

Article

A Hybrid Model Reduction Method for Dual-Continuum Model with Random Inputs

Lingling Ma 

School of Mathematics and Statistics, Hunan First Normal University, Changsha 410205, China;
hudalingling@126.com

Abstract

In this paper, a hybrid model reduction method for solving flows in fractured media is proposed. The approach integrates the Generalized Multiscale Finite Element Method (GMsFEM) with a novel variable-separation (VS) technique. Compared with many widely used variable-separation methods, the proposed model reduction method shares their merits but has lower computation complexity and higher efficiency. Within this framework, we can get the low-rank variable-separation expansion of dual-continuum model solutions in a systematic enrichment manner. No iteration is performed at each enrichment step. The expansion is constructed using two sets of basis functions: stochastic basis functions and deterministic physical basis functions, both derived from offline, model-oriented computations. To efficiently construct the stochastic basis functions, the original model is used to learn stochastic information. Meanwhile, the deterministic physical basis functions are trained using solutions obtained by applying an uncoupled GMsFEM to the dual-continuum system at a select number of optimal samples. Once these bases are established, the online evaluation for each new random sample becomes highly efficient, allowing for the computation of a large number of stochastic realizations at minimal cost. To demonstrate the performance of the proposed method, two numerical examples for dual-continuum models with random inputs are presented. The results confirm that the hybrid model reduction method is both efficient and achieves high approximation accuracy.

Keywords: dual-continuum model; uncoupled generalized multiscale finite element method; variable-separation method; model reduction



Academic Editors: Ali Cemal Benim,
Jeffrey S. Marshall, Sergey A.
Karabasov and Dimitris Drikakis

Received: 29 January 2026

Revised: 3 March 2026

Accepted: 10 March 2026

Published: 13 March 2026

Copyright: © 2026 by the author.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Numerous engineering disciplines—such as groundwater hydrology, hydrocarbon reservoir engineering, and composite materials science—study systems composed of multiple interacting continuous media. These systems exhibit complex behaviors arising from multi-component, multiscale, and multi-physics coupling, which traditional single-continuum models, relying on homogeneity assumptions, often fail to capture. Standard dual-continuum models address this limitation by coupling mass transfers between two separate continua of distinct scales. However, due to incomplete knowledge about the physical properties and the presence of measurement noise, such models inherently contain uncertainties. In this work, we consider the numerical simulation methods for dual-continuum problems and discuss their related applications.

For the numerical simulation of dual-continuum model, there exist some efficient methods, including the finite element method [1,2], finite volume method [3,4] and hybrid schemes that combine both finite element method and finite volume method [5,6]. In [7],

the authors employ a finite element method with standard linear basis functions for spatial approximation, and combine implicit and explicit temporal schemes regarding time discretization. However, due to the complex heterogeneity of porous media—particularly the presence of multiple scales and high contrast—some form of model reduction is often necessary for practical flow simulations. The application of homogenization theory provides one pathway to deriving dual-continuum models, as illustrated by [8], who extended the earlier framework of [9]. Their work established a general double-porosity model for single-phase flow in fractured reservoirs, explicitly describing transport within and between each continuum. Further extensions have been proposed to enhance the representation of inter-continuum interactions: Wu and Pruess [10] employed multiple interacting continua to better capture transient inter-continuum flow, and Yan et al. [11] later generalized this concept into a versatile multi-continuum framework capable of handling an arbitrary number of continua.

While such models capture flow both inside and across continua, their accuracy relies on a strong separation of scales. In cases of weak scale separation, homogenized effective parameters can lose physical meaning, significantly compromising model fidelity. To overcome the limitations of classical homogenization and directly incorporate multiscale heterogeneity in a computationally efficient manner, the Multiscale Finite Element Method (MsFEM) [12,13] and its generalization, the Generalized Multiscale Finite Element Method (GMsFEM) [14], have been developed. These approaches integrate fine-scale heterogeneity while reducing computational cost. For simulating gas transport in the shale formation, the work of I. Akkutlu et al. in [15] develop a coupled multiscale and multi-continuum approach. Furthermore, active research efforts continue to develop novel model reduction techniques, such as the Constraint Energy Minimizing Generalized Multiscale Finite Element Method (CEM-GMsFEM) [16–18]. Related numerical methods for multi-continuum systems [19], including the non-local multi-continuum method (NLMC) [20,21], are also being advanced. These approaches are particularly effective in addressing the challenges posed by high-contrast properties and multiscale features inherent in multi-continuum media.

When employing a multiscale approach to address heterogeneity in model parameters, solving models with uncertain parameters remains a significant challenge. Typically, such uncertain parameters are characterized by a finite set of random variables. This representation allows the modeling framework to be described by Parameterized Partial Differential Equations (PPDEs) [22,23]. The work [24] presents some reduced basis (RB) methods for fluid infiltration problems through certain porous media modeled as dual-continuum with localized uncertainties. In this work, we focus on developing the reduced basis approximations by combining the uncoupled GMsFEM method and the Variable-separation (VS) method for dual-continuum models with localized uncertainties. The VS method aims to approximate a solution in the form:

$$p(x, t; \mu) \approx \sum_{i=1}^N \zeta_i(\mu) P_i(t, x),$$

where N is the number of separated terms, $\{P_i(t, x)\}$ are deterministic basis functions of space x variable and time variable t , and $\{\zeta_i(\mu)\}$ are stochastic basis functions of the random parameter μ . This separated representation can also be achieved by other methods, such as the Proper Generalized Decomposition (PGD) [25–27] and the Empirical Interpolation Method (EIM) [28–30]. However, the VS method offers distinct advantages. Compared to PGD, which typically requires numerous iterations with an arbitrary initial guess at each enrichment step, the VS method is more computationally efficient. While the VS method shares similarities with EIM, it does not require the selection of optimal parameter values and interpolation

nodes. Furthermore, for online computation, the VS method directly evaluates the solution using the separated representation. In contrast, EIM requires solving an algebraic system based on pre-computed interpolation nodes and parameter values. Consequently, the VS method is not only easier to implement but also enables faster online computation.

This paper presents a framework that integrates a variable-separation technique with an uncoupled multiscale method, resulting in a novel and more powerful approach for solving dual-continuum problems with random inputs—beyond a simple combination of existing components. The remainder of the paper is structured as follows. In Section 2, we introduce some preliminaries and notations. Section 3 is about problem formulation, where we show the existence and uniqueness of the weak solution, and provide fine-scale finite element discretization. In Section 4, an overview of the VS method is given. In Section 5, we describe the uncoupled GMsFEM and hybrid model reduction method. Numerical examples are presented in Section 6 to demonstrate the effectiveness of the proposed hybrid model reduction method. Finally, conclusions and closing remarks are provided.

2. Mathematical Model

In this section, we present some preliminaries and notations for the rest of paper. We first define the close set $[t_0, t_f]$ as the time interval. Let $(\Omega, \mathcal{F}, \mathcal{P})$ be a complete probability space, where Ω is the space of elementary events, $\mathcal{F} \subset 2^\Omega$ is the σ -algebra, and $\mathcal{P}$ is the probability measure on Ω . For computation, the random field needs to be approximated by a prescribed number of random variables, $\mu(\omega) = \{\mu_1(\omega), \mu_2(\omega), \dots, \mu_n(\omega)\}$, i.e., $\mu(\cdot) : \Omega \rightarrow \Gamma \subset \mathbb{R}^n$ ($n \geq 1$). We use the random vector $\mu(\omega)$ to characterize the stochastic property of the random field.

We denote the space of the random variables with second order moments by $L^2_{\mathcal{P}}(\Omega)$, which is defined by

$$L^2_{\mathcal{P}}(\Omega) = \{\mu : \omega \in \Omega \rightarrow \xi(\omega) \in \mathbb{R}; \int_{\Omega} \mu(\omega)^2 \mathcal{P}(d\mu) < \infty\}.$$

The inner product in $L^2_{\mathcal{P}}(\Omega)$ is defined by $(\mu, \xi)_{L^2_{\mathcal{P}}(\Omega)} := \int_{\Omega} \mu(\omega) \xi(\omega) \mathcal{P}(d\mu)$, which induces the norm $\|\mu\|_{L^2}^2 = \|\mu\|_{L^2_{\mathcal{P}}}^2 := (\mu, \mu)_{L^2_{\mathcal{P}}}$.

Let $L^2(D)$ and $H^s(D)$ be the usual Lebesgue space and Sobolev space [31], respectively. With the notation defined above, we define the following tensor product Hilbert space

$$L^2(t_s, t_f; L^{\rho}_{\mathcal{P}}(\Gamma; H^s(D))) := \{v : [t_s, t_f] \times D \times \Gamma \rightarrow \mathbb{R} \mid \|v\|_{L^2(t_s, t_f; L^{\rho}_{\mathcal{P}}(\Gamma; H^s(D)))} < \infty\},$$

and the equipped norm is

$$\begin{aligned} \|v\|_{L^2(t_s, t_f; L^{\rho}_{\mathcal{P}}(\Gamma; H^s(D)))}^2 &= \int_{t_s}^{t_f} \|v\|_{L^{\rho}_{\mathcal{P}}(\Gamma; H^s(D))}^2 dt \\ &= \int_{t_s}^{t_f} \left(\int_{\Gamma} \|v(\cdot, \cdot; \gamma)\|_{H^s(D)}^2 \rho d\gamma \right) dt. \end{aligned}$$

To shorten the notation, we denote

$$\mathcal{H}^s(D_t) := L^2(t_s, t_f; L^{\rho}_{\mathcal{P}}(\Gamma; H^s(D))).$$

In particular,

$$\mathcal{H}_0^1(D_t) = \{v \in \mathcal{H}^1(D_t) : v|_{\partial D} = 0\}.$$

When $s = 0$, we employ the abbreviated notion $\mathcal{L}^2(D_t)$ to denote $\mathcal{H}^0(D_t)$ by convention.

For deterministic situation, the tensor spaces can be simplified by

$$\mathcal{L}^2(\Omega_t) := L^2(t_s, t_f; L^2(D)), \quad \mathcal{H}^1(\Omega_t) := L^2(t_s, t_f; H^1(D)).$$

3. Dual-Continuum Model

In this paper, we consider the dual-continuum model with localized uncertainties:

$$\begin{cases} \frac{\partial p_1(t, x; \mu)}{\partial t} - \operatorname{div}(\kappa_1 \nabla p_1(t, x; \mu)) + \sigma(x)(p_1(t, x; \mu) - p_2(t, x; \mu)) = f_1(t, x; \mu), \\ \frac{\partial p_2(t, x; \mu)}{\partial t} - \operatorname{div}(\kappa_2 \nabla p_2(t, x; \mu)) + \sigma(x)(p_2(t, x; \mu) - p_1(t, x; \mu)) = f_2(t, x; \mu), \end{cases} \quad (1)$$

in a computational domain $D \subset \mathbb{R}^2$. For $i = 1, 2$, let p_i denote the pressure, κ_i the permeability, and f_i the source term for the i -th continuum. The continua are coupled through mass exchange, with σ representing the strength of the mass transfer between them. A notable application of the dual-continuum model (1) is its ability to capture the global interactive effects between unresolved fractures and the surrounding matrix. In this work, we consider high-contrast, channelized media. The initial condition is prescribed as $p_i(x, t_0; \mu) = p_i^0(x; \mu)$ in D , and the boundary condition is given by $p_i(x, t; \mu) = g_i(t; \mu)$ on $\partial D \times [t_0, t_f]$. Here, we assume the permeability fields $\kappa_i(x)$ ($i = 1, 2$) are uniformly bounded, i.e.,

$$\underline{\kappa} \leq \kappa_i(x) \leq \bar{\kappa}, \quad \text{for } \forall x \in D.$$

and the transform term is also bounded, i.e.,

$$|\sigma(x)| \leq \bar{\sigma}, \quad \text{for } \forall x \in v.$$

3.1. Variational Formulation of the Dual-Continuum Model

The model (1) leads to the variational problem as follows: for any test functions v_i , $i = 1, 2$,

$$\begin{cases} (\frac{\partial p_1}{\partial t}, v_1) + a_1(p_1, v_1) + q(p_1 - p_2, v_1) = s_1(v_1) \\ (\frac{\partial p_2}{\partial t}, v_2) + a_2(p_2, v_2) + q(p_2 - p_1, v_2) = s_2(v_2), \end{cases} \quad (2)$$

where

$$\begin{aligned} (\frac{\partial p_1}{\partial t}, v_1) &= \int_D \frac{\partial p_1}{\partial t} v_1 dx, & (\frac{\partial p_2}{\partial t}, v_2) &= \int_D \frac{\partial p_2}{\partial t} v_2 dx, \\ a_1(p_1, v_1) &= \int_D \kappa_1 \nabla p_1 \nabla v_1 dx, & a_2(p_2, v_2) &= \int_D \kappa_2 \nabla p_2 \nabla v_2 dx, \\ q(p_m - p_f, v) &= \int_D \sigma(p_1 - p_2) v dx, & q(p_f - p_m, v) &= \int_D \sigma(p_2 - p_1) v dx. \end{aligned}$$

and

$$s_1(v) = \int_D f_1 v dx - \int_D \kappa_1 \nabla G_1 \nabla v dx, \quad s_2(v) = \int_D f_2 v dx - \int_D \kappa_2 \nabla G_2 \nabla v dx.$$

Here, G_1 and G_2 are continuous piecewise polynomials that match the values of g_1 and g_2 , respectively, at the boundary nodes on ∂D .

Moreover, we assume that both $a_1(\cdot, \cdot; \mu)$, $a_2(\cdot, \cdot; \mu)$, $s_1(v; \mu)$ and $s_2(v; \mu)$ are affinely dependent on the parameter vector μ , i.e., for some $Q_a, Q_b, N_a, M_a \in \mathbb{N}$, the following affine decompositions hold:

$$\begin{cases} a_1(w, v; \mu) = \sum_{i=1}^{Q_a} Q_1^i(\mu) a_1^i(w, v) & \forall w, v \in H^1(D), \forall \mu \in \Gamma, \\ a_2(w, v; \mu) = \sum_{j=1}^{Q_b} Q_2^j(\mu) a_2^j(w, v) & \forall w, v \in H^1(D), \forall \mu \in \Gamma, \\ s_1(v; \mu) = \sum_{n=1}^{N_a} X_n(\mu) s_{1,n}(v) & \forall v \in H^1(D), \forall \mu \in \Gamma, \\ s_2(v; \mu) = \sum_{m=1}^{M_a} Y_m(\mu) s_{2,m}(v) & \forall v \in H^1(D), \forall \mu \in \Gamma. \end{cases} \quad (3)$$

If the bilinear forms $a_1(\cdot, \cdot; \mu)$ and $a_2(\cdot, \cdot; \mu)$ are not affine, an affine approximation can still be constructed using the VS method introduced in [32] or the Empirical Interpolation Method (EIM; see, e.g., [28–30]).

3.2. The Approximation Based on FEM

To obtain a reference (truth) solution, the finite element method (FEM) is used for the spatial discretization of the dual-continuum model, while the implicit Euler method is employed for time integration.

Let $\mathcal{T}_h$ be a uniform partition of the spatial domain D . On a fine grid $\mathcal{T}_h$, the corresponding FE space is $\mathbf{V}^h(D) \subset H^1(D) \times H^1(D)$ and $\mathbf{V}_0^h(D) \subset \mathbf{V}^h(D)$ is the subspace with vanishing boundary values. Let N_h be the number of vertices, and the dimension of FE space be $\mathcal{N}$. Typically, the dimension is very large. Furthermore, we suppose that $M^h(D)$ is the finite dimensional subspace of $L^2(D)$. Thus, we define

$$\mathcal{V}^h(D_t) := L^2(t_s, t_f; \mathbf{V}^h(D)) \subset \mathcal{H}^1(D_t).$$

With the notations defined above, the Galerkin formulation of the system (2) is to find $(p_{1,h}, p_{2,h}) \in \mathcal{V}^h(D_t) \times \mathcal{V}^h(D_t)$ such that

$$\begin{cases} \left(\frac{\partial p_{1,h}}{\partial t}, v_{1,h} \right) + a_1(p_{1,h}, v_{1,h}) + q(p_{1,h} - p_{2,h}, v_{1,h}) = s_{1,h}(v_{1,h}) \\ \left(\frac{\partial p_{2,h}}{\partial t}, v_{2,h} \right) + a_2(p_{2,h}, v_{2,h}) + q(p_{2,h} - p_{1,h}, v_{2,h}) = s_{2,h}(v_{2,h}), \end{cases} \quad (4)$$

where both $v_{1,h}$ and $v_{2,h}$ are the respective test functions.

Applying the implicit Euler method to the system (4) yields the full discretized formulation of the dual-continuum model as follows:

$$\begin{cases} (p_{1,h}^{k+1}, v_{1,h}) + \Delta t a_1(p_{1,h}^{k+1}, v_{1,h}) + \Delta t q(p_{1,h}^{k+1} - p_{2,h}^{k+1}, v_{1,h}) = \Delta t s_{1,h}^{k+1}(v_{1,h}) + (p_{1,h}^k, v_{1,h}), \\ (p_{2,h}^{k+1}, v_{2,h}) + \Delta t a_2(p_{2,h}^{k+1}, v_{2,h}) + \Delta t q(p_{2,h}^{k+1} - p_{1,h}^{k+1}, v_{2,h}) = \Delta t s_{2,h}^{k+1}(v_{2,h}) + (p_{2,h}^k, v_{2,h}), \end{cases} \quad (5)$$

where $\Delta t := \frac{t_f - t_s}{N_t}$ is the time step, $t_k := k\Delta t$, $0 \leq k \leq N_t$. If the basis functions of space $V^h(D)$ are denoted by $\{\phi_j\}_{j=1}^{N_h}$, we can represent the variables in Equation (5) with the linear combination of the corresponding basis functions as

$$p_{1,h}^k = \sum_{i=1}^{N_h} p_{1,h}^{k,i}(\mu) \phi_i, \quad p_{2,h}^k = \sum_{i=1}^{N_h} p_{2,h}^{k,i}(\mu) \phi_i.$$

If affine assumption (3) holds, the full discretization (5) becomes

$$\left\{ \begin{array}{l} \sum_{i=1}^{N_h} P_{1,h}^{k+1,i}(\phi_i, \phi_l) + \Delta t \sum_{s=1}^{Q_a} \sum_{i=1}^{N_h} Q_1^s(\mu) P_{1,h}^{k+1,i} a_1^s(\phi_i, \phi_l) + \Delta t \sum_{i=1}^{N_h} P_{1,h}^{k+1,i} q(\phi_i, \phi_l) - \\ \Delta t \sum_{j=1}^{N_h} P_{2,h}^{k+1,i} q(\phi_j, \phi_l) = \Delta t \sum_{n=1}^{N_a} X_n(\mu) S_{1,n}^{k+1}(\phi_l) + \sum_{i=1}^{N_h} P_{1,h}^{k,i}(\phi_i, \phi_l) \\ \sum_{j=1}^{N_h} P_{2,h}^{k+1,j}(\phi_j, \phi_m) + \Delta t \sum_{t=1}^{Q_b} \sum_{j=1}^{N_h} Q_2^t(\mu) P_{2,h}^{k+1,j} a_2^t(\phi_j, \phi_m) + \Delta t \sum_{j=1}^{N_h} P_{2,h}^{k+1,j} q(\phi_j, \phi_m) - \\ \Delta t \sum_{i=1}^{N_h} P_{1,h}^{k+1,i} q(\phi_i, \phi_m) = \Delta t \sum_{r=1}^{M_a} Y_r(\mu) S_{2,r}^{k+1}(\phi_m) + \sum_{j=1}^{N_h} P_{1,h}^{k,j}(\phi_j, \phi_m), \end{array} \right. \quad (6)$$

where ϕ_l and ϕ_m are the test functions corresponding to the pressure variables.

Next, if we define

$$\left\{ \begin{array}{l} (M_{1,h})_{i,l} = (\phi_i, \phi_l), 1 \leq i, l \leq N_h, \\ (M_{2,h})_{i,l} = q(\phi_i, \phi_l), 1 \leq i, l \leq N_h, \\ (K_{1,h}^s)_{i,l} = a_1^s(\phi_i, \phi_l), 1 \leq i, l \leq N_h \\ (K_{2,h}^t)_{j,m} = a_2^t(\phi_j, \phi_m), 1 \leq j, m \leq N_h \\ (B_{1,n}^{k+1})_l = S_{1,n}^{k+1}(\phi_l), 1 \leq l \leq N_h \\ (B_{2,r}^{k+1})_m = S_{2,r}^{k+1}(\phi_m), 1 \leq m \leq N_h \\ K_{1,h} = \sum_{s=1}^{Q_a} Q_1^s(\mu) K_{1,h}^s, K_{2,h} = \sum_{t=1}^{Q_b} Q_2^t(\mu) K_{2,h}^t \\ B_1^{k+1} = \sum_{n=1}^{N_a} X_n(\mu) B_{1,n}^{k+1}, B_2^{k+1} = \sum_{r=1}^{M_a} Y_r(\mu) B_{2,r}^{k+1}, \end{array} \right. \quad (7)$$

then we can get the matrix form of full discrete system (6) as

$$\left\{ \begin{array}{l} (M_{1,h} + \Delta t K_{1,h}) P_{1,h}^{k+1} + \Delta t M_{2,h} (P_{1,h}^{k+1} - P_{2,h}^{k+1}) = \Delta t B_1^{k+1} + M_{1,h} P_{1,h}^k \\ (M_{1,h} + \Delta t K_{2,h}) P_{2,h}^{k+1} + \Delta t M_{2,h} (P_{2,h}^{k+1} - P_{1,h}^{k+1}) = \Delta t B_2^{k+1} + M_{2,h} P_{2,h}^k. \end{array} \right. \quad (8)$$

Combining these equations gives the matrix system

$$\begin{bmatrix} M_{1,h} + \Delta t K_{1,h} + \Delta t M_{2,h} & -\Delta t M_{2,h} \\ -\Delta t M_{2,h} & M_{1,h} + \Delta t K_{2,h} + \Delta t M_{2,h} \end{bmatrix} \begin{bmatrix} P_{1,h}^{k+1}(\mu) \\ P_{2,h}^{k+1}(\mu) \end{bmatrix} = \begin{bmatrix} F_1^{k+1}(\mu) \\ F_2^{k+1}(\mu) \end{bmatrix}. \quad (9)$$

Here, $P_{1,h}^{k+1}(\mu)$ and $P_{2,h}^{k+1}(\mu)$ denote the discrete approximations of p_1, p_2 at the $(k+1)$ -th time step, with the corresponding right-hand-side vectors given by

$$F_1^{k+1}(\mu) := \Delta t B_1^{k+1} + M_{1,h} P_{1,h}^k \quad F_2^{k+1}(\mu) := \Delta t B_2^{k+1} + M_{1,h} P_{2,h}^k.$$

Under the affine assumption (3), the mass matrices $M_{1,h}$ and $M_{2,h}$, the stiffness matrices $\{K_{1,h}^s\}_{s=1}^{Q_a}$ and $\{K_{2,h}^t\}_{t=1}^{Q_b}$, and the right-hand-side vectors $\{B_{1,n}^{k+1}\}_{n=1}^{N_a}$ and $\{B_{2,r}^{k+1}\}_{r=1}^{M_a}$ can be precomputed once during the offline stage, despite the high computational cost involved. In the online stage, for each new parameter sample $\mu \in \Gamma$, all coefficients $\{Q_1^s(\mu)\}_{s=1}^{Q_a}$, $\{Q_2^t(\mu)\}_{t=1}^{Q_b}$, $\{X_n(\mu)\}_{n=1}^{N_a}$ and $\{Y_r(\mu)\}_{r=1}^{M_a}$ need to be updated.

4. A Variable-Separation Method for the Stochastic Dual-Continuum Model

In this section, we present a variable-separation method applicable to the stochastic dual-continuum model. The corresponding stochastic solutions are expressed in the following form:

$$\begin{cases} p_1(t, x; \mu) \approx p_{1,N}(t, x; \mu) := \sum_{i=1}^N \xi_i(\mu) P_{1,i}(t, x) \\ p_2(t, x; \mu) \approx p_{2,N}(t, x; \mu) := \sum_{i=1}^N \lambda_i(\mu) P_{2,i}(t, x). \end{cases} \quad (10)$$

Here, N denotes the number of separation terms. The stochastic basis functions $\{\xi_i(\mu)\}_{i=1}^N$ and $\{\lambda_j(\mu)\}_{j=1}^N$ belong to the space $L^2_{\mathcal{P}}(\Gamma)$, while $\{P_{1,i}(t, x)\}_{i=1}^N$ and $\{P_{2,j}(t, x)\}_{j=1}^N$ are corresponding deterministic basis functions in $\mathcal{V}^h(D_t)$.

Let $\Xi \in \Gamma$ be a set of training samples. We define

$$\begin{cases} e_{p_1}(t, x; \mu) := p_1(t, x; \mu) - p_{1,k-1}(t, x; \mu), \\ e_{p_2}(t, x; \mu) := p_2(t, x; \mu) - p_{2,k-1}(t, x; \mu). \end{cases}$$

For $\forall v_1, v_2 \in V^h(D)$, we then have

$$\begin{cases} \left(\frac{\partial e_{p_1}}{\partial t}, v_1; \mu \right) + a_1(e_{p_1}, v_1; \mu) + q(e_{p_1} - e_{p_2}, v_1; \mu) \\ \quad = s_1(v_1; \mu) - \left(\frac{\partial p_{1,k-1}}{\partial t}, v_1; \mu \right) - a_1(p_{1,k-1}, v_1; \mu) - q(p_{1,k-1} - p_{2,k-1}, v_1; \mu) \\ \left(\frac{\partial e_{p_2}}{\partial t}, v_2; \mu \right) + a_2(e_{p_2}, v_2; \mu) + q(e_{p_2} - e_{p_1}, v_2; \mu) \\ \quad = s_2(v_2; \mu) - \left(\frac{\partial p_{2,k-1}}{\partial t}, v_2; \mu \right) - a_2(p_{2,k-1}, v_2; \mu) - q(p_{2,k-1} - p_{1,k-1}, v_2; \mu) \end{cases}$$

Let $r_{p_1}^k(v_1; \mu), r_{p_2}^k(v_2; \mu) \in \mathcal{V}^{h,*}(\Omega_t)$. We have

$$r_{p_1}^k(v_1; \mu) = \begin{cases} s_1(v_1; \mu), & k = 1, \\ s_1(v_1; \mu) - \left(\frac{\partial p_{1,k-1}}{\partial t}, v_1; \mu \right) - a_1(p_{1,k-1}, v_1; \mu) - q(p_{1,k-1} - p_{2,k-1}, v_1; \mu), & k \geq 2, \end{cases} \quad (11a)$$

$$r_{p_2}^k(v_2; \mu) = \begin{cases} s_2(v_2; \mu), & k = 1, \\ s_2(v_2; \mu) - \left(\frac{\partial p_{2,k-1}}{\partial t}, v_2; \mu \right) - a_2(p_{2,k-1}, v_2; \mu) - q(p_{2,k-1} - p_{1,k-1}, v_2; \mu), & k \geq 2, \end{cases} \quad (11b)$$

where $\mathcal{V}^{h,*}(D_t)$ is the dual space of $\mathcal{V}^h(D_t)$, and $\mathcal{M}^{h,*}(D_t)$ is the dual space of $\mathcal{M}^h(D_t)$. Then we have

$$\begin{cases} \left(\frac{\partial e_{p_1}}{\partial t}, v_1; \mu \right) + a_1(e_{p_1}, v_1; \mu) + q(e_{p_1} - e_{p_2}, v_1; \mu) = r_{p_1}^k(v_1; \mu) \\ \left(\frac{\partial e_{p_2}}{\partial t}, v_2; \mu \right) + a_2(e_{p_2}, v_2; \mu) + q(e_{p_2} - e_{p_1}, v_2; \mu) = r_{p_2}^k(v_2; \mu) \end{cases} \quad (12)$$

According to Riesz representation theory, there exists functions $\hat{e}_{p_1}^k(\mu), \hat{e}_{p_2}^k(\mu) \in \mathcal{V}^h(\Omega_t)$, such that

$$\begin{cases} (\hat{e}_{p_1}^k(\mu), v_1; \mu)_{\mathcal{H}^1} = r_{p_1}^k(v_1; \mu), \\ (\hat{e}_{p_2}^k(\mu), v_2; \mu)_{\mathcal{H}^1} = r_{p_2}^k(v_2; \mu). \end{cases} \quad (13)$$

Then we can rewrite Equation (12) as

$$\begin{cases} \left(\frac{\partial e_{p_1}}{\partial t}, v_1; \mu\right) + a_1(e_{p_1}, v_1; \mu) + q(e_{p_1} - e_{p_2}, v_1; \mu) = (\hat{e}_{p_1}^k(\mu), v_1; \mu)_{\mathcal{H}^1}, \\ \left(\frac{\partial e_{p_2}}{\partial t}, v_2; \mu\right) + a_2(e_{p_2}, v_2; \mu) + q(e_{p_2} - e_{p_1}, v_2; \mu) = (\hat{e}_{p_2}^k(\mu), v_2; \mu)_{\mathcal{H}^1}. \end{cases} \quad (14)$$

Consequently, the dual norm of the residuals $\{r_u^k(v_1; \mu)\}$, $\{r_\lambda^k(v_2; \mu)\}$, $\{r_f^k(\tilde{f}; \mu)\}$ can be evaluated through the Riesz representation,

$$\begin{cases} \|r_{p_1}^k(v_1; \mu)\|_{(\mathcal{H}^1)^*} := \sup_{v_1 \in \mathcal{H}^1} \frac{r_{p_1}^k(v_1; \mu)}{\|v_1\|_{\mathcal{H}^1}} = \|\hat{e}_{p_1}(\mu)\|_{\mathcal{H}^1}, \\ \|r_{p_2}^k(v_2; \mu)\|_{(\mathcal{H}^1)^*} := \sup_{v_2 \in \mathcal{H}^1} \frac{r_{p_2}^k(v_2; \mu)}{\|v_2\|_{\mathcal{H}^1}} = \|\hat{e}_{p_2}(\mu)\|_{\mathcal{H}^1}. \end{cases}$$

Next, we define an error estimator for the dual-continuum model (4) by

$$\Delta_k(x, t; \mu) := \sqrt{\|\hat{e}_{p_1}^k(\mu)\|_{\mathcal{H}^1}^2 + \|\hat{e}_{p_2}^k(\mu)\|_{\mathcal{H}^1}^2}.$$

The computation of the residuals is crucial to the VS procedure. To efficiently compute $\|\hat{e}_{p_1}^k(\mu)\|_{\mathcal{H}^1}$, $\|\hat{e}_{p_2}^k(\mu)\|_{\mathcal{H}^1}$, we apply an offline–online procedure presented in [33,34]. The detailed procedure is presented in Appendix A.

At step k , we choose

$$\mu_k := \begin{cases} \text{choose randomly in } \Xi, & k = 1, \\ \arg \max_{\mu \in \Xi} \Delta_k(\mu) & k \geq 2. \end{cases}$$

Let $e_{p_1}(x, t; \mu_k)$ and $e_{p_2}(x, t; \mu_k)$ be the solutions of system (12) with $\mu = \mu_k$. We take

$$p_{1,k}(t, x) = e_{p_1}(x, t; \mu_k), \quad p_{2,k}(t, x) = e_{p_2}(x, t; \mu_k).$$

We define the error expansions as

$$e_{p_1}(x, t; \mu) = \xi_k(\mu) p_{1,k}(t, x), \quad e_{p_2}(x, t; \mu) = \lambda_k(\mu) p_{2,k}(t, x).$$

Next, in the first equation of system (12) we set $v_1 = p_{1,k}(x, t_J)$, and in the second equation we set $v_2 = p_{2,k}(x, t_J)$, where

$$J := \arg \max_{j=1, \dots, N_t} \left(r_{p_1}^k(p_{1,k}(x, t_j); \mu_k) + r_{p_2}^k(p_{2,k}(x, t_j); \mu_k) \right).$$

By (3), (10), (12) and (A1), we can get the linear system with regard to $\xi_k(\mu)$ and $\lambda_k(\mu)$ as follows

$$\begin{cases} L^{11}(\mu) \xi_k(\mu) + L_s^{12}(\mu) \lambda_k(\mu) = r_{p_1}^k(p_{1,k}(x, t_J); \mu), \\ L^{21}(\mu) \lambda_k(\mu) + L^{22}(\mu) \xi_k(\mu) = r_{p_2}^k(p_{2,k}(x, t_J); \mu). \end{cases} \quad (15)$$

The coefficients are defined as follows:

$$\begin{cases} L_{11}(\mu) = \left(\frac{\partial p_{1,k}(t, x)}{\partial t}, p_{1,k}(x, t_J) \right) + \sum_{i=1}^{Q_a} Q_1^i(\mu) a_1^i(p_{1,k}(t, x), p_{1,k}(x, t_J)) + q(p_{1,k}(t, x), p_{1,k}(x, t_J)) \\ L_{12}(\mu) = -q(p_{2,k}(t, x), p_{1,k}(x, t_J)) \\ L_{21}(\mu) = \left(\frac{\partial p_{2,k}(t, x)}{\partial t}, p_{2,k}(x, t_J) \right) + \sum_{j=1}^{Q_b} Q_2^j(\mu) a_2^j(p_{2,k}(t, x), p_{2,k}(x, t_J)) + q(p_{2,k}(t, x), p_{2,k}(x, t_J)) \\ L_{22}(\mu) = -q(p_{1,k}(t, x), p_{2,k}(x, t_J)) \end{cases}$$

and

$$L_g^{11}(\gamma) = \beta(f^k(t, x), f_J^k(x)), \quad L_g^{12}(\gamma) = -(\lambda^{2k-1}(t, x), f_J^k(x)), \quad L_g^{13}(\gamma) = -(\lambda^{2k}(t, x), f_J^k(x)).$$

The right terms have the following affine expressions:

$$\begin{aligned} r_{p_1}^k(p_{1,k}(x, t_J); \mu) &= \sum_{n=1}^{N_a} X_n(\mu) s_{1,n}(p_{1,k}(x, t_J)) - \sum_{i=1}^{k-1} \xi_i(\mu) \left(\frac{\partial p_{1,i}(t, x)}{\partial t}, p_{1,k}(x, t_J) \right) \\ &\quad - \sum_{s=1}^{Q_a} \sum_{i=1}^{k-1} Q_1^s(\mu) \xi_i(\mu) a_1^s(p_{1,i}(t, x), p_{1,k}(x, t_J)) \\ &\quad - \sum_{i=1}^{k-1} \xi_i(\mu) q(p_{1,i}(t, x), p_{1,k}(x, t_J)) + \sum_{j=1}^{k-1} \lambda_j(\mu) q(p_{2,j}(t, x), p_{1,k}(x, t_J)), \\ r_{p_2}^k(p_{2,k}(x, t_J); \mu) &= \sum_{m=1}^{M_a} Y_m(\mu) s_{2,m}(p_{2,k}(x, t_J)) - \sum_{j=1}^{k-1} \lambda_j(\mu) \left(\frac{\partial p_{2,i}(t, x)}{\partial t}, p_{2,k}(x, t_J) \right) \\ &\quad - \sum_{t=1}^{Q_b} \sum_{j=1}^{k-1} Q_2^t(\mu) \lambda_j(\mu) a_2^t(p_{2,i}(t, x), p_{2,k}(x, t_J)) \\ &\quad - \sum_{j=1}^{k-1} \lambda_j(\mu) q(p_{2,j}(t, x), p_{2,k}(x, t_J)) + \sum_{i=1}^{k-1} \xi_i(\mu) q(p_{1,i}(t, x), p_{2,k}(x, t_J)). \end{aligned}$$

Therefore, the stochastic basis functions $\xi_k(\mu)$, $\lambda_k(\mu)$ can be obtained by system (15). We outline the VS method for stochastic dual-continuum problems in Algorithm 1.

Algorithm 1 The VS method for stochastic dual-continuum problems

Input: The stochastic dual-continuum problem (3), a set of training samples $\Xi \in \Gamma$, and the error tolerance ε_0 .

Output: The variable-separation representations $p_{1,N}(t, x; \mu) := \sum_{i=1}^N \xi_i(\mu) P_{1,i}(t, x)$, $p_{2,N}(t, x; \mu) := \sum_{i=1}^N \lambda_i(\mu) P_{2,i}(t, x)$.

1: **Initialization:** Residuals $r_{p_1}^1(\bar{u}; \mu) := s_1(v_1; \mu)$, $r_{p_2}^1(\bar{u}; \mu) := s_2(v_2; \mu)$, a random sample $\mu_1 \in \Xi$, the iteration counter $k = 1$;

2: Calculate the deterministic physical basis functions $P_{1,k}(t, x) = e_{p_1}(x, t; \mu_k)$, $P_{2,k}(t, x) = e_{p_2}(x, t; \mu_k)$ by solving (12) with $\mu = \mu_k$;

3: Compute the stochastic basis functions $\xi^k(\mu)$, $\lambda^k(\mu)$ by (15);

4: Update Ξ with $\Xi = \Xi \setminus \mu_k$, and take the approximation $p_{1,k}(x, t; \mu) := \sum_{i=1}^k \xi_i(\mu) P_{1,i}(t, x)$ and $p_{2,k}(x, t; \mu) := \sum_{i=1}^k \lambda_i(\mu) P_{2,i}(t, x)$;

5: $k = k + 1$;

6: Update the residuals $r_{p_1}^k(v_1; \mu)$ and $r_{p_2}^k(v_2; \mu)$, and compute $\Delta_k(x, t; \mu)$;

7: Take μ_k as $\mu_k := \arg \max_{\mu \in \Xi} \Delta_k(x, t; \mu)$;

8: If $\Delta_k \geq \varepsilon_0$, go back to Step 2; otherwise, **terminate**.

9: $N = k$.

5. The Hybrid Model Reduction Method Based on Variable-Separation

Derived from the governing equations of the dual-continuum model, the stochastic and deterministic basis functions used in the VS framework (Section 4) are computed via a traditional finite element method on a fine grid, employing a backward Euler temporal discretization. If the dual-continuum governing equations have strong multiscale features, then we have to use a very fine mesh to resolve the features in all scales. This computation of computing snapshots may be very expensive. To overcome the difficulty and improve the offline computation, we can use the uncoupled Generalized Multiscale Finite Element Method (uncoupled GMsFEM) to compute the basis functions. In this section, we briefly present the uncoupled GMsFEM and the hybrid model reduction method.

5.1. Uncoupled GMsFEM

For the parameterized dual-continuum model, when constructing the multiscale reduced basis space using the uncoupled Generalized Multiscale Finite Element Method (GMsFEM), the multiscale basis functions on each continuum are constructed independently, taking into account only the permeability $\kappa_i(x; \mu)$ while neglecting the effect of the interaction term σ .

Before presenting the framework of uncoupled GMsFEM approximation for the dual-continuum model, we first establish the numerical discretization setting. Let H be the characteristic size of a generic coarse cell. We consider that the spatial domain D to be uniformly partitioned by a coarse mesh $\mathcal{T}_H$. The partition $\mathcal{T}_H$ is referred to as a coarse grid and a generic element K within it is called a coarse element. Moreover, a corresponding fine-grid partition, denoted by $\mathcal{T}_h$, is obtained by refining the coarse mesh $\mathcal{T}_H$, where $h > 0$ represents the fine mesh size. Let N_h denote the number of elements in $\mathcal{T}_H$. The vertices of the coarse grid are denoted by $\{x_j\}_{j=1}^{N_c}$, where N_c is the total number of coarse nodes. We define the neighborhood ω_j as the union of all coarse elements containing x_j in their closure, i.e.,

$$\omega_j = \cup \{K_i \in \mathcal{T}_H | x_j \in \overline{K_i}\}.$$

As illustrated in Figure 1, the computational domain D is equipped with a coarse mesh partition (the element in red) and a fine mesh partition (the element in green). The region illustrates the neighborhood ω_j associated with the coarse node N_i marked by a red circle.

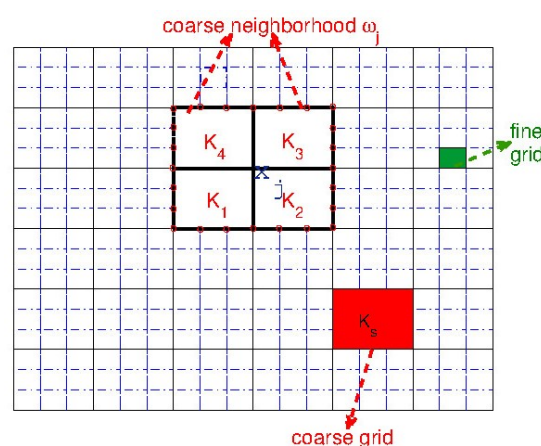


Figure 1. The fine grid, coarse grid K_s and the coarse neighborhood of node x_j .

More specially, on each coarse neighborhood ω_j , for each i -th continuum, we first solve the following local eigenvalue problem:

$$\begin{cases} -\operatorname{div}(\kappa_i(x)\nabla\psi_{\ell,i}^{(j)}) = \lambda_{i,\ell}\widetilde{\kappa(x)}\psi_{\ell,i}^{(j)} & \text{in } \omega_j, \\ \kappa_i(x) \cdot \nabla\psi_{\ell,i}^{(j)} \cdot n = 0 & \text{on } \partial\omega_j, \end{cases} \quad (16)$$

where $\widetilde{\kappa_i(x)} = \kappa_i(x) \sum_{i=1}^{N_c} H^2 |\nabla\chi_{j,i}|^2$. $\{\chi_{j,i}\}_{i=1}^{N_c}$ represents the partition of unity functions defined in the local support ω_j , which is defined as follows:

$$\begin{cases} -\operatorname{div}(\kappa_i(x)\nabla\chi_{j,i}) = 0 & \forall K \text{ in } \omega_j \\ \chi_{j,i} = \chi_{j,i}^0 & \text{on } \partial K, \end{cases}$$

To construct local multiscale basis functions on ω_j corresponding to the i -th continuum ($i = 1, 2$), we now solve local spectral problems: find the eigenfunctions $\psi_{\ell,i}^{(j)} \in V_h(\omega_j)$ and $\lambda_{i,\ell} \in \mathbb{R}$ such that

$$a_i^{(j)}(\psi_{\ell,i}^{(j)}, v_i) = \lambda_{i,\ell} s_i^{(j)}(\psi_{\ell,i}^{(j)}, v_i),$$

for all $v_i \in V_h(\omega_j)$, where $s_i^{(j)}(\psi_{\ell,i}^{(j)}, v_i)$ is defined as follows

$$s_i^{(j)}(u_i, v_i) = \int_{\omega_j} \widetilde{\kappa_i} u_i v_i dx.$$

We denote the basis functions defined on the fine grid as $\{\phi_{j,m}\}_{m=1}^{N_h}$, and then the matrix form of eigenvalue problem (16) can be written as

$$A\psi_{\ell,i}^{(j)} = \lambda_{\ell,i}^{(j)} S\psi_{\ell,i}^{(j)}. \quad (17)$$

Here, the definition of matrices A and S are

$$(A)_{m,n} = (\kappa_i(x)\nabla\phi_{j,m}, \nabla\phi_{j,n})_{L^2(\omega_j)}, \quad (S)_{m,n} = (\widetilde{\kappa_i(x)}\phi_{j,m}, \phi_{j,n})_{L^2(\omega_j)},$$

respectively.

By solving the eigenvalue problem (17), we can get a series of eigenvalues $\{\lambda_{\ell,i}^{(j)}\}_{i=1}^{N_h}$ and eigenfunctions $\{\psi_{\ell,i}^{(j)}\}_{i=1}^{N_h}$. Then we choose the M_i lowermost eigenvalues and the corresponding eigenvectors of the eigenvalue problem. Then we can get the local snapshot space:

$$V_{\text{ms}}^i(\omega_j) := \operatorname{span}\{\psi_{\ell,i}^{(j)}, 1 \leq \ell \leq M_j\}.$$

Further, we use the partition of unity functions to paste the snapshot functions and get the multiscale basis function space of i -th continuum:

$$V_{\text{ms}}^i(D) := \sum_{j=1}^{N_c} V_{\text{ms}}^i(\omega_j) = \operatorname{span}\{\psi_{\ell,i}^{(j),\text{ms}} : \psi_{\ell,i}^{(j),\text{ms}} = \chi_{j,i}\psi_{\ell,i}^{(j)}, 1 \leq j \leq N_c, 1 \leq \ell \leq M_i\}.$$

Finally, the global multiscale corresponding to the dual-continuum model as

$$V_{\text{ms}}(D) = V_{\text{ms}}^1(D) \times V_{\text{ms}}^2(D).$$

The multiscale basis functions can be repeatedly used online. We can use a matrix R_{re}^i to store these basis functions, i.e.,

$$R_{re}^i = [\psi_{1,i}^{(1),\text{ms}}, \dots, \psi_{M_i,i}^{(1),\text{ms}}, \dots, \psi_{1,i}^{(N_c),\text{ms}}, \dots, \psi_{M_i,i}^{(N_c),\text{ms}}].$$

Then define the global matrix in the dual-continuum model as

$$R_{re} = [R_{re}^1, R_{re}^2].$$

At the online stage, we use the precomputed multiscale basis functions to compute dual-continuum problem on the coarse grid.

5.2. The Hybrid Model Reduction Method

In this subsection, we will construct the hybrid model reduction method by combining the variable-separation method and uncoupled GMSFEM, which gives the solution in the following form:

$$\begin{cases} p_1^{vs-GMSFEM}(t, x; \mu) := \sum_{i=1}^N \tilde{\zeta}_i(\mu) P_{1,i}^{gms}(t, x) \\ p_2^{vs-GMSFEM}(t, x; \mu) := \sum_{i=1}^N \lambda_i(\mu) P_{2,i}^{gms}(t, x). \end{cases} \quad (18)$$

In the separation form (18), the physical basis functions $\{P_{1,i}^{gms}(t, x)\}_{i=1}^N$ and $\{P_{2,i}^{gms}(t, x)\}_{i=1}^N$ are learned using the uncoupled GMSFEM.

$$\begin{cases} (R_{re}^1)^T (M_{1,h} + \Delta t K_{1,h}) R_{re}^1 P_{1,h}^{gms,k+1} + \Delta t [(R_{re}^1)^T M_{2,h} R_{re}^1 P_{1,h}^{gms,k+1} - \\ (R_{re}^1)^T M_{2,h} R_{re}^2 P_{2,h}^{gms,k+1}] = \Delta t (R_{re}^1)^T B_1^{k+1} + (R_{re}^1)^T M_{1,h} R_{re}^1 P_{1,h}^{gms,k} \\ (R_{re}^2)^T (M_{1,h} + \Delta t K_{2,h}) R_{re}^2 P_{2,h}^{gms,k+1} + \Delta t [(R_{re}^2)^T M_{2,h} R_{re}^2 P_{2,h}^{gms,k+1} - \\ (R_{re}^2)^T M_{2,h} R_{re}^1 P_{1,h}^{gms,k+1}] = \Delta t (R_{re}^2)^T B_2^{k+1} + (R_{re}^2)^T M_{2,h} R_{re}^2 P_{2,h}^{gms,k}. \end{cases} \quad (19)$$

To simplify the symbols, we denote the matrices by

$$\begin{cases} A_1 = (R_{re}^1)^T (M_{1,h} + \Delta t K_{1,h}) R_{re}^1, \\ A_2 = (R_{re}^2)^T (M_{1,h} + \Delta t K_{2,h}) R_{re}^2, \\ C_{11} = (R_{re}^1)^T M_{2,h} R_{re}^1, \\ C_{12} = (R_{re}^1)^T M_{2,h} R_{re}^2, \\ C_{22} = (R_{re}^2)^T M_{2,h} R_{re}^2, \\ C_{21} = (R_{re}^1)^T M_{2,h} R_{re}^1, \\ F_1^{gms,k+1}(\mu) = \Delta t (R_{re}^1)^T B_1^{k+1} + (R_{re}^1)^T M_{1,h} R_{re}^1 P_{1,h}^{gms,k}, \\ F_2^{gms,k+1}(\mu) = \Delta t (R_{re}^2)^T B_2^{k+1} + (R_{re}^2)^T M_{2,h} R_{re}^2 P_{2,h}^{gms,k}. \end{cases}$$

Combining these equations gives the following matrix system:

$$\begin{bmatrix} A_1 + \Delta t C_{11} & \Delta t C_{12} \\ \Delta t C_{22} & A_2 + \Delta t C_{22} \end{bmatrix} \begin{bmatrix} P_{1,h}^{gms,k+1}(\mu) \\ P_{2,h}^{gms,k+1}(\mu) \end{bmatrix} = \begin{bmatrix} F_1^{gms,k+1}(\mu) \\ F_2^{gms,k+1}(\mu) \end{bmatrix}. \quad (20)$$

Here, the terms $P_{1,h}^{gms,k+1}(\mu)$ and $P_{2,h}^{gms,k+1}(\mu)$ represent the discrete numerical approximations for p_1, p_2 at the time step $(k+1)$, obtained using the uncoupled GMSFEM.

In Algorithm 2, we outline the main steps for the hybrid model reduction method based on the VS method.

Algorithm 2 Hybrid model reduction method for the stochastic dual-continuum model

Input: The stochastic dual-continuum model (4), a set of training samples $\Xi \in \Gamma$, and the error tolerance ε_0 .

Output: The hybrid reduced model solutions $p_1^{vs-GMsFEM}(t, x; \mu) := \sum_{i=1}^N \xi_i(\mu) P_{1,i}^{gms}(t, x)$, $p_2^{vs-GMsFEM}(t, x; \mu) := \sum_{i=1}^N \lambda_i(\mu) P_{2,i}^{gms}(t, x)$.

1: **Initialization:** Residuals $r_{p_1}^1(\tilde{u}; \mu) := s_1(v_1; \mu)$, $r_{p_2}^1(\tilde{u}; \mu) := s_2(v_2; \mu)$, a random sample $\mu_1 \in \Xi$;

2: Run the algorithm corresponding to system (20) similar to Algorithm 1 to obtain the physical basis functions $\{P_{1,k}^{gms}(t, x)\}$ and $\{P_{2,k}^{gms}(t, x)\}$ with $\mu = \mu_k$;

3: Compute the stochastic basis functions $\xi^k(\mu)$, $\lambda^k(\mu)$ similar to (15);

4: Update Ξ with $\Xi = \Xi \setminus \mu_k$, and take the approximation $p_1^{hb}(x, t; \mu) := \sum_{i=1}^k \xi_i(\mu) P_{1,i}^{gms}(t, x)$

and $p_2^{hb}(x, t; \mu) := \sum_{i=1}^k \lambda_i(\mu) P_{2,i}^{gms}(t, x)$

5: $k=k+1$;

6: Update the residuals $r_{p_1}^k(v_1; \mu)$ and $r_{p_2}^k(v_2; \mu)$, and compute $\Delta_k(x, t; \mu)$;

7: Take μ_k as $\mu_k := \arg \max_{\mu \in \Xi} \Delta_k(x, t; \mu)$;

8: If $\Delta_k \geq \varepsilon$, go back to Step 2; otherwise, **terminate**.

9: $N = k$.

6. Numerical Results

In this section, we present two numerical examples to illustrate the applicability of the proposed reduced basis methods for solving the stochastic dual-continuum models. In Section 6.1, we consider an example to illustrate performance of the hybrid model reduction method for solving the dual-continuum model. In Section 6.2, we use the proposed model reduction method to a dual-continuum model with high-dimensional random inputs. Before presenting the individual examples, we describe the computational domain, that is, spatial domain $D = [0, 1]^2$. The time interval is from 0 to 1.

Let p_1^{MR} and p_2^{MR} be the solutions of the model reduced methods, and $(p_{1,h}, p_{2,h})$ be the reference solutions, which are solved by FEM on a fine grid. Then the relative mean errors in the weighted L^2 norm are defined by

$$e_j = \frac{1}{N} \sum_{i=1}^N \frac{\|p_j^{MR}(x; \mu_i) - p_j^h(x; \mu_i)\|_{L^2}}{\|p_j^h(x; \mu_i)\|_{L^2}}, \quad (21)$$

where $j = 1, 2$, $\|v\|_{L^2}^2 = \int_D v^2 dx$, and N is the number of random samples.

6.1. Experiment 1

To demonstrate the efficacy of the hybrid model reduction method, we first consider the dual-continuum with random inputs as follows.

$$\begin{cases} \frac{\partial p_1}{\partial t} - \operatorname{div}(\kappa_1(x; \mu) \nabla p_1) + \sigma(x; \mu)(p_1 - p_2) = f_1(t, x; \mu), \\ \frac{\partial p_2}{\partial t} - \operatorname{div}(\kappa_2(x; \mu) \nabla p_2) + \sigma(x; \mu)(p_2 - p_1) = f_2(t, x; \mu), \end{cases} \quad (22)$$

where the diffusion coefficients, the transfer coefficient, and the source terms are defined by

$$\begin{cases} \kappa_1(x; \mu) = (1 + \mu^2)(1 + x_1)(1 + x_2) + (1 + \cos(\mu) + \mu^4)\kappa_{12}(x), \\ \kappa_2(x; \mu) = (1 + 2\mu^2)(1 + x_1^2)(1 + x_2^2) + (1 + \sin(\mu) + \mu^2)\kappa_{22}(x), \\ \sigma(x; \mu) = (1 + \mu)(1 + x_1) + (1 + \mu^2)x_1^2x_2^2, \\ f_1(t, x; \mu) = 2\mu^2(1 - x_1 - x_2 - t) + 2\mu^4t^2x_1^2x_2^2(1 - x_1 - x_2 - t) \\ \quad - 2\mu^2(1 + \mu^2)\left((1 + 2x_1^3)(1 + x_2^4)t + (1 + x_1)(1 + 2x_2 + t)\right) \\ f_2(t, x; \mu) = 0. \end{cases}$$

Here $x = (x_1, x_2) \in D$ and the random variable μ obeys the beta distribution, i.e., $\mu \sim \text{Beta}(1, 1)$. $\kappa_{12}(x)$ and $\kappa_{22}(x)$ are both high-contrast functions independent of μ , whose maps are depicted in Figure 2.

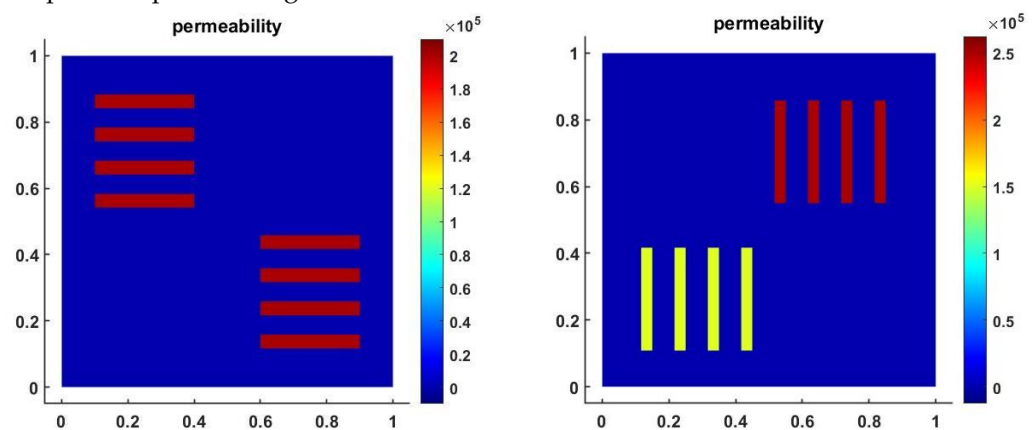


Figure 2. High-contrast captions are duplicated. Please confirm. functions $\kappa_{12}(x)$ (left) and $\kappa_{22}(x)$ (right).

The boundary conditions and initial condition are set as

$$\begin{cases} g_1(x) = x_1^2 + x_2^2, \\ g_2(x) = (x - 1)_1^2 + (x - 1)_2^2, \\ p_1(x, t_0) = x_1, \\ p_2(x, t_0) = x_2, \end{cases}$$

respectively. The partition of spatial domain is defined on 120×120 uniform grid to compute the reference solution. The model reduction solutions are computed on 10×10 coarse mesh. The time step is $\Delta t = 0.02$ up to the final simulation time $T = 1$. For offline computation in this example, we randomly choose 200 random samples for the training set Ξ and select 5 optimal samples. Figure 3 shows the mean profiles of the solutions $p_i(x, t; \mu)$, $i = 1, 2$ for the reference, GMsFEM and hybrid model reduction methods. The results demonstrate that the mean profiles from both reduction methods are nearly identical to the reference.

To evaluate the performance of the hybrid model reduction method, we compute the average relative L^2 error over an ensemble of 1000 randomly selected parameter samples.

The numerical setup is configured as follows: the number of local basis functions per coarse element is fixed at $L = 5$, and the number of variable-separation terms is also set to five. To systematically investigate the impact of the coarse mesh resolution on the approximation accuracy, we conduct a series of experiments with varying coarse mesh sizes, specifically $H = \{1/2, 1/4, 1/8, 1/10, 1/12\}$. The relative L^2 errors of variables $p_1(x; \mu)$ and $p_2(x; \mu)$ by different methods are listed in Table 1.

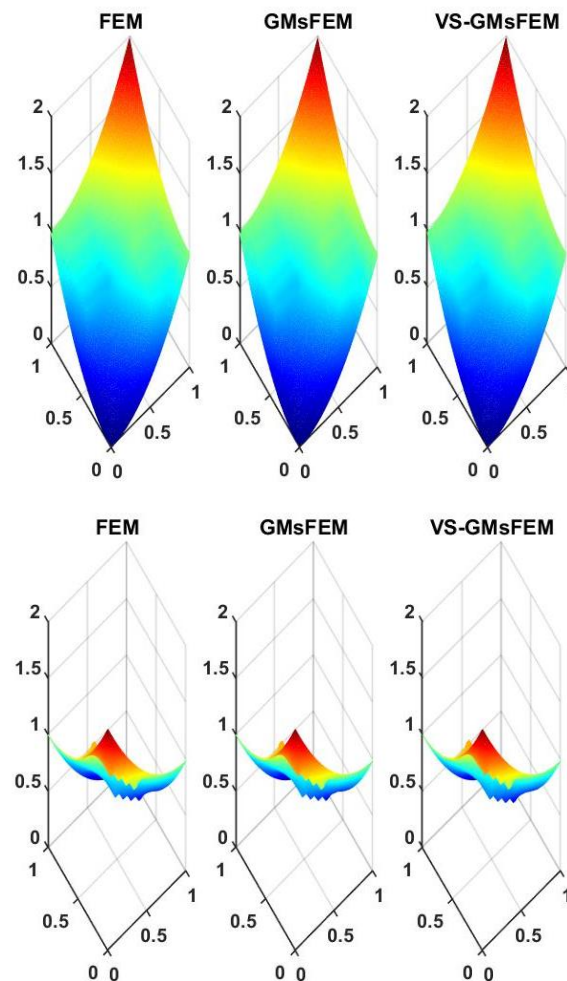


Figure 3. Figures of $p_1(t, x; \mu)$ (1st row) and $p_2(t, x; \mu)$ (2nd row) by different methods.

Table 1. The relative L^2 errors with different coarse mesh size H for $p_1(x; \mu)$ and $p_2(x; \mu)$.

Coarse Mesh Size	$H = 1/2$	$H = 1/4$	$H = 1/8$	$H = 1/10$	$H = 1/12$	$H = 1/15$
$e_{p_1}^{fg}$	5.0389×10^{-1}	1.2556×10^{-1}	1.3176×10^{-1}	3.2681×10^{-3}	3.4457×10^{-3}	1.2881×10^{-3}
$e_{p_1}^{fv}$	5.0337×10^{-1}	1.2563×10^{-1}	1.3191×10^{-1}	4.6316×10^{-3}	4.1590×10^{-2}	5.9059×10^{-3}
$e_{p_2}^{fg}$	2.2118×10^{-1}	1.3833×10^{-1}	8.7244×10^{-2}	2.5336×10^{-3}	1.8137×10^{-3}	1.4234×10^{-3}
$e_{p_2}^{fv}$	2.2208×10^{-1}	1.4099×10^{-1}	8.6095×10^{-2}	7.0355×10^{-3}	8.9305×10^{-3}	5.4730×10^{-3}

The results in the table indicate that: (1) the approximation accuracy of both the hybrid method and GMSFEM improves—that is, the errors for $p_1(t, x; \mu)$ and $p_2(t, x; \mu)$ decays as the coarse grid is refined; and (2) when compared with GMSFEM, the errors of the reduced-order model are of the same order of magnitude as those of the reference solutions.

To visualize the relative L^2 errors of variables $p_1(t, x; \mu)$ and $p_2(t, x; \mu)$, we plot the time evolution of these errors for two randomly selected samples in Figure 4.

For both the GMSFEM and the hybrid model reduction method, the relative errors tend to plateau as the number of time layers increase.

To compare the computational efficiency and accuracy, we describe the computation setting and list the CPU time and L^2 errors for the variables in Table 2 when we use the FE, GMSFEM and hybrid model reduction methods.

From the table, we see that: (1) when compared with standard FE method in fine grid, both the GMsFEM and hybrid model reduction method can significantly improve the efficiency in online stage; (2) although the accuracy of the hybrid model reduction method is worse than GMsFEM, the computation efficiency of hybrid model reduction method is much better than GMsFEM.

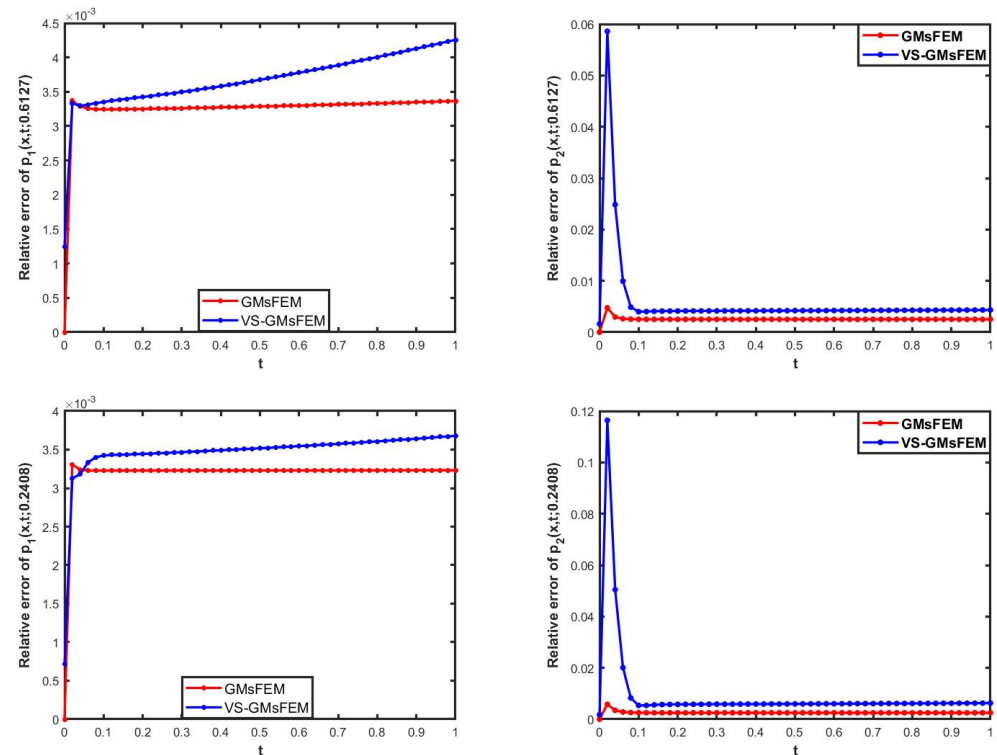


Figure 4. The average relative errors corresponding to the time t .

Table 2. Comparison of the CPU time for dual-continuum model with FEM in fine grid and hybrid model reduction method.

Computational Setting		FEM	GMsFEM	Hybrid Reduced Model
Number of online basis functions		14,641	5	5
Offline stage:	Total time for snapshot space	/	1.007×10^1 s	1.007×10^1 s
	Total time for basis functions	/	/	2.583×10^1 s
Online stage:	Average time for each optimal solution	2.834 s	4.325 s	4.971×10^{-2} s
Approximation accuracy of $p_1(x, t; \mu)$		/	3.268×10^{-3} s	5.396×10^{-2} s
Approximation accuracy of $p_2(x, t; \mu)$		/	2.534×10^{-3} s	1.908×10^{-2} s

6.2. Experiment 2

In this subsection, we consider the dual-continuum with random inputs as follows.

$$\begin{cases} \frac{\partial p_1}{\partial t} - \text{div}(\kappa_1(x; \mu) \nabla p_1) + \sigma(x; \mu)(p_1 - p_2) = f_1(t, x; \mu), \\ \frac{\partial p_2}{\partial t} - \text{div}(\kappa_2(x; \mu) \nabla p_2) + \sigma(x; \mu)(p_2 - p_1) = f_2(t, x; \mu), \end{cases} \quad (23)$$

where the diffusion coefficients, the transfer coefficient, and the source terms are defined by

$$\begin{cases} \kappa_1(x; \mu) = (1 + 2\mu^2)(1 + \exp(x_1 + x_2)) + (2 + \sin(\mu))\kappa_{12}(x), \\ \kappa_2(x; \mu) = (2 + \exp(\mu))(2 + \sin(2x_1x_2)) + (1 + \sin(2\mu) + \mu^2)\kappa_{22}(x), \\ \sigma(x; \mu) = (1 + 2\mu)(1 + x_1) + (3 + \mu^2)x_1^2x_2^2, \\ f_1(t, x; \mu) = (\mu + \mu^2)(1 + x_1 + x_2 + t) + (\mu^4 - \mu^2)t^2x_1^2x_2^2(1 + x_1 + x_2 + t), \\ f_2(t, x; \mu) = (\mu - \mu^3)(x_1 + x_2 + 2) + (\mu^2 + \mu^4)2x_1x_2(x_1 + x_2 + 2). \end{cases}$$

Here $x = (x_1, x_2) \in D$ and the random variable μ obeys the beta distribution, i.e., $\mu \sim \text{Beta}(2, 2)$. $\kappa_{12}(x)$ and $\kappa_{22}(x)$ are both high-contrast functions independent of μ , whose maps are depicted in Figure 5.

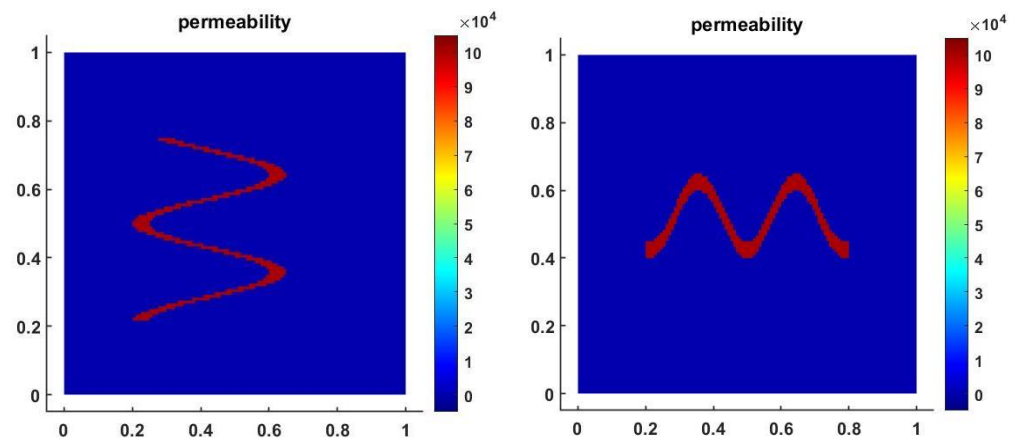


Figure 5. High-contrast functions $\kappa_{12}(x)$ (left) and $\kappa_{22}(x)$ (right).

The conductivity values in the background both are fixed to be 1, while the conductivity values in the channels are 10,000. In this experiment, we randomly take 2000 random samples to test the proposed iterative method. The boundary conditions and initial conditions are set as

$$\begin{cases} g_1(x) = 2 - x_1^2 - x_2^2, \\ g_2(x) = 2 + x_1 + x_2, \\ p_1(x, t_0) = 2 - x_1^2 - x_2^2, \\ p_2(x, t_0) = 2 + x_1 + x_2, \end{cases}$$

respectively. In this experiment, the solutions of the multi-continuum model for the three methods: FEM on 120×120 fine grid (reference solutions), GMsFEM on 12×12 coarse mesh size and the proposed model reduction method.

To evaluate the approximation accuracy for the hybrid model reduction method, we randomly selected 2000 parameter samples and compute the average relative L^2 -errors. Using a fixed number of separated terms ($N = 10$) and five local basis functions in GMsFEM, the mean and the variance of variables $p_1(t, x; \mu)$ and $p_2(t, x; \mu)$ are illustrated at a random time layer in Figures 6 and 7, respectively.

As shown in Figures 6 and 7, the mean and variance of $p_1(t, x; \mu)$ and $p_2(t, x; \mu)$ are in excellent agreement with the reference solutions. This strong correlation not only validates the accuracy of the proposed reduced model but also confirms the reliability of the solution obtained via the hybrid model reduction method.

To systematically assess the impact of the coarse mesh resolution on accuracy, we conduct experiments using progressively refined meshes with sizes $H = \{1/2, 1/4, 1/8, 1/12, 1/24\}$. We compare the relative L^2 errors of all methods in Table 3.

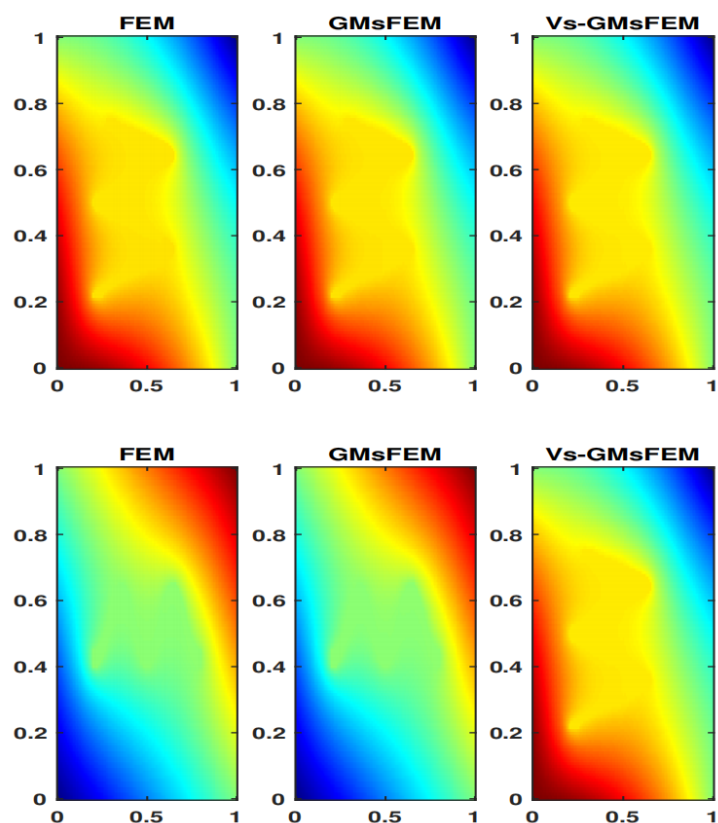


Figure 6. The mean of $p_1(t, x; \mu)$ (1st row) and $p_2(t, x; \mu)$ (2nd row) obtained via different methods.

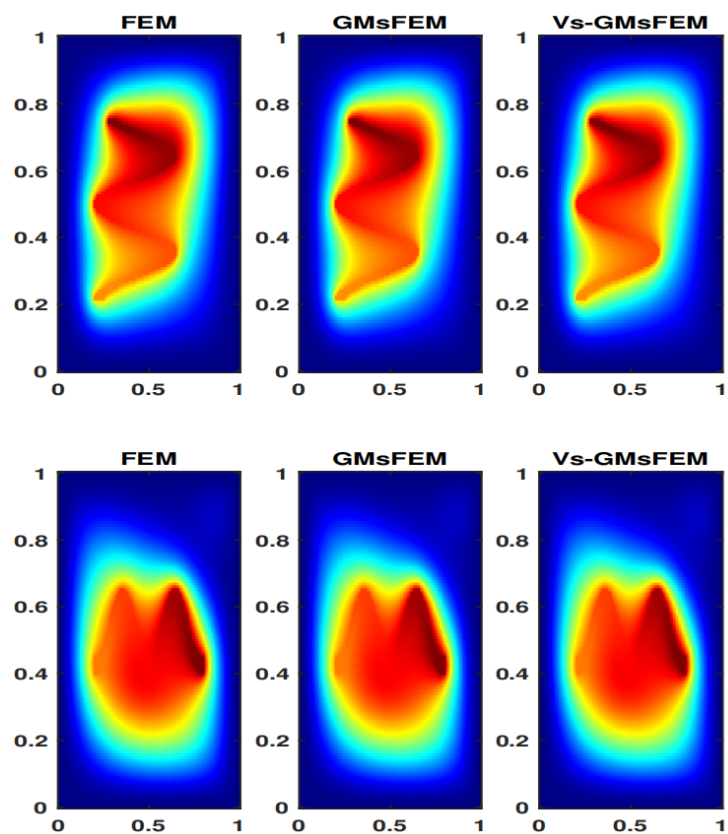


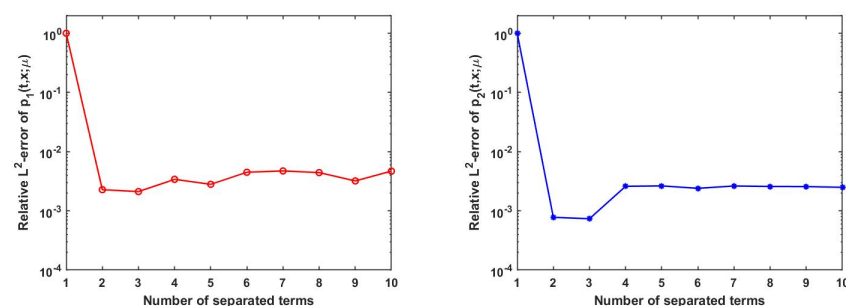
Figure 7. The variance of $p_1(t, x; \mu)$ (1st row) and $p_2(t, x; \mu)$ (2nd row) obtained via different methods.

Table 3. The relative L^2 errors with different coarse mesh size H for $p_1(t, x; \mu)$ and $p_2(t, x; \mu)$.

Coarse Mesh Size	$H = 1/2$	$H = 1/4$	$H = 1/8$	$H = 1/12$	$H = 1/24$
$e_{p_1}^{fs}$	1.8958×10^{-1}	1.7748×10^{-1}	2.0568×10^{-3}	1.2338×10^{-3}	5.3932×10^{-4}
$e_{p_1}^{fv}$	1.9047×10^{-1}	1.7539×10^{-1}	4.2623×10^{-3}	1.4946×10^{-2}	9.2871×10^{-3}
$e_{p_1}^{sv}$	1.7631×10^{-3}	1.9731×10^{-3}	3.5753×10^{-3}	1.4759×10^{-2}	9.2977×10^{-3}
$e_{p_2}^{fs}$	2.1234×10^{-1}	1.8409×10^{-2}	6.5206×10^{-4}	3.7393×10^{-4}	1.4397×10^{-4}
$e_{p_2}^{fv}$	2.1175×10^{-1}	1.8218×10^{-2}	1.4602×10^{-3}	2.0708×10^{-3}	1.4111×10^{-3}
$e_{p_2}^{sv}$	9.7625×10^{-4}	5.3918×10^{-4}	1.3870×10^{-3}	2.0860×10^{-3}	1.3949×10^{-3}

Similar to the results presented in the Table 1, Table 3 demonstrates that the approximation accuracy of both the hybrid method and GMsFEM improves—that is, their respective errors decrease—as the coarse mesh is refined.

In Figure 8, we depict the average relative L^2 -error versus number of the separated terms N for the hybrid model reduction method based on 2000 random samples.

**Figure 8.** The average relative L^2 -error of numerical solutions: $p_1(t, x; \mu)$ (left) and $p_2(t, x; \mu)$ (right) with different number of separated terms.

From the figure, we see that the average relative L^2 -error decays when the number of separated terms N increases. As the number of separated terms grows, it eventually stabilizes upon further growth of N .

7. Conclusions

In this paper, we present a hybrid model reduction method for solving stochastic dual-continuum models. In this work, the possible uncertainties considered originate in the PDE coefficient, the transfer term, and the source term. Direct fine-grid solutions of these equations are computationally challenging, especially in many-query scenarios. To dramatically improve efficiency, our method integrates the uncoupled Generalized Multiscale Finite Element Method (GMsFEM) with a low-rank variable-separation technique to construct compact representations of the solutions. This hybrid approach enables the computation of a large number of stochastic realizations at minimal cost. In the numerical experiments section, we apply the proposed model reduction method to dual-continuum models with random inputs. The results confirm that our hybrid method achieves an effective trade-off between computational efficiency and accuracy.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 12101217) and the Natural Science Foundation of Hunan Province (Grant No. 2022JJ40113).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The author declare no conflicts of interest

Appendix A. Online–Offline Computation of Residuals

Based on affine representations (3) and definition of residuals, we can express the residuals as

$$\begin{aligned}
 r_{p_1}^k(v_1; \mu) &= s_1(v_1; \mu) - \left(\frac{\partial p_{1,k-1}}{\partial t}, v_1; \mu \right) - a_1(p_{1,k-1}, v_1; \mu) - q(p_{1,k-1} - p_{2,k-1}, v_1; \mu) \\
 &= \sum_{n=1}^{N_a} X_n(\mu) s_{1,n}(v_1) - \sum_{i=1}^{k-1} \xi_i(\mu) \left(\frac{\partial P_{1,i}(t, x)}{\partial t}, v_1 \right) \\
 &\quad - \sum_{s=1}^{Q_a} \sum_{i=1}^{k-1} Q_1^s(\mu) \xi_i(\mu) a_1^s(P_{1,i}(t, x), v_1) \\
 &\quad - \sum_{i=1}^{k-1} \xi_i(\mu) q(P_{1,i}(t, x), v_1) + \sum_{j=1}^{k-1} \lambda_j(\mu) q(P_{2,j}(t, x), v_1), \\
 r_{p_2}^k(v_2; \mu) &= s_2(v_2; \mu) - \left(\frac{\partial p_{2,k-1}}{\partial t}, v_2; \mu \right) - a_2(p_{2,k-1}, v_2; \mu) - q(p_{2,k-1} - p_{1,k-1}, v_2; \mu) \\
 &= \sum_{m=1}^{M_a} Y_m(\mu) s_{2,m}(v_2) - \sum_{j=1}^{k-1} \lambda_j(\mu) \left(\frac{\partial P_{2,j}(t, x)}{\partial t}, v_2 \right) \\
 &\quad - \sum_{t=1}^{Q_b} \sum_{j=1}^{k-1} Q_2^t(\mu) \lambda_j(\mu) a_2^t(P_{2,j}(t, x), v_2) \\
 &\quad - \sum_{j=1}^{k-1} \lambda_j(\mu) q(P_{2,j}(t, x), v_2) + \sum_{i=1}^{k-1} \xi_i(\mu) q(P_{1,i}(t, x), v_2).
 \end{aligned} \tag{A1}$$

Based on (13) and the above expressions about residuals, we can get

$$\begin{aligned}
 (\hat{e}_{p_1}^k(\mu), v_1; \gamma)_{\mathcal{H}^1} &= \sum_{n=1}^{N_a} X_n(\mu) s_{1,n}(v_1) - \sum_{i=1}^{k-1} \xi_i(\mu) \left(\frac{\partial P_{1,i}(t, x)}{\partial t}, v_1 \right) \\
 &\quad - \sum_{s=1}^{Q_a} \sum_{i=1}^{k-1} Q_1^s(\mu) \xi_i(\mu) a_1^s(P_{1,i}(t, x), v_1) \\
 &\quad - \sum_{i=1}^{k-1} \xi_i(\mu) q(P_{1,i}(t, x), v_1) + \sum_{j=1}^{k-1} \lambda_j(\mu) q(P_{2,j}(t, x), v_1) \\
 (\hat{e}_{p_2}^k(\mu), v_1; \gamma)_{\mathcal{H}^1} &= \sum_{m=1}^{M_a} Y_m(\mu) s_{2,m}(v_2) - \sum_{j=1}^{k-1} \lambda_j(\mu) \left(\frac{\partial P_{2,j}(t, x)}{\partial t}, v_2 \right) \\
 &\quad - \sum_{t=1}^{Q_b} \sum_{j=1}^{k-1} Q_2^t(\mu) \lambda_j(\mu) a_2^t(P_{2,j}(t, x), v_2) \\
 &\quad - \sum_{j=1}^{k-1} \lambda_j(\mu) q(P_{2,j}(t, x), v_2) + \sum_{i=1}^{k-1} \xi_i(\mu) q(P_{1,i}(t, x), v_2).
 \end{aligned}$$

This implies that

$$\left\{ \begin{aligned}
 \hat{e}_{p_1}^k(\mu) &= \sum_{n=1}^{N_a} X_n(\mu) \mathcal{C}_{p_1}^n + \sum_{i=1}^{k-1} \xi_i(\mu) \mathcal{D}_{p_1}^i + \sum_{s=1}^{Q_a} \sum_{i=1}^{k-1} Q_1^s(\mu) \xi_i(\mu) \mathcal{L}_i^s \\
 &\quad + \sum_{i=1}^{k-1} \xi_i(\mu) \mathcal{G}_{p_1}^i + \sum_{j=1}^{k-1} \lambda_j(\mu) \mathcal{G}_{p_2}^j \\
 \hat{e}_{p_2}^k(\mu) &= \sum_{m=1}^{M_a} Y_m(\mu) \mathcal{C}_{p_2}^m + \sum_{i=1}^{k-1} \lambda_i(\mu) \mathcal{D}_{p_2}^i + \sum_{t=1}^{Q_b} \sum_{j=1}^{k-1} Q_2^t(\mu) \lambda_j(\mu) \mathcal{L}_j^t \\
 &\quad + \sum_{j=1}^{k-1} \lambda_j(\mu) \mathcal{G}_{p_2}^j + \sum_{i=1}^{k-1} \xi_i(\mu) \mathcal{G}_{p_1}^i.
 \end{aligned} \right. \tag{A2}$$

where $\mathcal{C}_{p_1}^n$ is the Riesz representation of $s_{1,n}(v_1)$, i.e., $(\mathcal{C}_{p_1}^n, v_1)_{\mathcal{H}^1} = s_{1,n}(v_1)$ for any $v_1 \in \mathcal{V}_h(\Omega)$, $1 \leq n \leq N_a$. $\mathcal{C}_{p_2}^m$ is the Riesz representation of $s_{2,m}(v_2)$, i.e., $(\mathcal{C}_{p_2}^m, v_2)_{\mathcal{H}^1} = s_{2,m}(v_2)$ for any $v_2 \in \mathcal{V}_h(\Omega)$, $1 \leq m \leq N_b$. $\mathcal{D}_{p_j}^i$ is the Riesz representation of $-\left(\frac{\partial P_{j,i}(t,x)}{\partial t}, v_j\right)$, i.e., $(\mathcal{D}_{p_j}^i, v_j)_{\mathcal{H}^1} = -\left(\frac{\partial P_{j,i}(t,x)}{\partial t}, v_j\right)$ for any $v_j \in \mathcal{V}_h(\Omega)$, where $1 \leq i \leq k-1$. $\mathcal{L}_i^s$ is the Riesz representation of $-a^s(P_{1,i}(t,x), v_1)$, i.e., $(\mathcal{L}_i^s, v_1)_{\mathcal{H}^1} = -a_1^s(P_{1,i}(t,x), v_1)$ for any $v_1 \in \mathcal{V}_h$, where $1 \leq i \leq k-1$ and $1 \leq s \leq Q_a$. $\mathcal{L}_i^t$ is the Riesz representation of $-a^t(P_{2,i}(t,x), v_2)$, i.e., $(\mathcal{L}_i^t, v_2)_{\mathcal{H}^1} = -a^t(P_{2,i}(t,x), v_2)$ for any $v_2 \in \mathcal{V}_h$, where $1 \leq i \leq k-1$ and $1 \leq t \leq Q_b$. $\mathcal{G}_{p_1}^j$ is the Riesz representation of $q(P_{1,i}(t,x), v_1)$, i.e., $(\mathcal{G}_{p_1}^j, v_s)_{\mathcal{H}^1} = -q(P_{1,i}(t,x), v_s)$ for any $\tilde{u} \in \mathcal{V}_h(\Omega)$, where $1 \leq j \leq k-1$, $s = 1, 2$.

Equation (A2) gives rise to

$$\begin{aligned} \|\hat{e}_{p_1}^k(\mu)\|_{\mathcal{H}^1}^2 &= \sum_{n=1}^{N_a} \sum_{n'=1}^{N_a} X_n(\mu) X_{n'}(\mu) (\mathcal{C}_{p_1}^n, \mathcal{C}_{p_1}^{n'})_{\mathcal{H}^1} + \sum_{i=1}^{k-1} \sum_{i'=1}^{k-1} \xi_i(\mu) \xi_{i'}(\mu) (\mathcal{D}_{p_1}^i, \mathcal{D}_{p_1}^{i'})_{\mathcal{H}^1} \\ &+ \sum_{i=1}^{k-1} \sum_{i'=1}^{k-1} \sum_{s=1}^{Q_a} \sum_{s'=1}^{Q_a} \xi_i(\mu) \xi_{i'}(\mu) Q_1^s(\mu) Q_1^{s'}(\mu) (\mathcal{L}_i^s, \mathcal{L}_{i'}^{s'})_{\mathcal{H}^1} + \sum_{i=1}^{k-1} \sum_{i'=1}^{k-1} \xi_i(\mu) \xi_{i'}(\mu) (\mathcal{G}_{p_1}^i, \mathcal{G}_{p_1}^{i'})_{\mathcal{H}^1} \\ &+ \sum_{j=1}^{k-1} \sum_{j'=1}^{k-1} \lambda_j(\mu) \lambda_{j'}(\mu) (\mathcal{G}_{p_2}^j, \mathcal{G}_{p_2}^{j'})_{\mathcal{H}^1} + 2 \sum_{n=1}^{N_a} \sum_{i=1}^{k-1} X_n(\mu) \xi_i(\mu) (\mathcal{C}_{p_1}^n, \mathcal{D}_{p_1}^i)_{\mathcal{H}^1}, \\ &+ 2 \sum_{n=1}^{N_a} \sum_{s=1}^{Q_a} \sum_{i=1}^{k-1} X_n(\mu) Q_1^s(\mu) \xi_i(\mu) (\mathcal{C}_{p_1}^n, \mathcal{L}_i^s)_{\mathcal{H}^1} + 2 \sum_{n=1}^{N_a} \sum_{i=1}^{k-1} X_n(\mu) \xi_i(\mu) (\mathcal{C}_{p_1}^n, \mathcal{G}_{p_1}^i)_{\mathcal{H}^1} \\ &+ 2 \sum_{n=1}^{N_a} \sum_{j=1}^{k-1} X_n(\mu) \lambda_j(\mu) (\mathcal{C}_{p_1}^n, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1} + 2 \sum_{i=1}^{k-1} \sum_{s=1}^{Q_a} \sum_{i'=1}^{k-1} \xi_i(\mu) Q_1^s(\mu) \xi_{i'}(\mu) (\mathcal{D}_{p_1}^i, \mathcal{L}_{i'}^s)_{\mathcal{H}^1} \\ &+ 2 \sum_{i=1}^{k-1} \sum_{i'=1}^{k-1} \xi_i(\mu) \xi_{i'}(\mu) (\mathcal{G}_{p_1}^i, \mathcal{G}_{p_1}^{i'})_{\mathcal{H}^1} + 2 \sum_{i=1}^{k-1} \sum_{j=1}^{k-1} \xi_i(\mu) \lambda_j(\mu) (\mathcal{G}_{p_1}^i, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1} \\ &+ 2 \sum_{i=1}^{k-1} \sum_{s=1}^{Q_a} \sum_{i'=1}^{k-1} \xi_i(\mu) Q_1^s(\mu) \xi_{i'}(\mu) (\mathcal{L}_i^s, \mathcal{G}_{p_1}^{i'})_{\mathcal{H}^1} + 2 \sum_{i=1}^{k-1} \sum_{s=1}^{Q_a} \sum_{j=1}^{k-1} Q_1^s(\mu) \xi_i(\mu) \lambda_j(\mu) (\mathcal{L}_i^s, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1} \\ &+ 2 \sum_{i=1}^{k-1} \sum_{j=1}^{k-1} \xi_i(\mu) \lambda_j(\mu) (\mathcal{G}_{p_1}^i, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1}, \end{aligned}$$

and

$$\begin{aligned} \|\hat{e}_{p_2}^k(\mu)\|_{\mathcal{H}^1}^2 &= \sum_{m=1}^{M_a} \sum_{m'=1}^{M_a} Y_m(\mu) Y_{m'}(\mu) (\mathcal{C}_{p_2}^m, \mathcal{C}_{p_2}^{m'})_{\mathcal{H}^1} + \sum_{i=1}^{k-1} \sum_{i'=1}^{k-1} \lambda_i(\mu) \lambda_{i'}(\mu) (\mathcal{D}_{p_2}^i, \mathcal{D}_{p_2}^{i'})_{\mathcal{H}^1} \\ &+ \sum_{j=1}^{k-1} \sum_{j'=1}^{k-1} \sum_{t=1}^{Q_b} \sum_{t'=1}^{Q_b} \lambda_j(\mu) \lambda_{j'}(\mu) Q_2^t(\mu) Q_2^{t'}(\mu) (\mathcal{L}_j^t, \mathcal{L}_{j'}^{t'})_{\mathcal{H}^1} + \sum_{i=1}^{k-1} \sum_{i'=1}^{k-1} \xi_i(\mu) \xi_{i'}(\mu) (\mathcal{G}_{p_1}^i, \mathcal{G}_{p_1}^{i'})_{\mathcal{H}^1} \\ &+ \sum_{j=1}^{k-1} \sum_{j'=1}^{k-1} \lambda_j(\mu) \lambda_{j'}(\mu) (\mathcal{G}_{p_2}^j, \mathcal{G}_{p_2}^{j'})_{\mathcal{H}^1} + 2 \sum_{m=1}^{M_a} \sum_{i=1}^{k-1} Y_m(\mu) \lambda_i(\mu) (\mathcal{C}_{p_2}^m, \mathcal{D}_{p_2}^i)_{\mathcal{H}^1}, \\ &+ 2 \sum_{m=1}^{M_a} \sum_{t=1}^{Q_b} \sum_{j=1}^{k-1} Y_m(\mu) Q_2^t(\mu) \lambda_j(\mu) (\mathcal{C}_{p_2}^m, \mathcal{L}_j^t)_{\mathcal{H}^1} + 2 \sum_{m=1}^{M_a} \sum_{j=1}^{k-1} Y_m(\mu) \lambda_j(\mu) (\mathcal{C}_{p_2}^m, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1} \\ &+ 2 \sum_{m=1}^{M_a} \sum_{i=1}^{k-1} Y_m(\mu) \xi_i(\mu) (\mathcal{C}_{p_2}^m, \mathcal{G}_{p_1}^i)_{\mathcal{H}^1} + 2 \sum_{i=1}^{k-1} \sum_{t=1}^{Q_b} \sum_{j=1}^{k-1} \lambda_i(\mu) Q_2^t(\mu) \lambda_j(\mu) (\mathcal{D}_{p_2}^i, \mathcal{L}_j^t)_{\mathcal{H}^1} \\ &+ 2 \sum_{i=1}^{k-1} \sum_{j=1}^{k-1} \lambda_i(\mu) \xi_j(\mu) (\mathcal{D}_{p_2}^i, \mathcal{G}_{p_1}^j)_{\mathcal{H}^1} + 2 \sum_{i=1}^{k-1} \sum_{j=1}^{k-1} \lambda_i(\mu) \lambda_j(\mu) (\mathcal{D}_{p_2}^i, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1} \\ &+ 2 \sum_{i=1}^{k-1} \sum_{t=1}^{Q_b} \sum_{j=1}^{k-1} \lambda_i(\mu) Q_2^t(\mu) \lambda_j(\mu) (\mathcal{L}_i^t, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1} + 2 \sum_{i=1}^{k-1} \sum_{t=1}^{Q_b} \sum_{j=1}^{k-1} \lambda_i(\mu) \xi_j(\mu) Q_2^t(\mu) (\mathcal{L}_i^t, \mathcal{G}_{p_1}^j)_{\mathcal{H}^1} \\ &+ 2 \sum_{i=1}^{k-1} \sum_{j=1}^{k-1} \xi_i(\mu) \lambda_j(\mu) (\mathcal{G}_{p_1}^i, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1}. \end{aligned}$$

$\|\hat{e}_{p_1}^k(\mu)\|_{\mathcal{H}^1}$ and $\|\hat{e}_{p_2}^k(\mu)\|_{\mathcal{H}^1}$ can be efficiently computed when offline–online procedure is applied. In the offline stage, we compute $\mathcal{C}_{p_1}^n, \mathcal{D}_{p_1}^i, \mathcal{L}_i^s, \mathcal{G}_{p_1}^i, \mathcal{G}_{p_2}^i, \mathcal{C}_{p_2}^m, \mathcal{D}_{p_2}^i, \mathcal{L}_i^t$ and store the quantities independent with stochastic inputs. We store $(\mathcal{C}_{p_1}^n, \mathcal{C}_{p_1}^{n'})_{\mathcal{H}^1}, (\mathcal{D}_{p_1}^i, \mathcal{D}_{p_1}^{i'})_{\mathcal{H}^1}, (\mathcal{L}_i^s, \mathcal{L}_{i'}^s)_{\mathcal{H}^1}, (\mathcal{G}_{p_1}^i, \mathcal{G}_{p_1}^{i'})_{\mathcal{H}^1}, (\mathcal{G}_{p_2}^j, \mathcal{G}_{p_2}^{j'})_{\mathcal{H}^1}, (\mathcal{C}_{p_1}^n, \mathcal{D}_{p_1}^i)_{\mathcal{H}^1}, (\mathcal{C}_{p_1}^n, \mathcal{L}_i^s)_{\mathcal{H}^1}, (\mathcal{C}_{p_1}^n, \mathcal{G}_{p_1}^i)_{\mathcal{H}^1}, (\mathcal{C}_{p_1}^n, \mathcal{G}_{p_2}^i)_{\mathcal{H}^1}, (\mathcal{D}_{p_1}^i, \mathcal{L}_{i'}^s)_{\mathcal{H}^1}, (\mathcal{G}_{p_1}^i, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1}, (\mathcal{L}_i^s, \mathcal{G}_{p_1}^{i'})_{\mathcal{H}^1}, (\mathcal{L}_i^s, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1}, (\mathcal{C}_{p_2}^m, \mathcal{C}_{p_2}^{m'})_{\mathcal{H}^1}, (\mathcal{D}_{p_2}^i, \mathcal{D}_{p_2}^{i'})_{\mathcal{H}^1}, (\mathcal{L}_j^t, \mathcal{L}_{j'}^t)_{\mathcal{H}^1}, (\mathcal{C}_{p_2}^m, \mathcal{D}_{p_2}^i)_{\mathcal{H}^1}, (\mathcal{C}_{p_2}^m, \mathcal{L}_j^t)_{\mathcal{H}^1}, (\mathcal{C}_{p_2}^m, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1}, (\mathcal{C}_{p_2}^m, \mathcal{G}_{p_1}^i)_{\mathcal{H}^1}, (\mathcal{D}_{p_2}^j, \mathcal{L}_j^t)_{\mathcal{H}^1}, (\mathcal{D}_{p_2}^j, \mathcal{G}_{p_1}^i)_{\mathcal{H}^1}, (\mathcal{D}_{p_2}^j, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1}, (\mathcal{L}_i^t, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1}, (\mathcal{L}_i^t, \mathcal{G}_{p_1}^i)_{\mathcal{H}^1}$ and $(\mathcal{G}_{p_1}^i, \mathcal{G}_{p_2}^j)_{\mathcal{H}^1}$ for the online stage, where $1 \leq n, n' \leq N_a, 1 \leq m, m' \leq M_a, 1 \leq i, i', j, j' \leq k-1, 1 \leq s, s' \leq Q_a$ and $1 \leq t, t' \leq Q_b$. In the online stage, we only need to compute the stochastic coefficient $X_n(\mu), \xi_i(\mu), \lambda(\mu), Q_1^s(\mu), Y^m(\mu), Q_2^t(\mu)$ and $\|\hat{e}_{p_1}^k(\mu)\|_{\mathcal{H}^1}, \|\hat{e}_{p_2}^k(\mu)\|_{\mathcal{H}^1}$ for any given μ .

References

1. Eikemo, B.; Lie, K.; Eigestad, G.; Dahle, H. Discontinuous Galerkin methods for advective transport in single-continuum models of fractured media. *Adv. Water Resour.* **2009**, *32*, 493–506. [\[CrossRef\]](#)
2. Erhel, J.; Dreuzy, J.; Poirriez, B. Flow simulation in three-dimensional discrete fracture networks. *SIAM J. Sci. Comput.* **2009**, *31*, 2688–2705. [\[CrossRef\]](#)
3. Bogdanov, I.; Mourzenko, V.; Thovert, J.; Adler, P. Two-phase flow through fractured porous media. *Phys. Rev. E* **2003**, *68*, 026703. [\[CrossRef\]](#) [\[PubMed\]](#)
4. Granet, S.; Fabrie, P.; Lemonnier, P.; Quintard, M. A two-phase flow simulation of a fractured reservoir using a new fissure element method. *J. Pet. Sci. Eng.* **2001**, *32*, 35–52. [\[CrossRef\]](#)
5. Geiger, S.; Matthai, S.; Niessner, J.; Helmig, R. Black-Oil simulations for three-component, three-phase flow in fractured porous media. *SPE J.* **2009**, *14*, 338–354. [\[CrossRef\]](#)
6. Matthai, S.; Mezentsev, A.; Belayneh, M. Finite element-node-centered finite-volume two-phase-flow experiments with fractured rock represented by unstructured hybrid-element meshes. *SPE Reserv. Eval. Eng.* **2007**, *10*, 740–756. [\[CrossRef\]](#)
7. Kalachikova, U.; Ammosov, D. Advancing wave equation analysis in dual-continuum systems: A partial learning approach with discrete empirical interpolation and deep neural networks. *J. Comput. Appl. Math.* **2024**, *443*, 115755. [\[CrossRef\]](#)
8. Arbogast, T.; Douglas, J., Jr.; Hornung, U. Derivation of the double porosity model of single phase flow via homogenization theory. *SIAM J. Math. Anal.* **1990**, *22*, 823–836. [\[CrossRef\]](#)
9. Barenblatt, G.; Zheltov, I.; Kochina, I. Basic concepts in the theory of seepage of homogeneous liquids in fissured rocks [strata]. *J. Appl. Math. Mech.* **1960**, *24*, 1286–1303. [\[CrossRef\]](#)
10. Wu, Y.; Pruess, K. A multiple-porosity method for simulation of naturally fractured petroleum reservoirs. *SPE Reserv. Eng.* **1988**, *3*, 327–336. [\[CrossRef\]](#)
11. Yan, B.; Alfi, M.; Cheng, A.; Cao, Y.; Wang, Y.; Killough, J. General Multi-Porosity simulation for fractured reservoir modeling. *J. Nat. Gas Sci. Eng.* **2016**, *33*, 777–791. [\[CrossRef\]](#)
12. Efendiev, Y.; Hou, T. *Multiscale Finite Element Methods: Theory and Applications*; Springer: New York, NY, USA, 2009.
13. Efendiev, Y.; Hou, T.; Ginting, V. Multiscale finite element methods for nonlinear problems and their applications. *Commun. Math. Sci.* **2004**, *2*, 553–589. [\[CrossRef\]](#)
14. Efendiev, Y.; Galvis, J.; Hou, T. Generalized multiscale finite element methods (GMsFEM). *J. Comput. Phys.* **2013**, *251*, 116–135. [\[CrossRef\]](#)
15. Akkutlu, I.; Efendiev, Y.; Vasilyeva, M.; Wang, Y. Multiscale model reduction for shale gas transport in a coupled discrete fracture and dual-continuum porous media. *J. Nat. Gas Sci. Eng.* **2017**, *48*, 65–76. [\[CrossRef\]](#)
16. Chung, E.; Efendiev, Y.; Leung, W. Constraint energy minimizing generalized multiscale finite element method. *Comput. Methods Appl. Mech. Eng.* **2018**, *339*, 298–319. [\[CrossRef\]](#)
17. Chung, E.; Efendiev, Y.; Leung, W. Fast online generalized multiscale finite element method using constraint energy minimization. *J. Comput. Phys.* **2018**, *355*, 450–463. [\[CrossRef\]](#)
18. Fu, S.; Chung, E.; Mai, T. Constraint energy minimizing generalized multiscale finite element method for nonlinear poroelasticity and elasticity. *J. Pet. Sci. Eng.* **2020**, *417*, 109569. [\[CrossRef\]](#)
19. Cheung, S.; Chung, E.; Efendiev, Y.; Leung, W.; Vasilyeva, M. Constraint energy minimizing generalized multiscale finite element method for dual continuum model. *arXiv* **2018**, arXiv:1807.10955. [\[CrossRef\]](#)
20. Chung, E.; Efendiev, Y.; Leung, W.; Vasilyeva, M.; Wang, Y. Non-local multi-continua upscaling for flows in heterogeneous fractured media. *J. Comput. Phys.* **2018**, *372*, 22–34. [\[CrossRef\]](#)

21. Vasilyeva, M.; Chung, E.; Cheung, S.; Wang, Y.; Prokopev, G. Nonlocal multicontinua upscaling for multicontinua flow problems in fractured porous media. *J. Comput. Appl. Math.* **2019**, *355*, 258–267. [\[CrossRef\]](#)
22. Modesto, D.; Zlotnik, S.; Huerta, A. Proper generalized decomposition for parameterized Helmholtz problems in heterogeneous and unbounded domains: Application to harbor agitation. *Comput. Methods Appl. Mech. Eng.* **2015**, *295*, 127–149. [\[CrossRef\]](#)
23. Pacciarini, P.; Rozza, G. Stabilized reduced basis method for parametrized advection–diffusion PDEs. *Comput. Methods Appl. Mech. Eng.* **2014**, *274*, 1–18. [\[CrossRef\]](#)
24. Li, Q.; Wang, Y.; Maria, V. Multiscale model reduction for fluid infiltration simulation through dual-continuum porous media with localized uncertainties. *J. Comput. Appl. Math.* **2018**, *336*, 127–146. [\[CrossRef\]](#)
25. Nouy, A. Recent developments in spectral stochastic methods for the numerical solution of stochastic partial differential equations. *Arch. Comput. Methods Eng.* **2009**, *16*, 251–285. [\[CrossRef\]](#)
26. Nouy, A.; Le Maitre, O. Generalized spectral decomposition for stochastic nonlinear problems. *J. Comput. Phys.* **2009**, *228*, 202–235. [\[CrossRef\]](#)
27. Nouy, A. Proper generalized decompositions and separated representations for the numerical solution of high dimensional stochastic problems. *Arch. Comput. Methods Eng.* **2010**, *17*, 403–434. [\[CrossRef\]](#)
28. Barrault, M.; Maday, Y.; Nguyen, N.; Patera, A. An empirical interpolation method: Application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathématique* **2004**, *339*, 667–672. [\[CrossRef\]](#)
29. Grepl, M.; Maday, Y.; Nguyen, N.; Patera, A. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM Math. Model. Numer. Anal.* **2007**, *41*, 575–605. [\[CrossRef\]](#)
30. Quarteroni, A.; Manzoni, A.; Negri, F. *Reduced Basis Methods for Partial Differential Equations*; Springer: New York, NY, USA, 2015.
31. Evans, L. *Partial Differential Equations*, 2nd ed.; American Mathematical Society: Providence, RI, USA, 1998.
32. Li, Q.; Jiang, L. A novel variable-separation method based on sparse and low rank representation for stochastic partial differential equations. *SIAM J. Sci. Comput.* **2017**, *39*, A2879–A2910. [\[CrossRef\]](#)
33. Jiang, L.; Li, Q. Reduced multiscale finite element basis methods for elliptic PDEs with parameterized inputs. *J. Comput. Appl. Math.* **2016**, *301*, 101–120. [\[CrossRef\]](#)
34. Rozza, G.; Huynh, D.; Patera, A. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Arch. Comput. Methods Eng.* **2008**, *15*, 229–275. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.