

Two-Step MV-DeepONet: Probabilistic Operator Learning for Uncertainty Propagation Driven by Random Input Fields

Yupei Nie^a Lei Wang^{a,*} Jiasen Liu^b

^a*School of Mathematics and Physics, North China Electric Power University, Beijing 102206, P.R. China*

^b*School of Energy Power and Mechanical Engineering, North China Electric Power University, Beijing 102206, P.R. China*

Abstract:

Forward uncertainty propagation in complex physical systems can induce structured covariance across field-valued outputs. For a probabilistic surrogate, the total predictive covariance comprises the covariance of conditional means across input realizations and the average conditional predictive covariance. Probabilistic DeepONet (Prob-DeepONet) provides lightweight uncertainty quantification by predicting pointwise Gaussian means and variances in a single forward pass, but its conditional predictive covariance is restricted to a diagonal form. To represent cross-location conditional dependence without explicitly parameterizing a full high-dimensional covariance matrix, we develop a two-step mean-variance DeepONet (two-step MV-DeepONet) through two principal modifications. First, two-step training is used to decouple output-basis learning from the input-to-coefficient mapping, together with basis orthogonalization and subspace rotation. Second, Gaussian probabilistic modeling is transferred from the high-dimensional physical output space to the low-dimensional rotated coefficient space. Mapping these probabilistic coefficients through the shared basis induces a generally non-diagonal conditional predictive covariance in the physical output space while retaining single-pass inference. A Frobenius-norm error decomposition and corresponding upper bound identify low-rank covariance compressibility, trunk-subspace approximation, finite-sample statistical error, and coefficient-space covariance estimation as the principal factors governing covariance recovery. Numerical experiments on three representative problems governed by partial differential equations (PDEs) and a hypersonic blunt-body aerothermal problem show improved generalization, more structured uncertainty bands, and accurate recovery of off-diagonal correlation patterns compared with Prob-DeepONet. Conformal calibration further improves empirical interval coverage, while two-step MV-DeepONet maintains sharper prediction intervals.

Keywords: Input uncertainty propagation; Two-step MV-DeepONet; Structured covariance recovery; Probabilistic operator learning; Uncertainty quantification

*Corresponding author: 50901924@ncepu.edu.cn

1 Introduction

Uncertainty quantification is essential for credible modeling and reliable design of complex engineering systems. A standard workflow generally includes uncertainty identification and characterization, uncertainty propagation, and uncertainty analysis [1]. Among these steps, input uncertainty propagation characterizes the variability of model outputs induced by random inputs and is therefore central to engineering design under uncertainty. In complex physical systems, uncertain inputs may arise from material properties, boundary conditions, external loads, and other sources of variability, and can be represented as random variables, stochastic processes, or random fields [2]. For a probabilistic surrogate under random inputs, the total predictive covariance consists of the covariance of the predicted means across input realizations and the average conditional predictive covariance over inputs. The former captures cross-location dependence induced by random inputs [3], while the latter may also contain structured dependence that is not explicitly represented under a diagonal covariance assumption [4].

Classical methods for input uncertainty propagation can be broadly classified into simulation methods, local approximation methods, most probable point methods, functional expansion methods, and numerical integration methods [5]. Among these approaches, polynomial chaos expansion and generalized polynomial chaos are widely used functional expansion methods for uncertainty propagation [6]. These methods can provide efficient approximations for smooth problems with moderate stochastic dimensions. However, their computational cost may increase rapidly with the number of uncertain inputs and the approximation order [7–9]. With the development of data-driven modeling, machine-learning approaches have increasingly been adopted for uncertainty propagation, including analytical approximations through trained neural networks [10, 11], interval-based propagation [12], and offline sampling combined with deterministic surrogates [13–15]. In addition to input-induced uncertainty, surrogate predictions may also be affected by limited or noisy training data, model inadequacy, and variability in the training procedure [16]. Representative approaches for quantifying such predictive uncertainty include Bayesian neural networks [17], Monte Carlo dropout [18], and deep ensembles [19]. Other approaches directly parameterize predictive distributions, including evidential deep learning [20] and mean–variance estimation (MVE) [21]. Among these, MVE provides a lightweight formulation that assumes an input-dependent Gaussian target distribution and jointly predicts its conditional mean and variance by minimizing the Gaussian negative log-likelihood (NLL) [21]. This formulation enables input-dependent predictive variance to be estimated in a single forward pass. However, joint mean–variance optimization may be unstable [22]; a mean-only warm-up with fixed variance can therefore be used to improve optimization stability [23].

Both classical input-uncertainty propagation methods and conventional machine-learning approaches to predictive uncertainty are commonly formulated in finite-dimensional spaces. For physical systems driven by random input fields, however, both the inputs and the corresponding responses are naturally function-valued, motivating operator-learning methods that directly approximate mappings between function spaces [24, 25]. Among representative neural operator architectures, Deep Operator Network (DeepONet) provides a flexible framework grounded in the universal approximation theory of nonlinear operators [26, 27]. DeepONet represents an operator through a branch network and a trunk network. The branch network maps sensor observations of the input function to input-dependent coefficients, whereas the trunk network maps output coordinates to output-domain basis functions. Their inner product reconstructs the operator output, thereby separating input-function encoding from output-space representation. DeepONet has demonstrated strong capability in a variety of scientific and engineering problems involving high-dimensional field prediction [28–32]. Nevertheless, standard deterministic DeepONet provides only point predictions and does not explicitly quantify predictive uncertainty [33], motivating extensions of DeepONet that incorporate uncertainty quantification for more reliable predictions.

Uncertainty quantification for DeepONet has been explored through Bayesian methods, randomized-prior ensembles, and calibration-based approaches. Bayesian methods provide a principled probabilistic framework but often require iterative inference and remain sensitive to prior specification and posterior approximation [34–36]. Randomized-prior ensembles provide an effective means of quantifying epistemic uncertainty, while their computational cost grows with ensemble size and their performance depends on careful tuning [37]. Calibration-based methods can improve empirical interval coverage, whereas they primarily operate as post-processing procedures and do not directly modify the underlying uncertainty representation [38, 39]. Recent extensions further broaden the scope to conditional field generation, stochastic operators, and bounded uncertainty [40–42]. Among lightweight alternatives, Probabilistic DeepONet (Prob-DeepONet) directly parameterizes the predictive distribution using a formulation analogous to MVE, jointly predicting the output mean and variance through a Gaussian NLL [21, 43]. Representative studies have employed Prob-DeepONet for post-fault trajectory prediction, uncertainty-guided sample selection, and active learning [43, 44]. Unlike Bayesian posterior sampling and model ensembles, Prob-DeepONet estimates predictive means and variances in a single forward pass, providing an efficient approach to uncertainty quantification in operator learning [43]. However, its commonly adopted pointwise Gaussian formulation models marginal means and variances independently at individual output locations, resulting in a diagonal conditional covariance structure. Extending Prob-DeepONet to represent cross-location conditional dependence while retaining its lightweight formulation would therefore provide a more expressive probabilistic operator-learning framework for engineering applications.

Accordingly, this study considers probabilistic operator learning under random input fields and focuses on recovering the total predictive covariance of high-dimensional output fields. The principal methodological target is the conditional predictive component of this covariance, for which the pointwise conditional-independence assumption in Prob-DeepONet limits the representation of cross-location dependence. To address this limitation, we develop a two-step MV-DeepONet framework through two principal modifications. First, the two-step training strategy of Lee and Shin [45] is incorporated to decouple the learning of output-space basis functions from the input-to-coefficient mapping. Second, probabilistic modeling is transferred from the high-dimensional physical output space to the low-dimensional modal coefficient space. Although a diagonal Gaussian model is retained in the coefficient space, the probabilistic modal coefficients are jointly mapped to multiple output locations through the shared basis functions, allowing a generally non-diagonal conditional predictive covariance to be represented in the physical output space. The resulting formulation avoids explicit learning or storage of the full high-dimensional covariance matrix while retaining the lightweight structure and single-pass inference of Prob-DeepONet, thereby providing a structured representation of the conditional component of the total predictive covariance. The decoupled training strategy follows the two-step DeepONet formulation of Lee and Shin [45], in which the output-space basis is first learned and orthogonalized, followed by the learning of the input-to-coefficient mapping. This separation facilitates the construction of a linearly independent and numerically stable output basis and provides a low-dimensional representation for the subsequent coefficient-space probabilistic regression. The two-step formulation has subsequently been applied to a range of physical and engineering problems, including brittle fracture, three-dimensional electromagnetic field prediction, stochastic mechanical metamaterials, poroelasticity, Riemann problems, and hypersonic aerothermal analysis [46–51]. In this study, this decoupling provides a methodological foundation for probabilistic modeling in the coefficient space and structured output covariance recovery.

This study proposes a two-step MV-DeepONet framework for forward uncertainty propagation driven by random input fields and focuses on structured predictive uncertainty characterized by the total predictive covariance of high-dimensional output fields. The main contributions are summarized as follows:

- (1) A two-step MV-DeepONet framework is developed for forward uncertainty propagation driven by random input fields. By decoupling output-space basis learning from the input-to-coefficient mapping and performing probabilistic regression in the low-dimensional coefficient space, the proposed method relaxes the pointwise conditional-independence assumption of Prob-DeepONet. Through the shared basis functions, each probabilistic modal coefficient simultaneously affects multiple output locations, allowing a generally non-diagonal conditional predictive covariance to be represented in the physical output space. As a supplementary extension, deep ensembles and post-hoc calibration are used to account for epistemic uncertainty and improve empirical interval coverage.
- (2) A theoretical analysis of output covariance recovery is established through an error decomposition and a corresponding upper bound. The analysis identifies the low-rank compressibility of the target covariance, the approximation quality of the learned trunk subspace, finite-sample statistical error, and coefficient-space covariance estimation as the principal factors governing recovery accuracy. The resulting low-rank formulation avoids explicit learning or storage of the full high-dimensional covariance matrix while preserving lightweight, single-pass inference.
- (3) The proposed framework is validated on the reaction–diffusion, Burgers, and Darcy equations, together with a hypersonic aerothermal prediction problem. Compared with Prob-DeepONet, the proposed method exhibits improved generalization to unseen inputs, produces tighter and more structured uncertainty bands, and recovers correlation patterns consistent with the underlying physical mechanisms. These results highlight the complementary roles of the two methodological components: the two-step strategy helps reduce the mutual compensation between basis functions and coefficients, while coefficient-space probabilistic modeling propagates uncertainty jointly to multiple output locations through the shared basis functions.

The remainder of this paper is organized as follows. Section 2 reviews the relevant DeepONet formulations and develops the two-step MV-DeepONet framework. Section 3 evaluates the proposed method in terms of mean prediction, predictive uncertainty, and output covariance recovery using several representative examples. Section 4 summarizes the paper and discusses directions for future work.

2 Method

This section first reviews the standard DeepONet architecture and its two-step training strategy, followed by the pointwise probabilistic formulation of Prob-DeepONet. Building on these foundations, we develop a two-step MV-DeepONet that performs Gaussian modeling in a rotated low-dimensional coefficient space and represents structured conditional predictive covariance in the physical output space through a shared basis representation. Finally, a Frobenius-norm error decomposition and the corresponding upper bound are derived to characterize the principal factors governing total predictive covariance recovery.

2.1 Deep Operator Network

DeepONet [27] is a neural operator architecture that learns a mapping between infinite-dimensional function spaces, i.e., an operator $\mathcal{G} : \mathcal{U} \rightarrow \mathcal{S}$ that maps $u \in \mathcal{U}$ to $s = \mathcal{G}(u) \in \mathcal{S}$, where u and s denote the input and output

functions, respectively. It approximates $\mathcal{G}$ through two subnetworks. Given the sampled input

$$\mathbf{u} = [u(x_1), u(x_2), \dots, u(x_m)]^\top \in \mathbb{R}^m, \quad (1)$$

the branch network produces an input-dependent coefficient vector

$$\mathbf{b}_\theta(\mathbf{u}) = [b_1(\mathbf{u}), \dots, b_p(\mathbf{u})]^\top \in \mathbb{R}^p. \quad (2)$$

For the discrete output coordinates $\{y_i\}_{i=1}^M$, the trunk network generates the basis matrix

$$\Phi_\omega = [\phi_j(y_i)]_{i=1, \dots, M; j=1, \dots, p} \in \mathbb{R}^{M \times p}. \quad (3)$$

The discretized output prediction is then expressed as

$$\hat{\mathbf{s}}(\mathbf{u}) = \Phi_\omega \mathbf{b}_\theta(\mathbf{u}) \in \mathbb{R}^M. \quad (4)$$

Thus, the trunk network learns basis functions over the output domain, while the branch network predicts their input-dependent coefficients; θ and ω denote their respective trainable parameters.

For K training samples, the standard DeepONet is trained end-to-end by minimizing

$$\min_{\theta, \omega} \frac{1}{K} \sum_{k=1}^K \left\| \Phi_\omega \mathbf{b}_\theta(\mathbf{u}^{(k)}) - \mathbf{s}^{(k)} \right\|_2^2. \quad (5)$$

2.2 Two-Step Training of DeepONet

For high-dimensional output fields, end-to-end DeepONet training requires the joint optimization of the branch and trunk networks in a large coupled problem. The two-step strategy decomposes this problem into two sequential subproblems, decoupling output-basis learning from input-dependent coefficient learning [45], thereby simplifying optimization and providing a stable coordinate system for subsequent probabilistic modeling in the coefficient space.

Step 1: Trunk Basis Learning. The first step learns the trunk network alone. To isolate basis learning from the nonlinear branch mapping, the branch output is temporarily replaced by a trainable coefficient matrix $\mathbf{A} \in \mathbb{R}^{p \times K}$, and the trunk is trained by

$$(\omega^*, \mathbf{A}^*) = \arg \min_{\omega, \mathbf{A}} \left\| \Phi_\omega \mathbf{A} - \mathbf{S} \right\|_F^2, \quad (6)$$

where $\Phi^* := \Phi_{\omega^*}$ and $\mathbf{S} = [\mathbf{s}^{(1)}, \dots, \mathbf{s}^{(K)}] \in \mathbb{R}^{M \times K}$ collects the K output snapshots columnwise. This yields the low-rank approximation $\hat{\mathbf{S}}^* = \Phi^* \mathbf{A}^* \approx \mathbf{S}$, in which the columns of Φ^* span the learned low-dimensional subspace of the output field.

Intermediate Step: Basis Orthogonalization. The optimized trunk basis matrix is orthogonalized through the QR factorization

$$\Phi^* = \mathbf{Q}^* \mathbf{R}^*, \quad (\mathbf{Q}^*)^\top \mathbf{Q}^* = \mathbf{I}_p, \quad (7)$$

where $\mathbf{Q}^* \in \mathbb{R}^{M \times p}$ is column-orthonormal and $\mathbf{R}^* \in \mathbb{R}^{p \times p}$ is upper triangular. The corresponding coefficient targets are defined as

$$\mathbf{C}^* = \mathbf{R}^* \mathbf{A}^* = [\mathbf{c}_1^*, \dots, \mathbf{c}_K^*] \in \mathbb{R}^{p \times K}. \quad (8)$$

Accordingly, the first-step reconstruction can be written as $\Phi^* \mathbf{A}^* = \mathbf{Q}^* \mathbf{C}^*$.

Step 2: Branch Coefficient Regression. With $\mathbf{Q}^*$ fixed, the branch network is trained to predict the coefficient vector associated with each input function. Denoting its prediction by $\hat{\mathbf{c}}_\theta(\mathbf{u}) \in \mathbb{R}^p$, the training objective is

$$\theta^* = \arg \min_{\theta} \frac{1}{K} \sum_{k=1}^K \left\| \hat{\mathbf{c}}_\theta(\mathbf{u}^{(k)}) - \mathbf{c}_k^* \right\|_2^2. \quad (9)$$

The resulting output prediction is reconstructed as

$$\hat{\mathbf{s}}(\mathbf{u}) = \mathbf{Q}^* \hat{\mathbf{c}}_{\theta^*}(\mathbf{u}). \quad (10)$$

The original operator-learning problem is thereby reduced to a finite-dimensional supervised regression problem in coefficient space.

2.3 Probabilistic Deep Operator Network

To further quantify predictive uncertainty, Prob-DeepONet [43] extends the deterministic DeepONet by imposing pointwise Gaussian distributions at each physical output location. For a discrete input representation $\mathbf{u}$, the Prob-

DeepONet prediction is modeled as

$$\hat{\mathbf{s}}(\mathbf{u}) \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_\eta(\mathbf{u}), \text{diag}(\hat{\boldsymbol{\sigma}}_\eta^2(\mathbf{u}))), \quad (11)$$

where $\eta = \{\theta, \omega\}$ denotes all trainable parameters, with $\theta = \{\theta_\mu, \theta_\sigma\}$ and $\omega = \{\omega_\mu, \omega_\sigma\}$. Prob-DeepONet adopts separate branch-trunk paths for the mean and log-variance of the output field:

$$\hat{\boldsymbol{\mu}}_\eta(\mathbf{u}) = \Phi_{\mu, \omega_\mu} \mathbf{b}_{\mu, \theta_\mu}(\mathbf{u}) \in \mathbb{R}^M, \quad \hat{\boldsymbol{\ell}}_\eta(\mathbf{u}) = \Phi_{\sigma, \omega_\sigma} \mathbf{b}_{\sigma, \theta_\sigma}(\mathbf{u}) \in \mathbb{R}^M, \quad \hat{\boldsymbol{\sigma}}_\eta^2(\mathbf{u}) = \exp(\hat{\boldsymbol{\ell}}_\eta(\mathbf{u})) \in \mathbb{R}^M. \quad (12)$$

Here, the exponential is applied elementwise, ensuring strictly positive predictive variances. For K training samples, the model is trained by minimizing the pointwise Gaussian NLL:

$$\mathcal{L}_{\text{single}} = \frac{1}{2K} \sum_{k=1}^K \sum_{i=1}^M \left[\frac{(s_i^{(k)} - \hat{\mu}_{\eta, i}(\mathbf{u}^{(k)}))^2}{\hat{\sigma}_{\eta, i}^2(\mathbf{u}^{(k)})} + \log \hat{\sigma}_{\eta, i}^2(\mathbf{u}^{(k)}) \right]. \quad (13)$$

This formulation learns an input-dependent marginal distribution at each output location and provides pointwise prediction intervals for assessing marginal predictive reliability [43].

2.4 Two-Step MV-DeepONet for Covariance Recovery

To model the total predictive covariance of a probabilistic operator surrogate under random input functions, the proposed framework combines two-step basis learning [45] with Gaussian probabilistic modeling in a low-dimensional coefficient space. The proposed extension focuses on relaxing the pointwise conditional-independence assumption in Prob-DeepONet while retaining a computationally tractable covariance representation.

2.4.1 Structured Output Covariance under Random Inputs

Let $\mathbf{U}$ denote a random input function and let $\hat{\mathbf{s}}$ denote the field-valued prediction of a probabilistic operator surrogate. By the law of total covariance, the predictive covariance can be decomposed as

$$\text{Cov}(\hat{\mathbf{s}}) = \text{Cov}_{\mathbf{U}}(\mathbb{E}[\hat{\mathbf{s}} | \mathbf{U}]) + \mathbb{E}_{\mathbf{U}}[\text{Cov}(\hat{\mathbf{s}} | \mathbf{U})]. \quad (14)$$

The first term characterizes the variation of the predicted mean across random input realizations, and the second term represents the average conditional predictive covariance of the surrogate.

To illustrate why the first term in Eq. (14) may exhibit cross-location dependence, consider a random perturbation δu applied to the input function u . The corresponding output field is $s = \mathcal{G}(u + \delta u)$. A first-order expansion around u gives

$$s \approx \mathcal{G}(u) + \nabla \mathcal{G}(u) \delta u, \quad (15)$$

where $\nabla \mathcal{G}(u)$ denotes the linearized sensitivity of the operator. The resulting output covariance can therefore be approximated as

$$\text{Cov}(s) \approx \nabla \mathcal{G}(u) \text{Cov}(\delta u) \nabla \mathcal{G}(u)^\top. \quad (16)$$

Equation (16) shows that the covariance of the input perturbation is transformed by the sensitivity of the physical operator. For operators defined by partial differential equations or complex engineering systems, the same input perturbation can affect multiple output locations through differential operators, boundary conditions, or global physical constraints. Consequently, the sensitivities at different output locations generally share common components of the input perturbation, resulting in a generally non-diagonal $\text{Cov}(s)$. Regarding the second term, the pointwise Gaussian assumption in Prob-DeepONet yields a diagonal covariance:

$$\hat{\boldsymbol{\Sigma}}_{\text{Prob}}(\mathbf{u}) = \text{diag}(\hat{\sigma}_{\eta, 1}^2(\mathbf{u}), \dots, \hat{\sigma}_{\eta, M}^2(\mathbf{u})), \quad \left[\hat{\boldsymbol{\Sigma}}_{\text{Prob}}(\mathbf{u}) \right]_{ij} = 0, \quad i \neq j. \quad (17)$$

which estimates the marginal uncertainty at each output location but does not represent covariance between distinct locations for a fixed input. Consequently, the conditional predictive covariance, corresponding to the second term in Eq. (14), is restricted to a diagonal form and therefore does not explicitly represent cross-location dependence within this component, motivating the two-step MV-DeepONet developed below.

2.4.2 Probabilistic Extension Based on the Two-Step Strategy

Step 1: Trunk Basis Learning. The trunk training follows the original two-step DeepONet strategy [45] without structural modification. This step provides a stable low-dimensional output subspace for the subsequent probabilistic extension.

Intermediate Step: Orthogonalization and Subspace Rotation. The branch network models the coefficients with a diagonal Gaussian covariance, which implicitly assumes uncorrelated coefficients. Since basis orthogonalization

does not generally diagonalize the empirical coefficient covariance, an additional orthogonal rotation is introduced within the learned subspace.

After first-step trunk training and QR orthogonalization, the orthonormal basis $\mathbf{Q}^*$ yields the reconstruction

$$\hat{\mathbf{S}}^* = \Phi^* \mathbf{A}^* = \mathbf{Q}^* (\mathbf{R}^* \mathbf{A}^*) \approx \mathbf{S}, \quad (18)$$

where $\mathbf{Q}^* \in \mathbb{R}^{M \times p}$ is column-orthonormal and $\mathbf{R}^* \in \mathbb{R}^{p \times p}$ is upper triangular.

The training outputs are then centered as

$$\bar{\mathbf{s}} = \frac{1}{K} \sum_{k=1}^K \mathbf{s}^{(k)}, \quad \mathbf{x}^{(k)} = \mathbf{s}^{(k)} - \bar{\mathbf{s}}, \quad \mathbf{X} = [\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(K)}] \in \mathbb{R}^{M \times K}. \quad (19)$$

Their coefficients under the orthonormal basis $\mathbf{Q}^*$ are

$$\mathbf{C}_Q = (\mathbf{Q}^*)^\top \mathbf{X} = [\mathbf{c}_Q^{(1)}, \dots, \mathbf{c}_Q^{(K)}] \in \mathbb{R}^{p \times K}, \quad (20)$$

with empirical covariance

$$\hat{\mathbf{B}}_Q = \frac{1}{K} \mathbf{C}_Q \mathbf{C}_Q^\top \in \mathbb{R}^{p \times p}. \quad (21)$$

Because $\hat{\mathbf{B}}_Q$ is generally non-diagonal, an eigendecomposition is performed to rotate the coefficient coordinates and diagonalize their empirical covariance, thereby providing a representation better aligned with the subsequent diagonal Gaussian approximation:

$$\hat{\mathbf{B}}_Q = \hat{\mathbf{V}}_Q \hat{\mathbf{\Lambda}}_Q \hat{\mathbf{V}}_Q^\top, \quad \hat{\mathbf{V}}_Q^\top \hat{\mathbf{V}}_Q = \mathbf{I}_p, \quad (22)$$

where $\hat{\mathbf{\Lambda}}_Q = \text{diag}(\hat{\lambda}_{Q,1}, \dots, \hat{\lambda}_{Q,p})$ contains the eigenvalues of $\hat{\mathbf{B}}_Q$, and the rotated basis is defined as $\tilde{\mathbf{Q}}^* = \mathbf{Q}^* \hat{\mathbf{V}}_Q$. Since $\hat{\mathbf{V}}_Q$ is orthogonal, $\tilde{\mathbf{Q}}^*$ and $\mathbf{Q}^*$ span the same subspace and differ only in the orientation of the coordinate axes. The corresponding coefficient matrix is

$$\mathbf{C}_{\tilde{Q}} = (\tilde{\mathbf{Q}}^*)^\top \mathbf{X} = \hat{\mathbf{V}}_Q^\top \mathbf{C}_Q, \quad (23)$$

and its empirical covariance satisfies

$$\hat{\mathbf{B}}_{\tilde{Q}} = \frac{1}{K} \mathbf{C}_{\tilde{Q}} \mathbf{C}_{\tilde{Q}}^\top = \hat{\mathbf{V}}_Q^\top \hat{\mathbf{B}}_Q \hat{\mathbf{V}}_Q = \hat{\mathbf{\Lambda}}_Q. \quad (24)$$

The rotation preserves the learned output subspace while diagonalizing the empirical coefficient covariance, thereby providing coefficient coordinates better suited to the subsequent diagonal Gaussian modeling.

Step 2: Probabilistic Regression for Branch Coefficients. The second stage extends deterministic coefficient regression to probabilistic modeling. For a given input $\mathbf{u}$, the branch network predicts the coefficient mean and log-variance:

$$\hat{\boldsymbol{\mu}}_{c,\theta}(\mathbf{u}) = [\hat{\mu}_{c,1}(\mathbf{u}), \dots, \hat{\mu}_{c,p}(\mathbf{u})]^\top \in \mathbb{R}^p, \quad \hat{\boldsymbol{\ell}}_{c,\theta}(\mathbf{u}) = [\hat{\ell}_{c,1}(\mathbf{u}), \dots, \hat{\ell}_{c,p}(\mathbf{u})]^\top \in \mathbb{R}^p. \quad (25)$$

The predicted coefficient variances are defined by

$$\hat{\sigma}_{c,j}^2(\mathbf{u}) = \exp(\hat{\ell}_{c,j}(\mathbf{u})), \quad j = 1, \dots, p, \quad (26)$$

which ensures their strict positivity. The corresponding coefficient covariance matrix $\hat{\mathbf{\Lambda}}_{c,\theta}(\mathbf{u})$ is diagonal, with its j th diagonal entry given by $\hat{\sigma}_{c,j}^2(\mathbf{u})$. The modal coefficient vector is therefore modeled as

$$\hat{\mathbf{c}}_\theta(\mathbf{u}) \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_{c,\theta}(\mathbf{u}), \hat{\mathbf{\Lambda}}_{c,\theta}(\mathbf{u})). \quad (27)$$

The independence assumption in Eq. (27) is imposed on the modal coefficients in the rotated coefficient space rather than on the physical output variables.

For the k -th training sample, let $\mathbf{c}_Q^{(k)}$ denote the empirical coefficient target under $\tilde{\mathbf{Q}}^*$. The branch network is trained using the Gaussian NLL to predict the input-dependent coefficient means and variances:

$$\mathcal{L}_{\text{coef}} = \frac{1}{2K} \sum_{k=1}^K \sum_{j=1}^p \left[\frac{\left(c_{\tilde{Q},j}^{(k)} - \hat{\mu}_{c,j}(\mathbf{u}^{(k)}) \right)^2}{\hat{\sigma}_{c,j}^2(\mathbf{u}^{(k)})} + \log \hat{\sigma}_{c,j}^2(\mathbf{u}^{(k)}) \right]. \quad (28)$$

2.4.3 Output-Space Covariance Recovery

For a given input $\mathbf{u}$, the centered output field is reconstructed from the rotated orthonormal basis and the probabilistic modal coefficients as

$$\hat{\mathbf{x}}_\theta(\mathbf{u}) = \tilde{\mathbf{Q}}^* \hat{\mathbf{c}}_\theta(\mathbf{u}). \quad (29)$$

By the covariance propagation rule for linear transformations, the corresponding conditional predictive covariance in the physical output space is

$$\hat{\Sigma}_s(\mathbf{u}) = \text{Cov}(\hat{\mathbf{x}}_\theta(\mathbf{u}) | \mathbf{u}) = \tilde{\mathbf{Q}}^* \hat{\Lambda}_{c,\theta}(\mathbf{u}) (\tilde{\mathbf{Q}}^*)^\top. \quad (30)$$

Although $\hat{\Lambda}_{c,\theta}(\mathbf{u})$ is diagonal, the output covariance in Eq. (30) is generally non-diagonal. Each probabilistic modal coefficient affects multiple output locations through the shared basis, thereby inducing structured output covariance without directly predicting a full covariance matrix. For two output locations y_i and y_j , the corresponding covariance is

$$[\hat{\Sigma}_s(\mathbf{u})]_{ij} = \sum_{m=1}^p \tilde{q}_m^*(y_i) \hat{\sigma}_{c,m}^2(\mathbf{u}) \tilde{q}_m^*(y_j), \quad (31)$$

where $\tilde{q}_m^*(y_i)$ is the value of the m -th rotated basis function at y_i , and $\hat{\sigma}_{c,m}^2(\mathbf{u})$ is the predicted variance of the corresponding modal coefficient. For a given pair of distinct output locations, this sum may vanish under special configurations, including: (i) all basis functions vanish simultaneously at y_i or y_j , i.e., $\tilde{q}_m^*(y_\ell) = 0$ for all $m = 1, \dots, p$ and some $\ell \in \{i, j\}$, a strong coincidence condition uncommon for neural-network bases on general output fields; (ii) the signed modal contributions cancel exactly, $\sum_{m=1}^p \tilde{q}_m^*(y_i) \hat{\sigma}_{c,m}^2(\mathbf{u}) \tilde{q}_m^*(y_j) = 0$, which requires the branch-predicted variances and the basis values to satisfy a strict algebraic relation that is generally unstable across inputs and locations; or (iii) $\hat{\sigma}_{c,m}^2(\mathbf{u}) = 0$ for all m , corresponding to a deterministic limiting case. This exact degeneracy is excluded by the adopted log-variance parameterization. Excluding the aforementioned cases, there exists at least one mode m with $\tilde{q}_m^*(y_i) \tilde{q}_m^*(y_j) \hat{\sigma}_{c,m}^2(\mathbf{u}) \neq 0$. Thus the fluctuation of a probabilistic modal coefficient propagates through its associated shared basis function to multiple output locations, so random fluctuations at different locations are not independent but jointly driven by the same low-dimensional random coefficients.

Accordingly, the conditional mean and variance of the centered output at each output location are given by the following expressions, with the resulting network architecture shown in Fig. 1.

$$\mathbb{E}[\hat{x}_\theta(\mathbf{u})(y_i) | \mathbf{u}] = \sum_{m=1}^p \tilde{q}_m^*(y_i) \hat{\mu}_{c,m}(\mathbf{u}), \quad \text{Var}[\hat{x}_\theta(\mathbf{u})(y_i) | \mathbf{u}] = \sum_{m=1}^p (\tilde{q}_m^*(y_i))^2 \hat{\sigma}_{c,m}^2(\mathbf{u}). \quad (32)$$

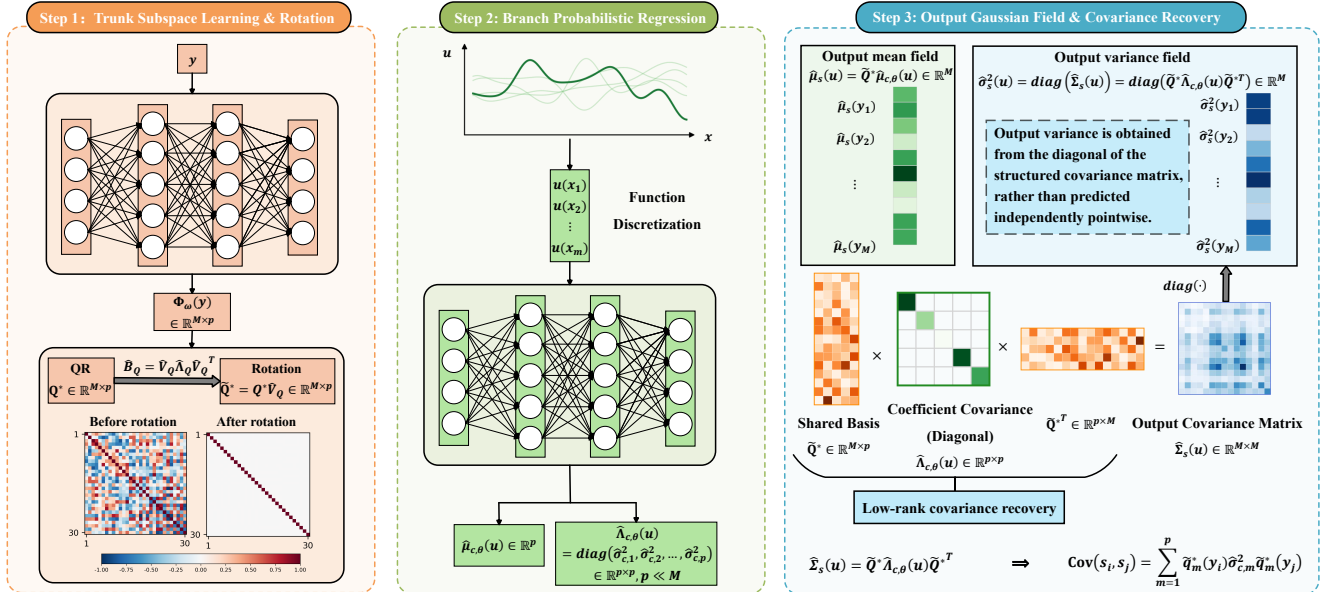


Fig. 1. Schematic of the two-step MV-DeepONet framework. Step 1 learns the output-space basis and applies QR orthogonalization and the subspace rotation $\tilde{\mathbf{Q}}^* = \mathbf{Q}^* \tilde{\mathbf{V}}_Q$. Step 2 predicts the coefficient mean $\hat{\mu}_{c,\theta}(\mathbf{u})$ and diagonal coefficient covariance $\hat{\Lambda}_{c,\theta}(\mathbf{u})$. Finally, the resulting coefficient distribution is mapped back to the physical output space through the shared rotated basis, producing the output mean and structured conditional predictive covariance.

By contrast, Prob-DeepONet represents predictive randomness independently at the physical output locations:

$$\hat{s}_i(\mathbf{u}) = \hat{\mu}_{\eta,i}(\mathbf{u}) + \hat{\sigma}_{\eta,i}(\mathbf{u})\varepsilon_i, \quad \varepsilon_i \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1). \quad (33)$$

Consequently,

$$\text{Cov}(\hat{s}_i(\mathbf{u}), \hat{s}_j(\mathbf{u}) \mid \mathbf{u}) = 0, \quad i \neq j. \quad (34)$$

The principal distinction between the two models therefore lies in the space where randomness is represented. Prob-DeepONet models conditionally independent randomness at individual output locations, whereas the proposed framework models low-dimensional random coefficients whose fluctuations are shared across the output domain through the basis functions.

At the dataset level, the law of total covariance decomposes the coefficient covariance into the covariance of the predicted coefficient means across inputs and the average conditional coefficient covariance. For K input samples, let $\bar{\hat{\mu}}_{c,\theta} = \frac{1}{K} \sum_{k=1}^K \hat{\mu}_{c,\theta}(\mathbf{u}^{(k)})$ denote the sample average of the predicted coefficient mean vectors. The covariance induced by the branch mean head is

$$\hat{\mathbf{B}}_\mu = \frac{1}{K} \sum_{k=1}^K \left[\hat{\mu}_{c,\theta}(\mathbf{u}^{(k)}) - \bar{\hat{\mu}}_{c,\theta} \right] \left[\hat{\mu}_{c,\theta}(\mathbf{u}^{(k)}) - \bar{\hat{\mu}}_{c,\theta} \right]^\top, \quad (35)$$

whereas the average covariance predicted by the branch variance head is

$$\hat{\mathbf{B}}_\sigma = \frac{1}{K} \sum_{k=1}^K \hat{\Lambda}_{c,\theta}(\mathbf{u}^{(k)}). \quad (36)$$

The total predictive covariance in the coefficient space is therefore $\hat{\mathbf{B}}_{2\text{step}} = \hat{\mathbf{B}}_\mu + \hat{\mathbf{B}}_\sigma$, and the corresponding covariance in the physical output space is

$$\hat{\Sigma}_{2\text{step}} = \tilde{\mathbf{Q}}^* \hat{\mathbf{B}}_{2\text{step}} (\tilde{\mathbf{Q}}^*)^\top = \tilde{\mathbf{Q}}^* (\hat{\mathbf{B}}_\mu + \hat{\mathbf{B}}_\sigma) (\tilde{\mathbf{Q}}^*)^\top. \quad (37)$$

Thus, dataset-level covariance recovery depends jointly on the predicted coefficient means and variances.

From a computational perspective, the proposed method does not require direct parameterization of the full $M \times M$ covariance matrix. Instead, it predicts p conditional modal variances and constructs a low-dimensional coefficient-space covariance, which is mapped to the physical output space through the shared basis. When $p \ll M$, this yields a low-rank covariance representation at reduced cost. Its diagonal entries provide pointwise uncertainty bands, while the factorized form supports cross-location correlation analysis.

2.4.4 Error Decomposition and Upper Bound for Output Covariance Recovery

To characterize the covariance recovery capability of the proposed model, this subsection derives a Frobenius-norm decomposition and a corresponding upper bound for the output covariance recovery error.

2.4.4.1 Basic Notation and Covariance Estimator

Let $\mathbf{U}$ denote the random input function and $\mathbf{u}$ a generic realization of $\mathbf{U}$, with corresponding discretized output field $\mathbf{s}(\mathbf{u}) \in \mathbb{R}^M$, mean field $\boldsymbol{\mu}_s = \mathbb{E}_{\mathbf{U}}[\mathbf{s}(\mathbf{U})]$, and centered field $\mathbf{x}(\mathbf{u}) = \mathbf{s}(\mathbf{u}) - \boldsymbol{\mu}_s$. The true input-induced output covariance is

$$\Sigma = \mathbb{E}_{\mathbf{U}} \left[\mathbf{x}(\mathbf{U}) \mathbf{x}(\mathbf{U})^\top \right] \in \mathbb{R}^{M \times M}. \quad (38)$$

Let the rotated trunk basis form the column-orthonormal matrix $\tilde{\mathbf{Q}}^* = [\tilde{\mathbf{q}}_1^*, \dots, \tilde{\mathbf{q}}_p^*] \in \mathbb{R}^{M \times p}$ with $(\tilde{\mathbf{Q}}^*)^\top \tilde{\mathbf{Q}}^* = \mathbf{I}_p$ and $p \ll M$, and let $\mathbf{P} = \tilde{\mathbf{Q}}^* (\tilde{\mathbf{Q}}^*)^\top$ be the associated orthogonal projector. In this learned subspace, the true modal coefficients and their covariance are

$$\mathbf{c}(\mathbf{u}) = (\tilde{\mathbf{Q}}^*)^\top \mathbf{x}(\mathbf{u}) \in \mathbb{R}^p, \quad \mathbf{B} = \text{Cov}(\mathbf{c}) = (\tilde{\mathbf{Q}}^*)^\top \Sigma \tilde{\mathbf{Q}}^* \in \mathbb{R}^{p \times p}. \quad (39)$$

The dataset-level total predictive covariance follows from the law of total covariance:

$$\text{Cov}(\hat{\mathbf{x}}_\theta) = \text{Cov}_{\mathbf{U}}(\mathbb{E}[\hat{\mathbf{x}}_\theta \mid \mathbf{U}]) + \mathbb{E}_{\mathbf{U}}[\text{Cov}(\hat{\mathbf{x}}_\theta \mid \mathbf{U})]. \quad (40)$$

With the trained basis $\tilde{\mathbf{Q}}^*$ fixed, linearity of conditional expectation and the covariance propagation rule for linear transformations give

$$\mathbb{E}[\hat{\mathbf{x}}_\theta \mid \mathbf{u}^{(k)}] = \tilde{\mathbf{Q}}^* \hat{\mu}_{c,\theta}(\mathbf{u}^{(k)}), \quad \text{Cov}(\hat{\mathbf{x}}_\theta \mid \mathbf{u}^{(k)}) = \tilde{\mathbf{Q}}^* \hat{\Lambda}_{c,\theta}(\mathbf{u}^{(k)}) (\tilde{\mathbf{Q}}^*)^\top, \quad (41)$$

from the coefficient distribution $\hat{\mathbf{c}}_\theta(\mathbf{u}^{(k)}) \sim \mathcal{N}(\hat{\mu}_{c,\theta}(\mathbf{u}^{(k)}), \hat{\Lambda}_{c,\theta}(\mathbf{u}^{(k)}))$, with predicted mean $\hat{\mu}_{c,\theta}(\mathbf{u}^{(k)}) \in \mathbb{R}^p$ and diag-

onal covariance $\widehat{\mathbf{\Lambda}}_{c,\theta}(\mathbf{u}^{(k)}) = \text{diag}(\hat{\sigma}_{c,1}^2(\mathbf{u}^{(k)}), \dots, \hat{\sigma}_{c,p}^2(\mathbf{u}^{(k)}))$.

Taking the sample covariance of the conditional means and the sample average of the conditional covariances over the K input samples gives, as established in the preceding subsection, the dataset-level estimators

$$\widehat{\mathbf{B}}_{2\text{step}} = \widehat{\mathbf{B}}_{\mu} + \widehat{\mathbf{B}}_{\sigma}, \quad \widehat{\mathbf{\Sigma}}_{2\text{step}} = \widetilde{\mathbf{Q}}^* \widehat{\mathbf{B}}_{2\text{step}} (\widetilde{\mathbf{Q}}^*)^\top, \quad (42)$$

where $\widehat{\mathbf{B}}_{\mu}$ and $\widehat{\mathbf{B}}_{\sigma}$ are the mean-head and variance-head coefficient covariances defined in Eqs. (35)–(36). Their sum defines the dataset-level coefficient covariance analyzed below.

2.4.4.2 Basic Error Decomposition

The recovery error is measured in the Frobenius norm as $\|\mathbf{\Sigma} - \widehat{\mathbf{\Sigma}}_{2\text{step}}\|_F$. Introducing the projected covariance $\mathbf{P}\mathbf{\Sigma}\mathbf{P} = \widetilde{\mathbf{Q}}^* \mathbf{B} (\widetilde{\mathbf{Q}}^*)^\top$ as an intermediate term gives

$$\mathbf{\Sigma} - \widehat{\mathbf{\Sigma}}_{2\text{step}} = (\mathbf{\Sigma} - \mathbf{P}\mathbf{\Sigma}\mathbf{P}) + \widetilde{\mathbf{Q}}^* (\mathbf{B} - \widehat{\mathbf{B}}_{2\text{step}}) (\widetilde{\mathbf{Q}}^*)^\top. \quad (43)$$

Because $\widetilde{\mathbf{Q}}^*$ has orthonormal columns, the triangle inequality yields

$$\|\mathbf{\Sigma} - \widehat{\mathbf{\Sigma}}_{2\text{step}}\|_F \leq \|\mathbf{\Sigma} - \mathbf{P}\mathbf{\Sigma}\mathbf{P}\|_F + \|\mathbf{B} - \widehat{\mathbf{B}}_{2\text{step}}\|_F. \quad (44)$$

Here, $\|\mathbf{\Sigma} - \mathbf{P}\mathbf{\Sigma}\mathbf{P}\|_F$ denotes the subspace projection error, while $\|\mathbf{B} - \widehat{\mathbf{B}}_{2\text{step}}\|_F$ denotes the total coefficient covariance estimation error.

Consider the eigendecomposition

$$\mathbf{\Sigma} = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^\top, \quad \mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_M), \quad \lambda_1 \geq \dots \geq \lambda_M \geq 0. \quad (45)$$

Let $\mathbf{W}_p \in \mathbb{R}^{M \times p}$ collect the leading p eigenvectors, and $\mathbf{P}^* = \mathbf{W}_p \mathbf{W}_p^\top$ be the projector onto the optimal rank- p principal subspace. By the Eckart–Young–Mirsky theorem [52],

$$\|\mathbf{\Sigma} - \mathbf{P}^* \mathbf{\Sigma} \mathbf{P}^*\|_F = \sqrt{\sum_{j>p} \lambda_j^2}. \quad (46)$$

Introducing $\mathbf{P}^* \mathbf{\Sigma} \mathbf{P}^*$ as an intermediate term, the projection error can be decomposed as

$$\|\mathbf{\Sigma} - \mathbf{P}\mathbf{\Sigma}\mathbf{P}\|_F \leq \sqrt{\sum_{j>p} \lambda_j^2} + \|\mathbf{P}^* \mathbf{\Sigma} \mathbf{P}^* - \mathbf{P}\mathbf{\Sigma}\mathbf{P}\|_F. \quad (47)$$

Applying the mixed-norm inequality to the second term, we obtain

$$\|\mathbf{P}^* \mathbf{\Sigma} \mathbf{P}^* - \mathbf{P}\mathbf{\Sigma}\mathbf{P}\|_F \leq \|\mathbf{P}^* - \mathbf{P}\|_2 \|\mathbf{\Sigma}\|_F \|\mathbf{P}^*\|_2 + \|\mathbf{P}\|_2 \|\mathbf{\Sigma}\|_F \|\mathbf{P}^* - \mathbf{P}\|_2 \leq 2 \|\mathbf{\Sigma}\|_F \|\mathbf{P} - \mathbf{P}^*\|_F. \quad (48)$$

Substituting this estimate into the preceding bound gives

$$\|\mathbf{\Sigma} - \mathbf{P}\mathbf{\Sigma}\mathbf{P}\|_F \leq \sqrt{\sum_{j>p} \lambda_j^2} + 2 \|\mathbf{\Sigma}\|_F \|\mathbf{P} - \mathbf{P}^*\|_F. \quad (49)$$

Consequently,

$$\|\mathbf{\Sigma} - \widehat{\mathbf{\Sigma}}_{2\text{step}}\|_F \leq \sqrt{\sum_{j>p} \lambda_j^2} + 2 \|\mathbf{\Sigma}\|_F \|\mathbf{P} - \mathbf{P}^*\|_F + \|\mathbf{B} - \widehat{\mathbf{B}}_{2\text{step}}\|_F. \quad (50)$$

For convenience, let T_1 , T_2 , and T_3 represent the first, second, and third terms on the right-hand side, respectively. Specifically, T_1 is the low-rank truncation error of the true covariance, T_2 denotes the discrepancy between the trunk-learned subspace and the optimal rank- p principal subspace, and T_3 quantifies the error in the dataset-level total coefficient covariance.

2.4.4.3 Trunk Subspace Error Term

We next analyze T_2 , which is governed by the subspace discrepancy $\|\mathbf{P} - \mathbf{P}^*\|_F$, with $\|\mathbf{\Sigma}\|_F$ serving as a problem-dependent scale factor. Let $\mathbf{X} = [\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(K)}] \in \mathbb{R}^{M \times K}$ be the centered training-output matrix, and let $\widehat{\mathbf{\Sigma}} = \frac{1}{K} \mathbf{X} \mathbf{X}^\top$ be the empirical covariance, with eigendecomposition

$$\widehat{\mathbf{\Sigma}} = \widehat{\mathbf{W}} \widehat{\mathbf{\Lambda}} \widehat{\mathbf{W}}^\top, \quad \widehat{\mathbf{\Lambda}} = \text{diag}(\widehat{\lambda}_1, \dots, \widehat{\lambda}_M), \quad \widehat{\lambda}_1 \geq \dots \geq \widehat{\lambda}_M \geq 0. \quad (51)$$

Let $\widehat{\mathbf{W}}_p \in \mathbb{R}^{M \times p}$ collect the leading p eigenvectors of $\widehat{\mathbf{\Sigma}}$, and define $\widehat{\mathbf{P}}^* = \widehat{\mathbf{W}}_p \widehat{\mathbf{W}}_p^\top$ as the projector onto the empirically optimal p -dimensional principal subspace. Then

$$\|\mathbf{P} - \mathbf{P}^*\|_F \leq \|\mathbf{P} - \widehat{\mathbf{P}}^*\|_F + \|\widehat{\mathbf{P}}^* - \mathbf{P}^*\|_F, \quad (52)$$

where the two terms represent trunk learning error and finite-sample statistical error, respectively.

Trunk Learning Error Define the reconstruction risk of a rank- p projector $\mathbf{P}$ as

$$\mathcal{R}_T(\mathbf{P}) = \frac{1}{K} \|\mathbf{X} - \mathbf{P}\mathbf{X}\|_F^2 = \text{tr} \left[(\mathbf{I}_M - \mathbf{P}) \widehat{\mathbf{\Sigma}} \right]. \quad (53)$$

The reconstruction risk of the empirically optimal principal subspace is similarly expressed through its orthogonal projector $\widehat{\mathbf{P}}^*$ as

$$\mathcal{R}_T(\widehat{\mathbf{P}}^*) = \text{tr} \left[(\mathbf{I}_M - \widehat{\mathbf{P}}^*) \widehat{\mathbf{\Sigma}} \right]. \quad (54)$$

Define the excess reconstruction risk of the trunk-learned subspace relative to the empirical optimum as

$$\Delta_T = \mathcal{R}_T(\mathbf{P}) - \mathcal{R}_T(\widehat{\mathbf{P}}^*) = \text{tr}(\widehat{\mathbf{P}}^* \widehat{\mathbf{\Sigma}}) - \text{tr}(\mathbf{P} \widehat{\mathbf{\Sigma}}) \geq 0. \quad (55)$$

To relate Δ_T to the discrepancy between the trunk-learned and empirical principal subspaces, express $\mathbf{P}$ in the eigenbasis of $\widehat{\mathbf{\Sigma}}$ by defining $\tilde{\mathbf{P}} = \widehat{\mathbf{W}}^\top \mathbf{P} \widehat{\mathbf{W}}$. Using the cyclic property of the trace gives

$$\text{tr}(\mathbf{P} \widehat{\mathbf{\Sigma}}) = \text{tr}(\mathbf{P} \widehat{\mathbf{W}} \widehat{\mathbf{\Lambda}} \widehat{\mathbf{W}}^\top) = \text{tr}(\widehat{\mathbf{W}}^\top \mathbf{P} \widehat{\mathbf{W}} \widehat{\mathbf{\Lambda}}) = \text{tr}(\widehat{\mathbf{\Lambda}} \tilde{\mathbf{P}}) = \sum_{j=1}^M \hat{\lambda}_j \tilde{p}_{jj}. \quad (56)$$

Let $\hat{\gamma}_p = \hat{\lambda}_p - \hat{\lambda}_{p+1} > 0$ denote the empirical spectral gap at the selected dimension.

The empirically optimal projector can then be represented in the eigenbasis of $\widehat{\mathbf{\Sigma}}$ as $\widehat{\mathbf{P}}^* = \widehat{\mathbf{W}} \mathbf{E}_p \widehat{\mathbf{W}}^\top$, with $\mathbf{E}_p = \begin{bmatrix} \mathbf{I}_p & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \in \mathbb{R}^{M \times M}$. Consequently,

$$\text{tr}(\widehat{\mathbf{P}}^* \widehat{\mathbf{\Sigma}}) = \text{tr}(\widehat{\mathbf{W}} \mathbf{E}_p \widehat{\mathbf{W}}^\top \widehat{\mathbf{W}} \widehat{\mathbf{\Lambda}} \widehat{\mathbf{W}}^\top) = \text{tr}(\mathbf{E}_p \widehat{\mathbf{\Lambda}}) = \sum_{j=1}^p \hat{\lambda}_j. \quad (57)$$

It follows that $\Delta_T = \sum_{j=1}^p \hat{\lambda}_j (1 - \tilde{p}_{jj}) - \sum_{j=p+1}^M \hat{\lambda}_j \tilde{p}_{jj}$. Using the eigenvalue ordering of $\widehat{\mathbf{\Sigma}}$ yields

$$\Delta_T \geq \hat{\lambda}_p \sum_{j=1}^p (1 - \tilde{p}_{jj}) - \hat{\lambda}_{p+1} \sum_{j=p+1}^M \tilde{p}_{jj} = (\hat{\lambda}_p - \hat{\lambda}_{p+1}) \sum_{j=1}^p (1 - \tilde{p}_{jj}) = \hat{\gamma}_p \sum_{j=1}^p (1 - \tilde{p}_{jj}). \quad (58)$$

Since $\mathbf{P}$ and $\widehat{\mathbf{P}}^*$ are rank- p orthogonal projectors,

$$\|\mathbf{P} - \widehat{\mathbf{P}}^*\|_F^2 = \text{tr} \left[(\mathbf{P} - \widehat{\mathbf{P}}^*)^\top (\mathbf{P} - \widehat{\mathbf{P}}^*) \right] = 2 \left[p - \text{tr}(\mathbf{P} \widehat{\mathbf{P}}^*) \right]. \quad (59)$$

Moreover, $\text{tr}(\mathbf{P} \widehat{\mathbf{P}}^*) = \text{tr}(\mathbf{P} \widehat{\mathbf{W}} \mathbf{E}_p \widehat{\mathbf{W}}^\top) = \text{tr}(\widehat{\mathbf{W}}^\top \mathbf{P} \widehat{\mathbf{W}} \mathbf{E}_p) = \text{tr}(\tilde{\mathbf{P}} \mathbf{E}_p) = \sum_{j=1}^p \tilde{p}_{jj}$. Therefore,

$$\|\mathbf{P} - \widehat{\mathbf{P}}^*\|_F^2 = 2 \sum_{j=1}^p (1 - \tilde{p}_{jj}) \leq \frac{2\Delta_T}{\hat{\gamma}_p}. \quad (60)$$

Equivalently,

$$\|\mathbf{P} - \widehat{\mathbf{P}}^*\|_F \leq \sqrt{\frac{2\Delta_T}{\hat{\gamma}_p}}.$$

This inequality shows that a small excess reconstruction risk implies a small discrepancy between the trunk-learned and empirical principal subspaces, provided that $\hat{\gamma}_p$ does not vanish. This conclusion is further supported by Theorem 3.5 of Lee and Shin [45], which establishes that a sufficiently expressive trunk network can attain the best rank- p approximation error of the training-output matrix. Together with the spectral-gap bound above, this result provides theoretical justification for the trunk network's capacity to recover the empirically optimal p -dimensional principal subspace.

Finite-Sample Statistical Error The finite-sample statistical error arises from the covariance perturbation $\widehat{\Sigma} - \Sigma$, where $\widehat{\Sigma} = \Sigma + (\widehat{\Sigma} - \Sigma)$. Under a non-vanishing true spectral gap $\gamma_p = \lambda_p - \lambda_{p+1} > 0$, Theorem 2 of Yu *et al.* [53], a variant of the Davis–Kahan $\sin \Theta$ theorem based on the population spectral gap, gives the following bound for the leading p -dimensional eigenspaces:

$$\left\| \sin \Theta \left(\mathbf{W}_p, \widehat{\mathbf{W}}_p \right) \right\|_F \leq \frac{2}{\gamma_p} \min \left(\sqrt{p} \left\| \widehat{\Sigma} - \Sigma \right\|_2, \left\| \widehat{\Sigma} - \Sigma \right\|_F \right) \leq \frac{2}{\gamma_p} \left\| \widehat{\Sigma} - \Sigma \right\|_F. \quad (61)$$

Combined with the identity relating the Frobenius distance between the orthogonal projectors to the principal angles, $\left\| \widehat{\mathbf{P}}^* - \mathbf{P}^* \right\|_F = \sqrt{2} \left\| \sin \Theta(\mathbf{W}_p, \widehat{\mathbf{W}}_p) \right\|_F$, this yields

$$\left\| \widehat{\mathbf{P}}^* - \mathbf{P}^* \right\|_F \leq \frac{2\sqrt{2} \left\| \widehat{\Sigma} - \Sigma \right\|_F}{\gamma_p}. \quad (62)$$

Substituting this estimate and the preceding trunk-learning bound into T_2 gives

$$T_2 \leq 2 \left\| \Sigma \right\|_F \sqrt{\frac{2\Delta_T}{\widehat{\gamma}_p}} + \frac{4\sqrt{2} \left\| \Sigma \right\|_F}{\gamma_p} \left\| \widehat{\Sigma} - \Sigma \right\|_F. \quad (63)$$

Thus, T_2 comprises the trunk-learning error relative to the empirical principal subspace and the finite-sample statistical error.

2.4.4.4 Learning Error in the Branch-Induced Total Coefficient Covariance

To bound T_3 , we introduce the empirical coefficient covariance $\widehat{\mathbf{B}} = (\widetilde{\mathbf{Q}}^*)^\top \widehat{\Sigma} \widetilde{\mathbf{Q}}^* \in \mathbb{R}^{p \times p}$ as an intermediate quantity. The triangle inequality gives

$$\left\| \mathbf{B} - \widehat{\mathbf{B}}_{2\text{step}} \right\|_F \leq \left\| \mathbf{B} - \widehat{\mathbf{B}} \right\|_F + \left\| \widehat{\mathbf{B}} - \widehat{\mathbf{B}}_{2\text{step}} \right\|_F. \quad (64)$$

Here, the first term represents the finite-sample statistical error, whereas the second term quantifies the branch learning error. Since $\mathbf{B} - \widehat{\mathbf{B}} = (\widetilde{\mathbf{Q}}^*)^\top (\Sigma - \widehat{\Sigma}) \widetilde{\mathbf{Q}}^*$, the first term satisfies $\left\| \mathbf{B} - \widehat{\mathbf{B}} \right\|_F \leq \left\| \Sigma - \widehat{\Sigma} \right\|_F$.

Let $\widehat{\mathbf{B}}_{2\text{step}}^{\text{NN}, \text{best}}$ denote the dataset-level covariance induced by the branch model in the considered function class that best approximates $\widehat{\mathbf{B}}$, and define

$$\eta_{\text{app}}^B = \left\| \widehat{\mathbf{B}} - \widehat{\mathbf{B}}_{2\text{step}}^{\text{NN}, \text{best}} \right\|_F, \quad \eta_{\text{opt}}^B = \left\| \widehat{\mathbf{B}}_{2\text{step}}^{\text{NN}, \text{best}} - \widehat{\mathbf{B}}_{2\text{step}} \right\|_F. \quad (65)$$

Here, η_{app}^B measures the approximation error of the branch-network function class relative to the empirical target, whereas η_{opt}^B measures the optimization-related discrepancy between this benchmark and the trained model. The former depends on network expressivity and target complexity; the latter is influenced by training configuration. Therefore,

$$T_3 \leq \left\| \Sigma - \widehat{\Sigma} \right\|_F + \eta_{\text{app}}^B + \eta_{\text{opt}}^B. \quad (66)$$

Thus, the total coefficient covariance error is governed jointly by finite-sample covariance estimation and branch probabilistic regression error.

2.4.4.5 Combined Error Bound

Combining the preceding results gives

$$\left\| \Sigma - \widehat{\Sigma}_{2\text{step}} \right\|_F \leq \sqrt{\sum_{j>p} \lambda_j^2} + 2 \left\| \Sigma \right\|_F \sqrt{\frac{2\Delta_T}{\widehat{\gamma}_p}} + \left(\frac{4\sqrt{2} \left\| \Sigma \right\|_F}{\gamma_p} + 1 \right) \left\| \widehat{\Sigma} - \Sigma \right\|_F + \eta_{\text{app}}^B + \eta_{\text{opt}}^B. \quad (67)$$

Under standard covariance concentration conditions, the empirical covariance satisfies the spectral-norm convergence rate $\left\| \widehat{\Sigma} - \Sigma \right\|_2 = \mathcal{O}_{\mathbb{P}}(K^{-1/2})$ [54]. For any $\mathbf{A} \in \mathbb{R}^{M \times M}$, $\left\| \mathbf{A} \right\|_F \leq \sqrt{M} \left\| \mathbf{A} \right\|_2$. Therefore,

$$\left\| \widehat{\Sigma} - \Sigma \right\|_F = \mathcal{O}_{\mathbb{P}} \left(\sqrt{\frac{M}{K}} \right). \quad (68)$$

Hence,

$$\left\| \Sigma - \widehat{\Sigma}_{2\text{step}} \right\|_F \leq \sqrt{\sum_{j>p} \lambda_j^2} + 2 \left\| \Sigma \right\|_F \sqrt{\frac{2\Delta_T}{\widehat{\gamma}_p}} + \mathcal{O}_{\mathbb{P}} \left[\left(\frac{4\sqrt{2} \left\| \Sigma \right\|_F}{\gamma_p} + 1 \right) \sqrt{\frac{M}{K}} \right] + \eta_{\text{app}}^B + \eta_{\text{opt}}^B. \quad (69)$$

The bound identifies four principal factors governing covariance recovery. First, the spectral-tail term $\sqrt{\sum_{j>p} \lambda_j^2}$ characterizes the low-rank compressibility of the true covariance. Second, the trunk-related contribution is determined by the excess reconstruction risk Δ_T , with $\hat{\gamma}_p$ linking this risk to the discrepancy between the learned and empirical principal subspaces. Third, finite-sample statistical error depends on the number of training samples, the output dimension, and the distribution of the training outputs. Fourth, $\eta_{\text{app}}^B + \eta_{\text{opt}}^B$ accounts for the branch network’s approximation and optimization errors in learning the empirical total coefficient covariance. Additionally, the spectral gaps $\hat{\gamma}_p$ and γ_p control the stability of the corresponding bounds. The true spectral gap is assumed to be non-vanishing, whereas the empirical spectral gap is verified numerically in each example. The factor $\|\Sigma\|_F$ reflects the overall covariance scale.

In summary, accurate and stable covariance recovery relies on rapid spectral decay of the true covariance, non-vanishing spectral gaps, controlled trunk and branch errors, and sufficiently accurate finite-sample covariance estimation.

3 Results and Discussions

To ensure consistent evaluation across the case studies, a unified set of metrics is used to assess mean prediction, covariance recovery, spatial correlation, and prediction intervals. As a supplementary enhancement module, deep ensembles and post-hoc uncertainty calibration are introduced to account for epistemic uncertainty and improve interval coverage. The low-dimensional coefficient-space formulation keeps the additional cost of finite ensembles computationally manageable.

- (1) **Normalized Frobenius covariance error.** Covariance estimation error is commonly evaluated under the operator and Frobenius norms, which characterize different aspects of estimation accuracy and lead to different optimal procedures [55]. Since this study focuses on the overall covariance structure, the normalized Frobenius error is adopted:

$$E_\Sigma = \frac{\|\hat{\Sigma}_{\text{pred}} - \hat{\Sigma}_{\text{ref}}\|_F}{\|\hat{\Sigma}_{\text{ref}}\|_F}, \quad (70)$$

where $\hat{\Sigma}_{\text{ref}}$ and $\hat{\Sigma}_{\text{pred}}$ denote the reference empirical covariance and model prediction, respectively. Following the decomposition in Eq. (50), three error curves are compared over p :

- a) **Oracle singular value decomposition (SVD) curve.** The optimal rank- p approximation $\hat{\Sigma}_{\text{SVD}}^{(p)}$ retains the leading p modes of $\hat{\Sigma}_{\text{ref}}$ and represents T_1 .
 - b) **Learned-subspace oracle curve.** The reference covariance is projected onto the trunk-learned subspace as $\tilde{\mathbf{Q}}^*[(\tilde{\mathbf{Q}}^*)^\top \hat{\Sigma}_{\text{ref}} \tilde{\mathbf{Q}}^*](\tilde{\mathbf{Q}}^*)^\top$. Using the reference coefficient covariance excludes branch error, so this curve represents $T_1 + T_2$.
 - c) **Final model prediction curve.** The complete model estimates $\hat{\Sigma}_{2\text{step}} = \tilde{\mathbf{Q}}^*(\hat{\mathbf{B}}_\mu + \hat{\mathbf{B}}_\sigma)(\tilde{\mathbf{Q}}^*)^\top$, which incorporates $T_1 + T_2 + T_3$.
- (2) **Multi-anchor correlation plots.** These plots assess off-diagonal dependence across the output domain. The correlation coefficient is

$$\rho_{ij} = \frac{\Sigma_{ij}}{\sqrt{\Sigma_{ii}\Sigma_{jj}}}. \quad (71)$$

Reference and predicted correlation profiles are compared at several anchor points to evaluate the recovered correlation range, decay, and local structure.

- (3) **Prediction interval coverage and width.** Prediction interval coverage probability (PICP) and mean prediction interval width (MPIW) measure calibration and sharpness, respectively [56]:

$$\text{PICP}_\gamma = \frac{1}{N_{\text{test}}M} \sum_{i=1}^{N_{\text{test}}} \sum_{m=1}^M \mathbb{I}\left(s_m^{(i)} \in \left[L_{\gamma,m}^{(i)}, U_{\gamma,m}^{(i)}\right]\right), \quad \text{MPIW}_\gamma = \frac{1}{N_{\text{test}}M} \sum_{i=1}^{N_{\text{test}}} \sum_{m=1}^M \left(U_{\gamma,m}^{(i)} - L_{\gamma,m}^{(i)}\right). \quad (72)$$

Here, N_{test} is the number of test samples, $\mathbb{I}(\cdot)$ is the indicator function, and the interval is defined by $L_{\gamma,m}^{(i)} = \hat{\mu}_m^{(i)} - \kappa_\gamma \hat{\sigma}_m^{(i)}$ and $U_{\gamma,m}^{(i)} = \hat{\mu}_m^{(i)} + \kappa_\gamma \hat{\sigma}_m^{(i)}$. Under the Gaussian assumption, $\kappa_\gamma = z_{(1+\gamma)/2}$; under split conformal prediction (CP), it is determined by the calibration-score quantile. PICP should approach γ , while a smaller MPIW indicates sharper intervals at comparable coverage.

3.1 Reaction–Diffusion Equation

Reaction–diffusion equations describe the spatiotemporal evolution of concentration fields governed by diffusion and local reactions [57], with applications in chemical reaction engineering [58], biological morphogenesis [59], and spatial ecology [60]. We consider

$$\begin{aligned} \frac{\partial s}{\partial t} &= \nu \frac{\partial^2 s}{\partial x^2} + ks^2 + u(x), & (x, t) &\in [0, 1] \times [0, 1], \\ s(x, 0) &= 0, & x &\in [0, 1], \\ s(0, t) &= s(1, t) = 0, & t &\in [0, 1]. \end{aligned} \quad (73)$$

where ν and k are the diffusion and reaction coefficients, respectively, and are both set to 0.01. The random source function $u(x)$ is mapped to the spatiotemporal solution field $s(x, t)$ through

$$\mathcal{G} : u(x) \mapsto s(x, t). \quad (74)$$

Sources of Uncertainty. Random source terms may arise from fluctuating generation rates, spatial heterogeneity, and environmental perturbations. Examples include stochastic gene expression in morphogen formation [61] and randomly distributed catalytic sources in systems with disordered kinetics [62]. Their propagation through the governing dynamics induces uncertainty and correlation across the output domain. Accordingly, this example evaluates the recovery of the mean response and output correlation structure under a random input function $u(x)$.

Data Generation. The source function is sampled from a Gaussian process with the squared-exponential covariance kernel

$$K(x, x') = \exp\left(-\frac{|x - x'|^2}{2\ell^2}\right), \quad (75)$$

where ℓ is the correlation length. For each realization, Eq. (73) is solved by a finite-difference scheme with the spatial and temporal domains each discretized into 100 uniformly spaced points, producing a 100×100 response field. A total of 2,200 samples are generated, including 1,000 training samples, 1,000 test samples, and 200 calibration samples.

3.1.1 Prediction Accuracy and Generalization Performance

Training and In-Distribution Accuracy

We first evaluate the basic fitting capability and generalization stability of the two methods within the training distribution. Both the training set and the in-distribution test set were generated with the correlation length $\ell = 0.2$. The training loss curves of the two methods are shown in Figs. 2 and 3. Both methods exhibit stable convergence, ensuring that the subsequent accuracy comparison is conducted under adequate training. For probabilistic branch training, a mean-only warm-up with fixed variance is adopted before jointly optimizing the mean and variance [23]; accordingly, the branch loss curve exhibits a distinct transition at the end of the warm-up stage and subsequently converges steadily under joint NLL optimization. The same training strategy is adopted in all subsequent examples.

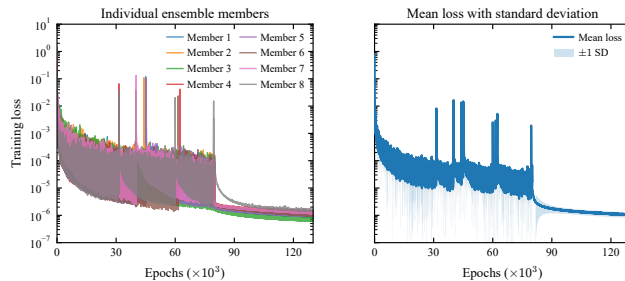


Fig. 2. Training loss curves of Prob-DeepONet: individual ensemble members (left) and the ensemble mean with a ± 1 standard deviation band (right).

The quantitative errors under the same input distribution are summarized in Table 1. Prob-DeepONet achieves lower training errors, whereas two-step MV-DeepONet yields lower mean and maximum in-distribution test errors. The relative degradation in predictive accuracy from the training set to unseen in-distribution samples is quantified by the ratio of the mean in-distribution test error to the mean training error. This ratio is $0.0129/0.0010 = 12.9$ for Prob-DeepONet and $0.0060/0.0037 \approx 1.62$ for two-step MV-DeepONet. These results indicate that the fixed orthogonal basis and subsequent coefficient regression provide stronger structural constraints and more stable in-distribution generalization. The out-of-distribution (OOD) results below further examine this behavior.

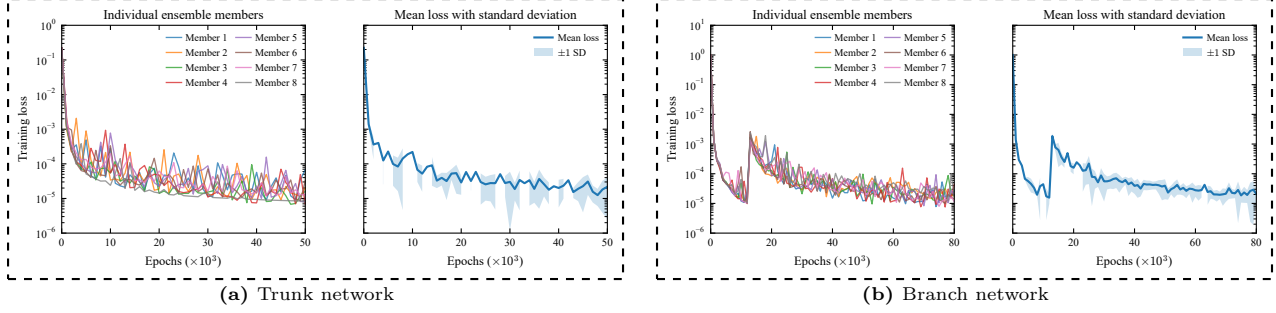


Fig. 3. Training loss curves of two-step MV-DeepONet for the trunk (a) and branch (b) networks: individual ensemble members (left) and the ensemble mean with a ± 1 standard deviation band (right).

Table 1. Mean and maximum relative L_2 errors on the training set and the in-distribution test set.

Metric	Prob-DeepONet	two-step MV-DeepONet
Mean relative L_2 error (training)	0.0010	0.0037
Maximum relative L_2 error (training)	0.0046	0.0099
Mean relative L_2 error (in-distribution)	0.0129	0.0060
Maximum relative L_2 error (in-distribution)	0.1025	0.0391

Out-of-Distribution Generalization Performance

Table 2 compares the OOD performance over input correlation lengths $\ell \in [0.001, 1.0]$, with $\ell = 0.2$ used for training. Both methods maintain low errors for moderate and large correlation lengths ($\ell \geq 0.18$), while their difference becomes evident for $\ell \leq 0.15$. Under the most severe OOD shifts, Prob-DeepONet yields errors of 44.15% at $\ell = 0.005$ and 109.64% at $\ell = 0.001$. The corresponding errors of two-step MV-DeepONet are 12.26% and 33.09%, respectively. Across all correlation lengths, two-step MV-DeepONet reduces the mean error from 13.43% to 5.82%, corresponding to a reduction of approximately 56.7%.

Table 2. Relative L_2 errors of Prob-DeepONet and two-step MV-DeepONet for out-of-distribution samples at different input correlation lengths.

Correlation length ℓ	Prob-DeepONet	two-step MV-DeepONet
1.0	0.488%	0.234%
0.8	0.461%	0.246%
0.6	0.413%	0.282%
0.5	0.435%	0.319%
0.4	0.318%	0.343%
0.3	0.368%	0.309%
0.25	0.542%	0.300%
0.22	0.513%	0.290%
0.18	0.708%	0.273%
0.15	2.932%	0.530%
0.1	7.391%	4.916%
0.08	6.401%	6.462%
0.05	9.882%	8.762%
0.03	20.576%	10.734%
0.01	19.245%	12.243%
0.005	44.152%	12.260%
0.002	17.289%	13.125%
0.001	109.639%	33.094%
Mean error	13.431%	5.818%

To illustrate the predictive behavior under varying degrees of distribution shift, Fig. 4 presents representative samples at four correlation lengths: $\ell = 0.5$ corresponds to a smooth-input regime with a large correlation length; $\ell = 0.15$ represents a mild distribution shift relatively close to the training distribution; $\ell = 0.05$ represents a moderate OOD condition; and $\ell = 0.005$ represents an extreme OOD condition.

Both methods remain accurate under mild distribution shifts. As ℓ decreases, the input source becomes increasingly oscillatory and induces more complex local variations in the solution. Under severe shifts, Prob-DeepONet exhibits pronounced prediction deviations, whereas two-step MV-DeepONet retains the principal solution structure with lower prediction errors.

The contrasting OOD behavior of the two methods is related to differences in both their learned output representations and their formulations of the input-to-coefficient mapping. Prob-DeepONet jointly optimizes the trunk basis and input-dependent coefficients, allowing them to compensate for each other on the training samples. This coupling

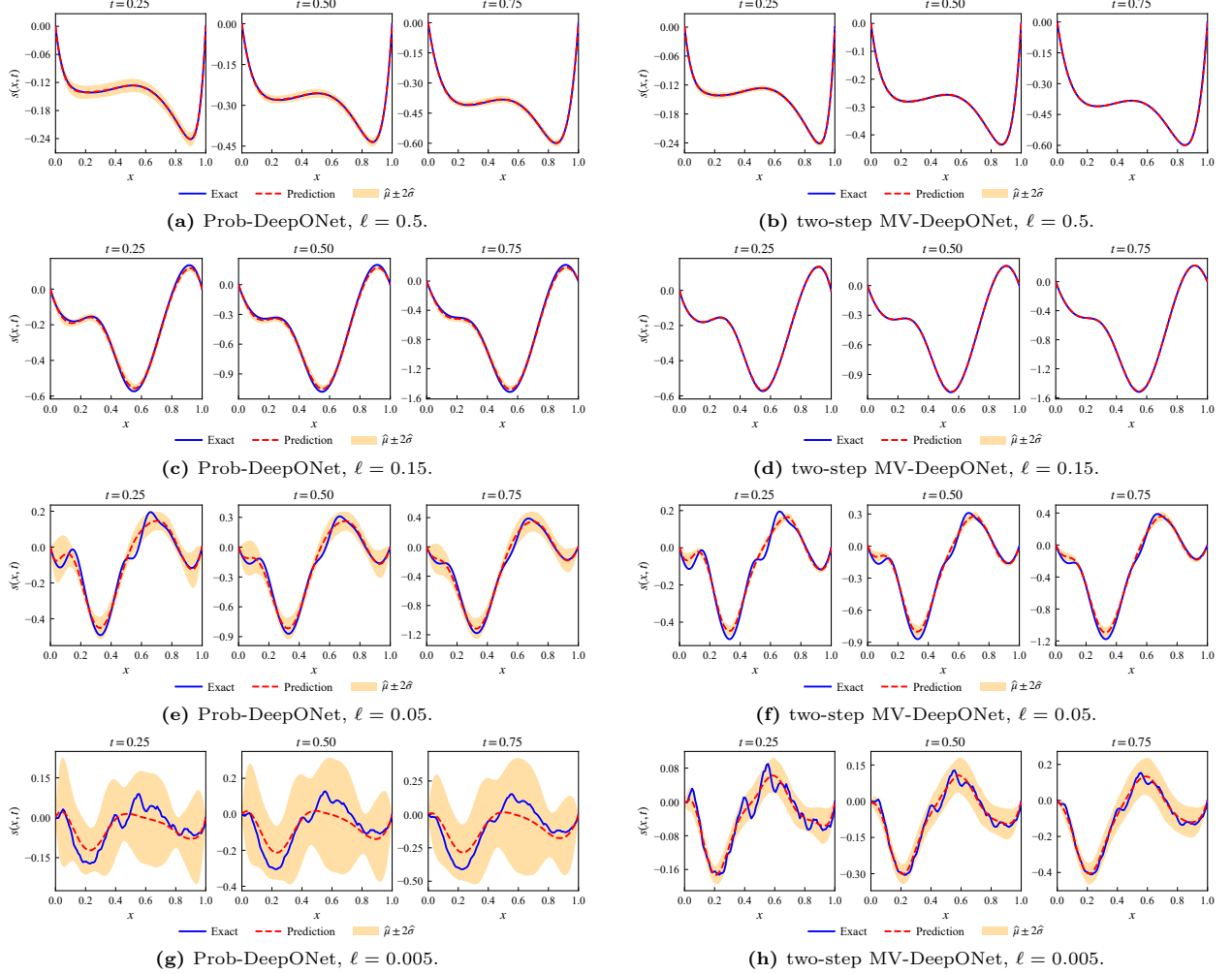


Fig. 4. Predictions and predictive uncertainty bands of Prob-DeepONet (left column) and two-step MV-DeepONet (right column) for representative out-of-distribution samples at different input correlation lengths.

facilitates training-data fitting but may produce a less stable representation when the input distribution changes. In two-step MV-DeepONet, a shared output basis is first learned from the output snapshots, orthogonalized, and then fixed, after which the branch network learns the corresponding modal coefficients. This decoupling provides a better-conditioned and more stable coordinate system for coefficient regression while reducing the co-adaptation between the basis functions and coefficients.

The role of the orthogonal basis can be expressed through the output error decomposition

$$\begin{aligned}\hat{\mathbf{s}} - \mathbf{s} &= \tilde{\mathbf{Q}}^* \hat{\mathbf{c}} - \mathbf{P}\mathbf{s} + \mathbf{P}\mathbf{s} - \mathbf{s} \\ &= \tilde{\mathbf{Q}}^* (\hat{\mathbf{c}} - \mathbf{c}_{\tilde{\mathbf{Q}}}) - (\mathbf{I} - \mathbf{P})\mathbf{s}.\end{aligned}\quad (76)$$

Here, $\hat{\mathbf{s}} = \tilde{\mathbf{Q}}^* \hat{\mathbf{c}}$ is the reconstructed prediction, $\hat{\mathbf{c}}$ is the coefficient vector predicted by the branch network, and $\mathbf{c}_{\tilde{\mathbf{Q}}} = (\tilde{\mathbf{Q}}^*)^\top \mathbf{s}$ denotes the projection coefficients of the true output. Moreover, $\mathbf{P} = \tilde{\mathbf{Q}}^* (\tilde{\mathbf{Q}}^*)^\top$ is the orthogonal projector onto the learned output subspace. The first term represents the coefficient regression error within the learned subspace, whereas the second is the component of the true output that cannot be represented by that subspace. Since these two terms lie in mutually orthogonal subspaces,

$$\|\hat{\mathbf{s}} - \mathbf{s}\|_2^2 = \left\| \tilde{\mathbf{Q}}^* (\hat{\mathbf{c}} - \mathbf{c}_{\tilde{\mathbf{Q}}}) \right\|_2^2 + \|(\mathbf{I} - \mathbf{P})\mathbf{s}\|_2^2. \quad (77)$$

Moreover, the orthonormality of $\tilde{\mathbf{Q}}^*$ gives

$$\left\| \tilde{\mathbf{Q}}^* (\hat{\mathbf{c}} - \mathbf{c}_{\tilde{\mathbf{Q}}}) \right\|_2 = \left\| \hat{\mathbf{c}} - \mathbf{c}_{\tilde{\mathbf{Q}}} \right\|_2, \quad (78)$$

so coefficient prediction errors are not amplified during output reconstruction.

For a general nonorthogonal trunk matrix Φ , as used in the jointly trained representation of Prob-DeepONet, only

$$\|\Phi(\hat{\mathbf{c}} - \mathbf{c}_\Phi)\|_2 \leq \|\Phi\|_2 \|\hat{\mathbf{c}} - \mathbf{c}_\Phi\|_2 \quad (79)$$

is guaranteed. Strongly correlated basis functions may therefore lead to an ill-conditioned coordinate representation and amplify coefficient errors during reconstruction.

Fixing the shared basis also introduces a structural regularization effect, encouraging the model to preserve dominant solution structures shared across the training samples rather than overadapting the output representation to specific local details. Consequently, when OOD inputs introduce previously unseen high-frequency components or local structures, two-step MV-DeepONet may not fully reconstruct these new details, but it can generally retain the principal structure of the solution field and exhibit more controlled error growth. By contrast, the training-specific compensation learned by Prob-DeepONet through joint optimization may become ineffective under severe distribution shifts, leading to larger prediction deviations.

A similar pattern is observed in the uncertainty bands. As shown in Fig. 4, the bands of Prob-DeepONet become increasingly broad as ℓ decreases and expand over much of the response profile under severe distribution shifts. In contrast, two-step MV-DeepONet generally produces tighter bands whose widths vary more coherently with the local solution structure.

This behavior is consistent with the covariance parameterizations of the two methods. Prob-DeepONet estimates the marginal variance separately at each output point under a pointwise diagonal covariance assumption and therefore does not explicitly couple uncertainty across different points in the output domain. Local prediction discrepancies may consequently be accommodated through pointwise variance inflation, producing broad uncertainty bands with limited organization across the output domain. In contrast, two-step MV-DeepONet predicts the variances of the modal coefficients and maps them to the output domain through the shared basis functions. Each modal coefficient variance contributes to the marginal uncertainty at multiple output points through the shared basis. This shared contribution induces coordinated variations in marginal uncertainty and off-diagonal covariance across the output domain.

3.1.2 Analysis of Output Covariance Recovery Accuracy

Output covariance recovery is evaluated using the normalized Frobenius error defined in Eq. (70). Multi-anchor correlation plots are further used to examine local correlation structures across the output domain. Since Prob-DeepONet assumes a pointwise diagonal covariance and does not explicitly represent off-diagonal dependence, this analysis focuses on two-step MV-DeepONet.

Covariance Error Decomposition and Quantitative Accuracy Analysis

Following the decomposition in Eq. (50), the Oracle SVD curve characterizes the intrinsic rank- p truncation error. The separation between the Oracle SVD and Learned-subspace oracle curves primarily reflects the trunk-subspace learning error, whereas the separation between the Learned-subspace oracle and Final model prediction curves primarily reflects the branch coefficient-covariance prediction error. Fig. 5 presents the three error curves, and Table 3 reports the minimum number of modes required to reach each target normalized Frobenius error.

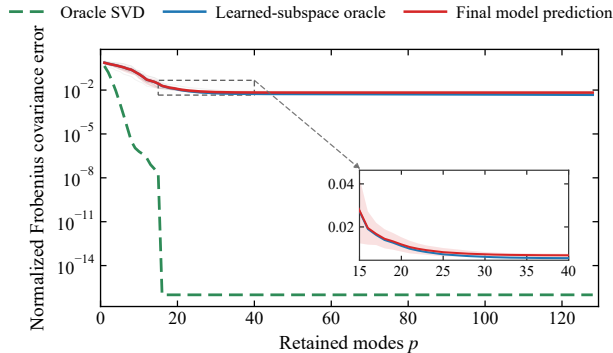


Fig. 5. Normalized Frobenius covariance error as a function of the number of retained modes.

(1) **Oracle SVD curve.** The Oracle SVD error decreases rapidly and reaches approximately 10^{-13} for $p \geq 15$, indicating that the low-rank compressibility assumption introduced in Section 2.4.4 is naturally satisfied in the reaction-diffusion problem, and the abrupt decrease near $p \approx 15$ also supports the nondegenerate spectral-gap assumption $\gamma_p > 0$. Once the effective numerical rank is reached, T_1 becomes negligible and is no longer the principal limitation on covariance recovery.

(2) **Comparison between the Learned-subspace oracle curve and the Final model prediction curve.**

Table 3. Minimum numbers of retained modes required to achieve different target normalized Frobenius covariance-error levels for the three covariance-recovery curves.

Target error	Oracle SVD	Learned-subspace oracle	Final model prediction
Error $\leq 20.00\%$	$p \geq 2$	$p \geq 9$	$p \geq 9$
Error $\leq 10.00\%$	$p \geq 3$	$p \geq 11$	$p \geq 11$
Error $\leq 5.00\%$	$p \geq 3$	$p \geq 13$	$p \geq 13$
Error $\leq 2.00\%$	$p \geq 4$	$p \geq 16$	$p \geq 16$
Error $\leq 1.00\%$	$p \geq 4$	$p \geq 21$	$p \geq 22$

The two curves are nearly indistinguishable over the full range of p . For $p \in [15, 40]$, their separation remains approximately 1×10^{-3} , while both errors decrease from approximately 2×10^{-1} to 6×10^{-3} . These results support the following observations:

1. The contribution of T_3 is substantially smaller than that of T_2 . This indicates that the branch approximation and optimization errors have been controlled to a relatively low level. Branch coefficient-covariance prediction is therefore not the principal limitation on covariance recovery.
2. The dominant contribution is T_2 , which is associated with the difference between the learned projector $\mathbf{P} = \tilde{\mathbf{Q}}^*(\tilde{\mathbf{Q}}^*)^\top$ and the true principal-subspace projector $\mathbf{P}^*$. As discussed in Section 2.4.4, this difference contains two main components. The first is the excess trunk reconstruction risk Δ_T relative to the empirically optimal projector $\hat{\mathbf{P}}^*$. The second arises from finite-sample covariance estimation, characterized by $\|\hat{\Sigma} - \Sigma\|_F = \mathcal{O}_\mathbb{P}(\sqrt{MK}^{-1/2})$, where M is the output dimension and K is the number of training samples. Its influence on the estimated subspace is further controlled by the corresponding spectral gap. For this problem, $M = 10^4$ and $K = 10^3$. The resulting relative statistical-error scale is

$$\frac{\sqrt{MK}^{-1/2}}{\|\hat{\Sigma}\|_F} \approx 2.0 \times 10^{-3}. \quad (80)$$

Because $\mathcal{O}_\mathbb{P}(\cdot)$ specifies only an asymptotic order, this value should be interpreted as an approximate scale rather than an exact error. It is about one third of the observed $T_2 \approx 6 \times 10^{-3}$, suggesting that Δ_T is the larger contribution within T_2 .

This conclusion concerns only the relative importance of the two contributions within T_2 and does not imply inadequate trunk training. The trunk loss in Fig. 3 stabilizes at a low level after approximately 50,000 epochs. Moreover, $T_2 \approx 6 \times 10^{-3}$ is of the same order as the 0.60% mean relative L_2 error of two-step MV-DeepONet on the in-distribution test set (Table 1). In principle, Δ_T could be further reduced by increasing the trunk-network capacity, extending the training duration, or adopting more refined optimization strategies. However, because the current covariance recovery error is already of the same order as the mean prediction error, further hyperparameter refinement would provide limited improvement while substantially increasing the training cost. The objective here is to determine whether the covariance matrix can be stably recovered to a practically useful level of accuracy, rather than to pursue the lowest attainable numerical error.

In summary, covariance recovery for the reaction–diffusion problem shows the clear hierarchy $T_1 \ll T_2, T_3 \ll T_2$. Within T_2 , the excess trunk reconstruction risk Δ_T is the principal source, whereas the finite-sample statistical contribution is comparatively small. These results are consistent with the theoretical decomposition in Section 2.4.4 and demonstrate that two-step MV-DeepONet recovers the output covariance with an error comparable in magnitude to its mean prediction error.

Recovery of Local Correlation Structures: Multi-Anchor Correlation Map Assessment

The normalized Frobenius error measures the overall covariance discrepancy but does not reveal whether local correlations across the output domain are accurately recovered. Multi-anchor correlation maps are therefore used for a more detailed structural assessment. Three anchor points are selected from the left, central, and right regions of the output domain $(x, t) \in [0, 1]^2$. For each anchor, the reference and predicted correlation maps are compared together with their absolute error. The resulting maps are shown in Fig. 6.

Overall pattern consistency. For all three anchors, the predicted maps reproduce the principal vertical correlation bands, the smooth decay away from the anchor location, and the weak negative correlations in the far field. The mean row-wise correlation error is approximately 6.97×10^{-2} , indicating a relatively small structural discrepancy. These results show that two-step MV-DeepONet recovers not only the global covariance magnitude but also its local correlation patterns.

Physical interpretation of the correlation structure. The vertical bands arise primarily from the time-independent random source $u(x)$, which continuously drives the system throughout its evolution. At a fixed spatial

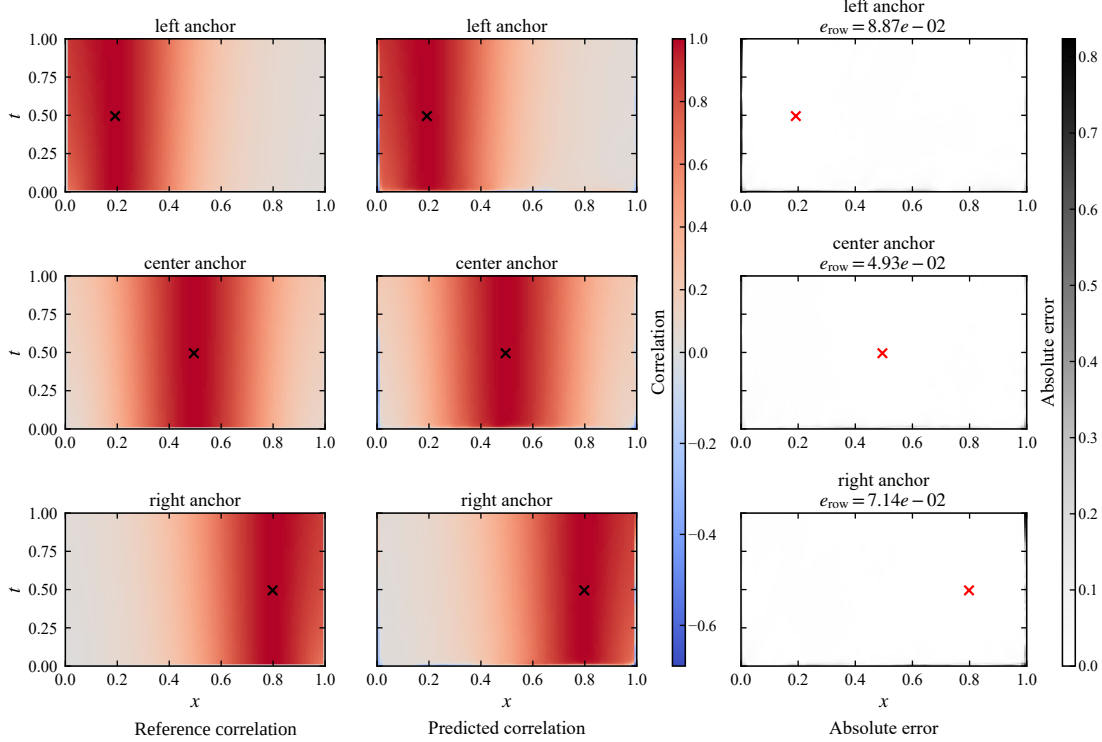


Fig. 6. Multi-anchor correlation maps showing the reference correlations, predicted correlations, and corresponding absolute errors for three anchor points, where e_{row} denotes the relative L_2 error of the predicted correlation map for each anchor point.

location $x = x_a$, responses at different times are influenced by similar local components of the same random source and therefore remain strongly correlated. This shared influence weakens as the spatial distance from x_a increases, leading to correlation decay away from the anchor. The combination of persistent temporal correlation and spatial decay produces the observed vertical-band structure.

The recovery of this physical structure highlights a core advantage of two-step MV-DeepONet. The trunk stage first learns the shared output-space basis $\mathbf{Q}^*$, which captures the dominant structures of the output field. The branch stage then models the input-dependent variances of the modal coefficients in the low-dimensional coefficient space. Mapping these coefficient uncertainties back through the shared basis induces an output covariance that naturally inherits the spatial patterns encoded by the columns of $\mathbf{Q}^*$.

3.1.3 Prediction Interval Calibration and Coverage Analysis

Accurate covariance recovery does not necessarily ensure nominal prediction interval coverage. For engineering uncertainty quantification, prediction intervals should provide reliable coverage while remaining sufficiently narrow. We therefore apply CP as a post-hoc calibration procedure and evaluate its effectiveness using the PICP and MPIW defined in Eq. (72).

Table 4. PICP and MPIW of Prob-DeepONet and two-step MV-DeepONet before and after CP calibration.

Method	Nominal coverage (%)	Before CP calibration		After CP calibration	
		PICP (%)	MPIW	PICP (%)	MPIW
Prob-DeepONet	90	99.96	4.62×10^{-2}	90.52	1.60×10^{-2}
	95	99.99	5.51×10^{-2}	95.30	2.01×10^{-2}
	99	100.00	7.24×10^{-2}	98.97	2.88×10^{-2}
two-step MV-DeepONet	90	98.99	1.60×10^{-2}	89.23	5.10×10^{-3}
	95	99.41	1.90×10^{-2}	94.21	6.95×10^{-3}
	99	99.81	2.50×10^{-2}	98.44	1.34×10^{-2}

Before calibration, both methods exhibit pronounced overcoverage, with PICP values close to 100% at all nominal levels. Their original intervals are therefore overly conservative. Prob-DeepONet also produces substantially wider intervals. At the 95% nominal level, its MPIW is approximately 2.9 times that of two-step MV-DeepONet. The latter thus provides more compact uncertainty intervals even before calibration. After CP calibration, the PICP values move closer to their nominal targets, while the MPIW values decrease substantially. This confirms that CP

effectively reduces excessive coverage and improves interval sharpness. The calibrated PICP values of Prob-DeepONet are slightly closer to the nominal levels, but its intervals remain considerably wider. At the 95% level, its calibrated MPIW is again approximately 2.9 times that of two-step MV-DeepONet.

Overall, both methods achieve coverage close to the prescribed levels after calibration. Two-step MV-DeepONet maintains markedly narrower intervals and thus achieves a better balance between coverage and interval sharpness in the reaction–diffusion problem.

3.2 Burgers Equation

The Burgers equation is a canonical nonlinear convection–diffusion model for studying nonlinear transport, shock formation, and simplified turbulence mechanisms [63]. Its quadratic convective term is analogous to that in the Navier–Stokes equations, making it a widely used benchmark in fluid dynamics, stochastic wave propagation, and numerical method validation [64]. We consider the one-dimensional viscous Burgers equation on $[0, 1]$:

$$\frac{\partial s}{\partial t} + s \frac{\partial s}{\partial x} = \nu \frac{\partial^2 s}{\partial x^2}, \quad (x, t) \in [0, 1] \times [0, 1], \quad (81)$$

subject to the periodic boundary conditions and the initial condition

$$s(0, t) = s(1, t), \quad \frac{\partial s}{\partial x}(0, t) = \frac{\partial s}{\partial x}(1, t), \quad s(x, 0) = u(x). \quad (82)$$

The viscosity coefficient is set to $\nu = 0.01$. Here, the random initial condition $u(x)$ is the input function, and the corresponding solution $s(x, t)$ is the output field. The target operator is

$$\mathcal{G} : u(x) \mapsto s(x, t). \quad (83)$$

Source of Uncertainty. The uncertainty arises from the random initial condition $u(x)$. Random initial velocity fields commonly occur in fluid transport [65] and the early evolution of turbulence [64]. Through nonlinear convection, these perturbations can alter waveform translation, steepening, and shock location. The resulting changes in the output distribution and correlation structure make this problem a suitable benchmark for nonlinear uncertainty propagation and covariance recovery.

Data Generation. The initial condition $u(x)$ is modeled as a Gaussian random field with covariance operator

$$25^2 (-\Delta + \tau^2 I)^{-2}, \quad (84)$$

where τ controls the correlation scale and spectral composition of the random field. A larger τ corresponds to a shorter characteristic length scale and richer high-frequency content. The training distribution is defined by $\tau = 25$. Consistent with the periodic boundary conditions, the random fields are generated using a Fourier spectral expansion and evaluated at $m = 101$ uniformly spaced spatial locations as inputs to the branch network. For each initial condition, Eq. (81) is solved on $(x, t) \in [0, 1]^2$ using a Fourier spectral method implemented in Chebfun. The solution is evaluated at 101 spatial and 101 temporal points, yielding a 101×101 spatiotemporal output matrix. A total of 2,000 samples are generated, comprising 1,000 training samples, 500 test samples, and 500 calibration samples.

3.2.1 Prediction Accuracy and Generalization Performance

Training and In-Distribution Accuracy

Figures 7 and 8 show stable convergence for both methods. Their training and in-distribution errors are reported in Table 5. Prob-DeepONet achieves lower training errors, whereas two-step MV-DeepONet yields lower mean and maximum in-distribution errors. In particular, the maximum in-distribution error decreases from 9.5806% to 5.6790%, indicating more stable performance on challenging in-distribution samples.

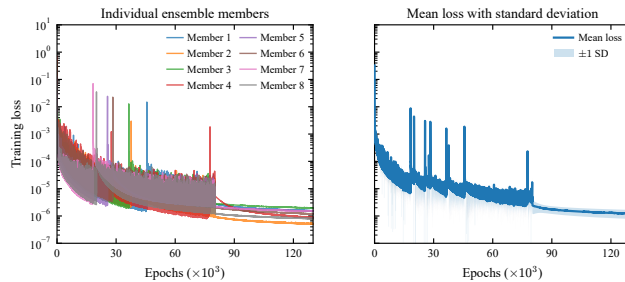


Fig. 7. Training loss curves of Prob-DeepONet: individual ensemble members (left) and the ensemble mean with a ± 1 standard deviation band (right).

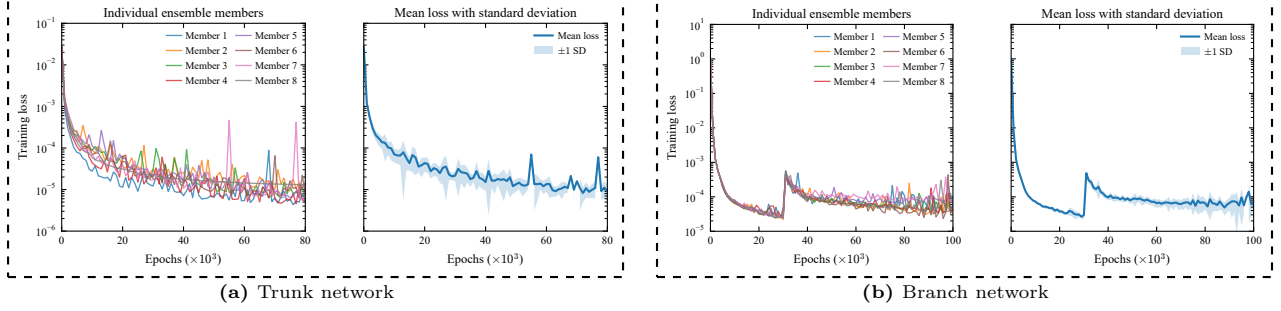


Fig. 8. Training loss curves of two-step MV-DeepONet for the trunk (a) and branch (b) networks: individual ensemble members (left) and the ensemble mean with a ± 1 standard deviation band (right).

Table 5. Mean and maximum relative L_2 errors on the training set and the in-distribution test set.

Metric	Prob-DeepONet	two-step MV-DeepONet
Mean relative L_2 error (training)	0.4440%	0.6040%
Maximum relative L_2 error (training)	2.6840%	3.8693%
Mean relative L_2 error (in-distribution)	1.6643%	1.5891%
Maximum relative L_2 error (in-distribution)	9.5806%	5.6790%

Out-of-Distribution Generalization Performance

In this example, OOD generalization is evaluated by varying the random-field parameter τ from its training value of 25. A larger τ corresponds to a shorter correlation scale and richer high-frequency content. The resulting relative L_2 errors are reported in Table 6.

Prob-DeepONet performs better only at $\tau = 5$ and 8, whereas two-step MV-DeepONet achieves lower errors for all remaining cases. Under the extreme shift $\tau = 400$, their errors are 293.932% and 68.931%, respectively. Across all conditions, two-step MV-DeepONet reduces the mean error from 59.2977% to 27.3800%, corresponding to a reduction of approximately 53.8%. Fig. 9 shows representative results at $\tau = 20, 64, 160$, and 400, spanning mild to extreme OOD conditions.

The qualitative differences observed for the challenging in-distribution samples become more pronounced under OOD shifts. Prob-DeepONet produces increasingly broad, nearly horizontal tubular uncertainty bands with limited adaptation to the local waveform, while its predictive mean deviates substantially under strong shifts. In contrast, two-step MV-DeepONet better preserves the overall solution profile and produces tighter, more structure-adaptive uncertainty bands that remain narrow in smooth regions and widen moderately near sharp transitions. Although both methods deteriorate under severe OOD conditions, the two-step model exhibits substantially greater predictive robustness.

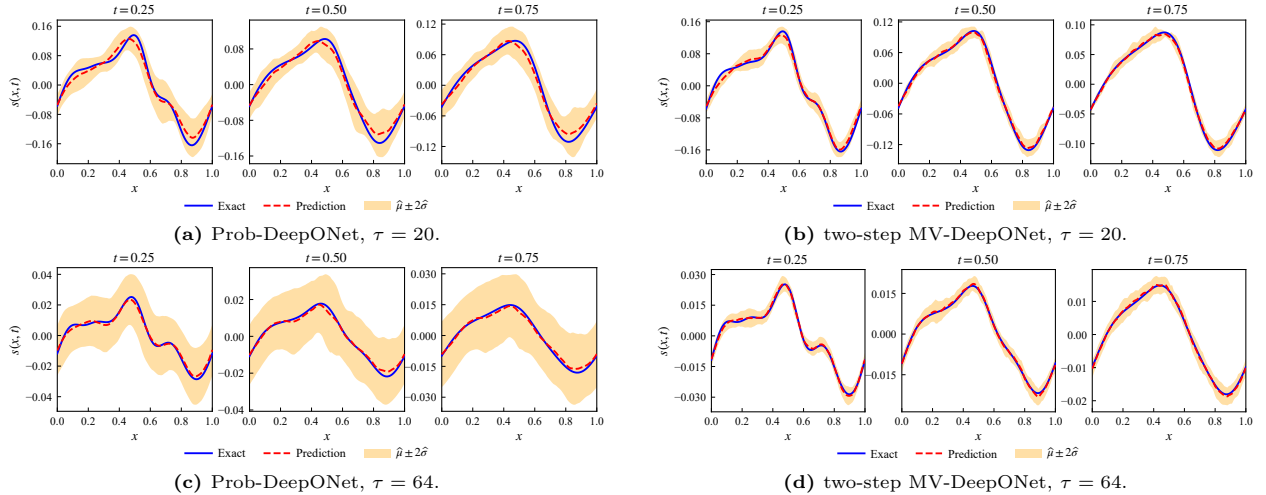


Fig. 9. Predictions and predictive uncertainty bands of Prob-DeepONet (left column) and two-step MV-DeepONet (right column) for representative out-of-distribution samples at different values of the random-field parameter τ .

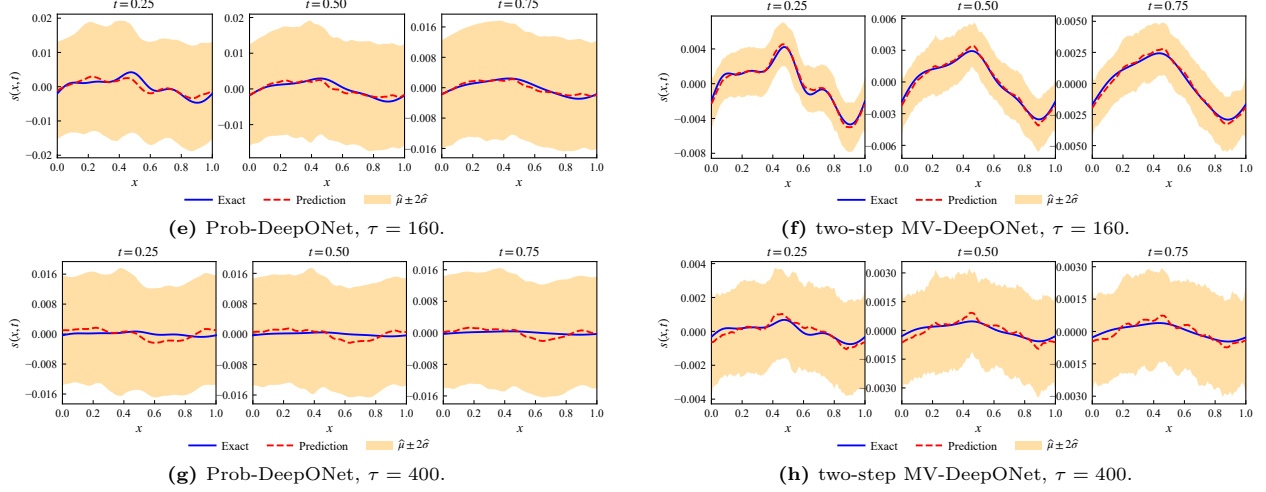


Fig. 9. Predictions and predictive uncertainty bands of Prob-DeepONet (left column) and two-step MV-DeepONet (right column) for representative out-of-distribution samples at different values of the random-field parameter τ (continued).

Table 6. Relative L_2 errors of Prob-DeepONet and two-step MV-DeepONet for out-of-distribution samples at different values of the random-field parameter τ .

Random-field parameter τ	Prob-DeepONet	two-step MV-DeepONet
5	27.758%	37.269%
8	17.160%	23.467%
12	15.349%	13.473%
16	18.630%	12.212%
20	19.809%	12.352%
25	20.117%	12.988%
32	21.091%	14.034%
40	23.625%	15.542%
50	23.754%	17.696%
64	26.339%	20.356%
80	32.651%	22.854%
100	39.915%	25.476%
128	45.414%	28.776%
160	54.637%	32.315%
200	75.009%	36.890%
256	121.548%	44.173%
320	190.620%	54.036%
400	293.932%	68.931%
Mean error	59.2977%	27.3800%

3.2.2 Analysis of Output Covariance Recovery Accuracy

This part further quantifies the output covariance recovery accuracy for the Burgers equation using the normalized Frobenius covariance error defined in Eq. (70).

Covariance Error Decomposition and Quantitative Accuracy Analysis

Figure 10 shows how the covariance errors vary with the number of retained modes p , while Table 7 reports the minimum numbers of modes required to reach different target normalized Frobenius errors.

Table 7. Minimum numbers of retained modes required to achieve different target normalized Frobenius covariance-error levels for the three covariance-recovery curves.

Target error	Oracle SVD	Learned-subspace oracle	Final model prediction
Error $\leq 20.00\%$	$p \geq 2$	$p \geq 8$	$p \geq 8$
Error $\leq 10.00\%$	$p \geq 2$	$p \geq 13$	$p \geq 13$
Error $\leq 5.00\%$	$p \geq 4$	$p \geq 19$	$p \geq 19$
Error $\leq 2.00\%$	$p \geq 4$	$p \geq 29$	$p \geq 30$
Error $\leq 1.00\%$	$p \geq 6$	$p \geq 43$	$p \geq 50$

(1) **Oracle SVD curve.** The Oracle SVD error falls below 5% for $p \geq 4$ and below 1% for $p \geq 6$, indicating strong low-rank compressibility of the reference covariance. Although its decay is slower than in the reaction-diffusion problem, low-rank truncation is not the main source of covariance recovery error.

(2) **Comparison between the Learned-subspace oracle curve and the Final model prediction curve.** The two curves follow similar decreasing trends over the full range of p . Their errors decline from approximately

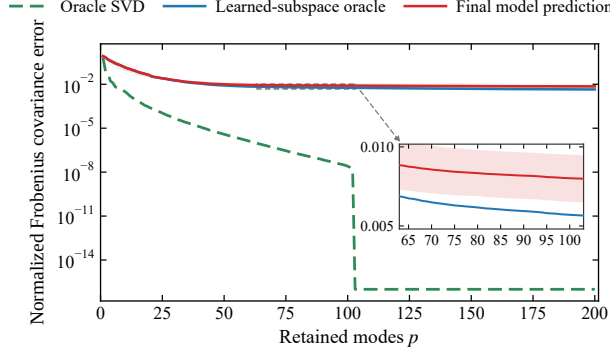


Fig. 10. Normalized Frobenius covariance error as a function of the number of retained modes.

2×10^{-1} to 8×10^{-3} . The enlarged inset shows that the Final model prediction remains approximately $1\text{--}2 \times 10^{-3}$ above the Learned-subspace oracle. Two observations follow:

1. The contribution of T_3 is clearly visible but remains smaller than that of T_2 . The branch approximation and optimization errors are not the bottleneck here.
2. The covariance recovery error is primarily governed by T_2 . For $M = 101 \times 101 = 10,201$ and $K = 10^3$, the corresponding relative statistical-error scale is

$$\frac{\sqrt{MK}^{-1/2}}{\|\hat{\Sigma}\|_F} \approx 1.74 \times 10^{-2}. \quad (85)$$

This estimate is larger than the observed $T_2 \approx 8 \times 10^{-3}$. Since the asymptotic relation provides only an order estimate, it does not permit a reliable separation of the contributions of Δ_T and finite-sample statistical error within T_2 . The final normalized Frobenius covariance error is approximately 8×10^{-3} , which is comparable in magnitude to the mean relative L_2 prediction error of 1.59% reported in Table 5. The method therefore achieves stable and practically useful covariance recovery despite the stronger nonlinearity of the Burgers problem.

In summary, covariance recovery for the Burgers equation exhibits the hierarchy $T_1 < T_2$ and $T_3 \lesssim T_2$. The overall recovery error is nonetheless of the same order as the mean prediction error, showing that the observed covariance recovery behavior is consistent with the theoretical decomposition in Section 2.4.4.

Recovery of Local Correlation Structures: Multi-Anchor Correlation Map Assessment

For the Burgers equation, three anchor points are selected from the left, central, and right regions of the output domain $(x, t) \in [0, 1]^2$. Fig. 11 compares the corresponding reference, predicted, and absolute-error correlation maps.

Overall pattern consistency. The predicted maps closely reproduce the reference patterns for all three anchors, with relatively small discrepancies in the absolute-error maps. The mean row-wise correlation error is approximately 1.27×10^{-2} , confirming accurate recovery of the local correlation structure.

Physical interpretation of the correlation structure. Unlike the vertical bands observed in the reaction–diffusion example, the Burgers correlation maps exhibit distinct horizontal bands centered near the anchor time $t = t_a$. They reflect coordinated variations across spatial locations at similar stages of the nonlinear evolution, while the correlation gradually weakens as the temporal distance from the anchor increases. The predicted maps accurately preserve this characteristic structure.

3.2.3 Prediction Interval Calibration and Coverage Analysis

CP is applied to calibrate the prediction intervals for the Burgers equation. The corresponding PICP and MPIW values before and after calibration are reported in Table 8.

Before calibration, both methods exhibit substantial overcoverage. After CP calibration, their PICP values approach the nominal levels, and their interval widths decrease markedly. At the 95% nominal level, the MPIW ratio between Prob-DeepONet and two-step MV-DeepONet decreases from approximately 4.1 to 2.3. Two-step MV-DeepONet therefore retains considerably sharper intervals while achieving comparable coverage.

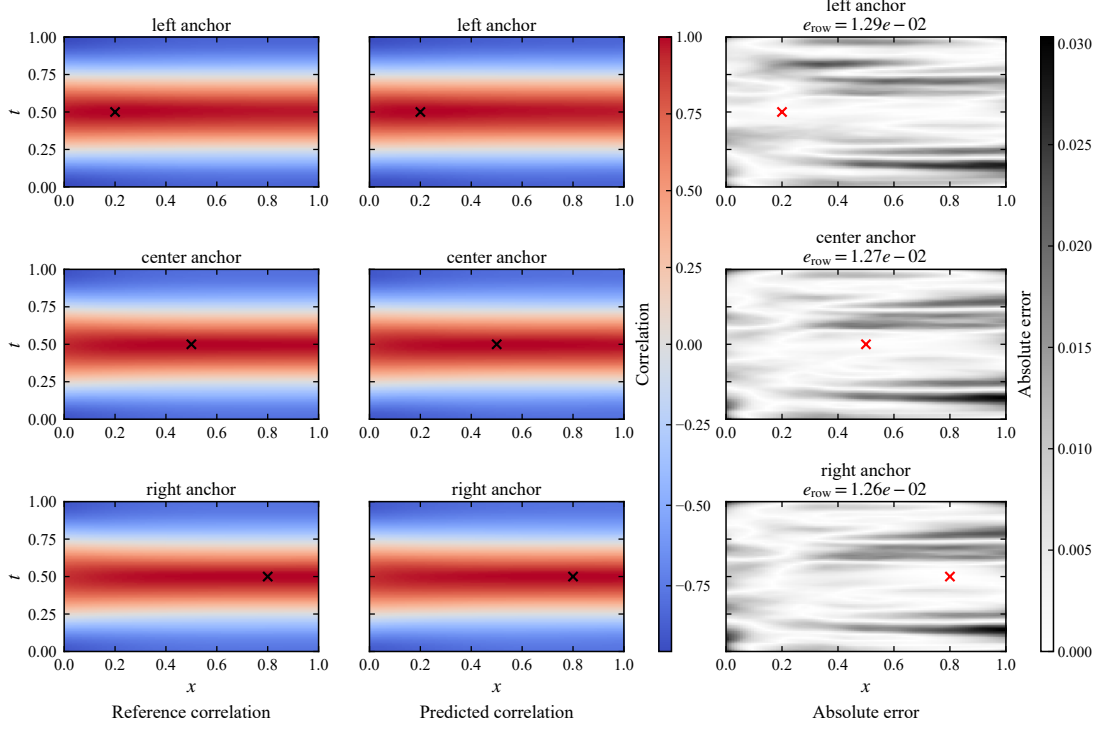


Fig. 11. Multi-anchor correlation maps showing the reference correlations, predicted correlations, and corresponding absolute errors for three anchor points, where e_{row} denotes the relative L_2 error of the predicted correlation map for each anchor point.

Table 8. PICP and MPIW of Prob-DeepONet and two-step MV-DeepONet before and after CP calibration.

Method	Nominal coverage (%)	Before CP calibration		After CP calibration	
		PICP (%)	MPIW	PICP (%)	MPIW
Prob-DeepONet	90	99.96	2.57×10^{-2}	90.52	4.68×10^{-3}
	95	99.98	3.06×10^{-2}	95.39	6.66×10^{-3}
	99	100.00	4.02×10^{-2}	99.11	1.21×10^{-2}
two-step MV-DeepONet	90	99.14	6.30×10^{-3}	90.10	2.12×10^{-3}
	95	99.44	7.51×10^{-3}	94.88	2.94×10^{-3}
	99	99.72	9.87×10^{-3}	98.82	5.54×10^{-3}

3.3 Darcy Equation

The Darcy equation is a classical elliptic model for steady incompressible flow through porous media, with applications in groundwater flow, seepage analysis, and contaminant transport modeling [66]. Its dependence on a spatially heterogeneous permeability field makes it a standard benchmark for uncertainty propagation in random media. We consider the two-dimensional steady Darcy equation on the unit square $\Omega = [0, 1]^2$:

$$\begin{cases} -\nabla \cdot (c(\mathbf{s}) \nabla u(\mathbf{s})) = f(\mathbf{s}), & \mathbf{s} \in \Omega, \\ u(\mathbf{s}) = 0, & \mathbf{s} \in \partial\Omega, \end{cases} \quad (86)$$

where $\mathbf{s} = (x, y)$ denotes the spatial coordinate, $u(\mathbf{s})$ is the pressure field, $c(\mathbf{s})$ is the permeability field, and $f(\mathbf{s}) = 50$ is the prescribed source term. Taking the random permeability $c(\mathbf{s})$ as the input and the solution $u(\mathbf{s})$ as the output response gives the target operator

$$\mathcal{G} : c(\mathbf{s}) \mapsto u(\mathbf{s}). \quad (87)$$

Source of Uncertainty. The uncertainty arises from the permeability field $c(\mathbf{s})$, which governs the flow response but cannot generally be observed throughout the domain. It is therefore commonly modeled as a spatially correlated random field [67]. Its variability induces spatially structured variations and correlations in the pressure field, making this problem suitable for evaluating field prediction and output covariance recovery under input uncertainty.

Data Generation. The permeability is modeled as the lognormal random field $c(\mathbf{s}) = \exp(a(\mathbf{s}))$, where $a(\mathbf{s})$ is a

zero-mean Gaussian process with the squared-exponential covariance kernel

$$\mathcal{C}(\mathbf{s}_1, \mathbf{s}_2) = \sigma^2 \exp \left(-\frac{\|\mathbf{s}_1 - \mathbf{s}_2\|_2^2}{2\ell^2} \right). \quad (88)$$

Here, $\sigma^2 = 1.0$ denotes the variance and $\ell = 0.25$ the correlation length. The latent field $a(\mathbf{s})$ is generated using a truncated Karhunen–Loève expansion with the leading 100 modes, after which $c(\mathbf{s})$ is obtained by exponentiation. For each permeability realization, Eq. (86) is solved using the FEniCS finite element solver. The pressure field is then interpolated onto a uniform 64×64 grid as the output label. A total of 20,000 input–output pairs are generated, comprising 16,000 training samples, 2,000 calibration samples, and 2,000 test samples.

3.3.1 Prediction Accuracy and Generalization Performance

Training and In-Distribution Accuracy

Figures 12 and 13 show stable convergence for both methods. The corresponding errors are summarized in Table 9.

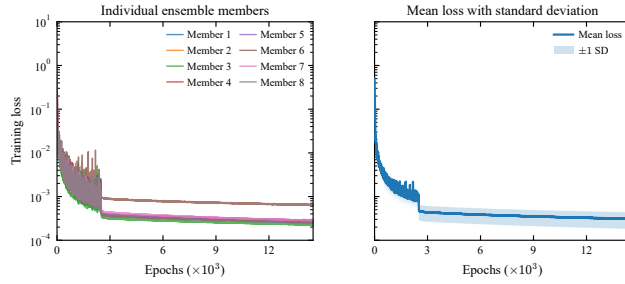


Fig. 12. Training loss curves of Prob-DeepONet: individual ensemble members (left) and the ensemble mean with a ± 1 standard deviation band (right).

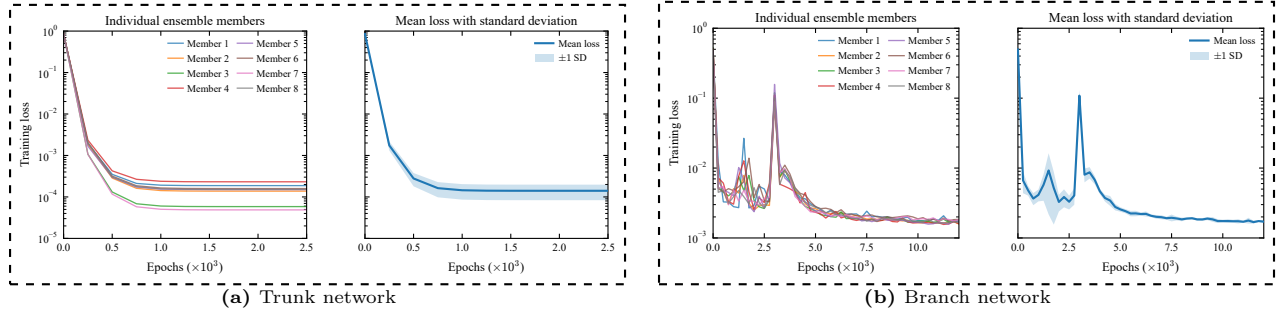


Fig. 13. Training loss curves of two-step MV-DeepONet for the trunk (a) and branch (b) networks: individual ensemble members (left) and the ensemble mean with a ± 1 standard deviation band (right).

Table 9. Mean and maximum relative L_2 errors on the training set and the in-distribution test set.

Metric	Prob-DeepONet	two-step MV-DeepONet
Mean relative L_2 error (training)	1.0936%	1.3587%
Maximum relative L_2 error (training)	7.0382%	8.9639%
Mean relative L_2 error (in-distribution)	3.4922%	1.7909%
Maximum relative L_2 error (in-distribution)	16.1542%	14.1562%

Although Prob-DeepONet achieves lower training errors, two-step MV-DeepONet reduces the mean in-distribution test error from 3.4922% to 1.7909% and the maximum in-distribution test error from 16.1542% to 14.1562%, demonstrating better in-distribution performance.

Out-of-Distribution Generalization Performance

For the Darcy equation, OOD performance is evaluated by varying the correlation length ℓ of the random permeability field from its training value $\ell = 0.25$. The relative L_2 errors are summarized in Table 10.

Prob-DeepONet exhibits large errors throughout the OOD range, with a mean error of 69.98%. Two-step MV-DeepONet reduces the mean error to 3.57%, approximately 1/20 of that of Prob-DeepONet, and remains below 10% except at $\ell = 0.105$. Fig. 14 compares representative results for rough and smooth OOD permeability fields.

As shown in Fig. 14, the bands produced by Prob-DeepONet exhibit irregular, locally jagged boundaries and often fail to cover the reference profiles, particularly under more severe OOD shifts. In contrast, the bands of two-step

MV-DeepONet are smoother and better aligned with the pressure-field structure. They remain narrow for smooth OOD inputs and broaden for rougher cases while generally maintaining coverage of the reference solutions.

Table 10. Relative L_2 errors of Prob-DeepONet and two-step MV-DeepONet for out-of-distribution samples at different input correlation lengths.

Correlation length ℓ	Prob-DeepONet	two-step MV-DeepONet
0.06	78.321%	7.141%
0.075	77.828%	8.765%
0.09	71.807%	6.959%
0.105	59.811%	10.691%
0.12	69.914%	5.246%
0.14	46.733%	4.429%
0.165	46.040%	3.860%
0.19	79.887%	2.651%
0.22	40.665%	3.070%
0.29	84.468%	3.084%
0.34	83.440%	0.701%
0.40	80.956%	1.401%
0.47	76.157%	1.801%
0.55	74.610%	0.743%
0.65	77.185%	0.844%
0.78	72.010%	1.200%
0.93	70.730%	0.571%
1.10	69.353%	1.097%
Mean error	69.98%	3.57%

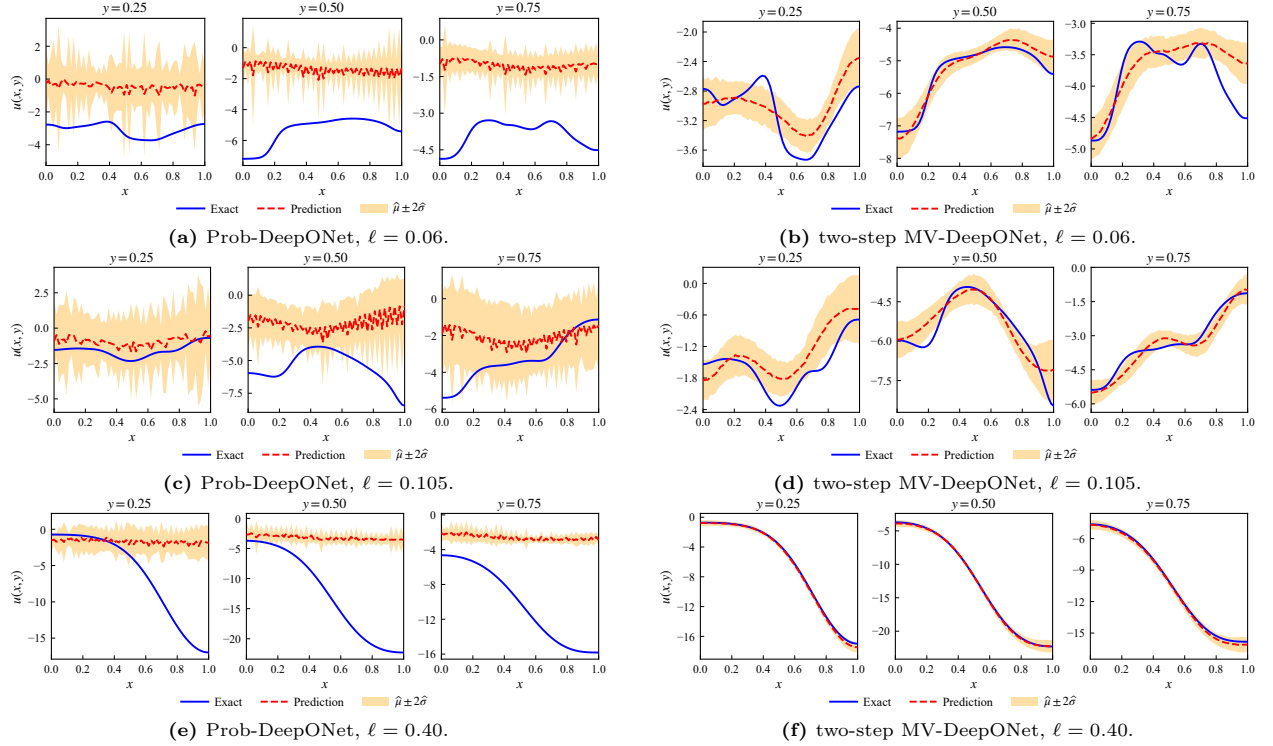


Fig. 14. Predictions and predictive uncertainty bands of Prob-DeepONet (left column) and two-step MV-DeepONet (right column) for representative out-of-distribution samples at different input correlation lengths.

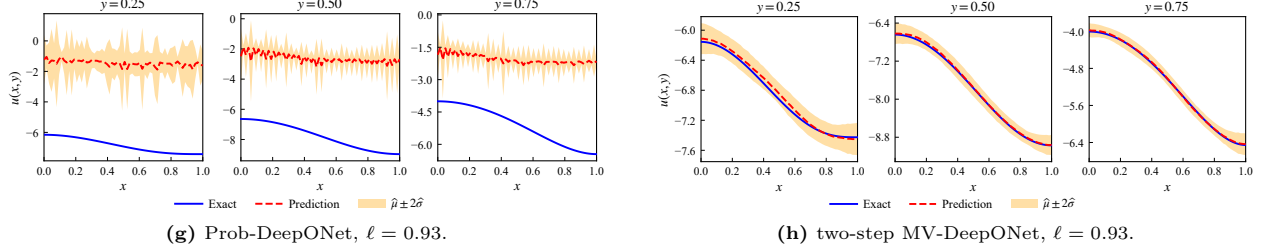


Fig. 14. Predictions and predictive uncertainty bands of Prob-DeepONet (left column) and two-step MV-DeepONet (right column) for representative out-of-distribution samples at different input correlation lengths (continued).

3.3.2 Analysis of Output Covariance Recovery Accuracy

This section further quantifies the output covariance recovery accuracy for the Darcy equation.

Covariance Error Decomposition and Quantitative Accuracy Analysis

Figure 15 shows the three normalized Frobenius covariance-error curves as a function of the number of retained modes p , and Table 11 reports the minimum numbers of modes required to reach different target normalized Frobenius errors. The main observations are as follows.

(1) **Oracle SVD curve.** As shown in Table 11, T_1 falls below 5% for $p \geq 4$ and below 1% for $p \geq 9$, indicating strong low-rank compressibility of the Darcy output covariance. Thus, the low-rank truncation error T_1 is not the main factor limiting covariance recovery.

(2) **Comparison between the Learned-subspace oracle curve and the Final model prediction curve.** The two curves exhibit similar decreasing trends over the full range of p , with the Final model prediction curve remaining only slightly above the Learned-subspace oracle curve. This leads to two main observations:

1. The contribution of T_3 is visible but small. The gap between the two curves remains much smaller than their overall error levels, indicating that branch prediction of the coefficient covariance is not the primary bottleneck.
2. The dominant contribution comes from T_2 . For $M = 64 \times 64 = 4096$ and $K = 2 \times 10^3$, the corresponding relative finite-sample statistical-error scale is

$$\frac{\sqrt{MK}^{-1/2}}{\|\hat{\Sigma}\|_F} \approx 6.2 \times 10^{-5}. \quad (89)$$

This value is far smaller than the observed $T_2 \approx 2 \times 10^{-3}$, indicating that T_2 is governed mainly by the trunk-subspace learning error. Compared with the first two examples, the Darcy problem requires more retained modes to achieve the same error levels, reflecting the greater complexity of the covariance structure induced by a two-dimensional random medium. Nevertheless, the final covariance error decreases steadily to the 10^{-2} level as p increases, which is comparable to the mean relative L_2 error of 1.7909% reported for two-step MV-DeepONet on the in-distribution test set.

Overall, covariance recovery for the Darcy problem follows the hierarchy $T_1 < T_2$ and $T_3 \lesssim T_2$, while the final recovery error is governed primarily by trunk-subspace learning. The resulting covariance recovery accuracy is comparable to the mean prediction accuracy, demonstrating the effectiveness of the proposed method for recovering output covariance in problems involving two-dimensional random media.

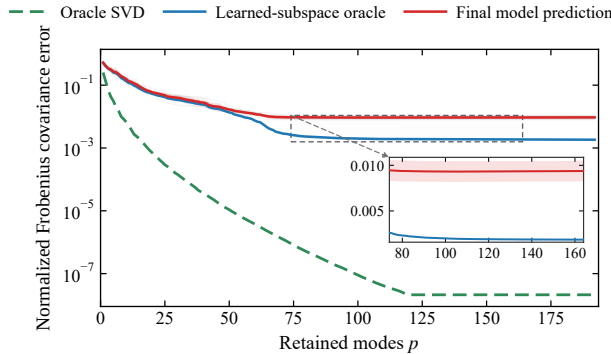


Fig. 15. Normalized Frobenius covariance error as a function of the number of retained modes.

Recovery of Local Correlation Structures: Multi-Anchor Correlation Map Assessment

Table 11. Minimum numbers of retained modes required to achieve different target normalized Frobenius covariance-error levels for the three covariance-recovery curves.

Target error	Oracle SVD	Learned-subspace oracle	Final model prediction
Error $\leq 20.00\%$	$p \geq 2$	$p \geq 8$	$p \geq 8$
Error $\leq 10.00\%$	$p \geq 3$	$p \geq 13$	$p \geq 15$
Error $\leq 5.00\%$	$p \geq 4$	$p \geq 22$	$p \geq 25$
Error $\leq 2.00\%$	$p \geq 6$	$p \geq 43$	$p \geq 47$
Error $\leq 1.00\%$	$p \geq 9$	$p \geq 56$	$p \geq 66$

Five anchor points (x_a, y_a) are selected in the physical domain $(x, y) \in [0, 1]^2$, representing the center, the left and bottom boundaries, a corner, and a transition region. Fig. 16 compares the corresponding reference, predicted, and absolute-error correlation maps.

Overall pattern consistency. The predicted maps reproduce the reference patterns for all five anchors, with a mean row-wise correlation error of approximately 2.38×10^{-2} . This result confirms accurate recovery of the local correlation structure of the Darcy pressure field.

Physical interpretation of the correlation structure. In contrast to the banded patterns of the preceding spatiotemporal examples, the Darcy maps exhibit localized two-dimensional correlation regions whose geometry depends on the anchor position. The interior anchor produces an approximately symmetric pattern with correlation decaying away from the anchor. Near a boundary or corner, this pattern becomes asymmetric and is confined by the domain geometry. These patterns show that the off-diagonal structure of the output covariance captures the spatial coupling induced by shared permeability perturbations through the elliptic operator. The strength, spatial extent, and geometry of the correlation patterns depend not only on spatial distance but also on the spatial correlation of the input field, the nonlocal action of the elliptic operator, and the boundary conditions.

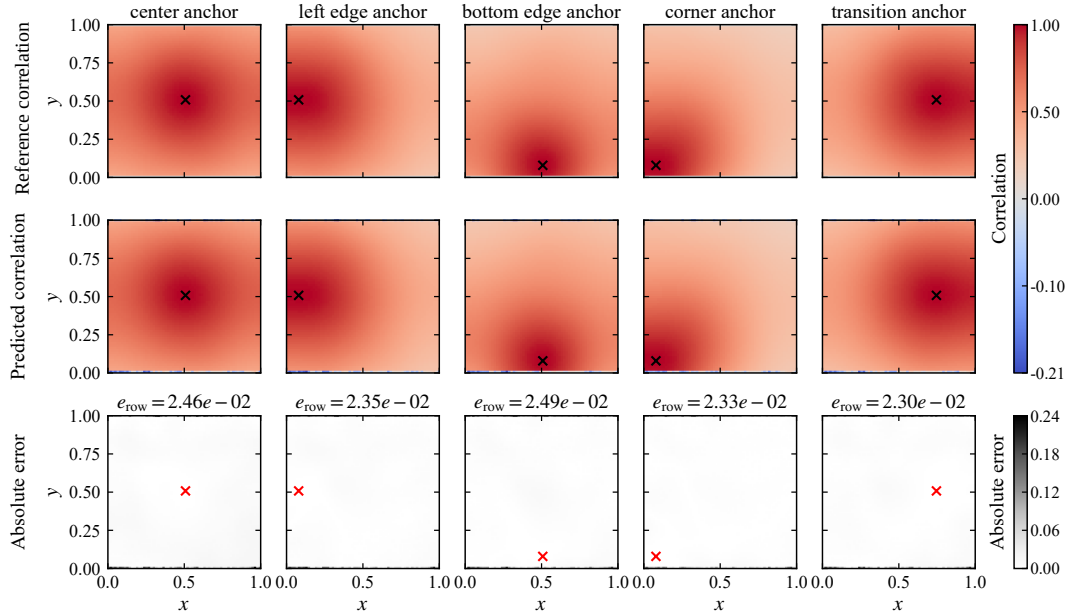


Fig. 16. Multi-anchor correlation maps showing the reference correlations, predicted correlations, and corresponding absolute errors for five anchor points, where e_{row} denotes the relative L_2 error of the predicted correlation map for each anchor point.

3.3.3 Prediction Interval Calibration and Coverage Analysis

CP is further applied to the prediction intervals for the Darcy problem. Table 12 summarizes the PICP and MPIW values before and after calibration.

Both methods exhibit pronounced overcoverage before calibration. CP moves their PICP values close to the nominal levels and substantially reduces the interval widths. Notably, the MPIW ratio between Prob-DeepONet and two-step MV-DeepONet remains large both before and after calibration, decreasing from approximately 12.0 to 10.0 at the 95% nominal coverage level. Thus, two-step MV-DeepONet achieves similar coverage with substantially narrower intervals.

Table 12. PICP and MPIW of Prob-DeepONet and two-step MV-DeepONet before and after CP calibration.

Method	Nominal coverage (%)	Before CP calibration		After CP calibration	
		PICP (%)	MPIW	PICP (%)	MPIW
Prob-DeepONet	90	99.14	9.89×10^{-2}	89.80	4.63×10^{-2}
	95	99.57	1.18×10^{-1}	94.86	5.93×10^{-2}
	99	99.84	1.55×10^{-1}	99.01	9.55×10^{-2}
two-step MV-DeepONet	90	98.37	8.23×10^{-3}	89.46	4.69×10^{-3}
	95	99.21	9.81×10^{-3}	94.63	5.92×10^{-3}
	99	99.76	1.29×10^{-2}	98.22	9.12×10^{-3}

3.4 Hypersonic Blunt-Body Aerothermal Modeling

Hypersonic aerothermal prediction involves coupled multiscale and multiphysics processes in a realistic engineering setting. During atmospheric reentry or hypersonic cruise, shock compression and viscous dissipation convert kinetic energy into internal energy, producing high temperatures within the shock layer. The resulting chemical reactions and vibrational relaxation may occur on time scales comparable to the flow time scale, leading to thermochemical nonequilibrium [68, 69].

A representative configuration is considered here: hypersonic laminar flow past a hemispherical blunt body. The numerical configuration follows MacLean *et al.* [70] and Dsouza *et al.* [71]. MacLean *et al.* compared LENS-XX surface heat-transfer measurements with Data-Parallel Line Relaxation (DPLR) predictions, whereas Dsouza *et al.* reproduced the same case in Ansys Fluent using a two-temperature nonequilibrium model and validated the predictions against the experimental data. This example is used to evaluate structured output covariance recovery under realistic aerothermal conditions.

Governing Equations and Constitutive Relations. Under the assumption of laminar flow, the three-dimensional thermochemical nonequilibrium Navier–Stokes equations are written in conservation form as follows [68]:

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_j)}{\partial x_j} &= 0, \\
\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial (\rho u_i u_j + p \delta_{ij})}{\partial x_j} &= \frac{\partial \tau_{ij}}{\partial x_j}, \\
\frac{\partial (\rho E)}{\partial t} + \frac{\partial [(\rho E + p) u_j]}{\partial x_j} &= \frac{\partial (u_i \tau_{ij})}{\partial x_j} - \frac{\partial (q_j^{TR} + q_j^V)}{\partial x_j} - \frac{\partial}{\partial x_j} \left(\sum_{n=1}^{N_S} \rho_n u_{nj}^D h_n \right), \\
\frac{\partial \rho_n}{\partial t} + \frac{\partial (\rho_n u_j)}{\partial x_j} &= - \frac{\partial (\rho_n u_{nj}^D)}{\partial x_j} + \dot{\omega}_n, \quad n = 1, \dots, N_S - 1, \\
\frac{\partial (\rho e_V)}{\partial t} + \frac{\partial (\rho e_V u_j)}{\partial x_j} &= \frac{\partial}{\partial x_j} \left(-q_j^V - \sum_{m=1}^{N_M} \rho_m u_{mj}^D e_{Vm} \right) + \sum_{m=1}^{N_M} (Q_{TVm} + \dot{\omega}_m e_{Vm}).
\end{aligned} \tag{90}$$

Here, E is the specific total energy, while q_j^{TR} and q_j^V denote the translational–rotational and vibrational heat fluxes, respectively. For species n , ρ_n , u_{nj}^D , h_n , and $\dot{\omega}_n$ denote the partial density, diffusion velocity, specific enthalpy, and production rate, respectively. The quantities e_V and Q_{TVm} represent the specific vibrational energy of the mixture and the energy exchange between the translational and vibrational modes of molecular species m , respectively. Closure of the governing equations is provided by the Park two-temperature model [72] and a finite-rate chemistry model [73, 74]. The former accounts for thermal nonequilibrium between the translational–rotational and vibrational–electronic energy modes, whereas the latter describes species production and depletion through finite-rate chemical reactions.

Sources of Uncertainty. Variations in freestream density, temperature, pressure, and velocity can alter the shock-layer structure and wall heat flux. Previous uncertainty quantification studies of hypersonic aerothermodynamics have treated these quantities as uncertain inputs, with freestream velocity and density identified as important contributors to surface-heating uncertainty [75–77]. Since wall heat flux is a critical design load for thermal protection systems, the present study represents freestream uncertainty through the Mach number, while the remaining freestream quantities are determined consistently from the prescribed atmospheric and flight conditions.

Computational Fluid Dynamics (CFD) Solver Validation. Since the dataset is generated by CFD simulations, the solver is first validated against the Mach 12.4 hemispherical blunt-body experiment of MacLean *et al.* [70]. The configuration uses the 7.6-cm-diameter stainless-steel test article shown in their Fig. 1(a), with mesh refinement

near the stagnation region and within the boundary layer [70]. The simulation employs an 11-species, 21-reaction thermochemical nonequilibrium model together with a two-temperature formulation. The freestream and boundary conditions are summarized in Table 13.

As shown in Fig. 17, the predicted wall heat flux agrees well with the experimental measurements and NASA DPLR results [70, 78]. The simulation captures both the stagnation-point heat-flux peak and its downstream decay, consistent with previous numerical results for the same configuration [71].

Table 13. Parameter settings for the high-fidelity aerothermal simulation.

Category	Parameter	Value
Physical configuration	Geometry	Stainless steel hemisphere, $D = 7.6$ cm
	Flow regime	Laminar
	Thermochemical model	11 chemical species, 21 elementary reactions, and a two-temperature model
Freestream conditions	Mach number	$Ma = 12.4$
	Temperature	$T_\infty = 535$ K
	Pressure	$p_\infty = 178.1$ Pa
	Velocity	$U_\infty = 5732$ m s ⁻¹
	Density	$\rho_\infty = 1.16 \times 10^{-3}$ kg m ⁻³
Boundary conditions	Inlet	Pressure far-field
	Outlet	Pressure outlet
	Wall	Isothermal wall, $T_w = 300$ K

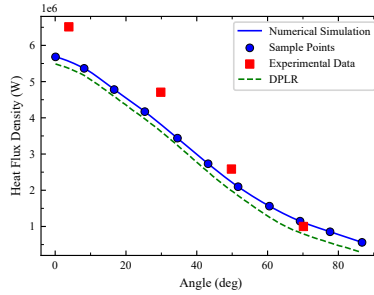


Fig. 17. Comparison of wall heat flux distributions obtained from the present CFD simulation, the LENS-XX experiment, and NASA DPLR. The angle θ is measured along the hemispherical surface from the stagnation point, with $\theta = 180s_{\text{arc}}/(\pi R)$, where s_{arc} is the surface arc length and R is the hemisphere radius. Thus, $\theta = 0^\circ$ denotes the stagnation point and $\theta = 90^\circ$ the hemisphere-cylinder junction.

Data Generation and Network Architecture

Based on the validated CFD model, the dataset construction and network design for this example are described below. Taking the freestream conditions as the source of uncertainty, Latin hypercube sampling is used to sample the freestream Mach number over $10.4 \leq Ma \leq 16.4$. A total of 179 independent samples are generated and divided into 125 training samples, 24 test samples, and 30 calibration samples. The trunk network takes the wall-node coordinates as input and learns basis functions for representing the wall heat flux distribution.

During the first training stage, a conventional mean squared error loss may smooth the stagnation-point heat flux peak. Since wall heat flux is related to the near-wall temperature gradients through

$$q_w = q_{\text{tr}} + q_{\text{ve}} = k_{\text{tr}} \nabla T_{\text{tr}} + k_{\text{ve}} \nabla T_{\text{ve}}, \quad (91)$$

a physics-informed weighting based on the translational-rotational temperature gradient is introduced into the trunk reconstruction loss:

$$(\omega^*, \mathbf{A}^*) = \arg \min_{\omega, \mathbf{A}} \|\mathbf{W}_{\nabla T} \odot (\Phi_\omega \mathbf{A} - \mathbf{S})\|_F^2. \quad \mathbf{W}_{\nabla T} = 1 + \varepsilon \tanh \left(\frac{|\nabla T_{\text{tr}}|}{\text{std}(|\nabla T_{\text{tr}}|)} \right). \quad (92)$$

The translational-rotational temperature gradient is adopted because it is numerically more stable than the vibrational-electronic temperature gradient and is more directly related to wall heat transfer under the cold-wall condition. The hyperbolic tangent limits excessive weight amplification, while ε controls the weighting strength. Setting $\varepsilon = 1.0$ gives $1 \leq \mathbf{W}_{\nabla T} < 2$. This weighted loss emphasizes the stagnation and post-shock regions without destabilizing the reconstruction loss. Wall heat flux depends not only on the freestream conditions but also strongly on the thermochemical state of the shock layer. A multi-branch architecture is therefore used [79]. Branch 1 receives the freestream Mach number, whereas Branch 2 receives features extracted from the mass fraction fields of the 11 chemical species.

The species fields are high dimensional and strongly correlated because of mass conservation, $\sum_{n=1}^{N_S} Y_n = 1$, and finite-rate chemical coupling. A variational autoencoder (VAE) is therefore used to encode the species mass fraction fields into a low-dimensional latent vector $\mathbf{z}$, which serves as the effective input to Branch 2. The outputs of the two branches are concatenated and processed by a multilayer perceptron (MLP). The resulting representation is then used by two-step MV-DeepONet to predict the wall heat flux distribution and recover its covariance structure. Details of the VAE encoding procedure are provided in Appendix A.

3.4.1 Prediction Accuracy and In-Distribution Generalization Performance

Compared with the preceding three standard benchmarks based on partial differential equations (PDEs), the hypersonic aerothermal example is more complex in both input dimensionality and the structure of the output field. The wall heat flux distribution over the hemispherical blunt body exhibits pronounced spatial gradients near the stagnation point and within the shock-affected region. This section compares the Prob-DeepONet and two-step MV-DeepONet in terms of training accuracy, in-distribution generalization, and uncertainty bands for challenging samples.

Training and In-Distribution Generalization Accuracy

Figures 18 and 19 show stable convergence for both methods. Their training and in-distribution test errors are summarized in Table 14.

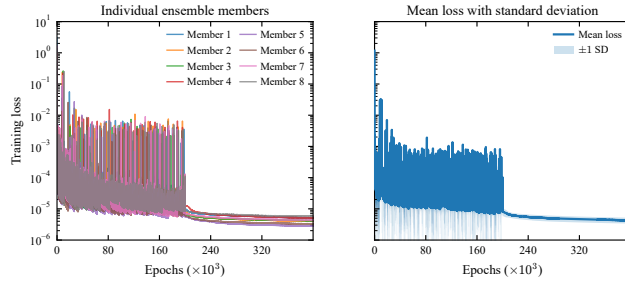


Fig. 18. Training loss curves of Prob-DeepONet: individual ensemble members (left) and the ensemble mean with a ± 1 standard deviation band (right).

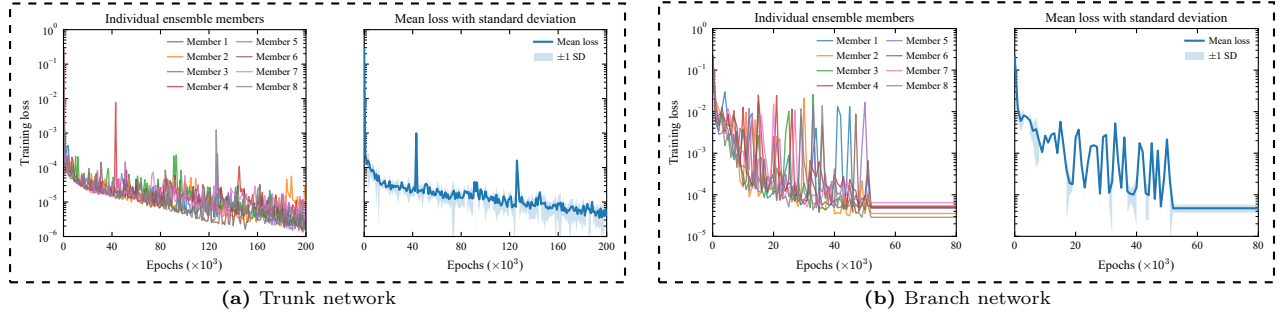


Fig. 19. Training loss curves of two-step MV-DeepONet for the trunk (a) and branch (b) networks: individual ensemble members (left) and the ensemble mean with a ± 1 standard deviation band (right).

Table 14. Mean and maximum relative L_2 errors on the training set and the in-distribution test set.

Metric	Prob-DeepONet	two-step MV-DeepONet
Mean relative L_2 error (training)	0.5496%	0.2555%
Maximum relative L_2 error (training)	0.8589%	0.3729%
Mean relative L_2 error (in-distribution test)	3.7174%	2.1064%
Maximum relative L_2 error (in-distribution test)	8.8195%	3.5545%

As reported in Table 14, two-step MV-DeepONet achieves lower mean and maximum relative L_2 errors on both the training and in-distribution test sets. The increases from training to test errors are 1.8509 and 3.1816 percentage points, respectively, compared with 3.1678 and 7.9606 percentage points for Prob-DeepONet. These results demonstrate that two-step MV-DeepONet achieves higher fitting accuracy and more stable in-distribution generalization, particularly for the worst-case samples.

Predictions and Uncertainty Bands for Challenging Samples

To further examine the predictive behavior of the two methods on challenging samples, the training and in-distribution test samples with the largest relative L_2 errors under each method are selected as stress-test cases.

Fig. 20 presents the reference fields, predicted fields, and relative error distributions for these samples. Fig. 21 shows the corresponding predictive means and $\pm 2\sigma$ uncertainty bands along selected spatial profiles.

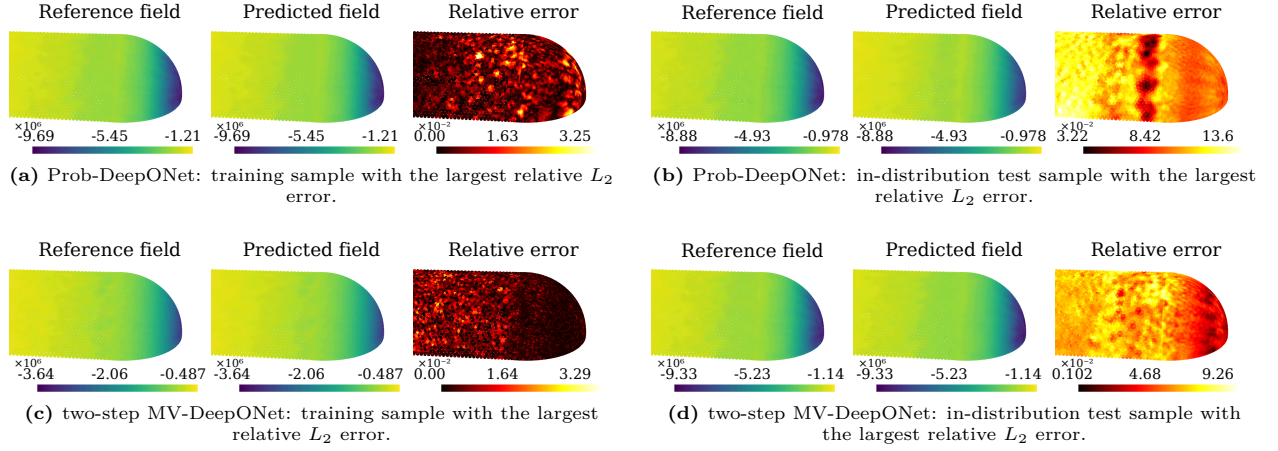


Fig. 20. Reference and predicted wall heat flux fields, together with the corresponding relative error distributions, for the training and in-distribution test samples with the largest relative L_2 errors. The first and second rows correspond to Prob-DeepONet and two-step MV-DeepONet, respectively.

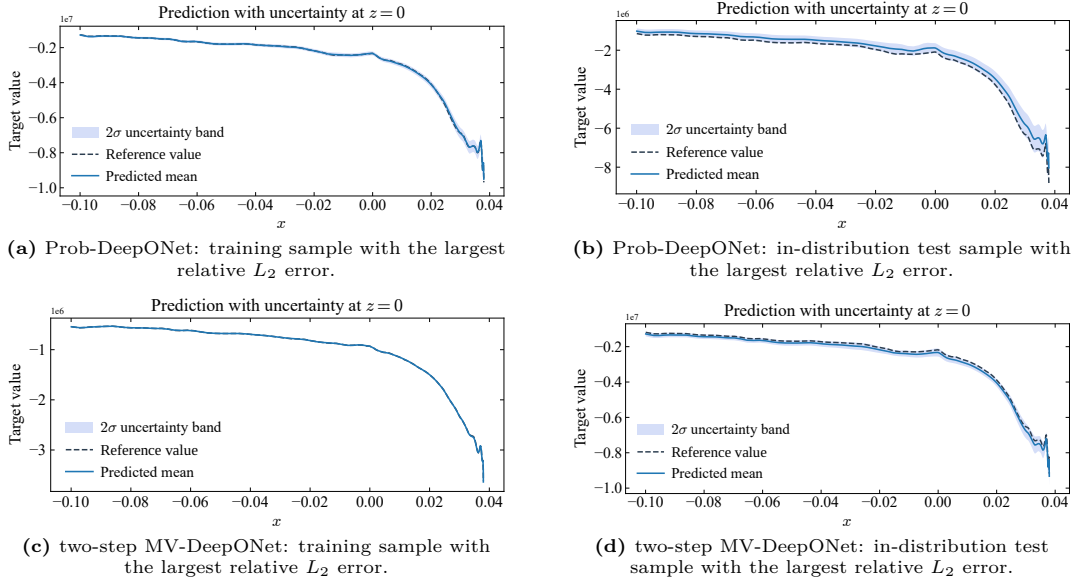


Fig. 21. Predicted mean and uncertainty bands for the training and in-distribution test samples with the largest relative L_2 errors. The uncertainty bands correspond to $\pm 2\sigma$. The first and second rows correspond to Prob-DeepONet and two-step MV-DeepONet, respectively.

Regarding the wall heat flux fields, both methods capture the dominant variation along the wall, but their relative error distributions differ markedly. For the most challenging in-distribution test sample, Prob-DeepONet exhibits localized errors with a patch-like pattern. In contrast, the errors of two-step MV-DeepONet vary more smoothly along the wall and generally increase from the stagnation region toward the downstream surface.

This difference can be explained by the different spaces in which probabilistic modeling is performed. Prob-DeepONet evaluates the Gaussian NLL pointwise in the physical output space. For an input $\mathbf{u}$ and the i -th output location y_i , its negative log-likelihood can be written as

$$\mathcal{L}_{\text{NLL}}^{\text{Prob}}(\mathbf{u}) = \frac{1}{2M} \sum_{i=1}^M \left[\log \hat{\sigma}_{s,i}^2(\mathbf{u}) + \frac{[s_i(\mathbf{u}) - \hat{\mu}_{s,i}(\mathbf{u})]^2}{\hat{\sigma}_{s,i}^2(\mathbf{u})} \right], \quad (93)$$

where $\hat{\mu}_{s,i}$ and $\hat{\sigma}_{s,i}^2$ denote the predictive mean and variance at y_i , respectively. The gradient of this loss with respect

to the predictive mean is

$$\frac{\partial \mathcal{L}_{\text{NLL}}^{\text{Prob}}}{\partial \hat{\mu}_{s,i}} = \frac{1}{M} \frac{\hat{\mu}_{s,i} - s_i}{\hat{\sigma}_{s,i}^2}. \quad (94)$$

Thus, the mean residual at each location is weighted by the corresponding pointwise variance. When the predicted variance at a particular location is large, the gradient and training penalty associated with the mean residual at that location may be reduced, so that the local mean error may remain insufficiently corrected. This pointwise mean–variance tradeoff may consequently produce relatively isolated error patches in local regions that are difficult to fit.

By contrast, two-step MV-DeepONet evaluates the negative log-likelihood in the modal coefficient space. For the m -th target modal coefficient $c_m^*(\mathbf{u})$, the loss is written as

$$\mathcal{L}_{\text{NLL}}^{2\text{step}}(\mathbf{u}) = \frac{1}{2p} \sum_{m=1}^p \left[\log \hat{\sigma}_{c,m}^2(\mathbf{u}) + \frac{[c_m^*(\mathbf{u}) - \hat{\mu}_{c,m}(\mathbf{u})]^2}{\hat{\sigma}_{c,m}^2(\mathbf{u})} \right], \quad (95)$$

and its gradient with respect to the predicted coefficient mean is

$$\frac{\partial \mathcal{L}_{\text{NLL}}^{2\text{step}}}{\partial \hat{\mu}_{c,m}} = \frac{1}{p} \frac{\hat{\mu}_{c,m} - c_m^*}{\hat{\sigma}_{c,m}^2}. \quad (96)$$

It follows that the predicted variance in two-step MV-DeepONet adjusts the weighting of modal coefficient mean errors rather than that of local residuals at individual output locations. Although a larger coefficient variance may likewise reduce the training penalty on the corresponding coefficient mean error, this tradeoff occurs at the modal-coefficient level rather than in the physical output space.

To further illustrate this distinction, let

$$\delta \boldsymbol{\mu}_c(\mathbf{u}) = \hat{\boldsymbol{\mu}}_c(\mathbf{u}) - \mathbf{c}^*(\mathbf{u}) \quad (97)$$

denote the coefficient mean error. When only the coefficient regression error is considered, the corresponding mean error in the physical output space is

$$\delta \boldsymbol{\mu}_s(\mathbf{u}) = \tilde{\mathbf{Q}}^* \delta \boldsymbol{\mu}_c(\mathbf{u}). \quad (98)$$

At any two output locations y_i and y_j , this relation gives

$$\delta \mu_s(y_i, \mathbf{u}) = \sum_{m=1}^p \tilde{q}_m^*(y_i) \delta \mu_{c,m}(\mathbf{u}), \quad \delta \mu_s(y_j, \mathbf{u}) = \sum_{m=1}^p \tilde{q}_m^*(y_j) \delta \mu_{c,m}(\mathbf{u}). \quad (99)$$

These expressions show that each coefficient mean error $\delta \mu_{c,m}$ is distributed across multiple output locations according to the spatial profile of the corresponding shared basis function $\tilde{q}_m^*$. If the shared basis functions vary smoothly over the output domain, the resulting mean-prediction error field also tends to vary coherently across locations, rather than forming isolated patch-like concentrations.

Consequently, transferring probabilistic modeling to the coefficient space and combining it with the two-step training strategy shifts the mean–variance tradeoff from the pointwise level in the physical output space to the low-dimensional modal level. This provides a structural explanation for the smoother and more spatially coordinated error distributions produced by two-step MV-DeepONet. The gradual increase in relative error away from the stagnation region may also be amplified by the decreasing magnitude of the reference wall heat flux.

For the training samples, the predictive means of both methods closely follow the reference profiles, and their uncertainty bands remain narrow. For the most challenging in-distribution test samples, the uncertainty bands widen as the prediction discrepancies increase, particularly in the region where the wall heat flux changes rapidly. The uncertainty band of Prob-DeepONet expands over a relatively broad portion of the profile, whereas that of two-step MV-DeepONet varies more coherently with the spatial variation of the wall heat flux field.

3.4.2 Analysis of Output Covariance Recovery Accuracy

This part further quantifies the output covariance recovery accuracy for the aerothermal example.

Covariance Error Decomposition and Quantitative Accuracy Analysis

Figure 22 shows how the normalized Frobenius covariance errors vary with the number of retained modes p , while Table 15 reports the minimum numbers of modes required to reach different target normalized Frobenius errors. The main observations are as follows.

(1) **Oracle SVD curve (T_1).** The Oracle SVD error falls below 1% with only one retained mode, below 10^{-4} for $p \geq 3$, and reaches 1.55×10^{-7} at $p = 10$. The reference covariance therefore has an approximately rank-one structure. Its physical origin and spectral characteristics are further discussed in Appendix B. Consequently, T_1 is

negligible compared with the model-dependent errors.

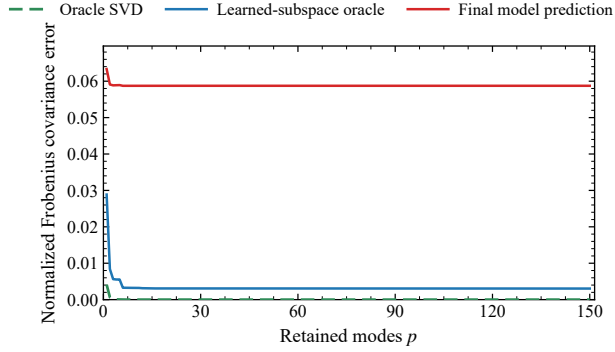


Fig. 22. Normalized Frobenius covariance error as a function of the number of retained modes.

Table 15. Minimum numbers of retained modes required to achieve different target normalized Frobenius covariance-error levels for the three covariance-recovery curves.

Target error	Oracle SVD	Learned-subspace oracle	Final model prediction
Error $\leq 20.00\%$	$p \geq 1$	$p \geq 1$	$p \geq 1$
Error $\leq 10.00\%$	$p \geq 1$	$p \geq 1$	$p \geq 1$
Error $\leq 5.00\%$	$p \geq 1$	$p \geq 1$	Not reached
Error $\leq 2.00\%$	$p \geq 1$	$p \geq 2$	Not reached
Error $\leq 1.00\%$	$p \geq 1$	$p \geq 2$	Not reached

(2) Comparison between the Learned-subspace oracle curve and the Final model prediction curve.

The two curves remain clearly separated over the full range of p . For $p \geq 5$, the Learned-subspace oracle error stabilizes at approximately 3.0×10^{-3} , whereas the Final model prediction error remains near 6.6×10^{-2} . This behavior differs from that observed in the preceding PDE examples and leads to two observations:

1. The final covariance recovery error is dominated by T_3 . The Learned-subspace oracle error is only approximately 0.3%, indicating that the trunk network accurately captures the dominant output subspace. By contrast, the large separation from the Final model prediction curve identifies branch coefficient-covariance regression as the principal bottleneck. This result is consistent with the approximately rank-one covariance structure. Since most covariance energy is concentrated in the dominant mode, bias in its predicted coefficient variance is transferred directly to the Frobenius covariance error. The limited training set and the complex inputs combining latent species-field features with freestream parameters may further increase the difficulty of variance regression. Accordingly, the Final model prediction reaches the 10% error level but not the 5% level, whereas the other two curves fall below 1% for $p \geq 2$.
2. For $M = 13,056$ and $K = 125$, the corresponding relative statistical-error scale is

$$\frac{\sqrt{MK}^{-1/2}}{\|\hat{\Sigma}\|_F} \approx 6.2 \times 10^{-2}. \quad (100)$$

This estimate is more than one order of magnitude larger than the observed $T_2 \approx 3.0 \times 10^{-3}$, showing that the order bound is loose in this case. It therefore does not permit a reliable ranking of the two contributions within T_2 .

Overall, covariance recovery in the aerothermal example follows the hierarchy $T_1 \ll T_2 \ll T_3$ in contrast to the T_2 -dominated behavior observed in the preceding PDE examples. These results show that when covariance energy is concentrated in a single dominant mode, recovery accuracy becomes particularly sensitive to the predicted variance of the corresponding coefficient.

Recovery of Local Correlation Structures: Multi-Anchor Correlation Map Assessment

For the aerothermal example, three anchor points are selected from the upstream, middle, and downstream regions of the wall along the flow direction. Fig. 23 compares the corresponding reference, predicted, and absolute-error correlation maps.

Overall pattern consistency. For all three anchors, the predicted and reference correlation maps show strong agreement in their overall spatial patterns. The spatial extent of the principal correlation region, the streamwise

variation in correlation strength, and the transition between regions of relatively higher and lower correlation are all well reproduced. The absolute errors remain small over most of the wall, with a mean row-wise correlation error of approximately 2.10×10^{-3} . Although the normalized Frobenius covariance error is approximately 6.6% and dominated by T_3 , the local correlation error is only about 0.2%. This difference indicates that the Frobenius error primarily reflects a bias in the magnitude of the dominant modal variance rather than an error in the recovered spatial correlation structure. The model therefore recovers the spatial correlation pattern accurately while slightly misestimating the overall covariance scale.

Physical interpretation of the correlation structure. The aerothermal maps exhibit globally coherent positive correlations along the wall. This behavior is consistent with the physical mechanism of the aerothermal problem. Since the freestream Mach number is the only sampled random input, its perturbation acts as a common forcing on the coupled flow and thermochemical response. Over broad regions of the wall, the heat flux responds to variations in Ma in the same direction. Consequently, increases or decreases in Ma produce coordinated changes in wall heat flux, giving rise to the extensive positive correlations observed in the correlation maps.

This structure differs from the vertical, horizontal, and localized correlation patterns observed in the preceding examples. These differences reflect the distinct mechanisms through which uncertainty enters and propagates in the corresponding physical systems.

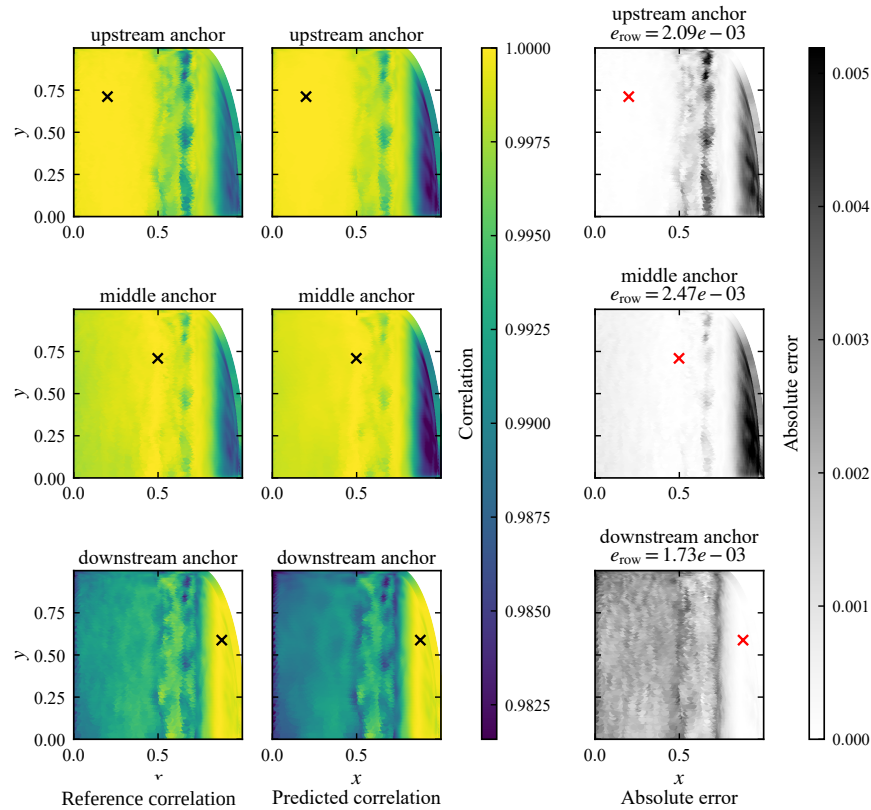


Fig. 23. Multi-anchor correlation maps showing the reference correlations, predicted correlations, and corresponding absolute errors for three anchor points, where e_{row} denotes the relative L_2 error of the predicted correlation map for each anchor point.

3.4.3 Prediction Interval Calibration and Coverage Analysis

CP is applied to calibrate the prediction intervals for the aerothermal example. Table 16 reports the corresponding PICP and MPIW values.

Before calibration, both methods produce conservative intervals with PICP values close to 100%. At the 90% and 95% nominal levels, CP moves the coverage close to the prescribed targets and reduces the interval widths. At the 95% level, the MPIW of Prob-DeepONet is approximately 1.74 times that of two-step MV-DeepONet before calibration and 1.65 times after calibration. At the 99% level, the calibration effect is less uniform: Prob-DeepONet retains 100% coverage, while the MPIW of two-step MV-DeepONet increases despite achieving a PICP of 99.14%. Nevertheless, its calibrated interval remains narrower than that of Prob-DeepONet. Overall, two-step MV-DeepONet provides comparable coverage with sharper intervals across the three nominal levels.

Table 16. PICP and MPIW of Prob-DeepONet and two-step MV-DeepONet before and after CP calibration.

Method	Nominal coverage (%)	Before CP calibration		After CP calibration	
		PICP (%)	MPIW (W m^{-2})	PICP (%)	MPIW (W m^{-2})
Prob-DeepONet	90	99.63	4.0469×10^5	90.58	2.0484×10^5
	95	99.99	4.8219×10^5	95.27	2.9337×10^5
	99	100.00	6.3373×10^5	100.00	4.8961×10^5
two-step MV-DeepONet	90	99.05	2.3197×10^5	90.63	1.4897×10^5
	95	99.99	2.7641×10^5	94.79	1.7783×10^5
	99	100.00	3.6326×10^5	99.14	4.0642×10^5

4 Conclusions

This paper developed a two-step MV-DeepONet framework for forward uncertainty propagation driven by random input fields, with particular emphasis on recovering the total predictive covariance and its structured conditional component. The framework was designed to relax the pointwise conditional-independence assumption of Prob-DeepONet by incorporating a two-step training strategy and transferring Gaussian probabilistic modeling from the physical output space to the low-dimensional modal coefficient space. The probabilistic modal coefficients jointly affect multiple output locations through the shared basis functions, thereby inducing a generally non-diagonal conditional predictive covariance without explicitly learning or storing the full high-dimensional covariance matrix. To further improve practical uncertainty estimation, deep ensembles and conformal prediction were incorporated to quantify model uncertainty and construct calibrated prediction intervals. More fundamentally, a Frobenius-norm error decomposition and corresponding upper bound were derived, identifying low-rank covariance compressibility, trunk-subspace approximation, finite-sample statistical error, and coefficient-space covariance estimation as the principal factors governing covariance recovery.

Numerical experiments on several PDE problems and a representative engineering problem validated the effectiveness of the proposed model. The results showed that although Prob-DeepONet fit the training data well, it exhibited larger generalization errors, wider prediction intervals, and more limited recovery of spatial correlation structures in challenging test and extrapolation scenarios, consistent with the limitations of its pointwise conditional covariance representation. Compared with the baseline, the two-step MV-DeepONet yielded more stable mean predictions, spatially structured uncertainty estimates, and improved covariance recovery. After calibration, both models approached the nominal coverage levels, while the proposed model achieved comparable coverage with narrower prediction intervals, indicating more compact and informative uncertainty estimates.

Future work may proceed in two directions. First, since the effectiveness of the proposed model depends partly on the low-rank compressibility of the output covariance and the retained modal dimension p , adaptive mode-selection strategies may be investigated to balance subspace expressiveness and coefficient-space probabilistic regression complexity. Second, this paper mainly adopted Gaussian probabilistic modeling in the modal coefficient space. For cases where the conditional coefficient-space predictive distributions exhibit non-Gaussian features, such as skewness, heavy tails, or multimodality, future work may adopt non-Gaussian coefficient-space distributions to improve robustness in scenarios involving strong nonlinearity, high-dimensional random inputs, and complex spatially correlated outputs.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT in order to improve readability and language. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

CRedit authorship contribution statement

Yupei Nie: Idea of this paper, Simulations and calculations, Analysis, Code, Conceptualization, Validation, Software, Wrote this paper. **Lei Wang:** Research direction, Idea of this paper, Algorithm, Total design scheme, Analysis, Conceptualization, Wrote this paper. **Jiasen Liu:** Numerical simulation of the aerothermal case.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability statements

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments

We express our sincere thanks to all the members of our discussion group for their valuable comments.

Appendix A. VAE-Based Reduction for the Aerothermal Case Study in Section 3.4

The aerothermal flow field contains mass fraction distributions for eleven chemical species. Directly using these fields as operator-learning inputs would substantially increase the input dimension, while the large differences in magnitude among species would complicate network training. A three-dimensional VAE is therefore employed to compress the species fields into a latent vector $\mathbf{z}$, which provides a compact input representation for the subsequent aerothermal prediction model. The VAE is used solely for dimensionality reduction. Its data processing, training objective, and reconstruction performance are described below.

A.1 Data Processing and Network Architecture

The original flow fields are defined on an unstructured mesh. The 11 species mass fraction fields are therefore first mapped onto a regular three-dimensional voxel grid using inverse distance weighting. The grid has dimensions $96 \times 26 \times 26$, and each sample contains 11 channels corresponding to the mass fraction distributions of the chemical species.

Because some voxels lie outside the valid flow domain, a binary mask $\mathbf{m}$ is introduced to identify the valid physical region, and the reconstruction error is evaluated only over this region. To improve training stability in the presence of large differences in magnitude among species, the mass fractions are first transformed using $\log_{10}$ and then standardized channelwise over the valid masked region:

$$x_{\text{std}} = \frac{\log_{10} (\max(x_{\text{raw}}, \varepsilon_{\log})) - \mu_c}{s_c}. \quad (\text{A.1})$$

Here, μ_c and s_c denote the mean and standard deviation of the c th channel over the valid region, and $\varepsilon_{\log}$ prevents the logarithm of zero.

The VAE consists of an encoder, a latent-variable sampling layer, and a decoder. The encoder uses three-dimensional convolutions, anisotropic downsampling, and residual blocks to map the standardized multichannel voxel field to the latent distribution parameters $\boldsymbol{\mu}$ and $\log \boldsymbol{\sigma}^2$. The latent vector is then generated using the reparameterization trick,

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \quad (\text{A.2})$$

The decoder reconstructs the three-dimensional multichannel field through upsampling, convolutional layers, and residual refinement blocks. The complete architecture is illustrated in Fig. A.1.

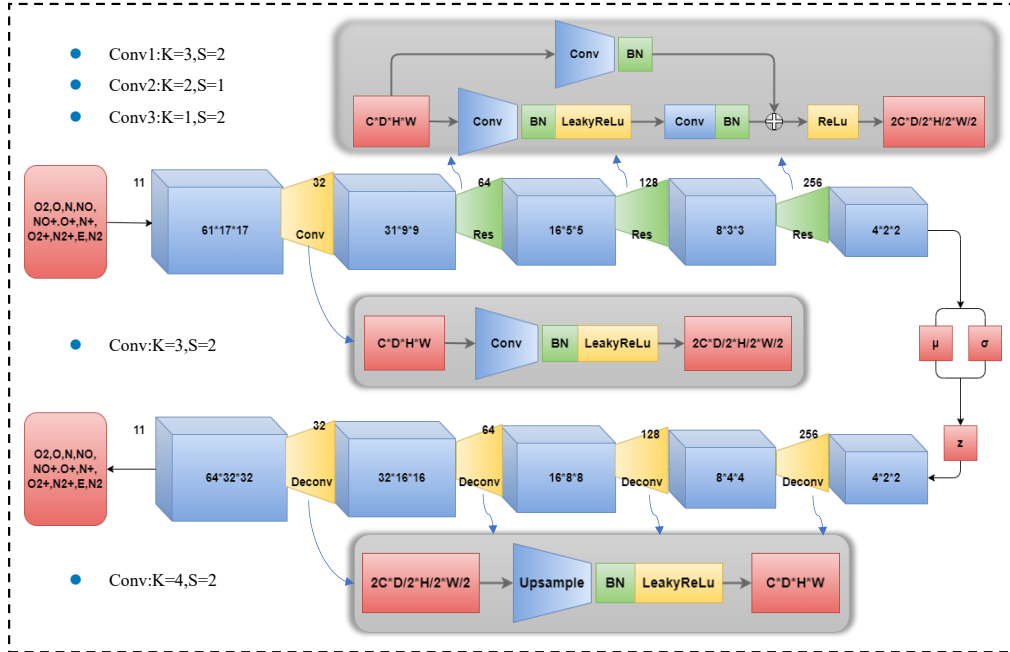


Fig. A.1. Schematic of the three-dimensional convolutional VAE architecture with residual blocks.

A.2 Training Objective and Hyperparameter Selection

The VAE is trained using a combination of the reconstruction loss and the KL regularization term:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \lambda_{\text{KL}} \mathcal{L}_{\text{KL}}, \quad (\text{A.3})$$

where λ_{KL} controls the strength of the Kullback–Leibler (KL) regularization.

The standard reconstruction loss is evaluated in the transformed network input space and is defined as

$$\mathcal{L}_{\text{recon}} = \frac{1}{N_b C} \sum_{n=1}^{N_b} \sum_{c=1}^C \frac{\|\mathbf{m} \odot (\hat{\mathbf{x}}_n^{(c)} - \mathbf{x}_n^{(c)})\|_2^2}{\|\mathbf{m}\|_1}, \quad (\text{A.4})$$

where $\mathbf{x}_n^{(c)}$ is the $\log_{10}$ -transformed and standardized input field, and $\hat{\mathbf{x}}_n^{(c)}$ is the corresponding decoder reconstruction in the same transformed space.

The KL regularization term is given by

$$\mathcal{L}_{\text{KL}} = -\frac{1}{2N_b d_z} \sum_{n=1}^{N_b} \sum_{j=1}^{d_z} (1 + \log \sigma_{n,j}^2 - \mu_{n,j}^2 - \sigma_{n,j}^2), \quad (\text{A.5})$$

where $\mu_{n,j}$ and $\sigma_{n,j}^2$ are the mean and variance of the j -th latent variable for the n -th sample, respectively.

A sensitivity study examines $d_z \in \{56, 64\}$ and $\lambda_{\text{KL}} \in \{0, 10^{-6}, 10^{-4}\}$. Fig. A.2 compares the final loss components and identifies $d_z = 64$ and $\lambda_{\text{KL}} = 10^{-4}$ as the best trade-off between reconstruction accuracy and latent-space regularization.

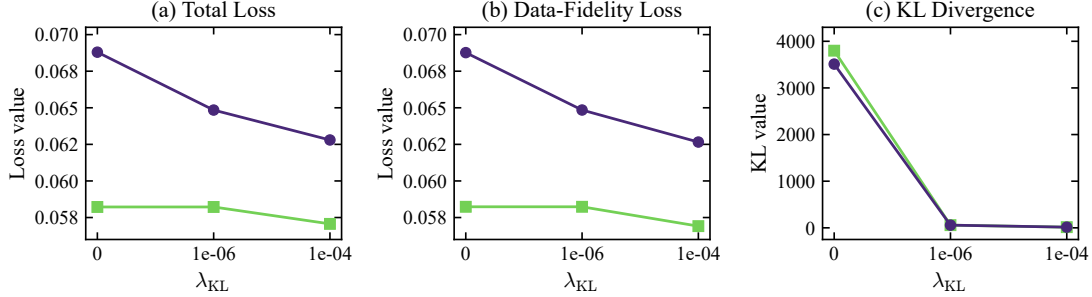


Fig. A.2. Hyperparameter sensitivity analysis for the VAE. The purple and green curves correspond to latent dimensions $d_z = 56$ and $d_z = 64$, respectively.

A.3 Reconstruction of Species Mass Fraction Fields

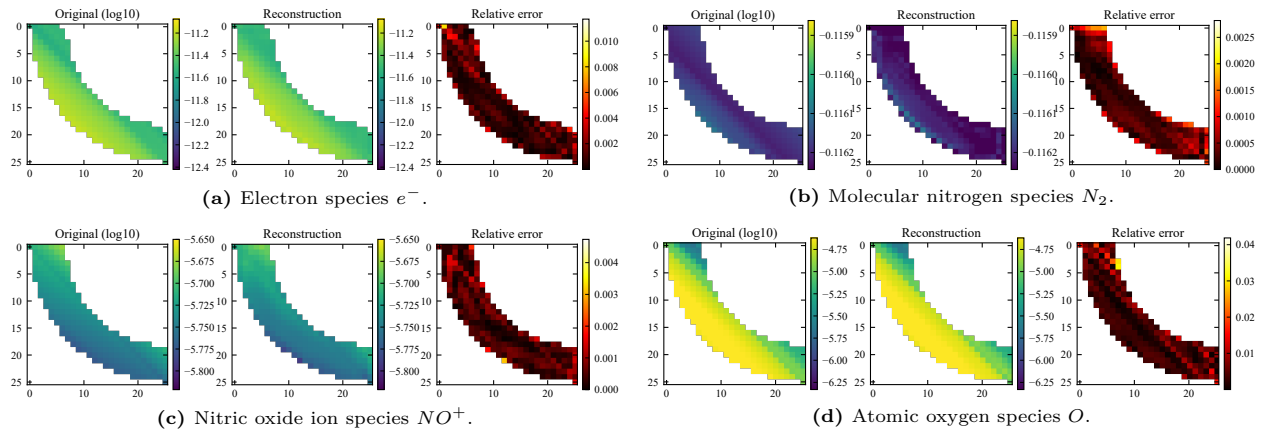


Fig. A.3. Representative slice reconstruction results for the species mass fraction fields. Within each subfigure, the original field, reconstructed field, and corresponding relative-error field are shown from left to right.

The reconstruction performance is illustrated using four representative species: e^- , N_2 , NO^+ , and O . These species cover electronic, molecular, ionic, and atomic components of the multispecies flow field. Fig. A.3 compares the original fields, reconstructed fields, and relative-error fields. Results for the remaining species exhibit similar reconstruction behavior and are omitted for conciseness.

Overall, the principal concentration and gradient patterns are well preserved, with errors mainly confined to regions of sharp variation. This confirms that the $d_z = 64$ latent representation preserves the essential information in the multispecies flow field. The VAE therefore provides a compact representation of the species mass fraction fields while maintaining satisfactory reconstruction accuracy, thereby reducing the input dimension and computational cost of the subsequent operator-learning model.

Appendix B. Approximate Rank-One Covariance Structure for the Aerothermal Case Study in Section 3.4

The aerothermal output covariance exhibits an approximately rank-one structure, which helps explain the distinct error hierarchy observed in the main text. This appendix provides a first-order explanation for this structure based on the one-dimensional effective stochastic input.

B.1 Effective Stochastic Dimension of the Input

In the aerothermal example, the freestream temperature, pressure, and density are fixed, and only the freestream Mach number varies randomly over $Ma \in [10.4, 16.4]$. For a prescribed Ma , the species mass fraction fields are deterministic CFD outputs, and the VAE latent vector $\mathbf{z}$ used by Branch 2 is therefore uniquely determined by Ma . Consequently, although the multi-branch model receives several input components, all samples lie on a one-dimensional manifold parameterized by Ma . The effective stochastic dimension is therefore one, unlike the random-field inputs used in the preceding PDE examples. This one-dimensional stochastic structure provides the basis for the approximately rank-one covariance obtained from the first-order Taylor expansion below.

B.2 First-Order Derivation of the Approximately Rank-One Covariance

Let the physical operator $\mathcal{G}$ map the input function $\mathbf{u}$ to the wall heat flux field $\mathbf{s} = \mathcal{G}(\mathbf{u}) \in \mathbb{R}^M$. For a perturbation $\delta\mathbf{u}$ about the mean input $\bar{\mathbf{u}}$, a first-order Taylor expansion gives

$$\mathbf{s} = \mathcal{G}(\bar{\mathbf{u}} + \delta\mathbf{u}) \approx \mathcal{G}(\bar{\mathbf{u}}) + \nabla\mathcal{G}(\bar{\mathbf{u}}) \delta\mathbf{u}, \quad (\text{B.1})$$

where $\nabla\mathcal{G}(\bar{\mathbf{u}})$ denotes the linearized sensitivity of the physical operator at $\bar{\mathbf{u}}$. Let $\boldsymbol{\mu}_s = \mathbb{E}[\mathbf{s}]$. The corresponding output covariance is approximated by

$$\boldsymbol{\Sigma} = \mathbb{E}[(\mathbf{s} - \boldsymbol{\mu}_s)(\mathbf{s} - \boldsymbol{\mu}_s)^\top] \approx \nabla\mathcal{G}(\bar{\mathbf{u}}) \text{Cov}(\delta\mathbf{u}) \nabla\mathcal{G}(\bar{\mathbf{u}})^\top. \quad (\text{B.2})$$

Under these conditions, the composite input function is completely determined by the single scalar Ma , namely $\mathbf{u} = \mathbf{u}(Ma)$. Let

$$Ma = \overline{Ma} + \delta Ma,$$

where $\overline{Ma} = \mathbb{E}[Ma]$ denotes the mean Mach number and δMa is the corresponding zero-mean fluctuation, satisfying $\mathbb{E}[\delta Ma] = 0$. Expanding this vector-valued function about $\overline{Ma}$ gives

$$\delta\mathbf{u} = \mathbf{u}(\overline{Ma} + \delta Ma) - \mathbf{u}(\overline{Ma}) \approx \left. \frac{\partial \mathbf{u}}{\partial Ma} \right|_{\overline{Ma}} \delta Ma =: \mathbf{v} \delta Ma \quad (\text{B.3})$$

where $\mathbf{v} = \left. \frac{\partial \mathbf{u}}{\partial Ma} \right|_{\overline{Ma}}$ is a deterministic vector in the input space and δMa is a scalar random variable. The first-order approximation to the input covariance is therefore

$$\text{Cov}(\delta\mathbf{u}) \approx \mathbb{E}[(\mathbf{v} \delta Ma)(\mathbf{v} \delta Ma)^\top] = \text{Var}(Ma) \mathbf{v} \mathbf{v}^\top. \quad (\text{B.4})$$

Substituting this result into Eq. (B.2) and defining $\mathbf{w} = \nabla\mathcal{G}(\bar{\mathbf{u}})\mathbf{v}$, we obtain

$$\boldsymbol{\Sigma} \approx \text{Var}(Ma) [\nabla\mathcal{G}(\bar{\mathbf{u}})\mathbf{v}] [\nabla\mathcal{G}(\bar{\mathbf{u}})\mathbf{v}]^\top = \text{Var}(Ma) \mathbf{w} \mathbf{w}^\top. \quad (\text{B.5})$$

The first-order approximation to the output covariance is therefore a scalar multiple of an outer product. Its rank is at most one and equals one whenever $\mathbf{w} \neq \mathbf{0}$. Hence, the one-dimensional input uncertainty directly induces an approximately rank-one output covariance after first-order propagation through the physical operator. This result is consistent with the observation in the main text that the Oracle SVD curve (T_1) approaches zero with only a few retained modes.

The vector

$$\mathbf{w} = \nabla\mathcal{G}(\bar{\mathbf{u}})\mathbf{v} = \left. \frac{\partial \mathbf{s}}{\partial Ma} \right|_{\overline{Ma}} \quad (\text{B.6})$$

has a direct physical interpretation. Its i th component represents the local sensitivity of the wall heat flux at the

i th wall grid point to a change in the freestream Mach number. Larger values of $|w_i|$ indicate regions with greater sensitivity to Ma , such as the region near the stagnation point.

To first order, the perturbation of the wall heat flux field satisfies

$$\delta \mathbf{s} \approx \mathbf{w} \delta Ma. \quad (\text{B.7})$$

Hence, all wall locations are driven by the same scalar fluctuation and differ only in their local sensitivities, producing coherent variations across the surface.

B.3 Interpretation of the Evaluation Metrics

Equation (B.5) directly gives the following first-order approximation to the correlation coefficient between any two wall locations i and j :

$$\rho_{ij} = \frac{\text{Cov}(s_i, s_j)}{\sqrt{\text{Var}(s_i) \text{Var}(s_j)}} \approx \frac{w_i w_j}{|w_i| |w_j|}, \quad w_i w_j \neq 0. \quad (\text{B.8})$$

Under these conditions, the wall heat flux exhibits a generally positive sensitivity to the freestream Mach number over the entire surface. Therefore, $w_i > 0$ and $\rho_{ij} \approx 1$ are expected. This agrees with Fig. 23, where the reference correlations are predominantly within 0.98–1.00.

The approximately rank-one structure also explains the sensitivity of the normalized Frobenius error. For $\Sigma \approx \lambda_1 \mathbf{u}_1 \mathbf{u}_1^\top$, nearly all covariance energy is concentrated in one dominant mode. Any relative error in this direction therefore contributes directly to the normalized Frobenius error and cannot be diluted by energy in other directions. By contrast, for a higher-rank compressible covariance, the energy is distributed across multiple directions, so an error in a single direction is naturally averaged and has a smaller relative effect. Therefore, for an approximately rank-one reference covariance, spatial correlation recovery provides a complementary metric that assesses whether the correlation structure is accurately recovered independently of the overall covariance magnitude.

Finally, this physical structure also highlights a limitation of the pointwise conditional covariance representation adopted in Prob-DeepONet. For an individual input $\mathbf{u}$, its conditional predictive covariance is

$$\hat{\Sigma}_{\text{Prob}}(\mathbf{u}) = \text{diag}(\hat{\sigma}_{s,1}^2(\mathbf{u}), \dots, \hat{\sigma}_{s,M}^2(\mathbf{u})). \quad (\text{B.9})$$

Thus, the conditional predictive component contains no off-diagonal dependence. The total predictive covariance additionally includes the covariance of the conditional means across random input realizations and therefore need not be diagonal. Nevertheless, when the reference response exhibits strong coherent cross-location correlations, restricting the conditional predictive covariance to a diagonal form limits the explicit representation of structured dependence in this component of the total predictive covariance.

By contrast, two-step MV-DeepONet models uncertainty in the modal coefficient space and maps the coefficient-space covariance to the physical output space through the shared basis functions, thereby allowing a generally non-diagonal conditional predictive covariance. This representation is better suited to capturing the strongly correlated response structure observed in the present aerothermal example.

References

- [1] S. Mohammadi, S. Cremaschi, Efficiency of uncertainty propagation methods for moment estimation of uncertain model outputs, *Comput. Chem. Eng.* 166 (2022) 107954.
- [2] W. Liu, T. Belytschko, A. Mani, Random field finite elements, *Int. J. Numer. Methods Eng.* 23 (1986) 1831–1845.
- [3] D. Xiu, G. Karniadakis, A new stochastic approach to transient heat conduction modeling with uncertainty, *Int. J. Heat Mass Transfer* 46 (2003) 4681–4693.
- [4] I. Simpson, S. Vicente, N. Campbell, Learning structured Gaussians to approximate deep ensembles, in: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2022, pp. 366–374.
- [5] S. Lee, W. Chen, A comparative study of uncertainty propagation methods for black-box-type problems, *Struct. Multidiscip. Optim.* 37 (2009) 239–253.
- [6] K. Abdi, B. Celse, K. McAuley, Propagating input uncertainties into parameter uncertainties and model prediction uncertainties—A review, *Can. J. Chem. Eng.* 102 (2024) 254–273.
- [7] R. Ghanem, P. Spanos, *Stochastic Finite Elements: A Spectral Approach*, Springer, New York, 1991.
- [8] D. Xiu, G. Karniadakis, The Wiener–Askey polynomial chaos for stochastic differential equations, *SIAM J. Sci. Comput.* 24 (2002) 619–644.
- [9] D. Xiu, J. Hesthaven, High-order collocation methods for differential equations with random inputs, *SIAM J. Sci. Comput.* 27 (2005) 1118–1139.
- [10] F. Petersen, A. Mishra, H. Kuehne, C. Borgelt, O. Deussen, M. Yurochkin, Uncertainty quantification via stable distribution propagation, in: *International Conference on Learning Representations*, 2024.
- [11] A. Shekhovtsov, B. Flach, Feed-forward propagation in probabilistic neural networks with categorical and max layers, in: *International Conference on Learning Representations*, 2018.
- [12] D. Betancourt, R. Muhanna, Interval deep learning for computational mechanics problems under input uncertainty, *Probab. Eng. Mech.* 70 (2022) 103370.
- [13] A. Sofi, G. Muscolino, F. Giunta, Propagation of uncertain structural properties described by imprecise Probability Density Functions via response surface method, *Probab. Eng. Mech.* 60 (2020) 103020.

- [14] R. Tripathy, I. Bilonis, M. Gonzalez, Gaussian processes with built-in dimensionality reduction: Applications to high-dimensional uncertainty propagation, *J. Comput. Phys.* 321 (2016) 191–223.
- [15] V. Giannella, F. Bardozzo, A. Postiglione, R. Tagliaferri, R. Sepe, E. Armentani, Neural networks for fatigue crack propagation predictions in real-time under uncertainty, *Comput. Struct.* 288 (2023) 107157.
- [16] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, M. Shahzad, W. Yang, R. Bamler, X. X. Zhu, A survey of uncertainty in deep neural networks, *Artif. Intell. Rev.* 56 (Suppl 1) (2023) 1513–1589.
- [17] R. Neal, *Bayesian Learning for Neural Networks*, Vol. 118 of Lecture Notes in Statistics, Springer, New York, 1996.
- [18] Y. Gal, Z. Ghahramani, Dropout as a Bayesian approximation: Representing model uncertainty in deep learning, in: *Proceedings of the 33rd International Conference on Machine Learning*, Vol. 48 of Proceedings of Machine Learning Research, 2016, pp. 1050–1059.
- [19] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, in: *Advances in Neural Information Processing Systems*, Vol. 30, 2017, pp. 6402–6413.
- [20] M. Sensoy, L. Kaplan, M. Kandemir, Evidential deep learning to quantify classification uncertainty, in: *Advances in Neural Information Processing Systems*, Vol. 31, 2018, pp. 3179–3189.
- [21] D. Nix, A. Weigend, Estimating the mean and variance of the target probability distribution, in: *Proceedings of the 1994 IEEE International Conference on Neural Networks (ICNN'94)*, Vol. 1, IEEE, 1994, pp. 55–60.
- [22] M. Seitzer, A. Tavakoli, D. Antic, G. Martius, On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks, in: *International Conference on Learning Representations*, 2022.
- [23] L. Sluijterman, E. Cator, T. Heskes, Optimal training of mean variance estimation neural networks, *Neurocomputing* 597 (2024) 127929.
- [24] U. Subedi, A. Tewari, Operator learning: A statistical perspective, *Annu. Rev. Stat. Appl.* 13 (2026) 123–148.
- [25] N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, A. Anandkumar, Neural operator: Learning maps between function spaces with applications to PDEs, *J. Mach. Learn. Res.* 24 (89) (2023) 1–97.
- [26] T. Chen, H. Chen, Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems, *IEEE Trans. Neural Netw.* 6 (4) (1995) 911–917.
- [27] L. Lu, P. Jin, G. Pang, Z. Zhang, G. E. Karniadakis, Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators, *Nat. Mach. Intell.* 3 (2021) 218–229.
- [28] S. Goswami, A. Bora, Y. Yu, G. Karniadakis, Physics-informed deep neural operator networks, in: T. Rabczuk, K.-J. Bathe (Eds.), *Machine Learning in Modeling and Simulation: Methods and Applications*, Springer, Cham, 2023, pp. 219–254.
- [29] I. Sahin, C. Moya, A. Mollaali, G. Lin, G. Paniagua, Deep operator learning-based surrogate models with uncertainty quantification for optimizing internal cooling channel rib profiles, *Int. J. Heat Mass Transfer* 219 (2024) 124813.
- [30] S. Goswami, M. Yin, Y. Yu, G. E. Karniadakis, A physics-informed variational DeepONet for predicting crack path in quasi-brittle materials, *Comput. Methods Appl. Mech. Eng.* 391 (2022) 114587.
- [31] S. Cai, Z. Wang, L. Lu, T. A. Zaki, G. E. Karniadakis, DeepM&Mnet: Inferring the electroconvection multiphysics fields based on operator approximation by neural networks, *J. Comput. Phys.* 436 (2021) 110296.
- [32] Z. Mao, L. Lu, O. Marxen, T. A. Zaki, G. E. Karniadakis, DeepM&Mnet for hypersonics: Predicting the coupled flow and finite-rate chemistry behind a normal shock using neural-network approximation of operators, *J. Comput. Phys.* 447 (2021) 110698.
- [33] L. Lu, X. Meng, S. Cai, Z. Mao, S. Goswami, Z. Zhang, G. E. Karniadakis, A comprehensive and fair comparison of two neural operators (with practical extensions) based on FAIR data, *Comput. Methods Appl. Mech. Eng.* 393 (2022) 114778.
- [34] G. Lin, C. Moya, Z. Zhang, B-DeepONet: An enhanced Bayesian DeepONet for solving noisy parametric PDEs using accelerated replica exchange SGLD, *J. Comput. Phys.* 473 (2023) 111713.
- [35] S. Garg, S. Chakraborty, VB-DeepONet: A Bayesian operator learning framework for uncertainty quantification, *Eng. Appl. Artif. Intell.* 118 (2023) 105685.
- [36] S. Lone, S. De, R. Nayek, α -VI DeepONet: A prior-robust variational Bayesian approach for enhancing DeepONets with uncertainty quantification, *Comput. Methods Appl. Mech. Eng.* 449 (2026) 118552.
- [37] Y. Yang, G. Kissas, P. Perdikaris, Scalable uncertainty quantification for deep operator networks using randomized priors, *Comput. Methods Appl. Mech. Eng.* 399 (2022) 115399.
- [38] C. Moya, A. Mollaali, Z. Zhang, L. Lu, G. Lin, Conformalized-DeepONet: A distribution-free framework for uncertainty quantification in deep operator networks, *Physica D* 471 (2025) 134418.
- [39] K. Kobayashi, S. Garg, F. Ahmed, S. Chakraborty, S. Alam, Distribution-free uncertainty-aware virtual sensing via conformalized neural operators, *arXiv preprint arXiv:2507.11574* (2025). doi:10.48550/arXiv.2507.11574.
- [40] L. Ma, L. Guo, H. Wu, T. Zhou, Deep set based operator learning with uncertainty quantification, *J. Comput. Phys.* 562 (2026) 115011.
- [41] P. Huynh, R. Archibald, F. Bao, Diffusion-based stochastic operator networks for uncertainty quantification in stochastic partial differential equations, *arXiv preprint arXiv:2605.17107* (2026). doi:10.48550/arXiv.2605.17107.
- [42] G. Faza, J. Wauters, F. Cuzzolin, H. Hallez, D. Moens, Direct interval propagation methods using neural-network surrogates for uncertainty quantification in physical systems surrogate model, *Knowl.-Based Syst.* 341 (2026) 115824.
- [43] C. Moya, S. Zhang, G. Lin, M. Yue, DeepONet-grid-UQ: A trustworthy deep operator framework for predicting the power grid's post-fault trajectories, *Neurocomputing* 535 (2023) 166–182.
- [44] N. Winovich, M. Daneker, L. Lu, G. Lin, Active operator learning with predictive uncertainty quantification for partial differential equations, *J. Comput. Phys.* 555 (2026) 114791.
- [45] S. Lee, Y. Shin, On the training and generalization of deep operator networks, *SIAM J. Sci. Comput.* 46 (2024) C273–C296.
- [46] E. Kiyani, M. Manav, N. Kadivar, L. D. Lorenzis, G. Karniadakis, Predicting crack nucleation and propagation in brittle materials using deep operator networks with diverse trunk architectures, *Comput. Methods Appl. Mech. Eng.* 441 (2025) 117984.
- [47] Q. Jiang, M. Salvadori, D. Ota, V. Shankar, K. Shukla, Complex valued deep operator network (DeepONet) $\mathcal{G}$ for three dimensional Maxwell's equations: $\mathcal{G} \in \mathbb{C}^{m \times n}$, *J. Comput. Phys.* 562 (2026) 114993.
- [48] H. Jin, B. Zhang, Q. Cao, E. Zhang, A. Bora, S. Krishnaswamy, G. Karniadakis, H. Espinosa, Characterization and inverse design of stochastic mechanical metamaterials using neural operators, *Adv. Mater.* 37 (2025) 2420063.

- [49] S. Park, Y. Shin, J. Choo, Deep operator network for surrogate modeling of poroelasticity with random permeability fields, arXiv preprint arXiv:2509.11966 (2025). doi:10.48550/arXiv.2509.11966.
- [50] A. Peyvan, V. Oommen, A. D. Jagtap, G. E. Karniadakis, RiemannONets: Interpretable neural operators for Riemann problems, *Comput. Methods Appl. Mech. Eng.* 426 (2024) 116996.
- [51] K. Shukla, J. Ratchford, L. Bravo, V. Oommen, N. Plewacki, A. Ghoshal, G. Karniadakis, Deep operator learning-based surrogate models for aerothermodynamic analysis of AEDC hypersonic waverider, arXiv preprint arXiv:2405.13234 (2024). doi:10.48550/arXiv.2405.13234.
- [52] C. Eckart, G. Young, The approximation of one matrix by another of lower rank, *Psychometrika* 1 (1936) 211–218.
- [53] Y. Yu, T. Wang, R. Samworth, A useful variant of the Davis–Kahan theorem for statisticians, *Biometrika* 102 (2015) 315–323.
- [54] V. Koltchinskii, K. Lounici, Concentration inequalities and moment bounds for sample covariance operators, *Bernoulli* 23 (1) (2017) 110–133.
- [55] T. Cai, C. Zhang, H. Zhou, Optimal rates of convergence for covariance matrix estimation, *Ann. Stat.* 38 (2010) 2118–2144.
- [56] A. Khosravi, S. Nahavandi, D. Creighton, A. F. Atiya, Comprehensive review of neural network-based prediction intervals and new advances, *IEEE Trans. Neural Netw.* 22 (2011) 1341–1356.
- [57] P. Fife, *Mathematical Aspects of Reacting and Diffusing Systems*, Vol. 28 of Lecture Notes in Biomathematics, Springer, Berlin, Heidelberg, 1979.
- [58] R. Aris, *The mathematical theory of diffusion and reaction in permeable catalysts*, Vol. 1: The theory of the steady state, Clarendon Press, Oxford, 1975.
- [59] A. Turing, The chemical basis of morphogenesis, *Bull. Math. Biol.* 52 (1990) 153–197.
- [60] R. Cantrell, C. Cosner, *Spatial ecology via reaction–diffusion equations*, John Wiley & Sons, Chichester, 2004.
- [61] J. England, J. Cardy, Morphogen gradient from a noisy source, *Phys. Rev. Lett.* 94 (2005) 078101.
- [62] M. Vlad, D. Rothman, J. Ross, Random channel kinetics for reaction–diffusion systems, *Physica D* 239 (2010) 739–745.
- [63] G. Whitham, *Linear and Nonlinear Waves*, Wiley, New York, 1974.
- [64] J. Bec, K. Khanin, Burgers turbulence, *Phys. Rep.* 447 (2007) 1–66.
- [65] G. Buendia, G. Viswanathan, V. Kenkre, Multifractality of random walks in the theory of vehicular traffic, *Phys. Rev. E* 78 (2008) 056110.
- [66] Y. Rubin, *Applied Stochastic Hydrogeology*, Oxford University Press, New York, 2003.
- [67] V. Godoy, L. Zuquette, J. Gómez-Hernández, Stochastic analysis of three-dimensional hydraulic conductivity upscaling in a heterogeneous tropical soil, *Comput. Geotech.* 100 (2018) 174–187.
- [68] D. Passiatore, L. Sciacovelli, P. Cinnella, G. Pascazio, Thermochemical non-equilibrium effects in turbulent hypersonic boundary layers, *J. Fluid Mech.* 941 (2022) A21.
- [69] C. Williams, M. D. Renzo, P. Moin, Turbulence–chemistry interaction in a non-equilibrium hypersonic boundary layer, *J. Fluid Mech.* 1017 (2025) A30.
- [70] M. MacLean, E. Marineau, R. Parker, A. Dufrene, M. Holden, P. DesJardin, Effect of surface catalysis on measured heat transfer in expansion tunnel facility, *J. Spacecr. Rockets* 50 (2013) 470–475.
- [71] J. B. Dsouza, N. Castelino, V. Viti, H. H. Vu, S. Gao, Numerical study of the effects of thermo-chemical non-equilibrium and surface catalysis on two hypersonic re-entry bodies, in: *AIAA SciTech 2024 Forum*, AIAA, 2024, AIAA Paper 2024-2086.
- [72] C. Park, Assessment of two-temperature kinetic model for ionizing air, *J. Thermophys. Heat Transf.* 3 (1989) 233–244.
- [73] R. Gupta, J. Yos, R. Thompson, K.-P. Lee, A review of reaction rates and thermodynamic and transport properties for an 11-species air model for chemical and thermal nonequilibrium calculations to 30000 K, NASA Reference Publication 1232, NASA Langley Research Center, Hampton, VA (1990).
- [74] P. Gnoffo, R. Gupta, J. Shinn, Conservation equations and physical models for hypersonic air flows in thermal and chemical nonequilibrium, NASA Technical Paper 2867, NASA Langley Research Center, Hampton, VA (1989).
- [75] J. A. Rataczak, I. D. Boyd, J. W. McMahon, Surrogate models for hypersonic aerothermodynamics and aerodynamics using gaussian process regression, in: *AIAA SciTech 2024 Forum*, AIAA, 2024, AIAA Paper 2024-0461.
- [76] M. Capriati, A. Cortesi, T. Magin, P. Congedo, Stagnation point heat flux characterization under numerical error and boundary conditions uncertainty, *Eur. J. Mech. B Fluids* 95 (2022) 221–230.
- [77] J. Lu, J. Li, Z. Song, W. Zhang, C. Yan, Uncertainty and sensitivity analysis of heat transfer in hypersonic three-dimensional shock waves/turbulent boundary layer interaction flows, *Aerosp. Sci. Technol.* 123 (2022) 107447.
- [78] M. Wright, D. Bose, G. Candler, A data parallel line relaxation method for the Navier–Stokes equations, *AIAA J.* 36 (1998) 1603–1609.
- [79] P. Jin, S. Meng, L. Lu, MIONet: Learning multiple-input operators via tensor product, *SIAM J. Sci. Comput.* 44 (6) (2022) A3490–A3514.