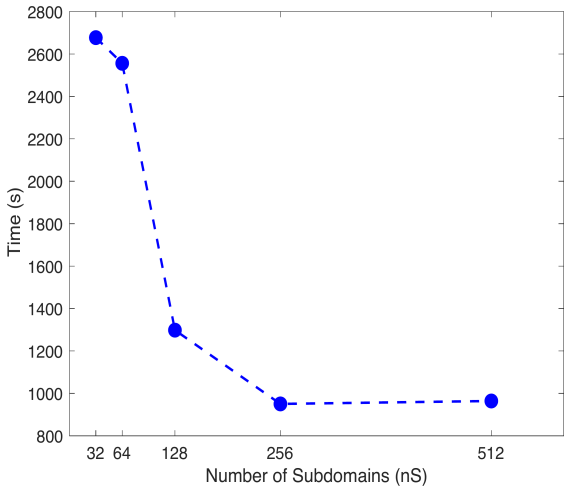


(d) $t = 0.8\,\mathrm{s}$

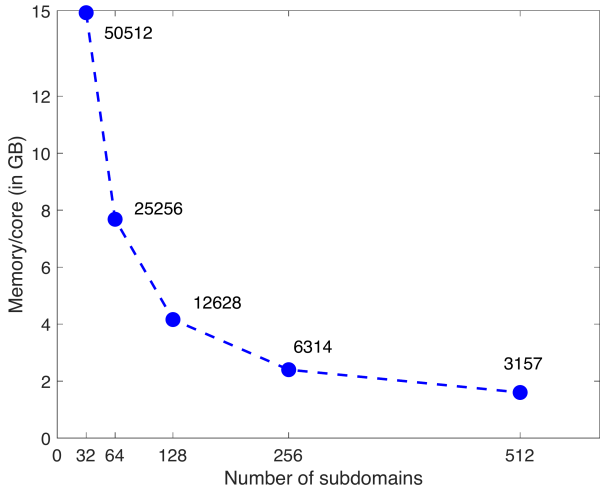
Fig. 4. Spatio-temporal evolution snapshots of the acoustic pressure field computed using the stochastic Galerkin method. The left column denotes the mean acoustic pressure field distribution, denoted as u, and the right column denotes the standard deviation field distribution, denoted as *sd*. Contours map the spatial variations in pressure magnitude across the two-dimensional domain at four sequential time steps



(a)) strong numerical scalability



(b) strong parallel scalability

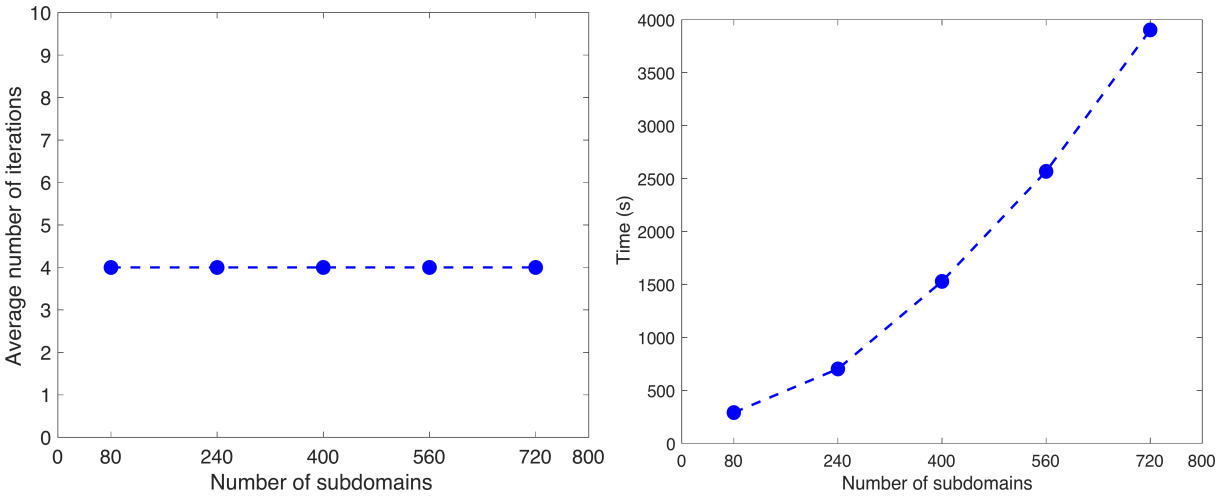


(c) memory/core with an increasing number of subdomains

Fig. 5. Strong scalability and memory consumption for a 7-random-variable case (120 PC terms) on a fixed mesh of 13 472 vertices (1.61 million total degrees of freedom) plotted against the number of subdomains (nS): (a) average conjugate gradient iterations over all time steps, (b) total simulation runtime in seconds, and (c) memory consumption per core in gigabytes. Note that graph in (c) also shows the number of spatial vertices corresponding to each subdomain size

size of 37 267 vertices and the time step of 3.9×10^{-3} are used. The time to solution is reported for a total of 128 time steps. The number of PCE terms is increased from 20 (3 random variables) to 220 (9 random variables) case with a fixed number of subdomains equal to 720. The total problem size in this case is 8.19 million. The average iteration over all time steps for the solver are reported in Fig. 7, which shows a constant value of four iterations.

The scalability with increasing order of expansions is shown in Fig. 8. For a fixed number of random variables equal to 3, the order of output PCE is increased from 2 to 9 leading to 220 terms and a total problem size of 8.19 million. The number of processes is increased to keep the problem size per core fixed at 6 200 degrees of freedom. Fig. 8 shows excellent weak numerical scalability with an increasing order of expansion. However, the time to solution increases even with a fixed problem size per core which is due to the strong stochastic coupling introduced



(a) average number of conjugate gradient iterations over all time steps

(b) total simulation runtime in seconds

Fig. 6. Weak scalability for a 3-random-variable case (20 PCE terms) with a fixed mesh size of approx. 3 400 degrees of freedom per core, scaled from 80 to 720 processes (up to 2.4 million global degrees of freedom)

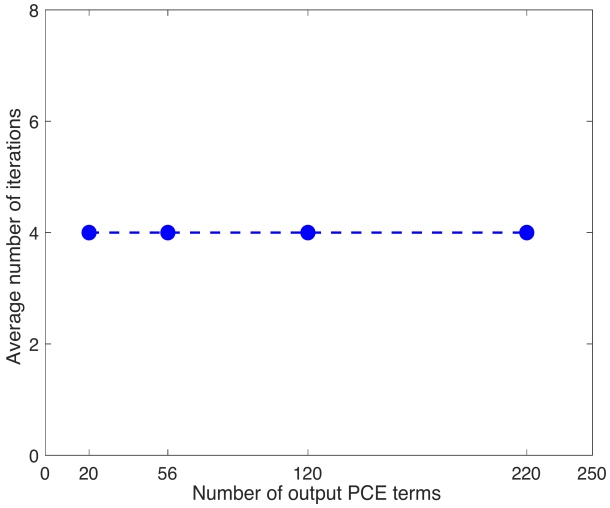
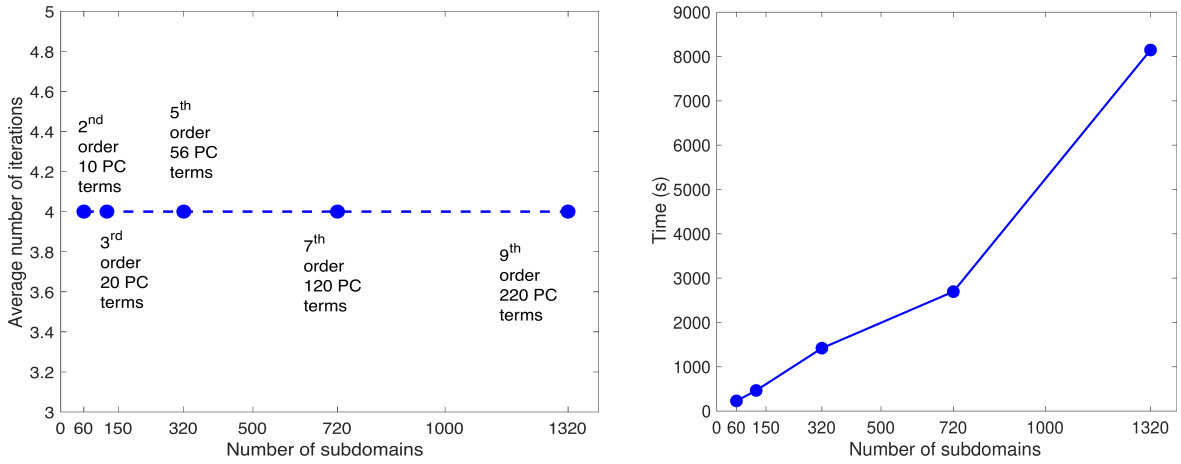


Fig. 7. Numerical scalability of the probabilistic two-level Neumann-Neumann solver evaluated against an increasing number of output PCE terms (from 20 terms up to 220 terms) for a fixed number of 720 subdomains

by higher-order PCE. Moreover, the use of collective communication operations in parallel algorithms contributes significantly to the increase in time. This can be improved by point-to-point communications which only transfer data between individual processes as necessary. Even with an increase in the time to solution, the application of DD-based solvers remains necessary for solving memory-intensive stochastic problems.

7. Conclusion

This article covered the uncertainty quantification for acoustic wave propagation on a two-dimensional random medium using the stochastic Galerkin method. The fine temporal, spatial



(a) average iterations per time step over all time steps

(b) total simulation runtime in seconds

Fig. 8. Numerical and parallel scalability evaluated against an increasing order of expansion (from 2nd order up to 9th order PCE, yielding 10 to 220 terms) for a fixed 3-random-variable input case. The problem size per computational core is kept fixed at 6 200 degrees of freedom as the number of subdomains scales from 60 to 1 320

or stochastic resolutions required to capture high frequency oscillations increase the computational cost and memory requirements for these systems. Non-overlapping domain decomposition-based methods are utilised which offer a way to decompose the spatial domain and distribute the memory and computations to several cores/processes. A conjugate gradient iterative solver is used for the coupled system of equations involving a symmetric positive definite stochastic coefficient matrix. To expedite the convergence of the solver, a probabilistic version of the two-level Neumann-Neumann preconditioner is developed. The solver has excellent numerical scalability with respect to mesh size and stochastic parameters. However, improvements to parallel scalability need further investigations into the architecture of the solver to tackle strong stochastic coupling. The combination of the stochastic Galerkin method and the domain decomposition-based solvers is effective in solving uncertainty quantification problems for high-resolution time-dependent systems.

Acknowledgements

This work was supported by the Ontario Trillium Scholarship for International Doctoral Students. The author thanks the supervisors and collaborators for their mentorship and advice during the course of this work. This research was enabled in part by support provided by *Calcul Québec*, *SHARCNET*, *SciNet*, and the *Digital Research Alliance of Canada* (alliancecan.ca).

Data availability

The source code supporting the findings of this study is openly available in the GitHub repository: https://github.com/sudhipv/stochastic_wave_propagation. This repository contains the parallel solver framework for wave propagation in random media. Specific scripts and documentation related to the stochastic Galerkin method and the domain decomposition-based solvers for the acoustic problem are provided within the repository structure.

References

[1] Bathe, K.-J., Finite element procedures, Prentice Hall, 2006.

[2] Chan, T. F., Mathew, T. P., Domain decomposition algorithms, Acta Numerica 3 (1994) 61–143. https://doi.org/10.1017/S0962492900002427

[3] Cros, J.-M., A preconditioner for the Schur complement domain decomposition method, Proceedings of the 14th International Conference on Domain Decomposition Methods, Cocoyoc, National Autonomous University of Mexico (UNAM), 2002, pp. 373–380.

[4] Dai, D., Epshteyn, Y., Narayan, A., Hyperbolicity-preserving and well-balanced stochastic Galerkin method for shallow water equations, SIAM Journal on Scientific Computing 43 (2) (2021) A929–A952. https://doi.org/10.1137/20M1360736

[5] Debusschere, B. J., Najm, H. N., Pébayt, P. P., Knio, O. M., Ghanem, R. G., Le Maître, O. P., Numerical challenges in the use of polynomial chaos representations for stochastic processes, SIAM Journal on Scientific Computing 26 (2) (2005) 698–719. https://doi.org/10.1137/S1064827503427741

[6] Delgado, P., Kumar, V., A stochastic Galerkin approach to uncertainty quantification in poroelastic media, Applied Mathematics and Computation 266 (2015) 328–338. https://doi.org/10.1016/j.amc.2015.04.127

[7] Desai, A., Scalable domain decomposition algorithms for uncertainty quantification in high performance computing, Ph.D. thesis, Carleton University, Ottawa, 2019. https://doi.org/10.22215/etd/2019-13428

[8] Desai, A., Khalil, M., Pettit, C., Poirel, D., Sarkar, A., Scalable domain decomposition solvers for stochastic PDEs in high performance computing, Computer Methods in Applied Mechanics and Engineering 335 (2018) 194–222. https://doi.org/10.1016/j.cma.2017.09.006

[9] Eldred, M. S., Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design, Proceedings of the 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Palm Springs, AIAA – American Institute of Aeronautics and Astronautics, 2009, pp. 1–37. https://doi.org/10.2514/6.2009-2274

[10] Elman, H. C., Miller, C. W., Phipps, E. T., Tuminaro, R. S., Assessment of collocation and Galerkin approaches to linear diffusion equations with random data, International Journal for Uncertainty Quantification 1 (1) (2011) 19–33. https://doi.org/10.1615/Int.J.UncertaintyQuantification.v1.i1.20

[11] Farhat, C., Crivelli, L., Roux, F.-X., A transient FETI methodology for large-scale parallel implicit computations in structural mechanics, International Journal for Numerical Methods in Engineering 37 (11) (1994) 1 945–1 975. https://doi.org/10.1002/nme.1620371111

[12] Ghanem, R., The nonlinear Gaussian spectrum of log-normal stochastic processes and variables, Journal of Applied Mechanics 66 (4) (1999) 964–973. https://doi.org/10.1115/1.2791806

[13] Ghanem, R. G., Spanos, P. D., Stochastic finite elements: A spectral approach, Springer, 1991. https://doi.org/10.1007/978-1-4612-3094-6

[14] Gottlieb, D., Xiu, D., Galerkin method for wave equations with uncertain coefficients, Communications in Computational Physics 3 (2) (2008) 505–518. https://doi.org/10.4208/cicp.2008.v3.p505

[15] Hughes, T. J. R., The finite element method: Linear static and dynamic finite element analysis, Dover Publications, 2000.

[16] Humar, J., Dynamics of structures, CRC press, 2012. https://doi.org/10.1201/b11772

[17] Khalil, M., Bayesian inference for complex and large-scale engineering systems, Ph.D. thesis, Carleton University, Ottawa, 2013. https://doi.org/10.22215/etd/2013-08572

[18] Kwon, S.-B., Bathe, K.-J., Noh, G., An analysis of implicit time integration schemes for wave propagations, Computers & Structures 230 (2020) No. 106188. https://doi.org/10.1016/j.compstruc.2019.106188

[19] Langtangen, H. P., Linge, S., Finite difference computing with PDEs: A modern software approach, Springer Nature, 2017. https://doi.org/10.1007/978-3-319-55456-3

[20] Le Maître, O. P., Knio, O. M., Spectral methods for uncertainty quantification: With applications to computational fluid dynamics, Springer, 2010. https://doi.org/10.1007/978-90-481-3520-2

[21] Mandel, J., Dohrmann, C. R., Convergence of a balancing domain decomposition by constraints and energy minimization, Numerical Linear Algebra with Applications 10 (7) (2003) 639–659. https://doi.org/10.1002/nla.341

[22] Mathew, T. P. A., Domain decomposition methods for the numerical solution of partial differential equations, Lecture Notes in Computational Science and Engineering Vol. 61, Springer, 2008. https://doi.org/10.1007/978-3-540-77209-5

[23] Mishra, S., Schwab, C., Šukys, J., Multi-level Monte Carlo finite volume methods for uncertainty quantification of acoustic wave propagation in random heterogeneous layered medium, Journal of Computational Physics 312 (2016) 192–217. https://doi.org/10.1016/j.jcp.2016.02.014

[24] Motamed, M., Nobile, F., Tempone, R., A stochastic collocation method for the second order wave equation with a discontinuous random speed, Numerische Mathematik 123 (3) (2013) 493–536. https://doi.org/10.1007/s00211-012-0493-5

[25] Nath, K., Dutta, A., Hazra, B., Long duration response evaluation of linear structural system with random system properties using time dependent polynomial chaos, Journal of Computational Physics 418 (2020) No. 109596. https://doi.org/10.1016/j.jcp.2020.109596

[26] Noh, G., Ham, S., Bathe, K.-J., Performance of an implicit time integration scheme in the analysis of wave propagations, Computers & Structures 123 (2013) 93–105. https://doi.org/10.1016/j.compstruc.2013.02.006

[27] Padillath Vasudevan, S. S., Scalable domain decomposition methods for nonlinear and time-dependent stochastic systems, Ph.D. thesis, Carleton University, Ottawa, 2023. https://doi.org/10.22215/etd/2023-15817

[28] Reynolds, D. D., Engineering principles of acoustics: Noise and vibration control, Allyn and Bacon, Boston, 1981.

[29] Sarkar, A., Benabbou, N., Ghanem, R., Domain decomposition of stochastic PDEs: Theoretical formulations, International Journal for Numerical Methods in Engineering 77 (5) (2009) 689–701. https://doi.org/10.1002/nme.2431

[30] Smith, B. F., Domain decomposition methods for partial differential equations, In: D. E. Keyes, A. Sameh, V. Venkatakrishnan (editors), Parallel numerical algorithms, Springer, 1997, pp. 225–243. https://doi.org/10.1007/978-94-011-5412-3_8

[31] Smith, R. C., Uncertainty quantification: Theory, implementation, and applications, Computational Science & Engineering Vol. 12, SIAM – Society for Industrial and Applied Mathematics, 2013. https://doi.org/10.1137/1.9781611973228

[32] Subber, W., Domain decomposition methods for uncertainty quantification, Ph.D. thesis, Carleton University, Ottawa, 2013. https://doi.org/10.22215/etd/2013-09627

[33] Toselli, A., Widlund, O. B., Domain decomposition methods – Algorithms and theory, Springer Series in Computational Mathematics Vol. 34, Springer, 2004. https://doi.org/10.1007/b137868

[34] Xiu, D., Fast numerical methods for stochastic computations: A review, Communications in Computational Physics 5 (2–4) (2009) 242–272. https://doi.org/10.4208/cicp.2009.v5.p242

[35] Xiu, D., Karniadakis, G. E., The Wiener-Askey polynomial chaos for stochastic differential equations, SIAM Journal on Scientific Computing 24 (2) (2002) 619–644. https://doi.org/10.1137/S1064827501387826

[36] Zakian, P., Khaji, N., A stochastic spectral finite element method for wave propagation analyses with medium uncertainties, Applied Mathematical Modelling 63 (2018) 84–108. https://doi.org/10.1016/j.apm.2018.06.027

[37] Zampieri, E., Pavarino, L. F., Implicit spectral element methods and Neumann-Neumann preconditioners for acoustic waves, Computer Methods in Applied Mechanics and Engineering 195 (19–22) (2006) 2 649–2 673. https://doi.org/10.1016/j.cma.2005.06.005