INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 11, Number 1, Pages 193–212

MAXWELL SOLUTIONS IN MEDIA WITH MULTIPLE RANDOM INTERFACES

CHANG-YEOL JUNG, BONGSUK KWON, ALEX MAHALOV AND THIEN BINH NGUYEN

(Communicated by Roger Temam)

Abstract. A hybrid operator splitting method is developed for computations of two-dimensional transverse magnetic Maxwell equations in media with multiple random interfaces. By projecting the solutions into the random space using the polynomial chaos (PC) projection method, the deterministic and random parts of the solutions are solved separately.
There are two independent stages in the algorithm: the Yee scheme with domain decomposition implemented on a staggered grid for the deterministic part and the Monte Carlo sampling in the post-processing stage. These two stages of the algorithm are subject of computational studies. A parallel implementation is proposed for which the computational cost grows linearly with the number of random interfaces. Output statistics of Maxwell solutions are obtained including means, variance and time evolution of cumulative distribution functions (CDF). The computational results are presented for several configurations of domains with random interfaces.
The novelty of this article lies in using level set functions to characterize the random interfaces and, under reasonable assumptions on the random interfaces (see Figure 1), the dimensionality issue from the PC expansions is resolved (see Sections 1.1.2 and 1.2).

Key words. Maxwell Equations, Evolution of probability distribution, Monte Carlo simulation, Stochastic partial differential equation, random media, random interface, Polynomial chaos.

1. Introduction

Time evolution of waves in random media has important applications in a wide range of areas such as medical imaging, wave scattering, radar detection, ionospheric plasmas and photonics devices (see e.g. [11]). Although the problem under consideration here is a forward problem, our approach reveals the effects of random inputs and provide some insights on inverse problems, e.g. reconstruction of the interior of a human body from MRI or Ultrasound, recovery of interior structural parameters of machines from non-destructive measurements, ionospheric dynamics and related problems.

In this article, we study the evolution of the cumulative distribution functions (CDF) in time of electromagnetic (EM) fields. The randomness of the EM fields is inherited from the randomness of the locations of interfaces, i.e., it is uncertain where two or more different media interface (e.g. [4], [6], [9], [25]). In particular, the permeability and permittivity fluctuate randomly in space (independently of time) around their mean values. The EM fields with a single interface were studied and simulated in [12]. It has been demonstrated that the polynomial chaos expansion (PCE) methods are superior to Monte-Carlo methods in a number of applications (see [1], [2], [7], [8], [10], [17], [18], [27], [28], [29], [30], [32], [22]). Here, we extend the single interface to two or multiple interfaces which are described as level sets, $\{z(x,y) = \xi_i\}$ where the level function $z = z(x,y)$ is a function of x, y, and ξ_i are

Received by the editors July 5, 2013 and, in revised form, July 13, 2013.
2000 *Mathematics Subject Classification.* 11K45, 65C20, 65C30, 82B31.
Alex Mahalov was partially supported by AFOSR grant FA9550-11-1-0220. Chang-Yeol Jung was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (NRF-2012R1A1B3001167).

independent random variables or parameters (see Figure1). The case where the level function z depends on some random variables will be considered elsewhere. For simplicity, here we consider only a deterministic level function.

We note that conventional PCE methods have some severe limitations. If the number of random variables increases, the computational cost will grow exponentially as indicated in the polynomial chaos expansions (1.2), (1.10) and (1.19) below where the polynomials pertaining to each random variable are multiplied in a tensor product form. Thus the number of unknown coefficients (PC modes) grows exponentially and the computations are very expensive. Monte-Carlo methods are then more feasible. To avoid this curse of dimensionality, in this article, along with the time explicit scheme we will update the PC modes in each interval of the level $z = z(x, y)$ (see Figure 1, and also e.g. (1.18) for two random parameters). The computational cost then grows linearly as explained in Sections 1.1.2 and 1.2 below. See also the Conclusion below. This substantially reduces the computational time with parallel computing as demonstrated in Figure 5 below.

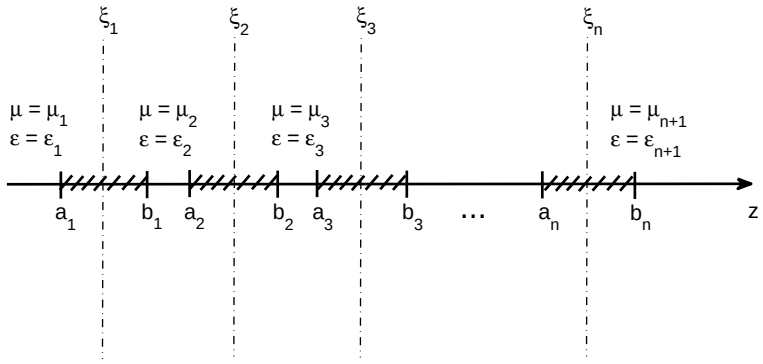

FIGURE 1. Multiple random interfaces $\{z(x, y) = \xi_i\}$ described by a level set function $z(x, y)$ where ξ_i are distributed over disjoint (a_i, b_i), $i = 1, 2, \cdots, n$.

We consider the following two-dimensional transverse magnetic Maxwell equations (e.g. [3], [19], [24]): for $(x, y, t) \in \mathbb{R}^2 \times (0, \infty)$,

$$\begin{cases} \dfrac{\partial H_1}{\partial t} = -\dfrac{1}{\mu}\dfrac{\partial E_3}{\partial y}, \\ \dfrac{\partial H_2}{\partial t} = \dfrac{1}{\mu}\dfrac{\partial E_3}{\partial x}, \\ \dfrac{\partial E_3}{\partial t} = \dfrac{1}{\epsilon}\dfrac{\partial H_2}{\partial x} - \dfrac{1}{\epsilon}\dfrac{\partial H_1}{\partial y}. \end{cases} \tag{1.1}$$

The initial conditions are $H_1(x, y, 0) = h_1(x, y)$, $H_2(x, y, 0) = h_2(x, y)$, $E_3(x, y, 0) = e_3(x, y)$, where $H = (H_1, H_2, 0)^T$ is the magnetic field, $E = (0, 0, E_3)^T$ is the electric

field, and h_1, h_2, e_3 are smooth functions. The boundary conditions will be specified below.

The parameters (permeability, permittivity) are given $(\mu, \epsilon) = (\mu_1, \epsilon_1)$ for $z(x, y) < \xi_1$, $(\mu, \epsilon) = (\mu_i, \epsilon_i)$ for $\xi_{i-1} < z(x, y) < \xi_i$, $i = 2, 3, \cdots, n$ and $(\mu, \epsilon) = (\mu_{n+1}, \epsilon_{n+1})$ for $z(x, y) > \xi_n$ where $\mu_i, \epsilon_i > 0$, μ_i's and ϵ_i's may be distinct (see Figure1). Here, ξ_i's are random variables and independent. Hence, we do not consider the correlations between the random variables ξ_i.

The case with correlated random variables will be studied in the forthcoming articles.

Remark 1.1. If an additive white noise (e.g. Gaussian) forcing drives System (1.1), which varies with time, the computational cost grows exponentially with the number of time steps because the PC expansion (1.2) is constructed in each time step. In this article, we do not consider the noise varying with time and this study will appear elsewhere.

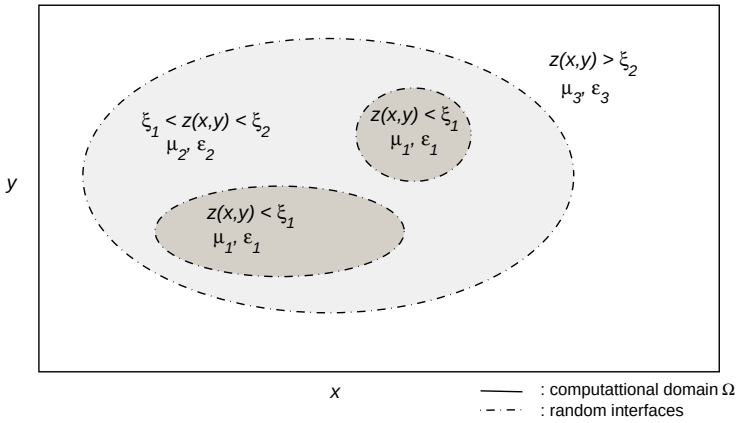

FIGURE 2. Two random interfaces for the model (1.1).

1.1. 2- random interfaces. We begin with two random interfaces (see Figure 2). We consider two cases. The first one is that the two random interfaces depend on a single random variable. In this case, a one-dimensional PC is used to approximate the randomness. The other one is that the two random interfaces depend on different random variables which are independent.

1.1.1. Case 1. In this section we consider the case $\xi_1 = \xi \in (a, b)$, $\xi_2 = \xi + \delta$, $\delta > b - a > 0$, where ξ is a uniform random variable over (a, b), for which its probability density function (PDF) is $f(\xi) = \frac{1}{b-a}\chi_{(a,b)}(\xi)$. Here $\chi_{(a,b)}$ is the characteristic function of (a, b). To deal with the random interfaces $\{z(x, y) = \xi_1\}$ and $\{z(x, y) = \xi_2\}$, employing the so-called Legendre polynomial chaos (PC) we

write:

$$\text{(1.2)}\qquad \begin{cases} H_1 = \sum_{k=0}^{N} H_{1k}(x,y,t)P_k(\xi), \\ H_2 = \sum_{k=0}^{N} H_{2k}(x,y,t)P_k(\xi), \\ E_3 = \sum_{k=0}^{N} E_{3k}(x,y,t)P_k(\xi), \end{cases}$$

where $P_k = P_k^{a,b}$ are shifted Legendre polynomials. Here,

$$\text{(1.3)}\qquad P_k(\xi) = P_k^{a,b}(\xi) = \widetilde{P}_k\left(\frac{2\xi - a - b}{b-a}\right),$$

where $\widetilde{P}_k$ are the standard Legendre polynomials with span $(-1,1)$.

Note that the number of PC modes in (1.2) does not grow in time and is fixed as N. This is because the fluctuations of the random variable(s) ξ or ξ_i below do not depend on t, x, y and depend only on the span of ξ. See Remark 1.1. Substituting the PC expansions (1.2) in Eq. $(1.1)_1$, multiplying $P_i(\xi)$ and integrating over (a,b) in ξ we obtain the PC mode equations for H_{1i}:

$$\text{(1.4)}\qquad \frac{\partial H_{1i}}{\partial t} = -\sum_{k=0}^{N} c_{ik}^{\mu}\frac{\partial E_{3k}}{\partial y},$$

$$\text{(1.5)}\qquad c_{ik}^{\mu} = \frac{\int_{-\infty}^{\infty} \frac{1}{\mu} P_i(\xi)P_k(\xi)\chi_{(a,b)}(\xi)d\xi}{\int_a^b P_i^2(\xi)d\xi}.$$

The PC mode equations for H_{2i}, E_{3i} similarly follow.

Depending on the value of z the coefficients $c_{ik}^{\mu} = c_{ik}^{\mu}(x,y)$ can be computed as follows. We note that $\mu = \mu_1$ for $z < \xi$, $\mu = \mu_2$ for $\xi < z < \xi + \delta$, and $\mu = \mu_3$ for $z > \xi + \delta$ (the parameter ε follows similarly). As in Figure 1, $a_1 = a$, $b_1 = b$, $a_2 = a + \delta$ and $b_2 = b + \delta$ with $n = 2$. Using $\int_a^b P_i(\xi)P_k(\xi)d\xi = \dfrac{b-a}{2i+1}\delta_{ik}$, we first obtain that

$$\text{(1.6a)}\qquad c_{ik}^{\mu} = \delta_{ik}\begin{cases} \mu_1^{-1} & \text{if } z < a, \\ \mu_2^{-1} & \text{if } b < z < a + \delta, \\ \mu_3^{-1} & \text{if } z > b + \delta, \end{cases}$$

and

(1.6b)

$$c_{ik}^{\mu} = \begin{cases} \frac{1}{\mu_1}\delta_{ik} + \left(\frac{1}{\mu_2} - \frac{1}{\mu_1}\right)\frac{2i+1}{b-a}\int_a^z P_i(\xi)P_k(\xi)d\xi & \text{if } a < z < b, \\ \frac{1}{\mu_2}\delta_{ik} + \left(\frac{1}{\mu_3} - \frac{1}{\mu_2}\right)\frac{2i+1}{b-a}\int_a^{z-\delta} P_i(\xi)P_k(\xi)d\xi & \text{if } a + \delta < z < b + \delta, \end{cases}$$

where the integrations $\int_a^z P_i(\xi)P_k(\xi)d\xi$ are given in explicit forms as in [12], [13] which can be evaluated with low computational cost.

We deduce the PC mode equations: for $\mathbf{H}_1(x,y,t) = (H_{10}, \cdots, H_{1N})$, $\mathbf{H}_2(x,y,t) = (H_{20}, \cdots, H_{2N})$ and $\mathbf{E}_3(x,y,t) = (E_{30}, \cdots, E_{3N})$,

$$\text{(1.7)}\qquad \begin{cases} \mathbf{H}_{1t} = -\Lambda^{\mu}(z)\mathbf{E}_{3y}, \\ \mathbf{H}_{2t} = \Lambda^{\mu}(z)\mathbf{E}_{3x}, \\ \mathbf{E}_{3t} = \Lambda^{\epsilon}(z)\left(\mathbf{H}_{2x} - \mathbf{H}_{1y}\right), \end{cases}$$

where the initial conditions are $\mathbf{H}_i(x,y,0) = (h_i(x,y),0,\cdots,0)$, $i=1,2$, and $\mathbf{E}_3(x,y,0) = (e_3(x,y),0,\cdots,0)$. The matrices $\Lambda^\mu(z) = \Lambda^\mu(z(x,y)) = (c^\mu_{ik})$ and $\Lambda^\epsilon(z) = \Lambda^\epsilon(z(x,y)) = (c^\epsilon_{ik})$, c^ϵ_{ik} is the c^μ_{ik} in (1.6) with μ_1 and μ_2 being replaced by, respectively, ϵ_1 and ϵ_2. Note that the entries c^μ_{ik}, c^ϵ_{ik} are continuous in z and we expect more regular solutions than in (1.1).

To compute the solutions of the system (1.7), we may incorporate some classical numerical methods (e.g. [14], [15]). Among others, we apply the Finite-Difference Time-Domain (FDTD) method by K. Yee (see [31], [24], and [12]). 2^{nd}-order centered finite difference is employed on a staggered grid for the space discretizations of the magnetic and electric fields. On this staggered grid, the magnetic and electric fields are located on the sides and at the center of each grid cell, respectively. The time derivatives are approximated in a same manner: the magnetic fields are first updated to the half time step; then the electric field is marched to the next time step based on the updated values of the former fields. For our Eqs. (1.7), we use the following notations for the magnetic fields $\mathbf{H}_{1,2}$ and electric field $\mathbf{E}_3$ at the node points on the staggered grid as in Figure 4: for $I,J,n \in \mathbb{Z}^+$, on the space domain $\Omega = (-2,2)\times(0,1)$,

$$
\text{(1.8)}\qquad \begin{cases} \mathbf{H}^{n-\frac12}_{1,i,j+\frac12} = \mathbf{H}_1(x_i, y_{j+\frac12}, t_{n-\frac12}), & i=0:I,\ j=0:J-1,\\ \mathbf{H}^{n-\frac12}_{2,i+\frac12,j} = \mathbf{H}_2(x_{i+\frac12}, y_j, t_{n-\frac12}), & i=0:I-1,\ j=0:J,\\ \mathbf{E}^{n}_{3,i,j} = \mathbf{E}_3(x_i, y_j, t_n), & i=0:I, j=0:J, \end{cases}
$$

where $x_p = -2+p\Delta x$, $y_q = q\Delta y$ and $t_r = r\Delta t$, $p,q,r\in\mathbb{Z}^+$, $p\in[0,I]$, $q\in[0,J]$, $r\in[0,T]$; and $\Delta x = 4/I$, $\Delta y = 1/J$ are the space steps and Δt is the time step. We note that $\mathbf{H}_{1,2}$, $\mathbf{E}_3$ are vectors of the deterministic PC modes defined in Eqs. (1.7). We then discretize the system (1.7) as follows: for $n\in\mathbf{Z}^+$,

(1.9)

$$
\begin{cases} \mathbf{H}^{n+\frac12}_{1,i,j+\frac12} = \mathbf{H}^{n-\frac12}_{1,i,j+\frac12} - \dfrac{\Delta t}{\Delta y}\Lambda^\mu_{i,j+\frac12}\left[\mathbf{E}^n_{3,i,j+1} - \mathbf{E}^n_{3,i,j}\right],\\ \quad i=1:I-1,\ j=0:J-1,\\ \mathbf{H}^{n+\frac12}_{2,i+\frac12,j} = \mathbf{H}^{n-\frac12}_{2,i+\frac12,j} + \dfrac{\Delta t}{\Delta x}\Lambda^\mu_{i+\frac12,j}\left[\mathbf{E}^n_{3,i+1,j} - \mathbf{E}^n_{3,i,j}\right],\\ \quad i=0:I-1,\ j=1:J-1,\\ \mathbf{E}^{n+1}_{3,i,j} = \mathbf{E}^n_{3,i,j} + \Lambda^\epsilon_{i,j}\left\{\dfrac{\Delta t}{\Delta x}\left[\mathbf{H}^{n+\frac12}_{2,i+\frac12,j} - \mathbf{H}^{n+\frac12}_{2,i-\frac12,j}\right] - \dfrac{\Delta t}{\Delta y}\left[\mathbf{H}^{n+\frac12}_{1,i,j+\frac12} - \mathbf{H}^{n+\frac12}_{1,i,j-\frac12}\right]\right\},\\ \quad i=1:I-1,\ j=1:J-1, \end{cases}
$$

where $\Lambda^\mu_{i,j+\frac12} = \Lambda^\mu(z(x_i, y_{j+\frac12}))$, $\Lambda^\mu_{i+\frac12,j} = \Lambda^\mu(z(x_{i+\frac12}, y_j))$ and $\Lambda^\epsilon_{i,j} = \Lambda^\epsilon(z(x_i,y_j))$.

1.1.2. Case 2. In this section we consider two random variables which determine two random interfaces, respectively. The random variables $\xi_i\in(a_i,b_i)$, $i=1,2$, are uniformly distributed over (a_i,b_i) and independent where $a_1<\xi_1<b_1<a_2<\xi_2<b_2$, and the PDFs are, respectively, $f_1(\xi_1) = \frac{1}{b_1-a_1}\chi_{(a_1,b_1)}(\xi_1)$, $f_2(\xi_2) = \frac{1}{b_2-a_2}\chi_{(a_2,b_2)}(\xi_2)$.

To deal with the random interfaces $\{z(x,y)=\xi_1\}$ and $\{z(x,y)=\xi_2\}$, employing the PC expansions we write:

$$\text{(1.10)}\qquad \begin{cases} H_1 = \displaystyle\sum_{k=0}^{N_1}\sum_{l=0}^{N_2} H_{1kl}(x,y,t)P_k^{a_1,b_1}(\xi_1)P_l^{a_2,b_2}(\xi_2), \\ H_2 = \displaystyle\sum_{k=0}^{N_1}\sum_{l=0}^{N_2} H_{2kl}(x,y,t)P_k^{a_1,b_1}(\xi_1)P_l^{a_2,b_2}(\xi_2), \\ E_3 = \displaystyle\sum_{k=0}^{N_1}\sum_{l=0}^{N_2} E_{3kl}(x,y,t)P_k^{a_1,b_1}(\xi_1)P_l^{a_2,b_2}(\xi_2). \end{cases}$$

Substituting the PC expansions in (1.1), multiplying by $P_i(\xi_1)P_j(\xi_2)$ and integrating over $(a_1,b_1)\times(a_2,b_2)$ we obtain the following modal equations:

$$\text{(1.11)}\qquad \frac{\partial H_{1ij}}{\partial t} = -\sum_{k=0}^{N_1}\sum_{l=0}^{N_2} c^{\mu}_{ijkl}\frac{\partial E_{3kl}}{\partial y},$$

$$\text{(1.12)}\qquad c^{\mu}_{ijkl} = \frac{\int_{\mathbb{R}^2}\frac{1}{\mu}P_i(\xi_1)P_k(\xi_1)\chi_{(a_1,b_1)}(\xi_1)P_j(\xi_2)P_l(\xi_2)\chi_{(a_2,b_2)}(\xi_2)d\xi_1 d\xi_2}{\int_{a_1}^{b_1}P_i^2(\xi_1)d\xi_1\int_{a_2}^{b_2}P_i^2(\xi_2)d\xi_2}.$$

Then the coefficients c^{μ}_{ijkl} are evaluated as follows:

$$\text{(1.13a)}\qquad c^{\mu}_{ijkl} = \delta_{ik}\delta_{jl}\begin{cases} \mu_1^{-1} & \text{if } z<a_1, \\ \mu_2^{-1} & \text{if } b_1<z<a_2, \\ \mu_3^{-1} & \text{if } z>b_2, \end{cases}$$

and

(1.13b)

$$c^{\mu}_{ijkl} = \begin{cases} \delta_{jl}\left[\frac{1}{\mu_1}\delta_{ik} + \left(\frac{1}{\mu_2}-\frac{1}{\mu_1}\right)\frac{2i+1}{b_1-a_1}\int_{a_1}^{z}P_i(\xi_1)P_k(\xi_1)d\xi_1\right] & \text{if } a_1<z<b_1, \\ \delta_{ik}\left[\frac{1}{\mu_2}\delta_{jl} + \left(\frac{1}{\mu_3}-\frac{1}{\mu_2}\right)\frac{2j+1}{b_2-a_2}\int_{a_2}^{z}P_j(\xi_2)P_l(\xi_2)d\xi_2\right] & \text{if } a_2<z<b_2, \end{cases}$$

Hence we similarly deduce the PC mode equations (1.7) and the Yee (FDTD) scheme (1.9). Here the corresponding vectors are $\mathbf{H}_1(x,y,t)=(H_{1,ij})$ (the indices i,j are in a dictionary order), $\mathbf{H}_2(x,y,t)=(H_{2,ij})$ and $\mathbf{E}_3(x,y,t)=(E_{3,ij})$, and the initial conditions are $H_{1,00}=h_1(x,y)$, $H_{2,00}=h_2(x,y)$, and $E_{3,00}=e_3(x,y)$. The matrices $\Lambda^{\mu}(z)=(c^{\mu}_{ijkl})$ and $\Lambda^{\epsilon}(z)=(c^{\epsilon}_{ijkl})$.

One disadvantage of the modal equation (1.11) is that it leads to a huge algebraic system when the Yee scheme (1.9) is applied since all the PC modes are arranged into vector forms, which in turn causes the coefficient matrices Λ^{μ} and Λ^{ε} to have very large sizes, i.e., $(1+N_1)(1+N_2)\times(1+N_1)(1+N_2)$. The case becomes worse when more interfaces are introduced into the modal problem. In the following part, we introduce a modification in deriving the PC modes which overcomes the drawback discussed above.

Substituting the PC expansions (1.10) in (1.1), multiplying by $P_i(\xi_1)P_j(\xi_2)$ and integrating over $(a_1,b_1)\times(a_2,b_2)$, if $a_1\le z<b_1$, since $\mu=\mu(\xi_1)$, $\varepsilon=\varepsilon(\xi_1)$ (see Figure 1), ξ_2 does not affect the parameters μ,ε and thus the integration with respect to ξ_2 has no effect in the resulting equations. Hence, we obtain:

$$\text{(1.14)}\qquad \frac{\partial H_{1ij}}{\partial t} = -c_i\sum_{k=1}^{N_1}\lambda_{ik}\frac{\partial E_{3kj}}{\partial y},\quad i=0\cdots N_1,$$

where

$$c_i = \left[\int_{a_1}^{b_1} (P_i(\xi_1))^2 d\xi_1\right]^{-1} = \frac{2i+1}{b_1 - a_1}, \tag{1.15}$$

and

$$\begin{aligned}\lambda_{ik} &= \int_{a_1}^{b_1} \frac{1}{\mu} P_k(\xi_1) P_i(\xi_1) d\xi_1 \\ &= \frac{1}{\mu_2}\int_{a_1}^{z} P_k(\xi_1) P_i(\xi_1) d\xi_1 + \frac{1}{\mu_1}\int_{z}^{b_1} P_k(\xi_1) P_i(\xi_1) d\xi_1.\end{aligned} \tag{1.16}$$

For other intervals, a similar procedure follows. Since $\lambda^{\mu}_{ik} = \lambda^{\mu}_{ki}$, c_i and λ^{μ}_{ik} do not change when we march the indexes, after the marching, we obtain,

$$\begin{cases} \dfrac{\partial \mathbf{H}_1}{\partial t} = -\dfrac{1}{\mu_1}\dfrac{\partial \mathbf{E}_3}{\partial y}, & z < a_1, \\ \dfrac{\partial \mathbf{H}_1}{\partial t} = -\Lambda^{\mu}_1 \dfrac{\partial \mathbf{E}_3}{\partial y}, & a_1 \le z < b_1, \\ \dfrac{\partial \mathbf{H}_1}{\partial t} = -\dfrac{1}{\mu_2}\dfrac{\partial \mathbf{E}_3}{\partial y}, & b_1 \le z < a_2, \\ \dfrac{\partial \mathbf{H}_1}{\partial t} = -\Lambda^{\mu}_2 \dfrac{\partial \mathbf{E}_3}{\partial y}, & a_2 \le z < b_2, \\ \dfrac{\partial \mathbf{H}_1}{\partial t} = -\dfrac{1}{\mu_3}\dfrac{\partial \mathbf{E}_3}{\partial y}, & b_2 \le z, \end{cases} \tag{1.17}$$

where $\mathbf{H}_1 = (H_{1kl})$, $\mathbf{E}_3 = (E_{3kl})$, $k = 1, \cdots, N_1$, $l = 1, \cdots, N_2$; and $\Lambda^{\mu}_{1,2} = (c_k \lambda_{kl})_{1,2}$ are matrices of sizes $(1+N_1)\times(1+N_1)$ and $(1+N_2)\times(1+N_2)$, respectively. The entries $(c_k\lambda_{kl})^{1,2}$ are similar to the coefficients c^{μ}_{ik} defined in $(1.6b)_1$, respectively with the removal of δ_{ik} and a, b replaced by appropriate a_1, b_1 and a_2, b_2, respectively. We notice that the sizes of $\Lambda^{\mu}_{(.)}$ are much reduced. Moreover, the computational cost is much reduced in case more random interfaces are introduced. H_2, E_3 follow similarly.

Remark 1.2.
It is noted that in the system (1.17), we treat the matrices $\mathbf{H}_1$, $\mathbf{E}_3$ as vectors by fixing one index and marching the other. Order of the indexes is important for intervals containing the random interfaces ξ_1, ξ_2, i.e., (a_1, b_1) and (a_2, b_2), respectively. In particular, for the interval (a_1, b_1), by fixing l, we march $\mathbf{H}_1$ by columns and solve for each column vector (H_{1il}), $i = 0, \cdots, N_1$ as in Eq. (1.14); whereas for the interval (a_2, b_2), we fix k and solve for each row vector (H_{1kj}), $j = 0, \cdots, N_2$.

Applying the Yee scheme to (1.17), we obtain an algebraic system as below, for $n, i, j \in \mathbb{Z}^+$,

(1.18a)
$$\begin{cases} \text{if } z < a_1, \\ \mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} = \mathbf{H}^{n-\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \dfrac{\Delta t}{\Delta y}\dfrac{1}{\mu_1}\left[\mathbf{E}^n_{3,i,j+1} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} = \mathbf{H}^{n-\frac{1}{2}}_{2,i+\frac{1}{2},j} + \dfrac{\Delta t}{\Delta y}\dfrac{1}{\mu_1}\left[\mathbf{E}^n_{3,i+1,j} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{E}^{n+1}_{1,i,j} = \mathbf{E}^n_{1,i,j} + \dfrac{1}{\epsilon_1}\left\{\dfrac{\Delta t}{\Delta x}\left[\mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} - \mathbf{H}^{n+\frac{1}{2}}_{2,i-\frac{1}{2},j}\right] - \dfrac{\Delta t}{\Delta y}\left[\mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \mathbf{H}^{n+\frac{1}{2}}_{1,i,j-\frac{1}{2}}\right]\right\}, \end{cases}$$

(1.18b)
$$\begin{cases} \text{if } a_1 \le z < b_1, \\ \mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} = \mathbf{H}^{n-\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \dfrac{\Delta t}{\Delta y}\Lambda^{\mu}_{1,i,j+\frac{1}{2}}\left[\mathbf{E}^n_{3,i,j+1} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} = \mathbf{H}^{n-\frac{1}{2}}_{2,i+\frac{1}{2},j} + \dfrac{\Delta t}{\Delta y}\Lambda^{\mu}_{1,i+\frac{1}{2},j}\left[\mathbf{E}^n_{3,i+1,j} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{E}^{n+1}_{1,i,j} = \mathbf{E}^n_{1,i,j} + \Lambda^{\epsilon}_{1,i,j}\left\{\dfrac{\Delta t}{\Delta x}\left[\mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} - \mathbf{H}^{n+\frac{1}{2}}_{2,i-\frac{1}{2},j}\right] - \dfrac{\Delta t}{\Delta y}\left[\mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \mathbf{H}^{n+\frac{1}{2}}_{1,i,j-\frac{1}{2}}\right]\right\}, \end{cases}$$

(1.18c)
$$\begin{cases} \text{if } b_1 \le z < a_2, \\ \mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} = \mathbf{H}^{n-\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \dfrac{\Delta t}{\Delta y}\dfrac{1}{\mu_2}\left[\mathbf{E}^n_{3,i,j+1} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} = \mathbf{H}^{n-\frac{1}{2}}_{2,i+\frac{1}{2},j} + \dfrac{\Delta t}{\Delta y}\dfrac{1}{\mu_2}\left[\mathbf{E}^n_{3,i+1,j} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{E}^{n+1}_{1,i,j} = \mathbf{E}^n_{1,i,j} + \dfrac{1}{\epsilon_2}\left\{\dfrac{\Delta t}{\Delta x}\left[\mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} - \mathbf{H}^{n+\frac{1}{2}}_{2,i-\frac{1}{2},j}\right] - \dfrac{\Delta t}{\Delta y}\left[\mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \mathbf{H}^{n+\frac{1}{2}}_{1,i,j-\frac{1}{2}}\right]\right\}, \end{cases}$$

(1.18d)
$$\begin{cases} \text{if } a_2 \le z < b_2, \\ \mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} = \mathbf{H}^{n-\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \dfrac{\Delta t}{\Delta y}\Lambda^{\mu}_{2,i,j+\frac{1}{2}}\left[\mathbf{E}^n_{3,i,j+1} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} = \mathbf{H}^{n-\frac{1}{2}}_{2,i+\frac{1}{2},j} + \dfrac{\Delta t}{\Delta y}\Lambda^{\mu}_{2,i+\frac{1}{2},j}\left[\mathbf{E}^n_{3,i+1,j} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{E}^{n+1}_{1,i,j} = \mathbf{E}^n_{1,i,j} + \Lambda^{\epsilon}_{2,i,j}\left\{\dfrac{\Delta t}{\Delta x}\left[\mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} - \mathbf{H}^{n+\frac{1}{2}}_{2,i-\frac{1}{2},j}\right] - \dfrac{\Delta t}{\Delta y}\left[\mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \mathbf{H}^{n+\frac{1}{2}}_{1,i,j-\frac{1}{2}}\right]\right\}, \end{cases}$$

(1.18e)
$$\begin{cases} \text{if } z \ge b_2, \\ \mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} = \mathbf{H}^{n-\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \dfrac{\Delta t}{\Delta y}\dfrac{1}{\mu_3}\left[\mathbf{E}^n_{3,i,j+1} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} = \mathbf{H}^{n-\frac{1}{2}}_{2,i+\frac{1}{2},j} + \dfrac{\Delta t}{\Delta y}\dfrac{1}{\mu_3}\left[\mathbf{E}^n_{3,i+1,j} - \mathbf{E}^n_{3,i,j}\right], \\ \mathbf{E}^{n+1}_{1,i,j} = \mathbf{E}^n_{1,i,j} + \dfrac{1}{\epsilon_3}\left\{\dfrac{\Delta t}{\Delta x}\left[\mathbf{H}^{n+\frac{1}{2}}_{2,i+\frac{1}{2},j} - \mathbf{H}^{n+\frac{1}{2}}_{2,i-\frac{1}{2},j}\right] - \dfrac{\Delta t}{\Delta y}\left[\mathbf{H}^{n+\frac{1}{2}}_{1,i,j+\frac{1}{2}} - \mathbf{H}^{n+\frac{1}{2}}_{1,i,j-\frac{1}{2}}\right]\right\}, \end{cases}$$

where $\Lambda^{\mu}_{(.),i,j+\frac{1}{2}} = \Lambda^{\mu}(z(x_i, y_{j+\frac{1}{2}}))$, $\Lambda^{\mu}_{(.),i+\frac{1}{2},j} = \Lambda^{\mu}(z(x_{i+\frac{1}{2}}, y_j))$, and $\Lambda^{\epsilon}_{.,i,j} = \Lambda^{\epsilon}(z(x_i, y_j))$.

Notice that the coefficient matrices $\Lambda^{(.)}_1(z)$ and $\Lambda^{(.)}_2(z)$ are independent of time. We just store those matrices only at the first step $n = 1$ and reuse them at $n \ge 2$.

1.2. n- random interfaces. We now extend the PC expansions (1.10) to the case with n- interfaces. Substituting the expansion

$$H_1 = \sum H_{1j_1\dots j_n}(x, y, t)P_{j_1}(\xi_1)\dots P_{j_n}(\xi_n), \tag{1.19}$$

into Eq. $(1.1)_1$, we obtain

$$\begin{aligned}&\sum \frac{\partial}{\partial t} H_{1j_1\dots j_n}(x,y,t)P_{j_1}(\xi_1)\dots P_{j_n}(\xi_n)\\ &\quad = -\frac{1}{\mu}\sum \frac{\partial}{\partial t} E_{3j_1\dots j_n}(x,y,t)P_{j_1}(\xi_1)\dots P_{j_n}(\xi_n).\end{aligned} \tag{1.20}$$

Here, $\sum = \sum_{j_1=1}^{N_1}\sum_{j_2=1}^{N_2}\cdots\sum_{j_n=1}^{N_n}$ and $\mu = \mu_i$ if $\xi_{i-1} \le z < \xi_i$ with $\xi_0 = -\infty$, $\xi_{n+1} = \infty$ as in Figure 1. Here, the random variables ξ_i are uniformly distributed over (a_i, b_i) with $a_{i+1} > b_i$, $i = 1, 2, \cdots, n$, and independent. The PDFs are respectively $f_i(\xi_i) = \frac{1}{b_i - a_i}\chi_{(a_i,b_i)}(\xi_i)$. Multiplying Eq. (1.20) by $P_{i_1}(\xi_i)\dots P_{i_n}(\xi_n)$ and integrating over $\Pi_{i=1}^n(a_i, b_i)$, we find that, if $a_1 \le z < b_1$, ξ_i, $i = 2, 3, \cdots, n$, do not affect the parameters μ, ε and we thud find that

$$\frac{\partial}{\partial t} H_{1i_1\dots i_n}(x,y,t) = -c_{i_1}\sum_{j_1=1}^{N_1}\lambda_{i_1,j_1}\frac{\partial}{\partial y}E_{3j_1,i_2\dots i_n}(x,y,t), \tag{1.21}$$

where

$$c_{i_1} = \left[\int_{a_1}^{b_1}(P_{i_1}(\xi_1))^2 d\xi_1\right]^{-1} = \frac{2i_1+1}{b_1-a_1}, \tag{1.22}$$

and

$$\begin{aligned}\lambda_{i1,j1} &= \int_{a_1}^{b_1}\frac{1}{\mu}P_{j_1}(\xi_1)P_{i_1}(\xi_1)d\xi_1\\ &= \frac{1}{\mu_2}\int_{a_1}^{z}P_{j_1}(\xi_1)P_{i1}(\xi_1)d\xi_1 + \frac{1}{\mu_1}\int_{z}^{b_1}P_{j1}(\xi_1)P_{i1}(\xi_1)d\xi_1.\end{aligned} \tag{1.23}$$

Note here that although we consider the n- tensor product of one-dimensional PC expansions, only an integration in one dimension needs to be computed to evaluate the coefficients in (1.21) and this makes the whole computational cost grows linearly. Furthermore, the explicit formula of the integration in (1.23) is available in [12].

For other intervals, the same procedure can be applied. In particular, if $b_1 \le z < a_2$, $\mu = \mu_2$, and hence

$$\frac{\partial}{\partial t} H_{i_1\dots i_n}(x,y,t) = -\frac{1}{\mu_2}\frac{\partial}{\partial y}E_{i_1,i_2\dots i_n}(x,y,t), \tag{1.24}$$

and H_2, E_3 follow similarly.

We can now discretize (1.21), (1.24) using the Yee scheme as in (1.18). The numerical simulations with 3 and 5 random interfaces are shown in Figs. 11, 12 below.

2. Statistics

Thanks to the orthogonality of Legendre polynomials, we can explicitly obtain the mean and variance. For example, we can obtain the mean and variance of E_3: for Case I, as in section 1.1.1,

$$\mathbb{E}(E_3) = \int_{-\infty}^{\infty} E_3 f_1(\xi)d\xi = E_{3,0}(x,y,t), \tag{2.1}$$

$$Var(E_3) = \int_{-\infty}^{\infty}(E_3)^2 f_1(\xi)d\xi - (\mathbb{E}(E_3))^2 = \sum_{i=1}^{N}\frac{(E_{3,i}(x,t))^2}{2i+1}. \tag{2.2}$$

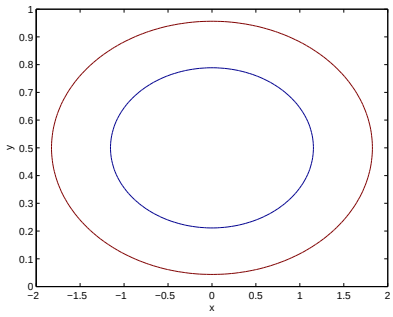

FIGURE 3. Two random interfaces $\{z(x,y)=\xi_1\}$ and $\{z(x,y)=\xi_2\}$; the case $\xi_1=-0.5$ (inner) and $\xi_2=1$ (outer) are plotted.

and, in general,

$$\mathbb{E}(E_3)=\int_{\mathbb{R}^n} E_3\prod_{i=1}^{n} f_i(\xi_i)d\xi = E_{3,0\cdots 0}(x,y,t), \tag{2.3}$$

$$\begin{aligned} Var(E_3) &= \int_{\mathbb{R}^n} (E_3)^2\prod_{i=1}^{n} f_i(\xi_i)d\xi - (\mathbb{E}(E_3))^2 \\ &= \sum_{\substack{i_1=0\\(i_1,\cdots,i_n)\neq(0,\cdots,0)}}^{N_1}\sum_{i_2=0}^{N_2}\cdots\sum_{i_n=0}^{N_n}\frac{(E_{3,i_1i_2\cdots i_n}(x,y,t))^2}{\Pi_{k=i_1}^{i_n}(2i_k+1)}, \end{aligned} \tag{2.4}$$

where $d\xi = d\xi_1\cdots d\xi_n$. Here, $f_i(\xi_i)=\frac{1}{b_i-a_i}\chi_{(a_i,b_i)}(\xi_i)$ are uniform probability density functions.

The Cumulative distribution functions (CDF) and density functions, e.t.c. can also be evaluated via the PC expansions (see [12]).

3. Numerical results

For the simulation purposes, we use the level set function z as below,

$$z(x,y)=\frac{3}{4}x^2+12(y-\frac{1}{2})^2-\frac{3}{2}, \tag{3.1}$$

for which two random interfaces $\{z(x,y)=\xi_1\}$ and $\{z(x,y)=\xi_2\}$, where ξ_1, ξ_2 are uniform random variables over (a_1,b_1) and (a_2,b_2), respectively, are plotted in Figure 3. The values of a_i, b_i, $i=1,2$, are specified in the simulations below.

Here and after, we use the computational domain $\Omega=(-2,2)\times(0,1)$. In Figs. 6 and 7, we consider the initial conditions:

$$H_1=H_2=0,\ E_3=(x+2)(x-2)y(y-1), \tag{3.2}$$

boundary conditions

$$E_3=0, \tag{3.3}$$

and the parameters,

$$\begin{cases} \mu_1=\varepsilon_1=1 & \text{if } z<\xi_1,\\ \mu_2=\varepsilon_2=2 & \text{if } \xi_1\le z<\xi_2,\\ \mu_3=\varepsilon_3=3 & \text{if } \xi_2\le z<\xi_3. \end{cases} \tag{3.4}$$

We notice that the boundary conditions for H_1, H_2 are not needed since following the Yee scheme (1.18), H_1, H_2 are first updated using the information of the previous time step of E_3; then E_3 is updated by the newly obtained H_1, H_2.

We consider the case 1 in section 1.1.1 of two random interfaces fluctuating with the same random parameter ξ and thickness δ between the two level sets. Two cases are taken into account. One for a small fluctuation with $a = -0.5$, $b = 0$, $\delta = 0.6$; and $a = -1$, $b = 0.8$, $\delta = 2$ for a large fluctuation. We observe the CDFs of the Maxwell solutions evolving in time for both cases as in Figure 6. It is shown that the fluctuation of the CDF of E_3 for the latter case is much wider than the former case. This is due to the wider variance of the random variable ξ in the latter case compared with that in the former one. The CDF plots are obtained via Monte-Carlo sampling at a post-processing stage after we obtain the PC mode solutions, i.e. PC expansion; see more explanations in Sec. 4 below and [12].

We now test with case 2 in which the two random variables ξ_1, ξ_2 are independent of each other. In this case, we use the intervals $a_1 = -1$, $b_1 = -0.1$, $a_2 = 0$, $b_2 = 0.9$. The CDF, mean and variance of E_3 at $t = 1.5$ are plotted in Figure 7. Here, we also observe the fluctuation of the CDF of the field E_3 when it evolves in time; and the fluctuation occurs in the regions locating the random interfaces (see Figure 7_3 showing the variance of E_3).

In Figure 8, for the compatible initial/boundary conditions, we use some known solutions for (1.1) as follows:

$$\begin{pmatrix} H_1 \\ H_2 \\ E_3 \end{pmatrix} = \begin{pmatrix} -\beta \\ \alpha \\ \sqrt{\dfrac{\mu}{\epsilon}} \end{pmatrix} \exp\left(i\omega \left(\frac{t}{\sqrt{\epsilon\mu}} + \alpha x + \beta y + \gamma \right) \right), \tag{3.5}$$

where $\alpha, \beta, \gamma, \omega \in \mathbb{R}$, $\alpha^2 + \beta^2 = 1$. We can then choose the compatible boundary conditions which satisfy (3.5) on $\partial\Omega$ and the compatible initial conditions (3.5) with $t = 0$. We try the real part of solutions (3.5) with $\alpha = \beta = 1/\sqrt{2}$, $\epsilon = \mu = 2$ and $\omega = 2$, $\gamma = 0$. Hence the initial conditions are: for all $(x, y) \in \Omega$,

$$\begin{cases} \begin{pmatrix} H_{1,00} \\ H_{2,00} \\ E_{3,00} \end{pmatrix} = \begin{pmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \\ 1 \end{pmatrix} cos\left(\sqrt{2}x + \sqrt{2}y\right), \\ H_{1kl} = H_{2kl} = E_{3kl} = 0 \quad \text{for } k, l = 1, \cdots, N. \end{cases} \tag{3.6}$$

Here we set the parameters $\mu_1 = \varepsilon_1 = 1$, $\mu_2 = \varepsilon_2 = 3$, $\mu_3 = \varepsilon_3 = 2$. Since $\partial\Omega \subset \{z(x, y) = \xi\}$ for all $\xi \in [-1, 1]$, the conditions (3.6) are compatible with the boundary conditions. For the boundary conditions, we only have to impose on E_3 field as indicated in (1.9) and (1.18) (i.e., the H- fields are updated from the E-field): for all (x,y) $\in \partial\Omega$, $t > 0$,

$$\begin{cases} E_{3,00} = cos\left(t + \sqrt{2}x + \sqrt{2}y\right), \\ E_{3,kl} = 0, \quad \text{for } k, l = 1, \cdots, N. \end{cases} \tag{3.7}$$

Notice also that the initial/boundary conditions are deterministic and thus only the conditions of the zero PC mode are imposed as in (3.7) and the conditions of other PC modes are zero.

The evolution in time of the electric field E_3 for the case 2 of the compatible initial and boundary conditions (3.6)–(3.7) is then simulated and the results are shown in Figure 9 at time $t = 0, 1, 2, 3$ respectively with two random interfaces. It is shown in the figures that even though we have deterministic initial and boundary conditions,

TABLE 1. Mean and variance of E_3 with 3 random interfaces at some points (x, y), $t = 1.5$ with different number of PC modes.

		PC		
	(x, y)	5	10	15
Mean	$(-0.8, 0.2)$	-0.4535	-0.4525	-0.4517
	$(0, 0.75)$	-0.6236	-0.6227	-0.6220
	$(-1.2, 0.8)$	-0.2212	-0.2200	-0.2194
	$(1.2, 0.5)$	-0.2645	-0.2631	-0.2625
	$(0.4, 0.4)$	-0.6585	-0.6572	-0.6565
Variance	$(-0.8, 0.2)$	4.25E-03	4.35E-03	4.35E-03
	$(0, 0.75)$	8.51E-04	8.89E-04	8.98E-04
	$(-1.2, 0.8)$	1.26E-03	7.27E-03	7.29E-03
	$(1.2, 0.5)$	5.19E-03	5.18E-03	5.17E-03
	$(0.4, 0.4)$	8.15E-03	9.15E-03	9.17E-03

the mean of E_3 becomes rough in the regions where the random interfaces fluctuate. This can be explained by the discontinuity of the parameters μ and ε across the interfaces.

We also check the decay of the PC modes of E_3 for both cases of initial/boundary conditions (3.2)–(3.6). The results are presented in Figure 10 with the z-axis is plotted in *log* scale. We notice that the PC modes decay exponentially. This implies that computations with only the first few modes, e.g., $N = 10$ in our numerical simulations, can produce a good approximation.

In Figure 11, we plot the CDF, mean and variance of E_3 at $x = 0, y = 0.75$ at $t = 1.5$ with 3 random interfaces $\{z(x, y) = \xi_1\}$, $\{z(x, y) = \xi_2\}$, $\{z(x, y) = \xi_3\}$. We use the same initial/boundary conditions as in (3.2), μ and ε are as in (3.4) with additional $\mu_4 = \varepsilon_4 = 4$ if $\xi_3 \le z < \xi_4$ and $a_1 = -1$, $b_1 = -0.5$, $a_2 = -0.4$, $b_2 = 1.4$, $a_3 = 1.5$, $b_3 = 2$. In Table 1, we compare the means and variances of the E_3 field at $t = 1.5$ at some specified points with different number of PC modes $PC = 5, 10, 15$. It is shown that these results are not much different from each other. This is due to the exponential decay of the PC modes (see Figure 10). Hence, for a simulation of increasing number of interfaces, a relatively small number of PC modes ($N = 5$) can be used for saving computational costs.

4. Parallel Computing

In this section, we present the parallelization of our numerical codes using the Message Passing Interface (MPI) library (see [16], [5], and [21]) for the purpose of saving computational time.

Before proceeding, we recall that our objective in this article is to study the evolution of the CDF in time of the electromagnetic fields in case there appear some uncertainties in the governing equations (1.1). By projecting the solutions of Eqs. (1.1) into the random space using the PC projection method, we were able to separate the deterministic and random parts of the solutions. We, then, could apply the well-known Yee scheme for solving the former parts numerically. In order to obtain the CDF evolution in time, we applied the Monte Carlo sampling method in a post-processing stage. To compare our Monte Carlo sampling (e.g. [12]) with the conventional one (e.g. [23]), we just note that the latter is applied as a preprocessing to the equations and then the resulting equations are deterministic for which we may use classical numerical methods. However, to get reliable statistics, large

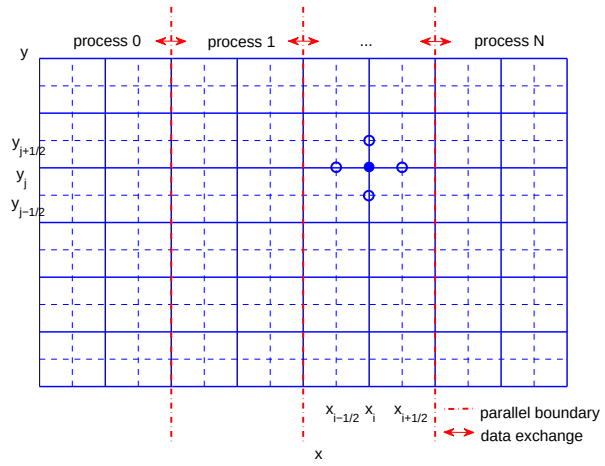

FIGURE 4. Domain decomposition for the parallel codes in a staggered grid.

samples are required and this causes a high cost in computations. Hence, there are two independent stages in the algorithm, i.e., the Yee scheme and the Monte Carlo sampling stages. Thus, it is natural that these two stages are objectives of our parallelization.

In the first stage, i.e., the Yee scheme, for parallel purposes, we apply the so-called domain decomposition in the x-direction to the $2D$ grid of the computational domain. We divide the computational domain into sub-domains and assign all computational loads relating to each of them to a process. We notice that between two neighboring sub-domains, there exists an interface in which data transfer is required between the two processes responsible for these sub-domains during the parallel computing (see Figure 4). We call these interfaces "parallel boundaries" to distinguish with the physical boundaries defined in the governing equations; and the data transfer among processes "parallel boundary update". Hence, for each time step, we need to update both the physical and parallel boundaries in each sub-domain. We note that since we parallel our computation based on the computational domain, not on the PC modes, the amounts of data transferring through the parallel boundaries are equal to each other and do not depend on the number of random interfaces and the fact that whether the sub-domains contain the random interfaces $\{z = \xi_i\}$ or not. We also notice that the Yee scheme is a type of multi-stage methods, i.e., at each time step, the update of $\mathbf{H}_1$, $\mathbf{H}_2$, and $\mathbf{E}_3$ in the system (1.18) consists of two sub-steps: firstly, the new $\mathbf{H}_1$ and $\mathbf{H}_2$ are updated based on the old $\mathbf{E}_3$ of the previous time step; then, the new $\mathbf{E}_3$ is updated following the new $\mathbf{H}_1$, $\mathbf{H}_2$ of the current time step. Hence, we need to have the parallel boundaries updated twice. Since we decompose the computational domain in the $x-$direction, only parallel boundaries for $\mathbf{H}_2$ and $\mathbf{E}_3$ need updating (see the Yee scheme in (1.18)).

The parallelization of the Monte Carlo sampling stage as a post-processing is simpler than that of the Yee scheme stage because we parallel our codes based on the number of samples used in the Monte Carlo method and all these samples are independent of each other. Thus we only need to divide the samples into portions

and assign each portion to a process. We, then, collect the results from all processes to obtain the aiming CDFs.

For numerical simulations, we test our parallel codes with two cases. In the first case, we simulate with two random variables or interfaces with initial and boundary data as in (3.6) and (3.7) to illustrate for the necessity of updating of both physical and parallel boundaries in each sub-domain. And in the second case, we test with three random interfaces with initial and boundary conditions (3.2) and (3.3) aiming for the saving of computational time. The numerical results of these tests are similar to those of the serial cases shown in Figs. 8, 11, respectively.

In Figure 5, we plot the computational time (in minutes) for different numbers of processes used in the parallel computation. The result obtained from the number of processes 1 is equivalent to that of a serial code. We conclude that using parallel computation, we can save a considerable amount of time. As shown in the figure, a parallel code with 8 processes is about 8 times faster than a serial code. The increase of computational time in case of two random variables or interfaces with 12 processes can be explained by the increase of the data transferring cost among the processes. Hence, in our problem, we conclude that parallel codes using 8 processes are more optimal for cases of 2 random interfaces; whereas for 3 and 5 interfaces, 16 processes yield faster computational time.

In Figure 12, we further test with the case of 5 random interfaces with initial / boundary conditions as in (3.2)–(3.3) and other parameters are as below:

$$\text{(4.1)} \qquad \begin{cases} \mu_1 = \varepsilon_1 = 1, \\ \mu_2 = \varepsilon_2 = 2, \\ \mu_3 = \varepsilon_3 = 3, \\ \mu_4 = \varepsilon_4 = 4, \\ \mu_5 = \varepsilon_5 = 1, \\ \mu_6 = \varepsilon_6 = 2, \end{cases} \quad \text{and} \quad \begin{cases} \xi_1 \in (a_1, b_1) = (-1, -0.5), \\ \xi_2 \in (a_2, b_2) = (-0.4, -0.1), \\ \xi_3 \in (a_3, b_3) = (0, 0.5), \\ \xi_4 \in (a_4, b_4) = (1, 1.4), \\ \xi_5 \in (a_5, b_5) = (1.5, 2), \end{cases}$$

It is shown in the figure that the mean and variance of E_3 behave more wildly than those of lesser numbers of random interfaces. It is because with the introduction of 5 random interfaces, their fluctuations occur in most of the computational domain Ω.

5. Conclusion

We studied the uncertainty propagation from parameters pertaining to certain random media for which multiple media randomly interface. Hence, multiple random parameters are present in the system considered here, and then the computational cost via the PC expansions grows exponentially with the dimensionality, i.e. the number of random parameters (ξ_i in this article). To overcome such dimensionality issues, rather than the tensor products of one-dimensional nodal sets, the stochastic collocation on a sparse grid by e.g. the Smolyak algorithm can be considered (see e.g [28], [26]). However, this collocation method is for approximating the multi-dimensional integrations involved in the coefficients, e.g. in (1.12). To avoid the dimensionality problem, we compute the PC modes in each interval of the level set function z (see Figure 1) which makes the computational cost grow linearly with the dimensionality. Furthermore, evaluating the integrations via the explicit formula given in [12], we avoid also the integration errors with low computational cost. In the numerical simulations as in Figs. 5 and 12, the 5 random parameters or interfaces were considered. Since increasing the number of random

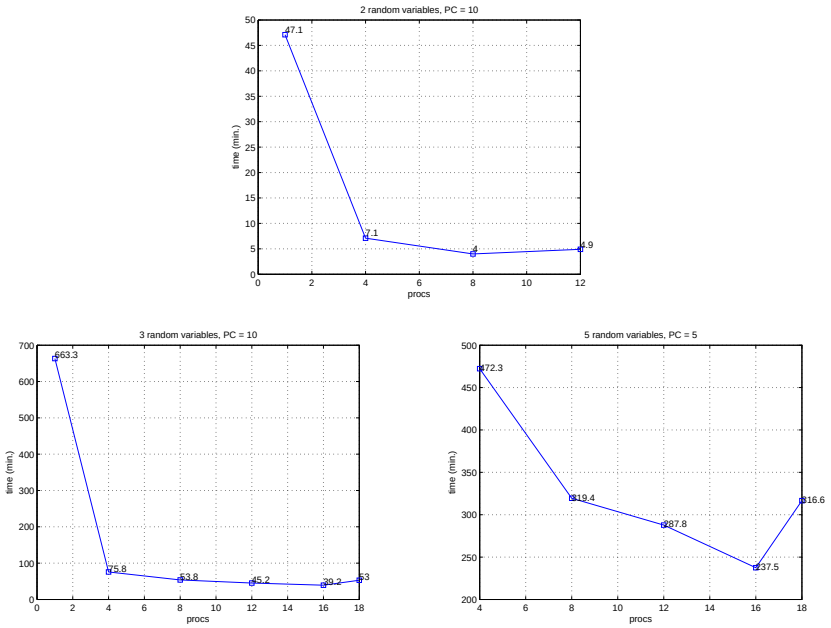

FIGURE 5. Computational time of different number of PC modes. x-axis is the number of processes used in the computation. Number of processes $= 1$ is equivalent to the serial codes.

parameters increases the computational time only linearly, with a linear growth cost, more random parameters can be taken into account in the (parallel) computations. We believe that our approach can be easily and efficiently applied to wave-type equations or conservation laws with multiple random parameters.

References

[1] Q.-Y. Chen, D. Gottlieb and J. S. Hesthaven, *Uncertainty analysis for the steady-state flows in a dual throat nozzle.* J. Comput. Phys. 204(2005), pp. 378–398.

[2] A. J. Chorin and O. H. Hald, *Stochastic tools in mathematics and science.* Springer, New York, 2006.

[3] C. Chauvière, J. S. Hesthaven and L. Lurati, *Computational modeling of uncertainity in time-domain electromagnetics.* SIAM J. Sci. Comput. 28(2006), pp. 751–775.

[4] J.-P. Fouque, J. Garnier, G. Papanicolaou, and K. Sølna, *Wave propagation and time reversal in randomly layered media.* Springer, New York, 2007.

[5] W. Gropp, E. Lusk, and A. Skjellum, *Using MPI: portable parallel programming with the message-passing interface.* 2nd ed., The MIT Press, Cambridge, Massachusetts, London, England, 1999.

[6] J. Garnier and G. Papanicolaou, *Pulse propagation and time reversal in random waveguides.* SIAM J. Appl. Math. 67(2007), pp. 1718-1739.

[7] R. G. Ghanem and P. D. Spanos, *Stochastic finite elements: a spectral approach.* Springer-Verlag, New York, 1991.

[8] D. Gottlieb and Dongbin Xiu, *Galerkin method for wave equations with uncertain coefficients.* Commun. Comput. Phys. 3(2008), pp. 505-518.

[9] A. Guadagnini, D. M. Tartakovsky and C. L. Winter, *Random domain decomposition for flow in heterogeneous stratified aquifers*, Stochastic Environ. Res. and Risk Assessment (SERRA), vol. 17, no. 6 (2003), pp. 394-407.

[10] T. Y. Hou, W. Luo, B. Rozovskii and H.-M. Zhou, *Wiener chaos expansions and numerical solutions of randomly forced equations of fluid mechanics.* Journal of computational physics. 216 (2006), 687–706.

[11] V. Isakov, *Inverse problems for partial differential equations.* Second edition. Applied Mathematical Sciences, 127, Springer-Verlag, New York, 2006.
[12] C. Jung, *Evolution of probability distribution in time for solutions of hyperbolic equations.* Journal of Scientific Computing. 41 (2009), No. 1, 13–48.
[13] C. Jung and A. Mahalov, *Wave propagation in one-dimensional random waveguides*, Discrete and continuous dynamical systems, 28, No. 1 (2010), pp. 147–159.
[14] C. Jung and R. Temam, *Finite volume approximation of one-dimensional stiff convection-diffusion equations*, Journal of Scientific Computing, v. 41 (2009), pp 384-410
[15] C. Jung and R. Temam, *Finite volume approximation of two-dimensional stiff problems*, International Journal of Numerical Analysis and Modeling. v.7 (2010), no.3, pp 462-476.
[16] G. E. Karniadakis and R. M. Kirby II, *Parallel scientific computing in C++ and MPI: a seamless approach to parallel algorithms and their implementation.* Cambridge University Press, 2003.
[17] O.M.Knio and O.P.Le Maître, *Uncertainty propagation in CFD using polynomial chaos decomposition*, Fluid dynamics research, 38 (2006), 616–640.
[18] G. Lin, X. Wan, C.-H. Su and G.E. Karniadakis, *Stochastic computational fluid mechanics.* Computing in Science and Engineering, 9 (2007), pp. 21-29
[19] A.S. Mahalov, *Mathematical investigation of periodic acoustical waveguides of an arbitrary shape.* Journal of mathematical analysis and applications. 127 (1986), no. 2, 569–576.
[20] O. P. Le Matre, Omar M. Knio, *Spectral Methods for Uncertainty Quantification with Applications to Computational Fluid Dynamics*, Springer, New York, 2010.
[21] P. S. Pacheco, *Parallel programming with MPI.* Morgan Kaufmann Publisher, Inc., San Francisco, California. An imprint of Elsevier, 1997.
[22] R. Pulch and D. Xiu, *Generalized polynomial chaos for a class of linear conservation laws*, J. Sci. Comput. 51 (2012), pp.. 293-312.
[23] C. P. Robert and G. Casella, *Monte Carlo statistical methods.* Springer Science+Business Media, New York, 2004.
[24] A. Taflove and S. Hagness, *Computational electromagnetics: the finite-difference time-domain method.* 3rd ed., Norwood, MA: Artech House, 2005.
[25] D. M. Tartakovsky and D. Xiu, *Numerical methods for differential equations in random domains*, SIAM J. Sci. Comput., vol. 28, no. 3 (2006), pp. 1167-1185.
[26] T. Tang and T. Zhou, *Convergence analysis for stochastic collocation methods to scalar hyperbolic equations with a random wave speed*, Commun. Comput. Phys., vol. 8, no. 1 (2010), pp. 226-248.
[27] D. Xiu, *Fast numerical methods for stochastic computations: A review.* Commun. Comput. Phys. 5 (2008), 242–272.
[28] D. Xiu and J. S. Hesthaven, *High-order collocation methods for differential equations with random inputs.* SIAM J. Sci. Comput. 27(2005), pp 1118-1139.
[29] D. Xiu and G. E. Karniadakis, *The Wiener-Askey polynomail chaos for stochastic differential equations.* SIAM J. Sci. Comput. 24(2002), pp 619-644.
[30] D. Xiu and G. E. Karniadakis, *Modeling uncertainty in steady state diffusion problems via generalized polynomial chaos.* Comput. Methods Appl. Mech. Engrg. 199(2002), pp 4927-4948.
[31] K. Yee, *Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media.* IEEE Transactions on Antennas and Propagation, 14, No. 5 (1966), pp 302-307.
[32] Z. Zhang, B. Rozovskii, M. V. Tretyakov, and G. E. Karniadakis, *A multistage wiener chaos expansion method for stochastic advection-diffusion-reaction equations*, SIAM J. Sci. Comput., 34, No. 2 (2012), pp. 914-A936.

Department of Mathematical Sciences, School of Natural Science, Ulsan National Institute of Science and Technology, UNIST-gil 50, Ulsan 689-798, Republic of Korea
E-mail: cjung@unist.ac.kr, bkwon@unist.ac.kr, thienbinh84@unist.ac.kr

School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85287-1804, USA
E-mail: alex.mahalov@asu.edu

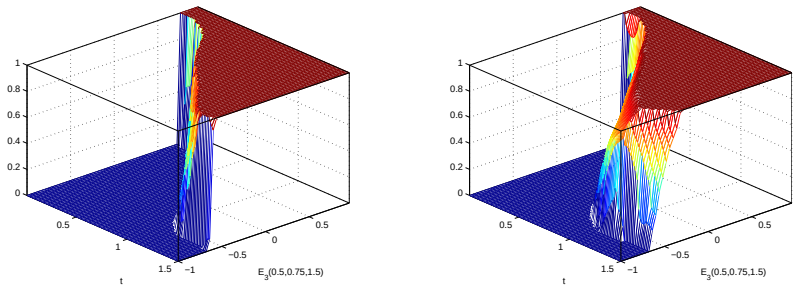

FIGURE 6. Case I: Evolution of Cumulative distribution function (CDF) of E_3 at $x = 0, y = 0.75$ with $N = 10$, (μ_i, ε_i), $\mu_1 = \varepsilon_1 = 1$, $\mu_2 = \varepsilon_2 = 2$ and $\mu_3 = \varepsilon_3 = 3$; LEFT: with a relatively small fluctuation $(a = -0.5, b = 0, \delta = 0.6)$; RIGHT: with a relatively large fluctuation $(a = -1, b = 0.8, \delta = 2)$.

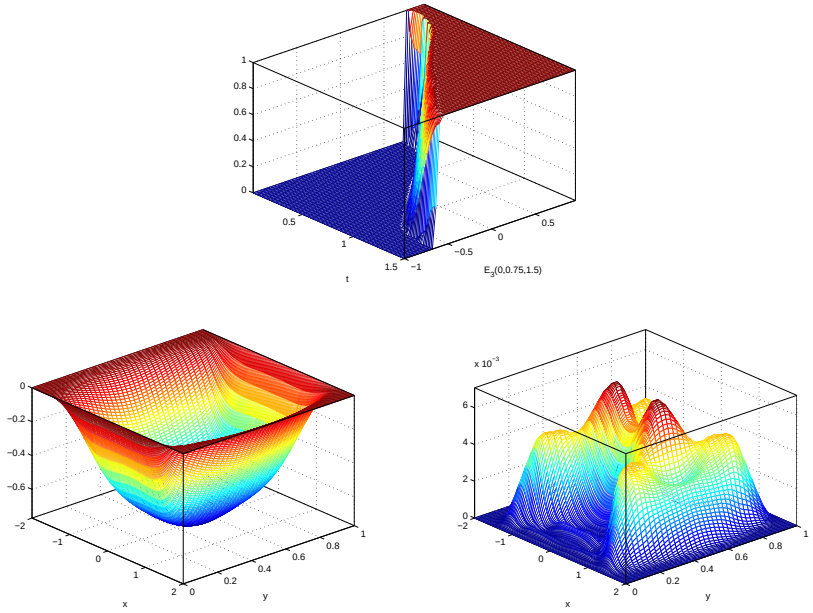

FIGURE 7. Case II: TOP: Evolution of Cumulative distribution function (CDF) of E_3 at $x = 0, y = 0.75$ with $N = 10$, initial and boundary conditions as in (3.2) and (3.3); BOTTOM: mean and variance of E_3 at $t = 1.5$, respectively.

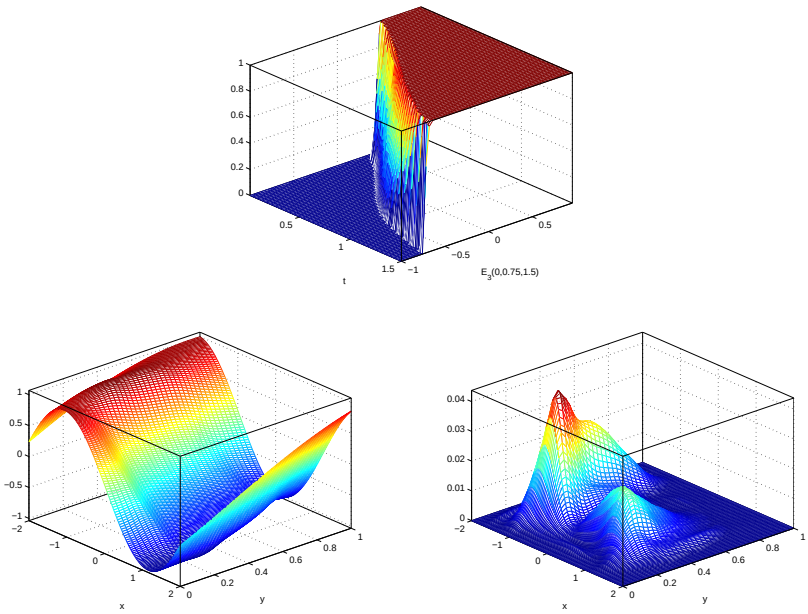

FIGURE 8. Case II: TOP: Evolution of Cumulative distribution function (CDF) of E_3 at $x = 0, y = 0.75$ with $N = 10$ using compatible initial/boundary conditions (3.6) and (3.7), μ and ε as in (3.4); BOTTOM: mean and variance of E_3 at $t = 1.5$, respectively.

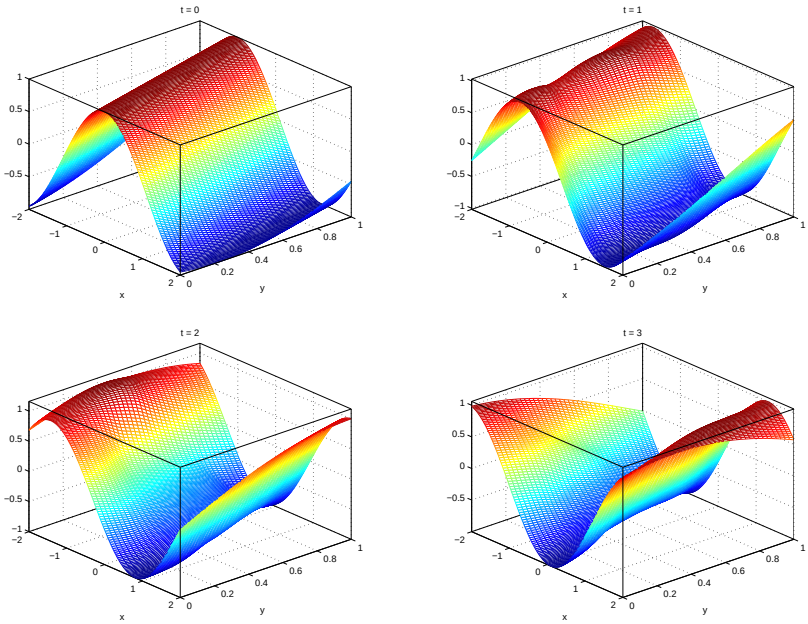

FIGURE 9. Evolution of the mean of E_3 with $N = 10$ for the case of compatible initial/boundary conditions (3.6)–(3.7) at $t = 0, 1, 2, 3$, respectively.

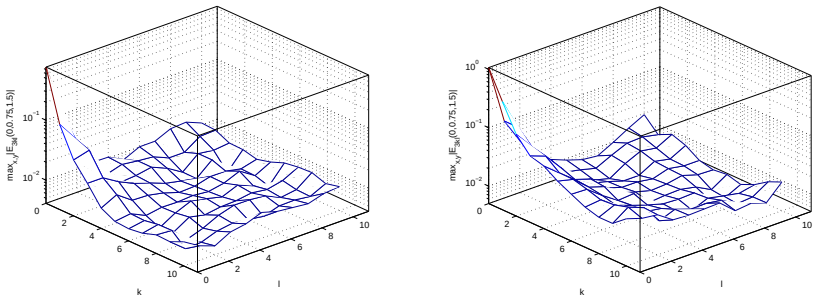

FIGURE 10. PC modes decay of E_3 with 2 random interfaces at $x = 0, y = 0.75$, $t = 1.5$. The z-axis is plotted in *log* scale. LEFT: initial/boundary conditions as in (3.2) and (3.3); RIGHT: compatible initial/boundary conditions (3.6) and (3.7).

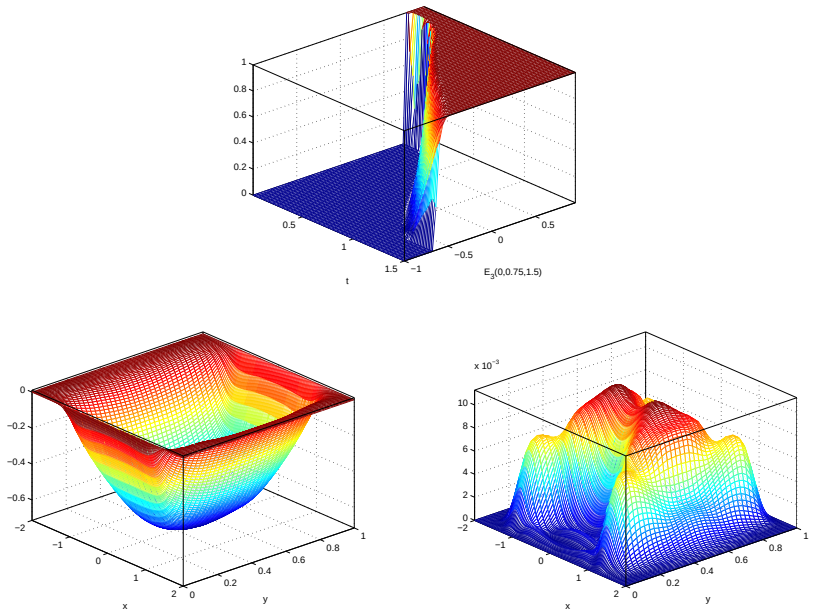

FIGURE 11. CDF, mean and variance of E_3 at $x = 0, y = 0.75$ with 3 independent random variables and 5 PC modes using initial/boundary conditions (3.2) and (3.3), μ and ε as in (3.4) with additional $\mu_4 = \varepsilon_4 = 4$ if $\xi_3 \le z < \xi_4$.

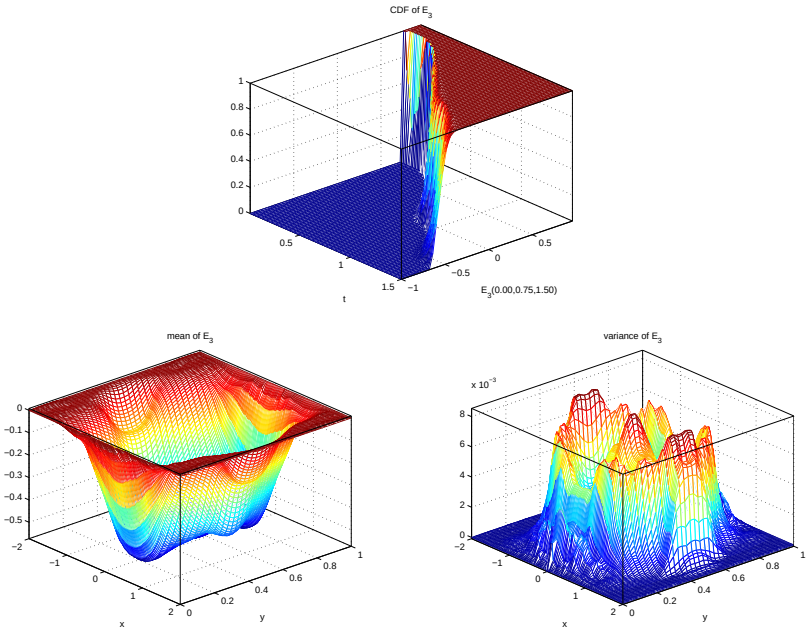

FIGURE 12. CDF, mean and variance of E_3 with 5 random interfaces with initial/boundary conditions (3.2)–(3.3) and other parameters as in (4.1). $PC = 5$.