Received: 26 January 2020 | Revised: 29 July 2020 | Accepted: 8 October 2020 | Published on: 22 January 2021

DOI: 10.1002/num.22742

RESEARCH ARTICLE

WILEY

Application of direct meshless local Petrov–Galerkin method for numerical solution of stochastic elliptic interface problems

Mostafa Abbaszadeh[1] | **Mehdi Dehghan**[1] | **Amirreza Khodadadian**[2,3] | **Clemens Heitzinger**[3,4]

[1]Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences, Amirkabir University of Technology, Tehran, Iran
[2]Institute of Applied Mathematics, Leibniz University of Hannover, Hanover, Germany
[3]Institute for Analysis and Scientific Computing, Vienna University of Technology (TU Wien), Vienna, Austria
[4]School of Mathematical and Statistical Sciences, Arizona State University, Tempe, Arizona, USA

Correspondence
Mostafa Abbaszadeh, Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences, Amirkabir University of Technology, No. 424, Hafez Avenue, 15914 Tehran, Iran.
Email: m.abbaszadeh@aut.ac.ir

Funding information
FWF (Austrian Science Fund) START Project no. Y660 PDE Models for Nanotechnology

Abstract

A truly meshless numerical procedure to simulate stochastic elliptic interface problems is developed. The meshless method is based on the generalized moving least squares approximation. This method can be implemented in a straightforward manner and has a very good accuracy for solving this kind of problems. Several realistic examples are presented to investigate the efficiency of the new procedure. Furthermore, compared with other meshless methods that have been developed, the present technique needs less CPU time.

KEYWORDS

complex computational domains, generalized moving least squares approximation, jump boundary conditions, stochastic elliptic interface problems, thin film elliptic interface problem

1 | INTRODUCTION

In many physical and biological problems, the system consists of different regions with specific physical properties that are separated by a surface or boundary between the materials. For instance interfaces within a viscous fluid domain [33], or two fluids with different viscosity (Hele-Shaw flow) [26], the interface between liquids and insulators [29, 31], the interface between SiO_2 and the conducting channel [30, 32], dielectric interfaces [20], and optical waveguides with arbitrarily curved interfaces [54] can be mentioned as important challenges. Some numerical techniques such as finite element, finite

volume, and finite difference methods are employed to solve the elliptic interface problems [11, 20, 21, 23, 25, 41, 49], and so on.

Authors of [44] developed an adaptive refinement scheme to reduce the geometry discretization error and provided higher-order enrichment functions for the interface-enriched generalized finite element method (FEM). The main aim of [43] is to solve the elliptic interface Poisson problem using discontinuous Galerkin (DG) formulation. A high-order hybridizable DG method for solving elliptic interface problems has been proposed in [27] such that the solution and gradient are nonsmooth because of jump conditions across the interface. Authors of [14] solved the thin film elliptic interface problems with variable coefficients by using a domain decomposition method.

Stochastic elliptic equations are interesting problems since they combine randomness and interfaces. In these problems, random coefficients such as the diffusion coefficient or the permittivity are taken into account. As a physical example, the authors of [46] studied the apparent conductivity of materials composed of multiple materials with random internal geometries and conductivities.

In [55], a Galerkin method was employed to convert the random problem into an uncoupled system of deterministic interface problems. A fast finite difference scheme was proposed to compute the variance of random solutions by using a low-rank approximation based on the pivoted Cholesky decomposition in [22]. The authors of [53] used the immersed FEMs (for spatial discretization) and a sparse grid collocation algorithm based on the Smolyak construction (for the probability space) to solve the stochastic interface problems.

A complex variable boundary element-free method is developed in [12] for solving the Helmholtz equation using regularized combined field integral equations. The main aim of [36] is to propose a complex variable boundary point interpolation method for numerical analysis of the nonlinear Signorini problem. The error of the finite point method is analyzed in [34] theoretically. Analysis of an augmented moving least squares approximation and the associated localized method of fundamental solutions are developed in [42]. The variational multiscale interpolating element-free Galerkin method is developed in [51] to obtain the numerical solution of the nonlinear Darcy–Forchheimer model. An element-free Galerkin method with a penalty for solving general second-order elliptic problems with mixed boundary conditions is presented in [52].

The authors of [17] proposed a novel truly meshless numerical method based on the interpolating stabilized MLS approximation for solving the one- and two-dimensional elliptic interface problems. The authors [37] studied the inherent instability of the interpolating moving least squares (IMLS) method and they developed the meshless Galerkin boundary node method based on IMLS for solving potential and Stokes problems. A meshless projection iterative method based on the moving Kriging interpolation is developed in [35] for solving nonlinear Signorini problems. The improved least-squares (IMLS) approximation is employed in [50] to present the improved element-free Galerkin method to simulate 3D elastoplasticity problems.

As mentioned in [47], the moment matrix in the MLS method may be singular for ill quality point sets and the computation of the inverse of the singular moment matrix is difficult. To overcome this problem, a regularized moving least-square method with a nonsingular moment matrix is proposed in [47]. The weighted orthogonal basis functions are applied in [48] for the improved interpolating moving least-square to obtain a diagonal moment matrix, which can overcome the difficulty caused by directly inversing singular or ill-conditioned matrices.

The main aim of [13] is to combine the dimension splitting method with the improved complex variable element-free Galerkin method to solve three-dimensional advection–diffusion problems. The technique used in [17] is based on the singular weight function to construct the shape functions. The meshless methods are proposed in [7] to simulate 2D interface heat equation with regular and irregular interface shapes. Authors of [28] developed a multiquadric radial basis function and

its integrated form for solving the elliptic problems with curved or closed interface. This paper aims to propose a local meshless technique without using a singular weight function. The developed method in [28] requires less CPU time than the method presented in [17]. The direct meshless local Petrov–Galerkin (DMLPG) method is employed in [2] for solving the fractional fourth-order partial differential equation on computational domains with complex shape. Authors of [15] developed a linear combination of shape functions of local radial basis functions collocation method and moving Kriging interpolation technique for solving some multi-dimensional problems such as Cahn–Hilliard, Swift–Hohenberg and phase field crystal equations. The DMLPG method has been employed in [6] to solve the stochastic Cahn–Hilliard–Cook and Swift–Hohenberg equations. The local weak form technique based on the reproducing kernel particle method is applied in [1] to simulate the two-dimensional nonstationary Boussinesq equations. Authors of [3] proposed a meshless weak form based on the Crank–Nicolson method and the interpolating stabilized element free Galerkin technique for solving the Oldroyd equation. Authors of [4] combined the upwind local radial basis function-differential quadrature method with the space-splitting idea for solving some conservation laws equations such as shallow water and Buckley–Leverett equations. An improvement of meshless boundary element method based on the shape functions of radial basis functions-QR is proposed in [5] for solving the two-dimensional elasticity problems. The meshless radial basis functions method is proposed in [40] to solve the time-fractional nonlinear Schrödinger equation in one and two dimensions which appear in quantum mechanics. The main aim of [18] is to propose a new numerical algorithm based on the interpolating element free Galerkin method for solving the magnetohydrodynamics equation. The moving Kriging element-free Galerkin method has been combined with the variational multiscale algorithm in [16] for solving some partial differential equations with discontinuous solutions.

In this paper, we also study elliptic equations and with discontinuous coefficients and randomness. To that end, we first develop the space discretization and in order to represent the random process (i.e., stochastic coefficients), a Karhunen–Loéve (KL) expansion [19] is used. We consider this model

$$-\nabla \cdot (\mathcal{G}(\mathbf{x},\omega)\,\nabla u(\mathbf{x},\omega)) = f(\mathbf{x},\omega) \qquad \text{in } D = D^+ \cup D^-, \tag{1.1a}$$

$$[u(\mathbf{x},\omega)]_\kappa = 0 \qquad \text{on } \Gamma, \tag{1.1b}$$

$$\left[\mathcal{G}(\cdot,\omega)\,\frac{\partial u(\cdot,\omega)}{\partial \mathbf{n}}\right]_\kappa = 0 \qquad \text{on } \Gamma, \tag{1.1c}$$

$$u(\mathbf{x},\omega) = g \qquad \text{on } \partial D. \tag{1.1d}$$

Here, D is the computational geometry consisting of D^+ and D^-, the interface is defined as $\Gamma := D^+ \cap D^-$, the random variable $\omega = (\omega_1, \omega_2, \ldots, \omega_n) \in \Omega$ is a random parameter in a probability space $(\Omega, \mathbb{A}, \mathbb{P})$, Ω denotes the set of elementary events, that is, the sample space, $\mathbb{A}$ is the σ-algebra of all possible events, and $\mathbb{P} : \mathbb{A} \to [0,1]$ is a probability measure. Furthermore, n indicates the unit outward normal vector of the interface κ. The jump $[u(\cdot,\omega)]$ is understood to be $u^-(\cdot,\omega) - u^+(\cdot,\omega)$ on κ in the sense of the trace of each sample ω. The diffusion coefficient $\mathcal{G}(x,\omega) : D \times \Omega \to \mathbb{R}$ is discontinuous across the interface and can be decomposed into the random field

$$\mathcal{G}(\mathbf{x},\omega) = \begin{cases} \mathcal{G}^-(\mathbf{x},\omega) & \mathbf{x} \in D^-, \\ \mathcal{G}^+(\mathbf{x},\omega) & \mathbf{x} \in D^+, \end{cases}$$

with continuous and bounded covariance functions. We assume that $\mathcal{G}(\cdot,\omega)$ is uniformly bounded and coercive, that is, there exist $\mathcal{G}_{\min}, \mathcal{G}_{\max} \in (0,\infty)$ such that

$$P\{\omega \in \Omega : \mathcal{G}(\mathbf{x},\omega) \in [\mathcal{G}_{\min}, \mathcal{G}_{\max}] \quad \forall \mathbf{x} \in D\} = 1.$$

Additionally, the stochastic function $f(\cdot,\omega)$ is square integrable with respect to $\mathbb{P}$.

The KL expansion is a useful tool for representing the stochastic process $\mathcal{G}(\omega, \mathbf{x})$. For the interface problems, it is characterized by its mean $\mathcal{G}_0^{\pm} = \int \mathcal{G}^{\pm}(\mathbf{x}, \omega)\, d\mathbb{P}(\omega)$ and its covariance

$$\mathrm{cov}_{\mathcal{G}^{\pm}}(\mathbf{x}, \mathbf{y}) = \int_{\Omega} \left(\mathcal{G}^{\pm}(\mathbf{x}, \omega) - \mathcal{G}_0^{\pm}(\mathbf{x})\right)\left(\mathcal{G}^{\pm}(\mathbf{y}, \omega) - \mathcal{G}_0^{\pm}(\mathbf{y})\right) d\mathbb{P}(\omega), \tag{1.2}$$

which leads to the decomposition

$$\mathcal{G}^{\pm}(\mathbf{x}, \omega) = \mathcal{G}_0^{\pm}(\mathbf{x}) + \sum_{n=1}^{\infty} \sqrt{\lambda_n^{\pm}}\, \mathcal{G}_n(\mathbf{x})^{\pm} \xi_n^{\pm}(\omega). \tag{1.3}$$

Here $\mathcal{G}_n^{\pm}$ are the orthogonal eigenfunctions and $\lambda_i^{\pm}$ are the corresponding eigenvalues of the eigenvalue problem

$$\int_{D^{\pm}} \mathrm{cov}_{\mathcal{G}^{\pm}}(\mathbf{x}, \mathbf{y})\, \mathcal{G}_n^{\pm}(\mathbf{y})\, d\mathbf{y} = \lambda_n^{\pm} \mathcal{G}_n^{\pm}(\mathbf{x}), \tag{1.4}$$

and $\left\{\xi_n^{\pm}(\omega)\right\}$ are mutually uncorrelated random variables satisfying

$$\mathbb{E}\left[\xi_n^{\pm}\right] = 0, \quad \mathbb{E}\left[\xi_n^{\pm} \xi_m^{\pm}\right] = \delta_{nm}, \tag{1.5}$$

where $\mathbb{E}$ indicates the expectation and the random variables are defined by

$$\xi_n^{\pm}(\omega) \coloneqq \frac{1}{\sqrt{\lambda_n^{\pm}}} \int_{D^{\pm}} \left[\mathcal{G}^{\pm}(\mathbf{x}, \omega) - \mathcal{G}_0^{\pm}(\mathbf{x})\right] \mathcal{G}_n^{\pm}(x)\, d\mathbf{x} \quad \forall n \in \mathbb{N}. \tag{1.6}$$

The KL expansion in the form of (6) is of little use because it is an infinite series. In practice, one adopts a finite series expansion, that is, an N-term truncation

$$\mathcal{G}^{\pm}(\mathbf{x}, \omega) \approx \mathcal{G}_0^{\pm}(\mathbf{x}, \omega) + \sum_{n=1}^{N} \sqrt{\lambda_n^{\pm}}\, \mathcal{G}_n(\mathbf{x})^{\pm} \xi_n^{\pm}(\omega). \tag{1.7}$$

The rest of the paper is organized as follows. In Section 2, we introduce the generalized moving least squares approximation. The implementation of the meshless method in one-dimensional and two-dimensional elliptic interface problems are explained in Sections 3 and 4, respectively. The numerical examples for both problems are given in Section 5. Finally, the conclusions are drawn in Section 6.

2 | GENERALIZED MOVING LEAST SQUARES APPROXIMATION

The traditional MLPG method is proposed by Atluri and his coauthors [8–10]. The DMLPG method is based upon the generalized moving least squares (GMLS) approximation. In recent years, a generalization of MLS approximation that it is called GMLS approximation is developed in [39]. As is mentioned in [38] the generalized form of the MLS approximation directly approximates boundary conditions and local weak forms. The GMLS approximation does not even have shape function. Instead, derivatives are estimated directly from nodal values, avoiding the inefficient detour via classical derivatives of shape functions. In MLPG technique and other MLS approximation based methods, the stiffness matrix is constructed by integrating over shape functions of MLS approximation and their derivatives. These shape functions are complicated and have no closed analytical forms. Then, a high-order numerical quadrature with many integration points is required to calculate the appeared integrals in the corresponding weak form. The DMLPG method avoids integration over shape functions of MLS and replaces it by the much cheaper integration over polynomials. Also, the DMPLG technique ignores shape functions completely. Altogether, the method is simpler, faster and more accurate

than the original MLPG method [38]. In [6], the method was used to solve time-dependent stochastic fourth-order problems. Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ be a set of scattered data and

$$\Theta_k(u) = \mathcal{G}_k, \quad 1 \le k \le M, \tag{2.1}$$

in which $\Theta_1, \Theta_2, \ldots, \Theta_M$ are linear functionals and the $\mathcal{G}_k$ are constants. Consider $\widehat{\Theta}_k$ as an approximation for Θ_k. Now, we want to calculate $\Upsilon_j(\Theta_k)$ such that

$$\Theta_k(u) \approx \widehat{\Theta}_k(u) = \sum_{j=1}^{M} \Upsilon_j(\Theta_k)\, u\left(x_j\right), \quad 1 \le k \le M. \tag{2.2}$$

The GMLS approximation [39] is based on approximating of $\Theta(u)$ associated with the linear functionals $\left\{\kappa_j(u)\right\}_{j=1}^{N}$ thus we set

$$\Theta(u) \approx \widehat{\Theta}(u) = \sum_{j=1}^{N} \Upsilon_j(\Theta)\, \kappa_j(u). \tag{2.3}$$

It is assumed that relation (2.3) is exact for a finite dimensional subspace of $\mathbf{P}_m^d = \text{span}\left\{p_1, p_2, \ldots, p_Q\right\}$, where p_i are polynomials. Therefore, we have

$$\sum_{j=1}^{N} \Upsilon_j(\Theta)\, \kappa_j(p) = \Theta(p) \quad \forall p \in \mathbf{P}_m^d. \tag{2.4}$$

Minimizing the weighted discrete L_2-norm

$$J(p) \coloneqq \sum_{j=1}^{N} \left(\kappa_j(u) - \kappa_j(p)\right)^2 W\left(\Theta; \kappa_j\right), \tag{2.5}$$

for finding

$$p^* \coloneqq \underset{p \in \mathbf{P}_m^d}{\text{argmin}}\, J(p), \tag{2.6}$$

yields the GMLS approximation, where $W(\lambda; \kappa_j)$ is a weight function. We have [39]

$$p(\mathbf{x}) = \sum_{i=1}^{Q} l_i p_i(\mathbf{x}). \tag{2.7}$$

Because of the linearity of the functionals κ_j, equation (2.7) is derived as

$$\kappa_j(p(\mathbf{x})) = \sum_{i=1}^{Q} l_i \kappa_j(p_i(\mathbf{x})) = \mathbf{p}_{\kappa_j}^T \mathbf{l}, \tag{2.8}$$

where

$$\mathbf{p}_{\kappa_j} \coloneqq \left(\kappa_j\left(p_1(\mathbf{x}), \kappa_j\left(p_2(\mathbf{x}), \ldots, \kappa_j\left(p_Q(\mathbf{x})\right)^T, \quad \mathbf{l} \coloneqq \left(l_1, l_2, \ldots, l_Q\right)^T. \right.\right.\right. \tag{2.9}$$

According to the above descriptions, Equation (2.5) can be rewritten as

$$J(\mathbf{p}) = \sum_{j=1}^{N} \left(\kappa_j(u) - \mathbf{p}_{\kappa_j}^T \mathbf{l}\right)^2 W\left(\Theta; \kappa_j\right). \tag{2.10}$$

Let

$$\mathbf{Q}^* \coloneqq \left(\mathbf{M}(\Theta; \kappa)\right)^{-1} \mathbf{B}(\Theta; \kappa)\, \widehat{\boldsymbol{\kappa}}(u), \tag{2.11}$$

be the solution of relation (2.10), where

$$\mathbf{M}(\Theta; \kappa) \coloneqq \mathbf{P}_\kappa \mathbf{W}(\Theta; \kappa)\, \mathbf{P}_\kappa^T,$$

$$\begin{aligned}
&\mathbf{B}(\Theta;\kappa) \coloneqq \mathbf{P}_{\kappa}\mathbf{W}\big(\Theta;\kappa\\
&\mathbf{P}_{\kappa} \coloneqq \left(p_{\kappa_1}, p_{\kappa_2}, \ldots, p_{\kappa_N}\right),\\
&\mathbf{W}(\Theta;\kappa) \coloneqq \operatorname{diag}\left(W(\Theta;\kappa_1), W\big(\Theta;\kappa_2\big), \ldots, W\big(\Theta;\kappa_N\big)\right)\\
&\widehat{\kappa}(u) \coloneqq (\kappa_1(u), \kappa_2(u), \ldots, \kappa_N(u))^T.
\end{aligned}$$

Therefore, $\widehat{\Theta}(u)$ will be the approximation of $\Theta(u)$ [39]

$$\Theta(u) = \Theta(p^*) = \mathbf{p}_{\Theta}^T\mathbf{Q}^* = \mathbf{p}_{\Theta}^T(\mathbf{M}(\Theta;\kappa))^{-1}\mathbf{B}(\Theta;\kappa)\,\widehat{\boldsymbol{\kappa}}(u). \tag{2.12}$$

Remark 2.1 As already mentioned, in this technique, we can apply the Dirichlet boundary condition exactly, that is, the boundary condition is imposed by some changes in the final coefficient matrix and also in the right-hand side vector of the final algebraic system of equations. Consider the following general problem

$$\begin{cases} \mathcal{L}u = f, & \text{in } \Omega,\\ u = g, & \text{on } \partial\Omega, \end{cases} \tag{2.13}$$

where $\mathcal{L}$ is a linear differential operator and also the functions f and g are known. After constructing the discretization of main equations, based on the interior and boundary integrals, we get the system of algebraic equations

$$\mathbf{AU} = \mathbf{F}. \tag{2.14}$$

Now, to apply the Dirichlet boundary condition the following steps must be performed:

1 Find rows of matrix $\mathbf{A}$ associated with the boundary nodes, for example the ith-row of matrix $\mathbf{A}$ that it is related to the ith-node on $\partial\Omega$. Set

$$A(i,:) = 0, \quad \text{i.e.,} \quad A(i,j) = 0 \ \text{ for every } \ j = 1, 2, \ldots, dim(\mathbf{A}),$$

$$A(i,i) = 1.$$

2 Substitute the value of the boundary condition into the ith-element on the right hand side vector F, that is one should put have

$$F(i) = g(\mathbf{x}_i).$$

According to the above steps, the Dirichlet boundary condition can be applied, exactly and also without interpolation error.

3 | ONE-DIMENSIONAL ELLIPTIC INTERFACE PROBLEMS

In this section, we explain the implementation of the GMLS approximation (for space discretization) for one-dimensional elliptic interface problems. Also, the KL expansion will be used for the stochastic discretization. First, we consider 1D elliptic interface problem

$$\begin{cases} -(\mathcal{G}u_x)_x + \kappa u = f + \nu\delta_{\alpha}, & x \in D = (a,b),\\ u(a) = u_1, & u(b) = u_2, \end{cases} \tag{3.1}$$

where $\mathcal{G}$ and ν are smooth functions on both $D^- = (0, \alpha)$ and $D^+ = (\alpha, 1)$. Furthermore

$$\mathcal{G} = \begin{cases} \mathcal{G}^- (x, \omega), & x \in D^-, \\ \mathcal{G}^+ (x, \omega), & x \in D^+, \end{cases} \tag{3.2}$$

and

$$\kappa = \begin{cases} \kappa^- (x, \omega), & x \in D^-, \\ \kappa^+ (x, \omega), & x \in D^+. \end{cases} \tag{3.3}$$

For the numerical plane, we assume

$$\begin{cases} -\frac{\partial}{\partial x}\left(\mathcal{G}^- \frac{\partial u^-}{\partial x}\right) + \kappa^- u^- = f^-, & x \in D^-, \\ u^- (a) = u_1, \end{cases} \tag{3.4}$$

and

$$\begin{cases} -\frac{\partial}{\partial x}\left(G^+ \frac{\partial u^+}{\partial x}\right) + \kappa^+ u^+ = f^+, & x \in D^+, \\ u^+ (b) = u_2, \end{cases} \tag{3.5}$$

with the following conditions

$$[u]_\alpha = g_1, \quad [\mathcal{G} u_x]_\alpha = g_2, \tag{3.6}$$

where

$$[u]_\alpha = u^+ (\alpha) - u^- (\alpha), \tag{3.7}$$

$$\left[\mathcal{G}\frac{\partial u}{\partial x}\right]_\alpha = \mathcal{G}^+ \frac{\partial u^+}{\partial x}\bigg|_{x=\alpha} - \mathcal{G}^- \frac{\partial u^-}{\partial x}\bigg|_{x=\alpha}. \tag{3.8}$$

To obtain a numerical algorithm, we consider

$$\left\{a = \sigma_1^-, \sigma_2^-, \ldots, \sigma_{N-1}^- < \sigma_N^- = \alpha\right\} \subset D^-, \left\{\alpha = \sigma_1^+, \sigma_2^+, \ldots, \sigma_{N-1}^+, \sigma_N^+ = b\right\} \subset D^+. \tag{3.9}$$

To apply the DMLPG2 method [39], we define

$$\Theta_{1,i}^- (u^-) := u^- \left(\sigma_i^-\right) \approx \widetilde{u}^- \left(\sigma_i^-\right) = \sum_{j=1}^{N} \Upsilon_{1,j}^- \left(\sigma_i^-\right) u^- \left(\sigma_i^-\right), \tag{3.10}$$

$$\Theta_{2,i}^- (u^-) := u_x^- \left(\sigma_i^-\right) \approx \widetilde{\Theta}_{2,i}^- (u^-) = \sum_{j=1}^{N} \Upsilon_{2,j}^- \left(\sigma_i^-\right) u^- \left(\sigma_i^-\right), \tag{3.11}$$

$$\Theta_{3,i}^- (u^-) := u_{xx}^- \left(\sigma_i^-\right) \approx \widetilde{\Theta}_{3,i}^- (u^-) = \sum_{j=1}^{N} \Upsilon_{3,j}^- \left(\sigma_i^-\right) u^- \left(\sigma_i^-\right), \tag{3.12}$$

$$\Theta_{1,i}^+ \left(u^+\right) := u^+ \left(\sigma_i^+\right) \approx \widetilde{u}^+ \left(\sigma_i^+\right) = \sum_{j=1}^{N} \Upsilon_{1,j}^+ \left(\sigma_i^+\right) u^+ \left(\sigma_i^+\right), \tag{3.13}$$

$$\Theta_{2,i}^- \left(u^+\right) := u_x^+ \left(\sigma_i^+\right) \approx \widetilde{\Theta}_{2,i}^+ \left(u^+\right) = \sum_{j=1}^{N} \Upsilon_{2,j}^+ \left(\sigma_i^+\right) u^+ \left(\sigma_i^+\right), \tag{3.14}$$

$$\Theta_{3,i}^- \left(u^+\right) := u_{xx}^- \left(\sigma_i^+\right) \approx \widetilde{\Theta}_{3,i}^+ \left(u^+\right) = \sum_{j=1}^{N} \Upsilon_{3,j}^+ \left(\sigma_i^+\right) u^+ \left(\sigma_i^+\right). \tag{3.15}$$

The GMLS approximation concludes

$$\begin{aligned}\boldsymbol{\Upsilon}_{k,:}^{-,T}\left(\mathbf{x}_i\right) &= \Theta_{k,i}^{-}\left(\mathbf{p}^T\right)\mathbf{M}^{-1}\mathbf{B}, \quad k=1,2,3,\\ \boldsymbol{\Upsilon}_{k,:}^{+,T}\left(\mathbf{x}_i\right) &= \Theta_{k,i}^{+}\left(\mathbf{p}^T\right)\mathbf{M}^{-1}\mathbf{B}, \quad k=1,2,3,\end{aligned} \tag{3.16}$$

and according to (2.12), we can obtain

$$\begin{aligned}\Theta_{1,i}^{\pm}\left(\mathbf{p}^T\right) &= \left(p_1\left(\sigma_i\right)p_2\left(\sigma_i\right)\ \ldots\ p_Q\left(\sigma_i\right)\right),\\ \Theta_{2,i}^{\pm}\left(\mathbf{p}^T\right) &= \left(\left.\frac{\partial p_1\left(x\right)}{\partial x}\right|_{x=\sigma_i}\left.\frac{\partial p_2\left(x\right)}{\partial x}\right|_{x=\sigma_i}\ \ldots\ \left.\frac{\partial p_Q\left(x\right)}{\partial x}\right)_{x=\sigma_i}\right),\\ \Theta_{3,i}^{\pm}\left(\mathbf{p}^T\right) &= \left(\left.\frac{\partial^2 p_1\left(x\right)}{\partial x^2}\right|_{x=\sigma_i}\left.\frac{\partial^2 p_2\left(x\right)}{\partial x^2}\right|_{x=\sigma_i}\ \ldots\ \left.\frac{\partial^2 p_Q\left(x\right)}{\partial x^2}\right|_{x=\sigma_i}\right).\end{aligned}$$

The boundary conditions can be applied by GMLS collocation. Let

$$\begin{aligned}\Theta_k^{-}\left(\sigma_N^{-}\right) &= u^{-}\left(\sigma_N^{-}\right), \quad \sigma_N^{-}\in\Gamma_I,\\ \Theta_k^{+}\left(\sigma_1^{+}\right) &= u^{+}\left(\sigma_1^{+}\right), \quad \sigma_1^{+}\in\Gamma_I,\end{aligned} \tag{3.17}$$

where

$$A_j^{-} = \Upsilon_j^{-'}\left(\sigma_N^{-}\right), \quad A_j^{+} = \Upsilon_j^{+'}\left(\sigma_1^{+}\right), \tag{3.18}$$

as well as

$$\begin{aligned}\boldsymbol{\Upsilon}^{-'\mathrm{T}}\left(\sigma_N^{-}\right) &= \Theta_k^{-}\left(\mathbf{p}^{\mathrm{T}}\right)\mathbf{M}^{-1}\left(\sigma_N^{-}\right)\mathbf{B}\left(\sigma_N^{-}\right),\\ \boldsymbol{\Upsilon}^{+'\mathrm{T}}\left(\sigma_1^{+}\right) &= \Theta_k^{+}\left(\mathbf{p}^{\mathrm{T}}\right)\mathbf{M}^{-1}\left(\sigma_1^{+}\right)\mathbf{B}\left(\sigma_1^{+}\right).\end{aligned} \tag{3.19}$$

For the Neumann boundary conditions we have

$$\begin{aligned}\xi_k^{-}\left(\sigma_N^{-}\right) &= \frac{\partial u}{\partial x}\left(\sigma_N^{-}\right), \quad \sigma_N^{-}\in\Gamma_I,\\ \xi_k^{+}\left(\sigma_1^{+}\right) &= \frac{\partial u}{\partial x}\left(\sigma_1^{+}\right), \quad \sigma_1^{+}\in\Gamma_I,\end{aligned} \tag{3.20}$$

where

$$B_j^{-} = c_j^{-'}\left(\sigma_N^{-}\right), \quad B_j^{+} = c_j^{+'}\left(\sigma_1^{+}\right), \tag{3.21}$$

and

$$\begin{aligned}\mathbf{c}^{-'\mathrm{T}}\left(\sigma_N^{-}\right) &= \xi_k^{-}\left(\mathbf{p}^{\mathrm{T}}\right)\mathbf{M}^{-1}\left(\sigma_N^{-}\right)\mathbf{B}\left(\sigma_N^{-}\right),\\ \mathbf{c}^{+'\mathrm{T}}\left(\sigma_1^{+}\right) &= \xi_k^{+}\left(\mathbf{p}^{\mathrm{T}}\right)\mathbf{M}^{-1}\left(\sigma_1^{+}\right)\mathbf{B}\left(\sigma_1^{+}\right).\end{aligned} \tag{3.22}$$

Substituting the above relations into Equations (3.4) and (3.5), respectively, gives

$$\mathbf{A}_1 = \begin{pmatrix} \Upsilon_1^{-'}\left(\sigma_1^{-}\right) & \Upsilon_2^{-'}\left(\sigma_1^{-}\right) & \ldots & \Upsilon_{N-1}^{-'}\left(\sigma_1^{-}\right) & \Upsilon_N^{-'}\left(\sigma_1^{-}\right)\\ \phi_1^{-}\left(\omega,\sigma_2^{-}\right) & \phi_2^{-}\left(\omega,\sigma_2^{-}\right) & & \phi_{N-1}^{-}\left(\omega,\sigma_2^{-}\right) & \phi_N^{-}\left(\omega,\sigma_2^{-}\right)\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ \phi_1^{-}\left(\omega,\sigma_{N-1}^{-}\right) & \phi_2^{-}\left(\omega,\sigma_{N-1}^{-}\right) & \ldots & \phi_{N-1}^{-}\left(\omega,\sigma_{N-1}^{-}\right) & \phi_N^{-}\left(\omega,\sigma_{N-1}^{-}\right)\\ -\Upsilon_1^{-'}\left(\alpha\right) & -\Upsilon_2^{-'}\left(\alpha\right) & \ldots & -\Upsilon_{N-1}^{-'}\left(\alpha\right) & -a_N^{-'}\left(\alpha\right) \end{pmatrix},$$

$$\mathbf{A}_2 = \begin{pmatrix} 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ \Upsilon_1^{+'}(\alpha) & \Upsilon_2^{+'}(\alpha) & \dots & \Upsilon_{N-1}^{+'}(\alpha) & \Upsilon_N^{+'}(\alpha) \end{pmatrix},$$

$$\mathbf{A}_3 = \begin{pmatrix} -c_1^{-'}(\alpha) & -c_2^{-'}(\alpha) & \dots & -c_{N-1}^{-'}(\alpha) & -c_N^{-'}(\alpha) \\ 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \end{pmatrix},$$

$$\mathbf{A}_4 = \begin{pmatrix} c_1^{+'}(\alpha) & c_2^{+'}(\alpha) & \dots & c_{N-1}^{+'}(\alpha) & c_N^{+'}(\alpha) \\ \phi_1^+(\omega,\sigma_2^+) & \phi_2^+(\omega,\sigma_2^+) & \dots & \phi_{N-1}^+(\omega,\sigma_2^+) & \phi_N^+(\omega,\sigma_2^+) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \phi_1^+(\omega,\sigma_{N-1}^+) & \phi_2^+(\omega,\sigma_{N-1}^+) & \dots & \phi_{N-1}^+(\omega,\sigma_{N-1}^+) & \phi_N^+(\omega,\sigma_{N-1}^+) \\ \Upsilon_1^{+'}(\sigma_N^+) & \Upsilon_2^{+'}(\sigma_N^+) & \dots & \Upsilon_{N-1}^{+'}(\sigma_N^+) & \Upsilon_N^{+'}(\sigma_N^+) \end{pmatrix},$$

where

$$\begin{aligned} \phi_j^-(\omega,\sigma_i^-) &= -\mathcal{G}_x^-(\omega,\sigma_i^-)\Upsilon_{2,j}^-(\sigma_i^-) - \mathcal{G}^-(\omega,\sigma_i^-)\Upsilon_{3,j}^-(\sigma_i^-) + \kappa^-(\omega,\sigma_i^-)\Upsilon_{1,j}^-(\sigma_i^-), \\ \phi_j^+(\omega,\sigma_i^+) &= -\mathcal{G}_x^+(\omega,\sigma_i^+)\Upsilon_{2,j}^+(\sigma_i^+) - \mathcal{G}^-(\omega,\sigma_i^+)\Upsilon_{3,j}^+(\sigma_i^+) + \kappa^+(\omega,\sigma_i^+)\Upsilon_{1,j}^+(\sigma_i^+). \end{aligned} \tag{3.23}$$

As a result, the matrix–vector form of the final system is

$$\mathbf{AU} = \mathbf{F}, \tag{3.24}$$

where

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{pmatrix}, \tag{3.25}$$

$$\mathbf{U}(i) = u_i^-, \quad \mathbf{U}(N+i) = u_i^+, \quad 1 \le i \le N, \tag{3.26}$$

$$\mathbf{F}(1) = u_1(a), \quad \mathbf{F}(2N) = u_2(b), \quad \mathbf{F}(i) = f^-(\sigma_i^-), \quad \mathbf{F}(i+N) = f^+(\sigma_i^+), \quad 2 \le i \le N-1. \tag{3.27}$$

In this work, the following weight function is employed

$$w_i(\mathbf{x}) = w(d_i) = \begin{cases} 1 - 6d_i^2 + 8d_i^3 - 3d_i^4, & d_i \le 1, \\ 0, & d_i > 1, \end{cases} \tag{3.28}$$

where $d_i = \frac{\|\mathbf{x}-\sigma_i\|}{\delta_i}$, and δ_i is its radius of influence domain. By following idea in [45], all points are separated in two distinct sets, that is, D^+ and D^-. Furthermore, sets D_κ and D_b contain the nodes on interface κ and on boundary ∂D, respectively. Now, we take the following weight function

$$w_i^\pm(\mathbf{x}_j) = \begin{cases} w_i(\mathbf{x}_j), & \mathbf{x}_i, \mathbf{x}_j \in D^\pm, \\ 0, & \text{otherwise}, \end{cases} \tag{3.29}$$

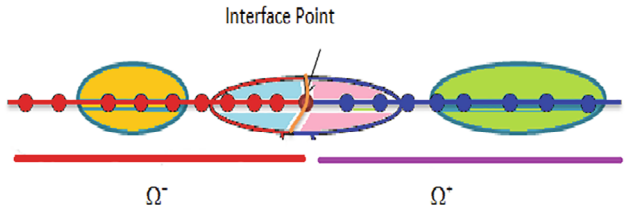

FIGURE 1 Computational domain Ω 1D case

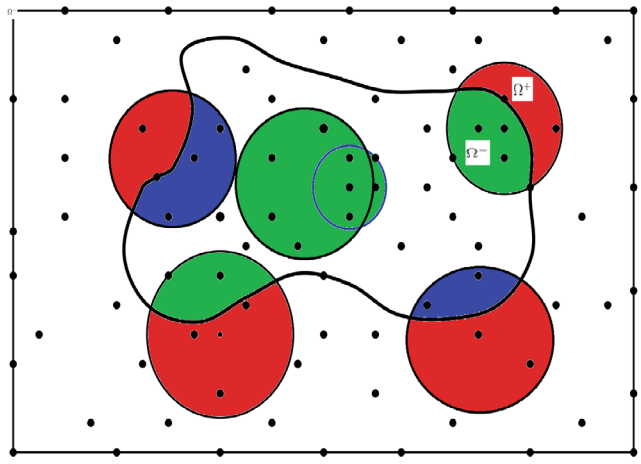

FIGURE 2 Computational domain Ω in 2D case

of the weight function [45]. Figures 1 and 2 illustrate computational domain D with interface κ in 1D and 2D cases.

4 | TWO-DIMENSIONAL ELLIPTIC STOCHASTIC INTERFACE PROBLEMS

Next, we consider the elliptic interface equation

$$\begin{cases} -\nabla \cdot (\mathcal{G}\nabla u) + k^2 u = f, & \text{in } \Omega^{\pm}, \\ u = g, & \text{on } \partial\Omega, \end{cases} \tag{4.1}$$

with function f that it may be discontinuous and also jump conditions

$$[u]_{\kappa}(\boldsymbol{\Theta}) \triangleq u^{+}(\boldsymbol{\Theta}) - u^{-}(\boldsymbol{\Theta}) = g_1(\boldsymbol{\Theta}), \quad \text{on } \Gamma, \tag{4.2}$$

$$\left[\mathcal{G}\frac{\partial u}{\partial \mathbf{n}_{\kappa}}\right]_{\kappa}(\boldsymbol{\Theta}) \triangleq \mathcal{G}^{+}\frac{\partial u^{+}}{\partial \mathbf{n}_{\kappa}}(\boldsymbol{\Theta}) - \mathcal{G}^{-}\frac{\partial u^{-}}{\partial \mathbf{n}_{\kappa}}(\boldsymbol{\Theta}) = g_2(\boldsymbol{\Theta}), \quad \text{on } \Gamma, \tag{4.3}$$

on the interface. The coefficients $\mathcal{G}$ and k can be considered as

$$\mathcal{G}(\boldsymbol{\Theta}) = \begin{cases} \mathcal{G}^{+}(\boldsymbol{\Theta}), & \boldsymbol{\Theta} \in \Omega^{+}, \\ \mathcal{G}^{-}(\boldsymbol{\Theta}), & \boldsymbol{\Theta} \in \Omega^{-}, \end{cases} \tag{4.4}$$

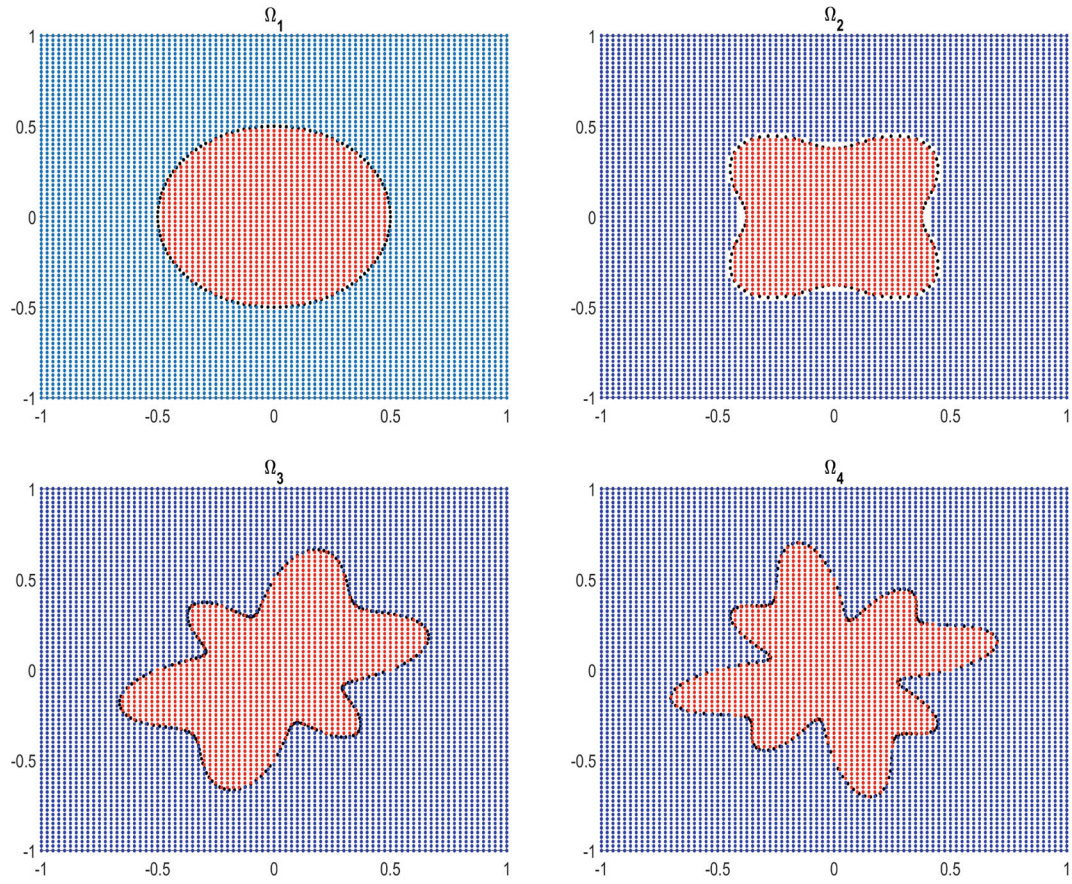

FIGURE 3 The computational geometry used for Example 2

$$k(\boldsymbol{\Theta}) = \begin{cases} k^{+}(\boldsymbol{\Theta}), & \boldsymbol{\Theta} \in \Omega^{+}, \\ k^{-}(\boldsymbol{\Theta}), & \boldsymbol{\Theta} \in \Omega^{-}. \end{cases} \tag{4.5}$$

Now, we have

$$\begin{cases} -\nabla \mathcal{G}^{-} \cdot \nabla u^{-} - \mathcal{G}^{-} \nabla^2 u^{-} + k^{2-} u^{-} = f^{-}, & \text{in } \Omega^{-}, \\ u^{-} = g^{-}, & \text{on } \partial\Omega^{-}, \end{cases} \tag{4.6}$$

and

$$\begin{cases} -\nabla \mathcal{G}^{+} \cdot \nabla u^{+} - \mathcal{G}^{+} \nabla^2 u^{+} + k^{2+} u^{+} = f^{+}, & \text{in } \Omega^{+}, \\ u^{+} = g^{+}, & \text{on } \partial\Omega^{+}. \end{cases} \tag{4.7}$$

We use two separate scattered points for Ω^{-} and Ω^{+}, namely

$$\mathbf{X}^{-} = \left\{\eta_i^{-}\right\}_{i=1}^{N}, \quad \mathbf{X}^{+} = \left\{\eta_i^{+}\right\}_{i=1}^{N}. \tag{4.8}$$

Next, we define the notations

$$\Theta_{1,i}^{-}(u^{-}) := u^{-}\left(\eta_i^{-}\right) \approx \widetilde{u}^{-}\left(\eta_i^{-}\right) = \sum_{j=1}^{N} \Upsilon_{1,j}^{-}\left(\eta_i^{-}\right) u^{-}\left(\eta_i^{-}\right), \tag{4.9}$$

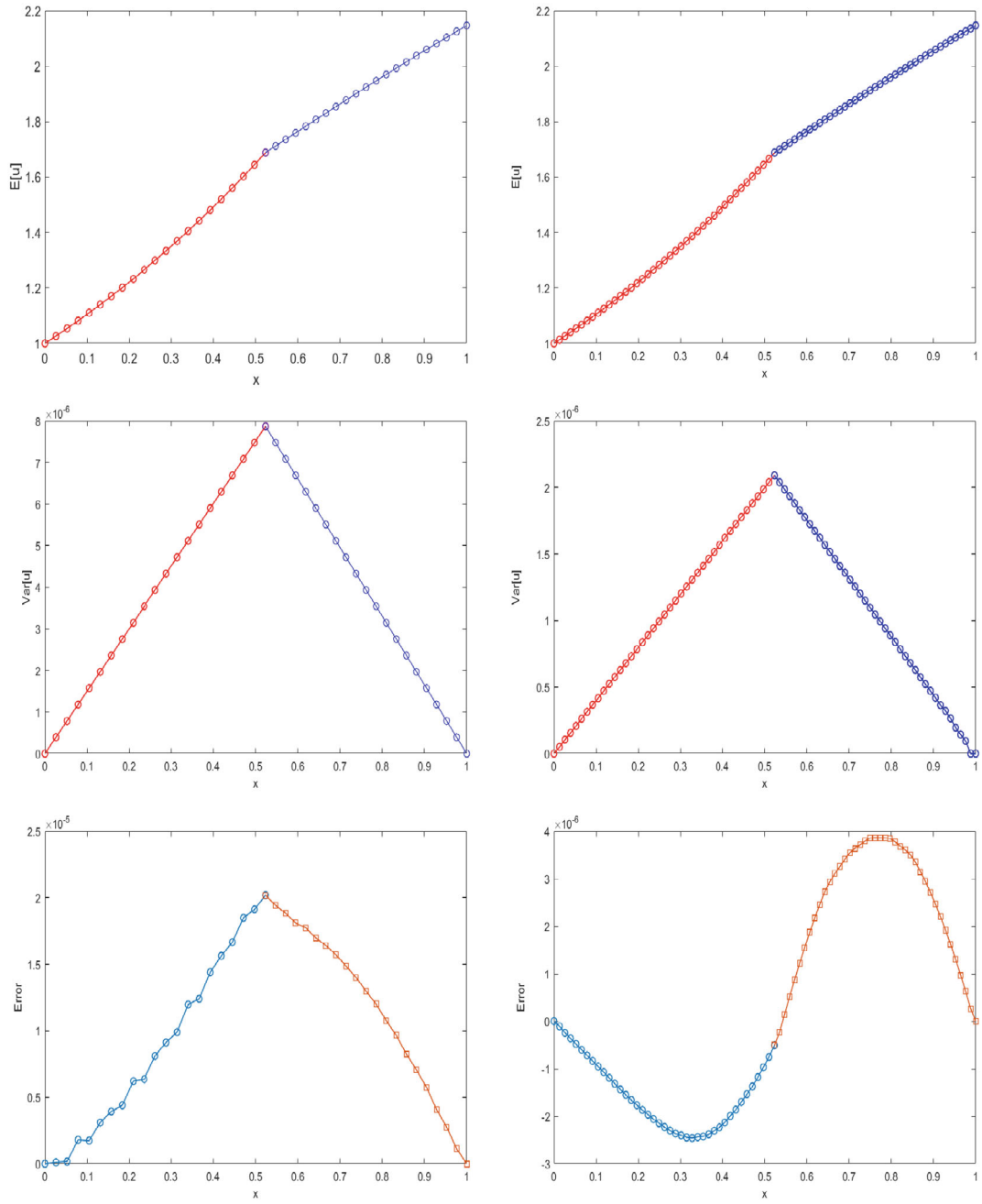

FIGURE 4 The expected value (first row), the variance (second row), and the error (third row) of the solution of Example 1 with $h = 1/40$ (first column) and $h = 1/100$ (second column)

$$\Theta_{2,i}^{-}(u^{-}) := u_{\eta}^{-}\left(\eta_i^{-}\right) \approx \widetilde{\Theta}_{2,i}^{-}(u^{-}) = \sum_{j=1}^{N} \Upsilon_{2,j}^{-}\left(\eta_i^{-}\right) u^{-}\left(\eta_i^{-}\right), \tag{4.10}$$

$$\Theta_{3,i}^{-}(u^{-}) := u_{xx}^{-}\left(\eta_i^{-}\right) \approx \widetilde{\Theta}_{3,i}^{-}(u^{-}) = \sum_{j=1}^{N} \Upsilon_{3,j}^{-}\left(\eta_i^{-}\right) u^{-}\left(\eta_i^{-}\right), \tag{4.11}$$

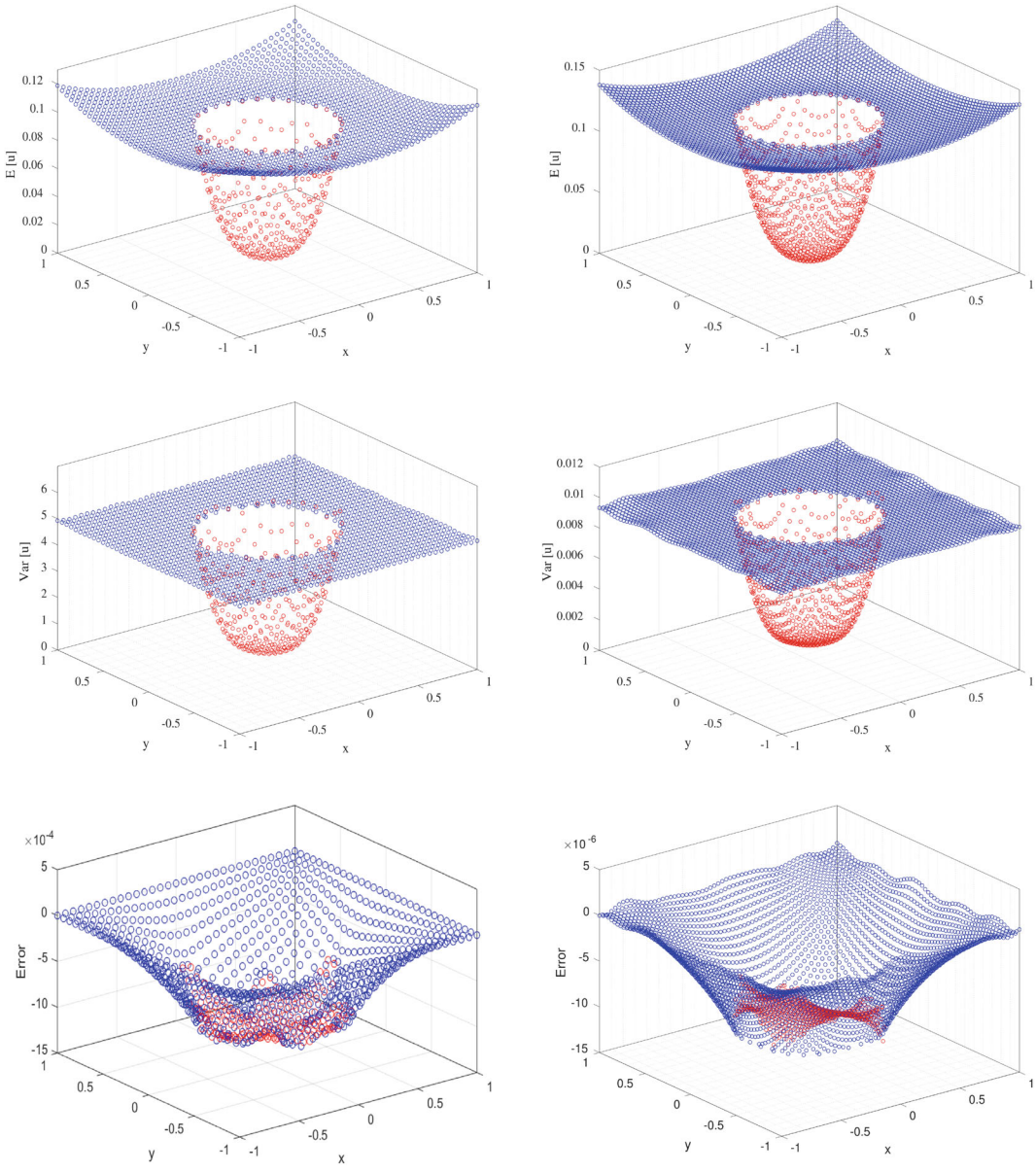

FIGURE 5 The expected value (first row), the variance (second row), and the error (third row) of the solution of Example 2 (geometry Ω_1) with $h = 1/40$ (first column) and $h = 1/100$ (second column)

$$\Theta_{4,i}^{-}\left(u^{-}\right) \coloneqq u_{yy}^{-}\left(\eta_i^{-}\right) \approx \widetilde{\Theta}_{4,i}^{-}\left(u^{-}\right) = \sum_{j=1}^{N} \Upsilon_{4,j}^{-}\left(\eta_i^{-}\right) u^{-}\left(\eta_i^{-}\right), \tag{4.12}$$

$$\Theta_{1,i}^{+}\left(u^{+}\right) \coloneqq u^{+}\left(\sigma_i^{+}\right) \approx \widetilde{u}^{+}\left(\eta_i^{+}\right) = \sum_{j=1}^{N} \Upsilon_{1,j}^{+}\left(\sigma_i^{+}\right) u^{+}\left(\eta_i^{+}\right), \tag{4.13}$$

$$\Theta_{2,i}^{-}\left(u^{+}\right) \coloneqq u_{x}^{+}\left(\eta_i^{+}\right) \approx \widetilde{\Theta}_{2,i}^{+}\left(u^{+}\right) = \sum_{j=1}^{N} \Upsilon_{2,j}^{+}\left(\eta_i^{+}\right) u^{+}\left(\eta_i^{+}\right), \tag{4.14}$$

$$\Theta_{3,i}^{-}\left(u^{+}\right) \coloneqq u_{xx}^{-}\left(\eta_i^{+}\right) \approx \widetilde{\Theta}_{3,i}^{+}\left(u^{+}\right) = \sum_{j=1}^{N} \Upsilon_{3,j}^{+}\left(\eta_i^{+}\right) u^{+}\left(\eta_i^{+}\right), \tag{4.15}$$

$$\Theta_{4,i}^{-}\left(u^{+}\right) \coloneqq u_{yy}^{-}\left(\eta_i^{+}\right) \approx \widetilde{\Theta}_{4,i}^{+}\left(u^{+}\right) = \sum_{j=1}^{N} \Upsilon_{4,j}^{+}\left(\eta_i^{+}\right) u^{+}\left(\eta_i^{+}\right). \tag{4.16}$$

Based on the GMLS technique, we have

$$\begin{aligned} \boldsymbol{\Upsilon}_{k,:}^{-,T}\left(\eta_i\right) &= \Theta_{k,i}^{-}\left(\mathbf{p}^{T}\right)\mathbf{M}^{-1}\mathbf{B}, \quad k = 1, 2, 3, 4, \\ \boldsymbol{\Upsilon}_{k,:}^{+,T}\left(\eta_i\right) &= \Theta_{k,i}^{+}\left(\mathbf{p}^{T}\right)\mathbf{M}^{-1}\mathbf{B}, \quad k = 1, 2, 3, 4, \end{aligned} \tag{4.17}$$

and according to (2.12), one can see

$$\begin{aligned} \Theta_{1,i}^{\pm}\left(\mathbf{p}^{T}\right) &= \left(p_1\left(\eta_i\right) p_2\left(\eta_i\right) \ \ldots \ p_Q\left(\eta_i\right)\right), \\ \Theta_{2,i}^{\pm}\left(\mathbf{p}^{T}\right) &= \left(\left.\frac{\partial p_1}{\partial x}\right|_{x=\sigma_i} \left.\frac{\partial p_2}{\partial x}\right|_{x=\sigma_i} \ \ldots \ \left.\frac{\partial p_Q}{\partial x}\right|_{x=\sigma_i}\right), \\ \Theta_{3,i}^{\pm}\left(\mathbf{p}^{T}\right) &= \left(\left.\frac{\partial^2 p_1}{\partial x^2}\right|_{x=\sigma_i} \left.\frac{\partial^2 p_2}{\partial x^2}\right|_{x=\sigma_i} \ \ldots \ \left.\frac{\partial^2 p_Q}{\partial x^2}\right|_{x=\sigma_i}\right), \\ \Theta_{4,i}^{\pm}\left(\mathbf{p}^{T}\right) &= \left(\left.\frac{\partial^2 p_1}{\partial y^2}\right|_{x=\sigma_i} \left.\frac{\partial^2 p_2}{\partial y^2}\right|_{x=\sigma_i} \ \ldots \ \left.\frac{\partial^2 p_Q}{\partial y^2}\right|_{x=\sigma_i}\right). \end{aligned}$$

Similar to the one-dimensional case, we obtain the system of equations

$$\mathbf{AU} = \mathbf{F}, \tag{4.18}$$

where

$$\mathbf{A} = \left(\begin{array}{c|c} \mathbf{A}_1 & \mathbf{A}2 \\ \hline \mathbf{A}_3 & \mathbf{A} \end{array}\right), \quad \mathbf{U} = \begin{pmatrix} \mathbf{u}_{if}^{-} \\ \mathbf{u}_{in}^{-} \\ \mathbf{u}_{if}^{+} \\ \mathbf{u}_{in}^{+} \\ \mathbf{u}_{b}^{+} \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \mathbf{g}_1\left(\boldsymbol{\Theta}^{-}\right) \\ \mathbf{f}^{-}\left(\boldsymbol{\Theta}^{-}\right) \\ \mathbf{g}_2\left(\boldsymbol{\Theta}^{-}\right) \\ \mathbf{f}^{+}\left(\boldsymbol{\Theta}^{+}\right) \\ \mathbf{g}\left(\boldsymbol{\Theta}^{+}\right) \end{pmatrix}, \tag{4.19}$$

and also

$$\mathbf{A}_1 = \begin{pmatrix} -\mathbf{I}_{(M_1-M_{if})\times(M_1-M_{if})} & \mathbf{0}_{(M_1-M_{if})\times M_{if}} \\ \mathbf{0}_{M_{if}\times(M_1-M_{if})} & \mathbf{D}_{M_{if}\times M_{if}}^{-} \end{pmatrix},$$

$$\mathbf{A}_2 = \begin{pmatrix} \mathbf{0}_{(M_1-M_{if})\times M_{if}} & \mathbf{0}_{(M_1-M_{if})\times(M_2-M_{if}-M_b)} & \mathbf{0}_{(M_1-M_{if})\times M_b} \\ \mathbf{0}_{M_{if}\times M_{if}} & \mathbf{0}_{M_{if}\times(M_2-M_{if}-M_b)} & \mathbf{I}_{M_{if}\times M_b} \end{pmatrix},$$

$$\mathbf{A}_3 = \begin{pmatrix} \mathbf{0}_{M_{if}\times(M_1-M_{if})} & \mathbf{0}_{M_{if}\times M_{if}} \\ \mathbf{0}_{(M_2-M_{if}-M_b)\times(M_1-M_{if})} & \mathbf{0}_{(M_2-M_{if}-M_b)\times M_{if}} \\ \mathbf{0}_{M_{if}\times(M_1-M_{if})} & -\mathbf{D}_{\mathbf{x}_{M_{if}\times M_{if}}}^{-} \end{pmatrix},$$

$$\mathbf{A}_4 = \begin{pmatrix} \mathbf{0}_{M_{if}\times M_{if}} & \mathbf{0}_{M_{if}\times(M_2-M_{if}-M_b)} & \mathbf{0}_{M_{if}\times M_b} \\ \mathbf{0}_{(M_2-M_{if}-M_b)\times M_{if}} & \mathbf{D}_{(M_2-M_{if}-M_b)\times(M_2-M_{if}-M_b)}^{+} & \mathbf{0}_{(M_2-M_{if}-M_b)\times M_b} \\ \mathbf{0}_{M_b\times M_{if}} & \mathbf{0}_{M_b\times(M_2-M_{if}-M_b)} & \mathbf{D}_{\mathbf{x}_{M_b\times M_b}}^{+} \end{pmatrix}.$$

TABLE 1 A comparison between the CPU time (s) used in the method of [17] and the present paper

	2D case		1D case	
h	Method of [17]	New Method	Method of [17]	New method
1/10	26	3	1	0.05
1/20	68	7	2	0.57
1/80	133	14	12	2
1/160	263	26	31	5
1/320	388	41	89	11
1/640	1031	81	179	24

5 | NUMERICAL SIMULATIONS

We demonstrate the effectiveness of this method using some examples. We consider the error norms

$$L_\infty = \max_{1\le j\le M-1} \left|\mathbb{E}\left[u_{\text{ex}}\right]\left(\mathbf{x}_j\right) - \mathbb{E}\left[u\right]\left(\mathbf{x}_j\right)\right|, \quad L_2 = \sqrt{\sum_{j=1}^{M-1}\left|\mathbb{E}\left[u_{\text{ex}}\right]\left(\mathbf{x}_j\right) - \mathbb{E}\left[u\right]\left(\mathbf{x}_j\right)\right|^2},$$

where u_{ex} indicates the exact solution. We select four regions, that is, Ω_1, Ω_2, Ω_3, and Ω_4 with complicated boundaries that are depicted in Figure 3. In all examples, the function f is assumed to be deterministic. The diffusion coefficient estimated by using the KL expansion (1.7) with the exponential covariance ($\exp(-\|x-y\|^2)$) with $N = 4$. Also, we consider different irregular computational regions to show the efficiency of the proposed method. In the current paper, let $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{\mathcal{N}}$ be a set of scattered data in $\Omega \subset \mathbb{R}^n$ with fill distance

$$h_{X,\Omega} = \sup_{x\in\Omega} \min_{1\le j\le\mathcal{N}} \|\varsigma - \varsigma_j\|_2. \tag{5.1}$$

The radius of the weight function is $r_i = 2.7h_{X,\,\Omega}$. Furthermore, $\mathcal{N}$ is the number of distributed nodes into the computational domain.

5.1 | Example 1

We consider 1D elliptic interface problem

$$\begin{cases} -\mathcal{G}\frac{\partial^2 u}{\partial x^2} = f, & x \in (0,1), \\ u(0) = 1, \quad u(1) = e^\alpha + \sin(1-\alpha), \end{cases} \tag{5.2}$$

where

$$[u]_\alpha = 0, \quad [\mathcal{G}u_x]_\alpha = 20000 - \exp(\alpha), \tag{5.3}$$

$$\mathcal{G}_0 = \begin{cases} 1, & x \in [0,\alpha], \\ 20000, & x \in [\alpha,1], \end{cases} \tag{5.4}$$

and

$$f(x) = \begin{cases} -e^x, & 0 \le x \le \alpha, \\ 2\times 10^4 \sin(x-\alpha), & \alpha \le x \le 1. \end{cases} \tag{5.5}$$

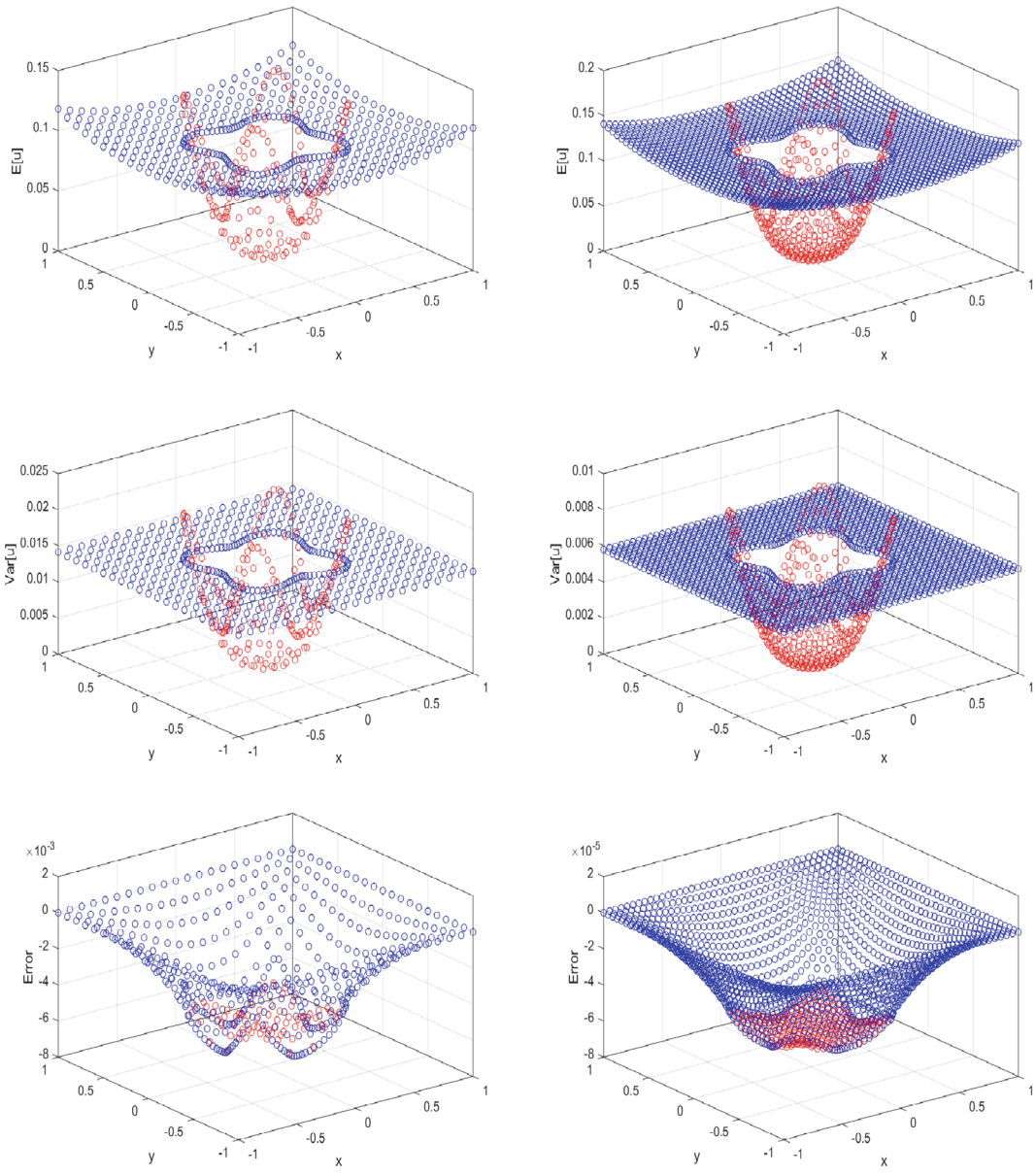

FIGURE 6 The expected value (first row), the variance (second row), and the error (third row) of the solution of the two-dimensional stochastic interface problem (geometry Ω_2) with $h = 1/40$ (first column) and $h = 1/60$ (second column)

The exact solution of deterministic case is [24]

$$u(x) = \begin{cases} -e^x, & 0 \le x \le \alpha, \\ e^\alpha + \sin(x - \alpha), & \alpha \le x \le 1. \end{cases} \tag{5.6}$$

Figure 4 shows the expected value, variance and the error of the solution of Example 1 with $h = 1/40$ and $h = 1/100$, where $\delta := 2.5h$ is used. The results show that a noticeable error reduction can be obtained with smaller mesh sizes. This fact will be shown for different mesh sizes in Figure 9.

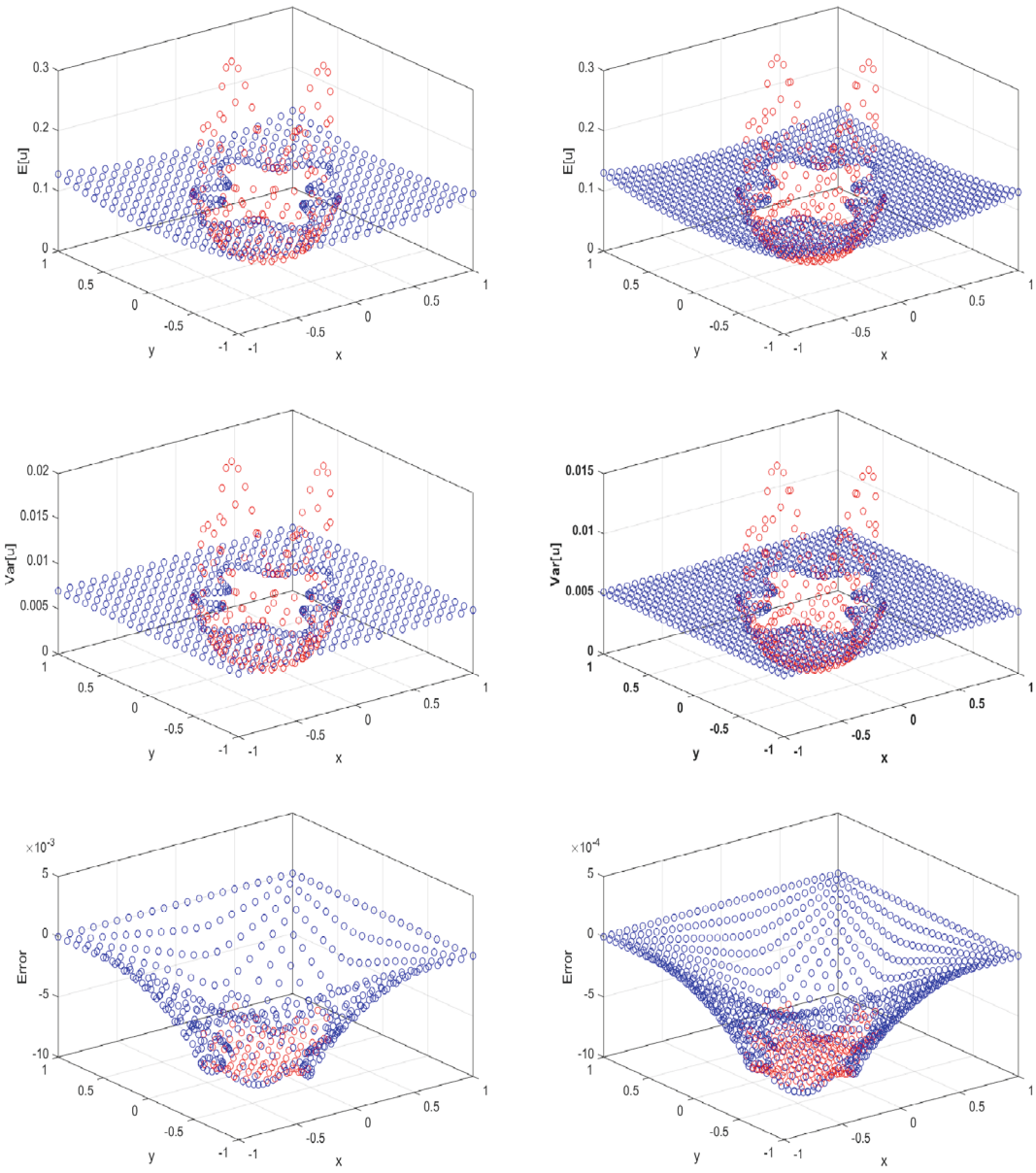

FIGURE 7 The expected value (first row), the variance (second row), and the error (third row) of the solution of the two-dimensional stochastic interface problem (geometry Ω_3) with $h = 1/40$ (first column) and $h = 1/60$ (second column)

5.2 | Example 2

We define the geometry $D := [-1, 1] \times [-1, 1]$, where a circle at the center of geometry with radius $r_0 = 0.5$ separates Ω^+ and Ω^- [53]. The exact solution is [53]

$$u = \begin{cases} \dfrac{r^3}{\mathcal{G}^-}, & \text{in } \Omega^+, \\ \dfrac{r^3}{\mathcal{G}^+} + \left(\dfrac{1}{\mathcal{G}^-} - \dfrac{1}{\mathcal{G}^+}\right) r_0^3, & \text{in } \Omega^-, \end{cases} \tag{5.7}$$

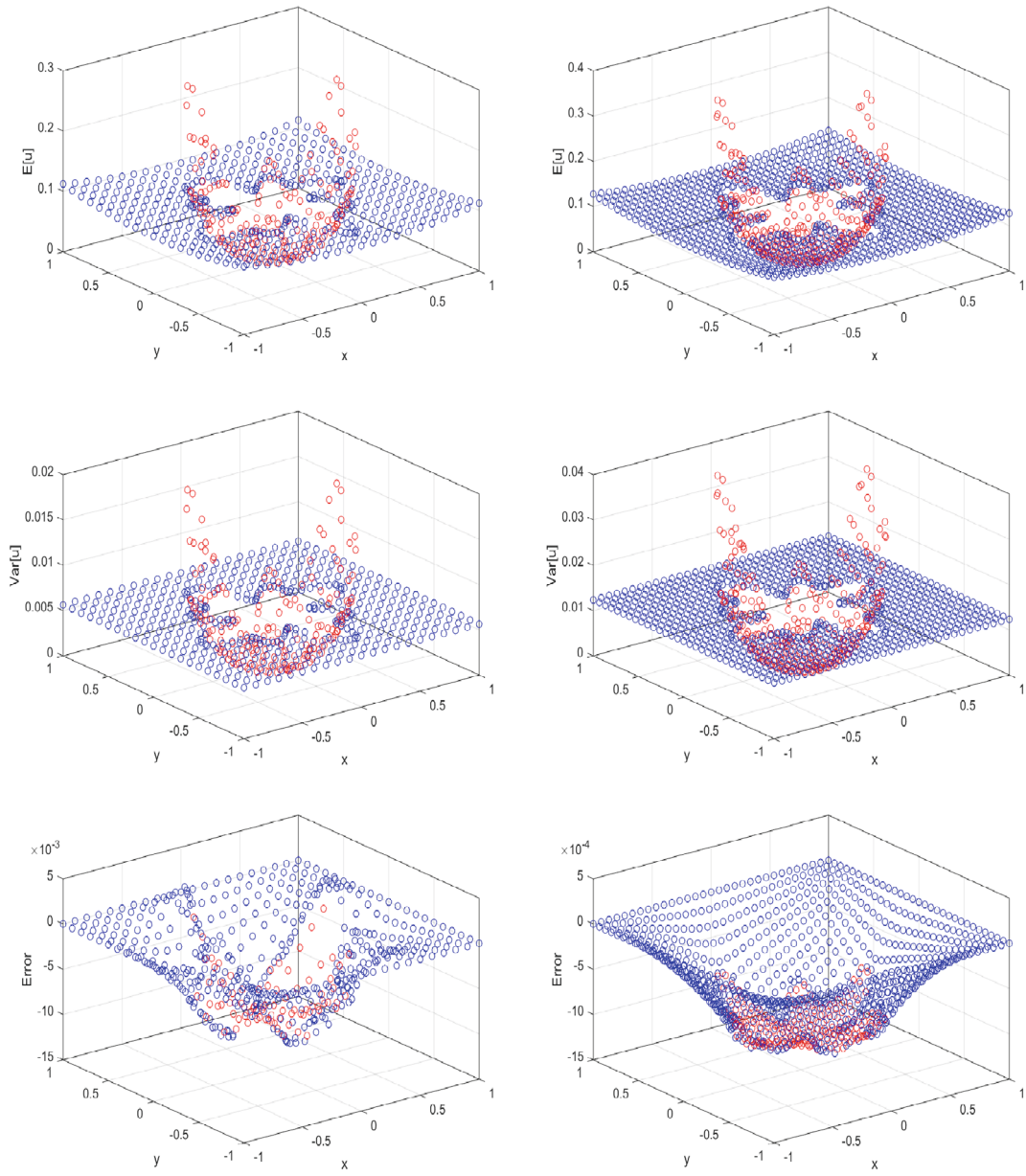

FIGURE 8 The expected value (first row), the variance (second row), and the error (third row) of the solution of the two-dimensional stochastic interface problem (geometry Ω_4) with $h = 1/40$ (first column) and $h = 1/60$ (second column)

where $r = \sqrt{x^2 + y^2}$ and we have $\mathcal{G}_0^+ := 100$ and $\mathcal{G}_0^- := 1$. Figure 5 shows the expected value, variance, and the error (with respect to the exact solution) for $h = 1/40$ and $h = 1/60$.

The results of this example show the accuracy and the efficiency of the meshless method. For more investigations, we consider the given coefficients and the geometry $D = [-1, 1] \times [-1, 1]$ of Example 2 with different interfaces. Here, since the exact solution cannot be calculated analytically, the numerical solution with $h = 1/500$ is considered. Again the expected value, the variance (for $N = 500$ samples)

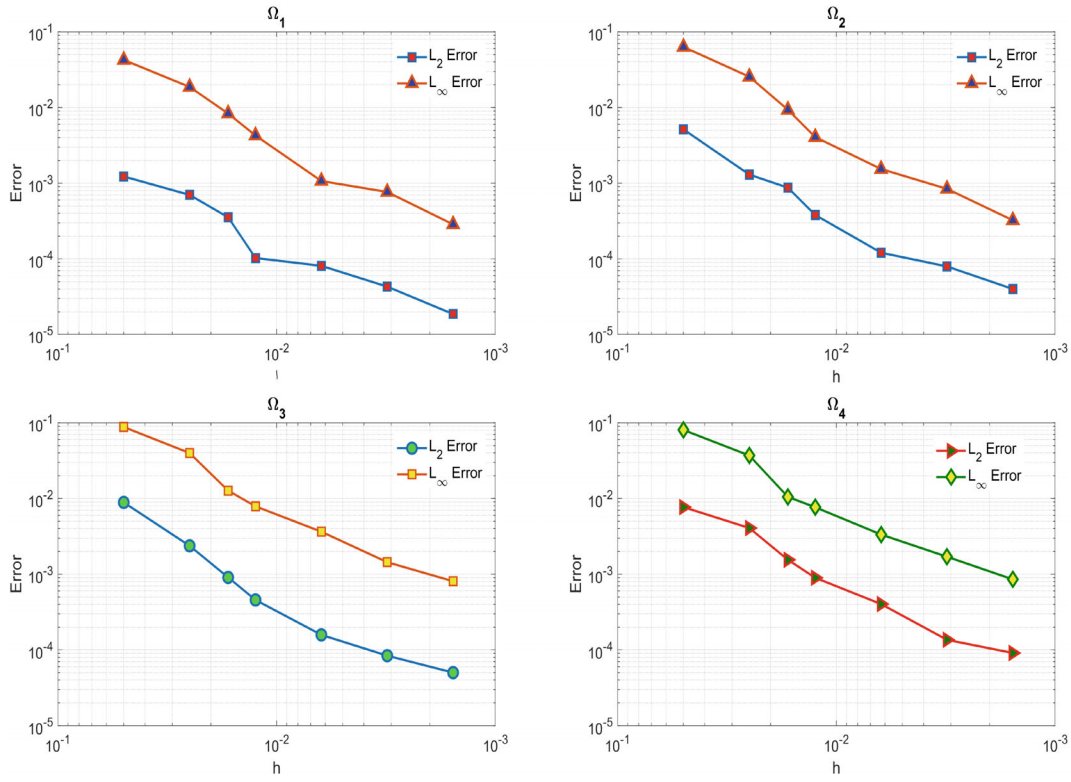

FIGURE 9 The computational error for four defined two-dimensional interface geometries. Here both the L_2 and the L_∞ error have been calculated

and the computational error for two different mesh sizes, that is, $h = 1/40$ and $h = 1/60$, are shown. The results are shown in Figures 6, 7, and 8 for Ω_2, Ω_3, and Ω_4, respectively.

For the four interfaces, the error estimations with respect to different mesh sizes are given in Figure 9. For all stochastic interface problems, the convergence rate is faster than $\mathcal{O}\left(h^1\right)$.

As we already mentioned, compared with the meshless method developed in [17], the present technique needs noticeably less CPU time. This fact has been validated in Examples 1 and 2. Table 1 shows the CPU time used for solving the stochastic problem given in both examples for different mesh sizes. The results demonstrate that a significant complexity reduction (more than 10 times in many cases) has been achieved which demonstrates the efficiency of the new meshless method.

6 | CONCLUSIONS

In the present work, a truly local meshless method has been developed to solve a new class of PDEs, that is, stochastic elliptic interface problems. The proposed method is based on the DMLPG2 method. In this technique, Dirichlet and Neumann boundary conditions are directly applied. For the first time, we used a local meshless method for solving the stochastic elliptic interface problems. Also, in order to represent the random process (i.e., stochastic coefficients), the Karhunen–Loéve expansion [19] is used. The numerical procedure presented is based on the GMLS approximation, thus the implementation of this technique is simple and efficient. It is worth mentioning, for solving the stochastic interface

problems, the present technique performs 10 times faster compared with the other meshless methods, for example, [17].

ACKNOWLEDGMENTS

The authors are very grateful to the reviewers for carefully reading this paper and for their comments and suggestions which have improved the paper. A. Khodadadian and C. Heitzinger acknowledge financial support by FWF (Austrian Science Fund) START Project no. Y660 PDE Models for Nanotechnology.

ORCID

Mostafa Abbaszadeh https://orcid.org/0000-0001-6954-3896
Mehdi Dehghan https://orcid.org/0000-0002-2573-9755
Amirreza Khodadadian https://orcid.org/0000-0003-2374-0557

REFERENCES

[1] M. Abbaszadeh and M. Dehghan, *The reproducing kernel particle Petrov–Galerkin method for solving two-dimensional nonstationary incompressible Boussinesq equations*, Eng. Anal. Bound. Elem. 106 (2019), 300–308.
[2] M. Abbaszadeh and M. Dehghan, *Direct meshless local Petrov–Galerkin (DMLPG) method for time-fractional fourth-order reaction–diffusion problem on complex domains*, Comput. Math. Appl. 79 (2020), 876–888.
[3] M. Abbaszadeh and M. Dehghan, *Investigation of the Oldroyd model as a generalized incompressible Navier–Stokes equation via the interpolating stabilized element free Galerkin technique*, Appl. Numer. Math. 150 (2020), 274–294.
[4] M. Abbaszadeh and M. Dehghan, *An upwind local radial basis functions-differential quadrature (RBFs–-DQ) technique to simulate some models arising in water sciences*, Ocean Eng. 197 (2020), 106844.
[5] M. Abbaszadeh and M. Dehghan, *A meshless numerical investigation based on the RBF-QR approach for elasticity problems*, AUT J. Math. Comput. (AJMC) 1 (2020), 1–15.
[6] M. Abbaszadeh, A. Khodadadian, M. Parvizi, M. Dehghan, and C. Heitzinger, *A direct meshless local collocation method for solving stochastic Cahn–Hilliard–Cook and stochastic Swift–Hohenberg equations*, Eng. Anal. Bound. Elem. 98 (2019), 253–264.
[7] M. Ahmad and S.-u. Islam, *Meshless analysis of parabolic interface problems*, Eng. Anal. Bound. Elem. 94 (2018), 134–152.
[8] S. N. Atluri, *The meshless method (MLPG) for domain and BIE discretizations*, Tech Science Press, USA, 2004, 680.
[9] S. N. Atluri and S. Shen, *The meshless local Petrov–Galerkin (MLPG) method: A simple & less-costly alternative to the finite element and boundary element methods*, Comput. Model. Eng. Sci. 3 (2002), 11–51.
[10] S. N. Atluri and T. Zhu, *A new meshless local Petrov–Galerkin (MLPG) approach in computational mechanics*, Comput. Mech. 22 (1998), 117–127.
[11] F. Brezzi, J. Douglas, and L. D. Marini, *Two families of mixed finite elements for second order elliptic problems*, Numer. Math. 47 (1985), 217–235.
[12] L. Chen and X. Li, *A complex variable boundary element-free method for the Helmholtz equation using regularized combined field integral equations*, Appl. Math. Lett. 101 (2020), 106067.
[13] H. Cheng, M. Peng, and Y. Cheng, *A hybrid improved complex variable element-free Galerkin method for three-dimensional advection–diffusion problems*, Eng. Anal. Bound. Elem. 97 (2018), 39–54.
[14] W. Dai and R. Nassar, *A domain decomposition method for solving thin film elliptic interface problems with variable coefficients*, Int. J. Numer. Methods Eng. 46 (1999), 747–756.
[15] M. Dehghan and M. Abbaszadeh, *The meshless local collocation method for solving multi-dimensional Cahn–Hilliard, Swift–Hohenberg and phase field crystal equations*, Eng. Anal. Bound. Elem. 78 (2017), 49–64.
[16] M. Dehghan and M. Abbaszadeh, *Variational multiscale element-free Galerkin method combined with the moving Kriging interpolation for solving some partial differential equations with discontinuous solutions*, Comput. Appl. Math. 37 (2018), 3869–3905.
[17] M. Dehghan and M. Abbaszadeh, *Interpolating stabilized moving least squares (MLS) approximation for 2D elliptic interface problems*, Comput. Methods Appl. Mech. Eng. 328 (2018), 775–803.

[18] M. Dehghan and M. Abbaszadeh, *Error analysis and numerical simulation of magnetohydrodynamics (MHD) equation based on the interpolating element free Galerkin (IEFG) method*, Appl. Numer. Math. 137 (2019), 252–273.
[19] R. G. Ghanem and P. D. Spanos, *Stochastic finite elements: A spectral approach*, Springer, New York, 1991.
[20] G. R. Hadley, *High-accuracy finite-difference equations for dielectric waveguide analysis I: Uniform regions and dielectric interfaces*, J. Lightwave Technol. 20 (2002), 1210–1218.
[21] I. Harari and J. Dolbow, *Analysis of an efficient finite element method for embedded interface problems*, Comput. Mech. 46 (2010), 205–211.
[22] H. Harbrecht and J. Li, *First order second moment analysis for stochastic interface problems based on low-rank approximation*, ESAIM: Math. Model. Numer. Anal. 47(5) (2013), 1533–1552.
[23] X. He, T. Lin, and Y. Lin, *Interior penalty bilinear IFE discontinuous Galerkin methods for elliptic equations with discontinuous coefficient*, J. Syst. Sci. Complex. 23 (2010), 467–483.
[24] X. He, T. Lin, and Y. Lin, *Immersed finite element methods for elliptic interface problems with non-homogeneous jump conditions*, Int. J. Numer. Anal. Model. 8 (2011), 284–301.
[25] P. Hessari, B.-C. Shin, and B. Jang, *Analysis of least squares pseudo-spectral method for the interface problem of the Navier–Stokes equations*, Comput. Math. Appl. 69 (2015), 838–851.
[26] T. Y. Hou, Z. Li, S. Osher, and H. Zhao, *A hybrid method for moving interface problems with application to the Hele–Shaw flow*, J. Comput. Phys. 134 (1997), 236–252.
[27] L. Huynh, N. Nguyen, J. Peraire, and B. Khoo, *A high-order hybridizable discontinuous Galerkin method for elliptic interface problems*, Int. J. Numer. Methods Eng. 93 (2013), 183–200.
[28] S.-u. Islam and M. Ahmad, *Meshless analysis of elliptic interface boundary value problems*, Eng. Anal. Bound. Elem. 92 (2018), 38–49.
[29] A. Khodadadian and C. Heitzinger, *Basis adaptation for the stochastic nonlinear Poisson–Boltzmann equation*, J. Comput. Electron. 15 (2016), 1393–1406.
[30] A. Khodadadian, M. Parvizi, and C. Heitzinger, *An adaptive multilevel Monte Carlo algorithm for the stochastic drift–diffusion–Poisson system*, Comput. Methods Appl. Mech. Eng. 368 (2020), 113163.
[31] A. Khodadadian, L. Taghizadeh, and C. Heitzinger, *Three-dimensional optimal multi-level Monte–Carlo approximation of the stochastic drift–diffusion–Poisson system in nanoscale devices*, J. Comput. Electron. 17 (2018), 76–89.
[32] A. Khodadadian, L. Taghizadeh, and C. Heitzinger, *Optimal multilevel randomized quasi-Monte-Carlo method for the stochastic drift–diffusion-Poisson system*, Comput. Methods Appl. Mech. Eng. 329 (2018), 480–497.
[33] A. T. Layton, *Using integral equations and the immersed interface method to solve immersed boundary problems with stiff forces*, Comput. Fluids 38 (2009), 266–272.
[34] X. Li and H. Dong, *Error analysis of the meshless finite point method*, Appl. Math. Comput. 382 (2020), 125326.
[35] X. Li and S. Li, *A meshless projection iterative method for nonlinear Signorini problems using the moving Kriging interpolation*, Eng. Anal. Bound. Elem. 98 (2019), 243–252.
[36] X. Li and S. Li, *A complex variable boundary point interpolation method for the nonlinear Signorini problem*, Comput. Math. Appl. 79 (2020), 3297–3309.
[37] X. Li and Q. Wang, *Analysis of the inherent instability of the interpolating moving least squares method when using improper polynomial bases*, Eng. Anal. Bound. Elem. 73 (2016), 21–34.
[38] D. Mirzaei and R. Schaback, *Direct meshless local Petrov–Galerkin (DMLPG) method: A generalized MLS approximation*, Appl. Numer. Math. 68 (2013), 73–82.
[39] D. Mirzaei, R. Schaback, and M. Dehghan, *On generalized moving least squares and diffuse derivatives*, IMA J. Numer. Anal. 32 (2012), 983–1000.
[40] A. Mohebbi, M. Abbaszadeh, and M. Dehghan, *The use of a meshless technique based on collocation and radial basis functions for solving the time fractional nonlinear Schrödinger equation arising in quantum mechanics*, Eng. Anal. Bound. Elem. 37 (2013), 475–485.
[41] M. Oevermann and R. Klein, *A Cartesian grid finite volume method for elliptic equations with variable coefficients and embedded interfaces*, J. Comput. Phys. 219 (2006), 749–769.
[42] W. Qu, C. M. Fan, and X. Li, *Analysis of an augmented moving least squares approximation and the associated localized method of fundamental solutions*, Comput. Math. Appl. 80 (2020), 13–30.
[43] S. J. Sherwin, R. Kirby, J. Peiró, R. Taylor, and O. Zienkiewicz, *On 2D elliptic discontinuous Galerkin methods*, Int. J. Numer. Methods Eng. 65 (2006), 752–784.
[44] S. Soghrati, C. A. Duarte, and P. H. Geubelle, *An adaptive interface-enriched generalized fem for the treatment of problems with curved interfaces*, Int. J. Numer. Methods Eng. 102 (2015), 1352–1370.
[45] A. Taleei and M. Dehghan, *An efficient meshfree point collocation moving least squares method to solve the interface problems with nonhomogeneous jump conditions*, Numer. Methods Partial Differ. Eq. 31 (2015), 1031–1053.
[46] D. M. Tartakovsky and A. Guadagnini, *Effective properties of random composites*, SIAM J. Sci. Comput. 26 (2004), 625–635.

[47] Q. Wang, W. Zhou, Y. Cheng, G. Ma, X. Chang, Y. Miao, and E. Chen, *Regularized moving least-square method and regularized improved interpolating moving least-square method with nonsingular moment matrices*, Appl. Math. Comput. 325 (2018), 120–145.
[48] Q. Wang, W. Zhou, Y. T. Feng, G. Ma, Y. Cheng, and X. Chang, *An adaptive orthogonal improved interpolating moving least-square method and a new boundary element-free method*, Appl. Math. Comput. 353 (2019), 347–370.
[49] K. Xia, M. Zhan, and G.-W. Wei, *MIB Galerkin method for elliptic interface problems*, J. Comput. Appl. Math. 272 (2014), 195–220.
[50] S. Yu, M. Peng, H. Cheng, and Y. Cheng, *The improved element-free Galerkin method for three-dimensional elastoplasticity problems*, Eng. Anal. Bound. Elem. 104 (2019), 215–224.
[51] T. Zhang and X. Li, *Variational multiscale interpolating element-free Galerkin method for the nonlinear Darcy–Forchheimer model*, Comput. Math. Appl. 79 (2020), 363–377.
[52] T. Zhang and X. Li, *Analysis of the element-free Galerkin method with penalty for general second-order elliptic problems*, Appl. Math. Comput. 380 (2020), 125306.
[53] Q. Zhang, Z. Li, and Z. Zhang, *A sparse grid stochastic collocation method for elliptic interface problems with random input*, J. Sci. Comput. 67(1) (2016), 262–280.
[54] S. Zhao, *High order matched interface and boundary methods for the Helmholtz equation in media with arbitrarily curved interfaces*, J. Comput. Phys. 229 (2010), 3155–3170.
[55] T. Zhou, *Stochastic Galerkin methods for elliptic interface problems with random input*, J. Comput. Appl. Math. 236 (2011), 782–792.

How to cite this article: Abbaszadeh M, Dehghan M, Khodadadian A, Heitzinger C. Application of direct meshless local Petrov–Galerkin method for numerical solution of stochastic elliptic interface problems. *Numer Methods Partial Differential Eq.* 2022;38:1271–1292. https://doi.org/10.1002/num.22742