Abstract

This paper introduces multivariate input-output models to predict the errors and dimensions of local parametric reduced-order models. We refer to these mappings as the MP-LROM models since their outputs are multivariate predictions of local reduced-order models characteristics. We employ Gaussian Processes and Artificial Neural Networks to construct approximations of these multivariate mappings. The predicted reduced-order models errors are compared against the multi-fidelity correction and ROMES predictions, whereas the predicted reduced-order dimensions are tested against the standard method based on the spectrum of snapshots matrix. Next, two applications of the newly proposed error models are considered. First, the error models are utilized by a sampling greedy algorithm to generate decompositions of one dimensional parametric domains with overlapping regions and associated local reduced-order models that are accurate to within an admissible prescribed threshold. Once a parametric domain decomposition is constructed, for any parametric configuration there exists a local reduced-order model whose solution is accurately estimated a-priori. Second, the error models are used to generate a hierarchy of the available local bases, local reduced-order and high-fidelity models producing the most accurate solutions for an arbitrary parametric configuration. This hierarchy is then employed in an attempt to construct more accurate reduced-order models using: (1) Lagrange interpolation of reduced bases in the matrix space; (2) Lagrange interpolation of reduced bases in the tangent space of the Grassmann manifold; (3) concatenation of reduced bases followed by a Gram-Schmidt orthogonalization process; (4) Lagrange interpolation of high-fidelity model solutions. Numerical results with a viscous Burgers model illustrate the potential of our input-output models to improve the design of parametric reduced-order models. For high-dimensional parametric spaces, the scalability challenge of MP-LROM models may be addressed using the active subspace method.

Keywords: local reduced-order models, Proper Orthogonal Decomposition, regression machine learning techniques, interpolation methods, Grassmann manifold.

1. Introduction

Many physical phenomena are described mathematically by partial differential equations (PDEs), and, after applying suitable discretization schemes, are simulated on a computer. PDE-based models frequently require calibration and parameter tuning in order to provide realistic simulation results. Recent developments in the field of uncertainty quantification [1–4] provide the necessary tools for validation of such models even in the context of variability and lack of knowledge of the input parameters. Techniques to propagate uncertainties through models include direct evaluation for linearly parameterized models, sampling methods such as Monte Carlo [5], Latin hypercube [6] and quasi-Monte Carlo techniques [7], perturbation methods [8–10] and spectral representation [1, 11, 12]. While stochastic Galerkin methods [1] are intrusive in nature, Monte Carlo sampling methods and stochastic collocations [11] do not require the modification of existing codes and hence they are non-intrusive. While uncertainty propagation techniques can measure the
impact of uncertain parameters on some quantities of interest, they often become infeasible due to the large number of model realizations requirement. Similar difficulties are encountered when solving Bayesian inference problems since sampling from posterior distribution is required.

For large-scale simulations, the variational \([13, 18]\) and ensemble \([19, 22]\) based data assimilation approaches are widely used in practice. Their efficiency decreases with increasing computational complexity of the underlying physical models. However, increasing model complexity is unavoidable as science fields progress. For example, finer space resolution of the underlying PDE models is one of the most important factors contributing to the one day/decade growth rate of the reliability of atmospheric weather predictions \([23, 24]\).

The need for computational efficiency motivated the development of surrogate models such as response surfaces, low resolution, and reduced-order models. Data fitting or response surface models \([2]\) are data-driven models. The underlying physics remain unknown and only the input-output behavior of the model is considered. Data fitting can use techniques such as regression, interpolation, radial basis function, Gaussian Processes, Artificial Neural Networks and other supervised machine-learning methods. The latter techniques can automatically detect patterns in data, and one can use them to predict future data under uncertainty in a probabilistic framework \([25]\). While easy to implement due to the non-intrusive nature, the prediction abilities may suffer since the governing physics are not specifically accounted for.

Low-fidelity models attempt to reduce the computational burden of the high-fidelity models by neglecting some of the physical aspects (e.g., replacing Navier-Stokes and Large Eddy Simulations with inviscid Euler’s equations and Reynolds-Averaged Navier-Stokes \([26, 28]\), or decreasing the spatial resolution \([29, 30]\)). The additional approximations, however, may considerably degrade the physical solution with only a modest decrease of the computational load.

Reduced basis \([31, 35]\) and Proper Orthogonal Decomposition \([36, 40]\) are two of the popular reduced-order modeling (ROM) strategies available in the literature. Data analysis is conducted to extract basis functions from experimental data or detailed simulations of high-dimensional systems (method of snapshots \([41, 43]\)), for subsequent use in Galerkin projections that yield low dimensional dynamical models. While these type of models are physics-based and therefore require intrusive implementations, they are usually more robust than data fitting and low-fidelity models. However, since surrogate model robustness depends heavily on the problem, it must be carefully analyzed especially for large-scale nonlinear dynamical systems.

ROM robustness in a parametric setting can be achieved by constructing a global basis \([44, 45]\), but this strategy generates large dimensional bases that may lead to slow reduced-order models. Local approaches have been designed for parametric or time domains generating local bases for both the state variables \([46, 47]\) and non-linear terms \([48, 49]\). A recent survey of state-of-the-art methods in projection-based parametric model reduction is available in \([50]\).

In this study, we propose input-output models to predict the characteristics of local parametric reduced-order models. Specifically, our input-output models are designed to provide multivariate predictions of local parametric reduced-order models errors and dimensions, so we refer to them as MP-LROM models. Let us consider a local parametric reduced-order model of dimension \(K_{POD}\) constructed using a high-fidelity solution associated with the parameter configuration \(\mu_p\). Let \(\varepsilon_{HF, \mu_p, K_{POD}}\) be the error of the low reduced-order model solution with respect to the high-fidelity solution for a viscosity parameter configuration \(\mu\).

Our first MP-LROM model consists in the mapping \(\{\mu, \mu_p, K_{POD}\} \mapsto \log \varepsilon_{HF, \mu_p, K_{POD}}\) and is designed to predict the error of a local parametric reduced-order model. Our proposed approach is inspired from the multi-fidelity correction \([51]\) and ROMES \([52]\) methodologies. Multi-fidelity correction \([26, 51, 53, 54]\) has been developed for low-fidelity models in the context of optimization. They simulate the input-output relation \(\mu \mapsto \varepsilon_{HF, \mu}\), where \(\varepsilon_{HF, \mu}\) is the low-fidelity model error depending on a global reduced basis with a constant reduced-order model dimension. The ROMES method \([52]\) introduced the concept of error indicators for global reduced-order models and generalized the multi-fidelity correction framework by approximating the mapping \(\rho(\mu) \mapsto \log \varepsilon_{HF, \mu}\). The error indicators \(\rho(\mu)\) include rigorous error bounds and reduced-order residual norms. No variation of the reduced basis dimension was taken into...
account. By estimating the log of the reduced-order model error instead of the error itself, the input-output map exhibits a lower variance as shown by our numerical experiments as well as those in [52].

The second proposed MP-LROM model addresses the issue of a-priori selection of the reduced basis dimension for a prescribed accuracy of the reduced solution. The standard approach is to analyze the spectrum of the snapshots matrix, and uses the largest singular value removed from the expansion to estimate the accuracy level [55]. To take into account the error due to the full-order-model equations projection in the reduced space, here we propose the mapping \( \{ \mu, \mu_p, \log \varepsilon^{HF}_{\mu, \mu_p, K_{POD}} \} \mapsto K_{POD} \) to predict the dimension of a local parametric reduced-order model given a prescribed error threshold.

To approximate the mappings \( \{ \mu, \mu_p, K_{POD} \} \mapsto \log \varepsilon^{HF}_{\mu, \mu_p, K_{POD}} \) and \( \{ \mu_p, \mu, \log \varepsilon^{HF}_{\mu, \mu_p, K_{POD}} \} \mapsto K_{POD} \), we proposed regression models constructed using Gaussian Processes (GP) [56, 57] and Artificial Neural Networks (ANN). In the case of one dimensional Burgers model, the resulted MP-LROM error models are accurate and their predictions are compared against those obtained by the multi-fidelity correction and ROMES models. The predicted dimensions of local reduced-order models using our proposed MP-LROM models are more accurate than those derived using the standard method based on the spectrum of snapshots matrix.

The second part of the paper focuses on applications of MP-LROM error models to construct decompositions of a parametric domain and generate a hierarchy of local bases, reduced-order models and high-fidelity trajectories.

By constructing decompositions of the parametric domain with overlapping feasible regions where local reduced-order models are accurate to within an admissible prescribed threshold, we address the robustness issue of Proper Orthogonal Decomposition (POD) local reduced-order models. The union of the feasible regions forms a decomposition of the parametric domain. Different thresholds lead to different domain decompositions. The essential ingredient is the MP-LROM error model which is used to sample the parametric domain and generates a feasible region where a specific local reduced-order model provides accurate solutions within a prescribed tolerance. We then use a greedy approach to sweep the parameter domain and cover it with such feasible regions. The current methodology is designed for one dimensional parametric spaces and it is applied to the viscous 1D-Burgers model. A decomposition for the viscosity domain is generated for various error thresholds. Once the decomposition is constructed there is no need to run the high-fidelity model again, since for each parameter value \( \mu \) there exists a parameter \( \mu_p \), and the associated reduced-order model (basis and reduced operators), whose solution error is accurately estimated a-priori. The dimension \( K_{POD} \) of the local basis is usually small since it depends only on one high-fidelity model trajectory.

In addition, we construct a hierarchy of the available local bases, local reduced-order and high-fidelity models for a specific parametric configuration \( \mu^* \). Similar MP-LROM error models are employed to predict the errors of reduced-order models and generate a hierarchy of local bases and models sorted from an accuracy point of view.

For the parametric configuration \( \mu^* \), one can combine the available hierarchy in an attempt to generate more accurate reduced-order models. Three different approaches are compared here; i.e., bases interpolation, bases concatenation, and high-fidelity model solutions interpolation using the hierarchy developed before. For the first method, we perform a Lagrangian interpolation of the bases in the matrix space [58], or linearly interpolate their projections onto some local coordinate systems [58, 59]. The second method follows the idea of the spanning ROM introduced in [60], where a projection basis is created by concatenating some of the available bases for an arbitrary parameter \( \mu^* \). The third method interpolates the associated high-fidelity solutions and then extracts the singular vectors to generate a new basis and local reduced-order model.

The remainder of the paper is organized as follows. Section 2 reviews the reduced-order modeling parametric framework. The MP-LROM models and the regression machine learning methods used in this study to approximate the MP-LROM mappings are described in details in Section 3. The applications of MP-LROM error models to construct decompositions of the parametric domain and generate a hierarchy of local bases, reduced-order models and high-fidelity trajectories are discussed in Section 4. Bases interpolation and concatenation and high-fidelity model solutions interpolation are discussed in Section 5. Section 6 describes the viscous 1D-Burgers
model, compares the performances of the MP-LROM and state of the art models and analyzes the MP-LROM error model applications to solve the proposed problems: (1) constructing a decomposition of the 1D-Burgers parametric domain; (2) designing a hierarchy of accurate local bases and models. The use of hierarchic local bases, local reduced-order and high-fidelity models for generating new bases via concatenation and interpolation is also described here. Conclusions are drawn in Section 4.

2. Parametrized reduced-order modeling

Proper Orthogonal Decomposition has been successfully applied in numerous applications such as compressible flow [61] and computational fluid dynamics [62–64], to mention a few. It can be thought of as a Galerkin approximation in the state variable built from functions corresponding to the solution of the physical system at specified time instances. A system reduction strategy for Galerkin models of fluid flows based on a partition in slow, dominant, and fast modes, has been proposed in [65]. Closure models and stabilization strategies for POD of turbulent flows have been investigated in [66, 67].

In this paper we consider discrete inner products (Euclidean dot product), though continuous products may be employed as well. Generally, an unsteady problem can be written in semi-discrete form as an initial value problem; i.e., as a system of nonlinear ordinary differential equations

$$\frac{d\bar{x}(\mu, t)}{dt} = F(x, t, \mu), \quad \bar{x}(\mu, 0) = x_0 \in \mathbb{R}^{N_{state}}, \quad \mu \in \mathcal{P}. \quad (1)$$

The input-parameter \(\mu\) typically characterizes the physical properties of the flow. By \(\mathcal{P}\) we denote the input-parameter space. For a given parameter configuration \(\mu_p\), we select an ensemble of \(N_t\) time instances of the flow \(x(\mu_p, t_1), \ldots, x(\mu_p, t_{N_t}) \in \mathbb{R}^{N_{state}}\), where \(N_{state}\) is the total number of discrete model variables, and \(N_t \in \mathbb{N}^+\). The POD method chooses an orthonormal basis \(U_{\mu_p} = [u_{1\mu_p}^T \cdots u_{K_{P POD}\mu_p}^T] \in \mathbb{R}^{N_{state} \times K_{P POD}},\) such that the mean square error between \(x(\mu_p, t_i)\) and the POD expansion \(\bar{x}_{\mu_p}^{POD}(t_i) = U_{\mu_p}^T \bar{x}_{\mu_p}(\mu, t_i), \bar{x}_{\mu_p}(\mu, t_i) = U_{\mu_p}^T x(\mu_p, t_i) \in \mathbb{R}^{K_{P POD}},\) is minimized on average. The POD space dimension \(K_{P POD} \ll N_{state}\) is appropriately chosen to capture the dynamics of the flow. Algorithm 1 describes the reduced-order basis construction procedure [68].

**Algorithm 1 POD basis construction**

1. Compute the singular value decomposition for the snapshots matrix \([x(\mu_p, t_1) \cdots x(\mu_p, t_{N_t})]\) = \(\bar{U}_{\mu_p} \Sigma_{\mu_p} \bar{V}_{\mu_p}^T\), with the singular vectors matrix \(\bar{U}_{\mu_p} = [u_{1\mu_p}^T \cdots u_{K_{P POD}\mu_p}^T]\).
2. Using the singular-values \(\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{N_t} \geq 0\) stored in the diagonal matrix \(\Sigma_{\mu_p}\), define \(I(m) = \sum_{i=1}^{m} \lambda_i^2 / \sum_{i=1}^{N_t} \lambda_i^2\). It is common to select \(\gamma = 0.99\). The basis \(U_{\mu_p}\) consists of the first \(K_{P POD}\) columns of \(\bar{U}_{\mu_p}\).

Next, a Galerkin projection of the full model state (1) onto the space \(X^{K_{P POD}}\) spanned by the POD basis elements is used to obtain the reduced-order model

$$\frac{d\bar{x}_{\mu_p}(\mu, t)}{dt} = U_{\mu_p}^T F(U_{\mu_p} \bar{x}_{\mu_p}(\mu, t), t, \mu), \quad \bar{x}_{\mu_p}(\mu, 0) = U_{\mu_p}^T x_0. \quad (2)$$

The notation \(\bar{x}_{\mu_p}(\mu, t)\) expresses the solution dependence on the varying parameter \(\mu\) and also on \(\mu_p\) the configuration whose associated high-fidelity trajectory was employed to generate the POD basis. While being accurate for \(\mu = \mu_p\), the reduced model (2) may lose accuracy when moving away from the initial setting. Several strategies have been proposed to derive a basis that spans the entire parameter space. These include the reduced basis method combined with the use of error estimates [34, 35, 69], global POD [70, 71], Krylov-based sampling methods [72, 73], and greedy techniques [74, 75]. The fundamental assumption used by these approaches is that a smooth low-dimensional global manifold characterizes the model solutions over the entire parameter domain. However, in order to ensure high accuracy of the reduced solution across the parameter space, the dimension of the reduced basis has to be increased in practice, leading to higher on-line computational costs. To alleviate this drawback we propose an alternative approach based on local...
parametric reduced-order models to be introduced in Subsection 4.1.

The efficiency of the POD-Galerkin technique is limited to linear or bilinear terms, since the projected nonlinear terms at every discrete time step still depend on the number of variables of the full model. In case of low-order polynomial nonlinearities, the tensorial POD technique [68] can be employed to efficiently remove the dependence on the full dimension by manipulating the order of computations. A considerable reduction in complexity is achieved by the Discrete Empirical Interpolation Method (DEIM) [76, 77], a discrete variation of Empirical Interpolation Method [78–80], gappy POD [81, 82] and collocation methods [83–85] for any type of nonlinear terms. A comparative analysis of Missing Point Estimation method [84], Gappy POD method, and Discrete Empirical Interpolation Method applied to a biological model is available in [86].

The quality of the reduced-order basis can be enhanced by applying a posteriori methods [87]. Among them, we mention here strategies relying on additional high-fidelity simulations [85, 88–90] and h-refinement analogy [91].

3. Multivariate prediction of local reduced-order models characteristics (MP-LROM)

We propose multivariate input-output models

$$\phi : z \mapsto y,$$

$$z \in \mathbb{R}^r$$, to predict characteristics $$y \in \mathbb{R}$$ of local parametric reduced-order models (2). Here we focus on predicting the error of the local reduced-order model

$$\varepsilon_{HF}^{\mu,\mu_p,K_{POD}} = \|x(\mu, t_1) - U_{\mu_p} \hat{x}_{\mu_p}(\mu, t_1) \cdot x(\mu, t_2) - U_{\mu_p} \hat{x}_{\mu_p}(\mu, t_2) \cdot \cdots x(\mu, t_{N_t}) - U_{\mu_p} \hat{x}_{\mu_p}(\mu, t_{N_t})\|_F,$$

where $$\| \cdot \|_F$$ denotes the Frobenius norm, and the dimension $$K_{POD}$$ of the reduced-order model as explained in details below. In this study, these input-output models are constructed using supervised machine learning techniques to be discussed in Section 3.3.

3.1. Error Model

Inspired from the multi-fidelity correction and ROMES methodologies we introduce an input-output model to predict the level of error $$\varepsilon_{HF}^{\mu,\mu_p,K_{POD}}$$ [3]. In contrast with ROMES and multi-fidelity correction models that predict the error of global reduced-order models with fixed dimensions, using univariate functions, here we propose a multivariate model

$$\phi_{MP-LROM}^{\varepsilon} : \{\mu, \mu_p, K_{POD}\} \mapsto \log \varepsilon_{HF}^{\mu,\mu_p,K_{POD}}$$

(5)

to predict the error of local reduced-order models [2] of various dimensions. Since the dimension of basis determines the level of error we include it among the input variables. To design models with reduced variances we look to approximate the logarithm of the error as suggested in [52].

For high-dimensional parametric spaces, ROMES method handles well the curse of dimensionality with their proposing univariate models. In combination with active subspace method [92], we can reduce the number of input variables in case the amount of variability in the parametric space is mild. This will increase our error model feasibility even for high-dimensional parametric space.

3.2. Dimension of the reduced basis

The basis dimension represents one of the most important characteristics of a reduced-order model. The reduced manifold dimension directly affects both the on-line computational complexity of the reduced-order model and its accuracy [93–95]. By increasing the dimension of the basis, the projection error usually decreases and the accuracy of the reduced-order model is enhanced. However this is
not necessarily valid as seen in [96 Section 5]. Nevertheless the spectrum of the snapshots matrix offers guidance regarding the choice of the reduced basis dimension when some prescribed reduced-order model error is desired. However the accuracy depends also on the ‘in-plane’ error, which is due to the fact that the full-order-model equations are projected on the reduced subspace [97, 98].

We seek to predict the dimension of the local reduced-order model (2) by accounting for both the orthogonal projection error onto the subspace, which is computable by the sum of squares of singular values, and the ‘in-plane’ error. As such we propose to model the mapping

$$\phi^{d}_{MP-LROM} : \{\mu_p, \log \varepsilon_{HF}^{\mu_p, \mu_p}, K_{POD}\} \mapsto K_{POD}. \quad (6)$$

Once such model is available, given a positive threshold \(\bar{\varepsilon}\) and a parametric configuration \(\mu_p\), we will be able to predict the dimension \(K_{POD}\) of the basis \(U_{\mu_p}\), such that the reduced-order model error satisfies

$$\| x(\mu_p, t_1) - U_{\mu_p} x_{\mu_p}(\mu_p, t_1) \cdot x(\mu_p, t_2) - U_{\mu_p} x_{\mu_p}(\mu_p, t_2) \cdot \ldots \cdot x(\mu_p, t_{N_t}) - U_{\mu_p} x_{\mu_p}(\mu_p, t_{N_t}) \|_F \approx \bar{\varepsilon}. \quad (7)$$

3.3. Supervised Machine Learning Techniques

In order to estimate the level of reduced-order model solution error \(\varepsilon^{HF}_{\mu_p, \mu_p, K_{POD}}\) and the reduced basis dimension \(K_{POD}\), we will use regression machine learning methods to approximate the maps \(\phi^{e}_{MP-LROM}\) and \(\phi^{d}_{MP-LROM}\) described in (5) and (6).

Artificial Neural Networks and Gaussian Processes are used to build a probabilistic model \(\phi : z \mapsto \hat{y}\), where \(\phi\) is a transformation function that learns through the input features \(z\) to estimate the deterministic output \(y\) [25]. These probabilistic models are approximations of the mappings introduced in (3). The input features \(z\) can be either categorical or ordinal. The real-valued random variable \(\hat{y}\) is expected to have a low variance and reduced bias. The features of \(z\) should be descriptive of the underlying problem at hand [99].

The accuracy and stability of estimations are assessed using the K-fold cross-validation technique. The samples are split into K subsets (“folds”), where typically \(3 \leq K \leq 10\). The model is trained on \(K - 1\) sets and tested on the \(K\)-th set in a round-robin fashion [25]. Each fold induces a specific error quantified as the average of the absolute values of the differences between the predicted and the \(K\)-th set values.

$$E_{\text{fold}} = \frac{\sum_{i=1}^{N} |\hat{y}^i - y^i|}{N}, \quad \text{VAR}_{\text{fold}} = \frac{\sum_{i=1}^{N} (\hat{y}^i - E_{\text{fold}})^2}{N - 1}, \quad \text{fold} = 1, 2, \ldots, K, \quad (8a)$$

where \(N\) is the number of test samples in the fold. The error is then averaged over all folds:

$$E = \frac{\sum_{\text{fold}=1}^{K} E_{\text{fold}}}{K}, \quad \text{VAR} = \frac{\sum_{\text{fold}=1}^{K} (E_{\text{fold}} - E)^2}{K - 1}. \quad (8b)$$

The variance of the prediction results (8a) accounts for the sensitivity of the model to the particular choice of data set. It quantifies the stability of the model in response to the new training samples. A smaller variance indicates more stable predictions, however, this sometimes translates into a larger bias of the model. Models with small variance and high bias make strong assumptions about the data and tend to underfit the truth, while models with high variance and low bias tend to overfit the truth [100]. The trade-off between bias and variance in learning algorithms is usually controlled via techniques such as regularization or bagging and boosting [99].

In what follows we briefly review the Gaussian Process and Artificial Neural Network techniques.
3.3.1. Gaussian process kernel method

A Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution [101]. A Gaussian process is fully described by its mean and covariance functions

\[ \phi(z) \sim \text{gp} \left( m(z), K \right), \]

where \( m(z) = \mathbb{E} \left[ \phi(z) \right] \), and \( K \) is the covariance matrix with entries \( K_{ij} = \mathbb{E} \left[ (\phi(z^i) - m(z^i)) \left( \phi(z^j) - m(z^j) \right) \right] \) [101].

In this work we employ the commonly used squared-exponential-covariance Gaussian kernel with

\[ k : \mathbb{R}^r \times \mathbb{R}^r \rightarrow \mathbb{R}, \quad k(z^i, z^j) = \sigma_o^2 \exp \left( -\frac{\|z^i - z^j\|^2}{2h^2} \right) + \sigma_n^2 \delta_{i,j}, \]

and \( K_{ij} = k(z^i, z^j) \) [101], where \( z^i \) and \( z^j \) are the pairs of data points in training or test samples, \( \delta \) is the Kronecker delta symbol and \( \| \cdot \| \) is some appropriate norm. The model [10] has three hyper-parameters. The length-scale \( h \) governs the correlation among data points. The signal variance \( \sigma_o^2 \in \mathbb{R} \) and the noise variance \( \sigma_n^2 \in \mathbb{R} \) govern the precision of variance and noise, respectively.

Consider a set of training data points \( Z = [z^1, z^2, \ldots, z^n] \in \mathbb{R}^{r \times n} \) and the corresponding noisy observations \( y = [y^1, y^2, \ldots, y^n] \in \mathbb{R}^{1 \times n} \),

\[ y^i = \phi(z^i) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N} \left( 0, \sigma_n^2 \right), \quad i = 1, \ldots, n. \]

Consider also the set of test points \( Z^* = [z^{i1}, z^{i2}, \ldots, z^{im}] \in \mathbb{R}^{r \times m} \) and the predictions \( \hat{y} = [\hat{y}^1, \hat{y}^2, \ldots, \hat{y}^m] \in \mathbb{R}^{1 \times m} \),

\[ \hat{y}^i = \phi(z^{i*}), \quad i = 1, \ldots, m. \]

For a Gaussian prior the joint distribution of training outputs \( y \) and test outputs \( \hat{y} \) is

\[ \begin{bmatrix} y^T \\ \hat{y}^T \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} m(Z)^T \\ m(Z^*)^T \end{bmatrix}, \begin{bmatrix} K & K^* \\ K^* & K^{**} \end{bmatrix} \right), \]

where

\[ m(Z) = [m(z^1), m(z^2), \ldots, m(z^n)] \in \mathbb{R}^{1 \times n}, \quad m(Z^*) = [m(z^{i1}), m(z^{i2}), \ldots, m(z^{im})] \in \mathbb{R}^{1 \times m}, \]

\[ K^* = (K_{ij})_{i=1,\ldots,n; \ j=1,\ldots,m} = k(z^i, z^{j*}) \quad \text{and} \quad K^{**} = (K_{ij})_{i=1,\ldots,m; \ j=1,\ldots,m} = k(z^{i*}, z^{j*}). \]

The predictive distribution represents the posterior after observing the data [99] and is given by

\[ p \left( \hat{y} | Z, y, Z^* \right) \sim \mathcal{N} \left( K^{*T} K^{-1} y, K^{**} - K^{*T} K^{-1} K^* \right), \]

where superscript \( T \) denotes the transpose operation.

The prediction of Gaussian process will depend on the choice of the mean and covariance functions, and on their hyper parameters \( h, \sigma_o^2 \) and \( \sigma_n^2 \) which can be inferred from the data

\[ \theta^* = [h, \sigma_o^2, \sigma_n^2] = \arg \min_{\theta} L(\theta), \]
by minimizing the marginal negative log-likelihood function

\[
L(\theta) = -\log p(y|Z, \theta) = \frac{1}{2} \log \det(K) + \frac{1}{2} (y - m(Z))^T K^{-1} (y - m(Z)) + \frac{n}{2} \log (2\pi).
\]

3.3.2. Artificial neural networks

The study of artificial neural networks (ANNs) begins in the 1910s in order to imitate human brain’s biological structure. Pioneering work was carried out by Rosenblatt, who proposed a three-layered network structure, the perceptron [102]. ANNs detect the pattern of data by discovering the input–output relationships. Applications include the approximation of functions, regression analysis, time series prediction, pattern recognition, and speech synthesis and recognition [103, 104].

ANNs consist of neurons and connections between the neurons (weights). Neurons are organized in layers, where at least three layers of neurons (an input layer, a hidden layer, and an output layer) are required for construction of a neural network.

The input layer distributes input signals \( z = [z_1 \ z_2 \ \cdots \ z_r] \) to the first hidden layer. For a neural network with \( L \) hidden layers and \( m^\ell \) neurons in each hidden layer, let \( \hat{y}^\ell = [\hat{y}^\ell_1 \ \hat{y}^\ell_2 \ \cdots \ \hat{y}^\ell_{m^\ell}] \) be the vector of outputs from layer \( \ell \), \( b^\ell = [b^\ell_1 \ b^\ell_2 \ \cdots \ b^\ell_{m^\ell}] \) the biases at layer \( \ell \), and \( w^\ell_j = [w^\ell_{j,1} \ w^\ell_{j,2} \ \cdots \ w^\ell_{j,m^\ell}] \) the weights connecting the neuron \( j \) to the input of that layer (output of previous layer). The vectors \( \hat{y}^\ell \) and \( w^\ell_j \) share the same dimension which varies along the layers depending on the number of input features, neurons and outputs. Then the feed-forward operation is

\[
x_j^{\ell+1} = w_j^{\ell+1} x^{\ell+1} + b_j^{\ell+1}, \quad \hat{y}^0 = z, \quad j = 1, \ldots, m^\ell.
\]

All products of previous layer output with current layer neuron weights will be summed and the bias value of each neuron will be added to obtain the vector \( x^\ell = [x_1^\ell \ x_2^\ell \ \cdots \ x_{m^\ell}^\ell] \). Then the final output of each layer will be obtained by passing the vector \( x^\ell \) through the transfer function \( \varphi \), which is a differentiable function and can be log-sigmoid, hyperbolic tangent sigmoid, or linear transfer function.

The training process of ANN adjusts the weights and the biases in order to reproduce the desired outputs when fed the given inputs. The training process via the back propagation algorithm [105] uses a gradient descent method to modify weights and thresholds such that the error between the desired output and the output signal of the network is minimized [106]. In supervised learning the network is provided with samples from which it discovers the relations of inputs and outputs. The output of the network is compared with the desired output, and the error is back-propagated through the network and the weights will be adjusted. This process is repeated during several iterations, until the network output is close to the desired output [107].

4. MP-LROM error model applications

In this section, we address a series of applications related to the construction of parametric local reduced-order models using the MP-LROM error model introduced in Section [3]. In particular we focus on designing the decomposition of the parametric domain and constructing a hierarchy of the local bases, local reduced-order and high-fidelity models producing the most accurate solutions for an arbitrary parametric configuration. We formulate them in detail below.

4.1. Designing the decomposition of the parametric domain

Motivated by the need of fast and accurate simulations along the entire parametric space, we propose an alternative to the global parametric approach where only a single basis and a single reduced order model is constructed. Our alternative relies on a series of local reduced bases and reduced order models whose solutions meet prescribed admissible error thresholds beyond the parametric configurations employed for bases’ construction. A parametric region, where a local reduced order model constructed using a single
high-fidelity model trajectory is accurate to within a prescribed threshold, is called a feasible region. We delimitate such a region by employing the MP-LROM error model and sampling the neighborhood of the parametric configuration used to construct the local reduced order model. Our solution consists in designing a reunion of feasible regions completely covering the entire parametric space. This decomposition of the parametric domain was obtained as the solution of the following problem.

**Problem 1 (Accurate local parametric ROMs).** For an arbitrary parameter configuration $\mu \in P$ construct a reduced-order model that provides an accurate and efficient approximation of the high-fidelity solution

$$\|x(\mu, t_1) - U_{\mu_p} \tilde{x}_{\mu_p}(\mu, t_1) \cdot x(\mu, t_2) - U_{\mu_p} \tilde{x}_{\mu_p}(\mu, t_2) \cdot \cdots \cdot x(\mu, t_{N_t}) - U_{\mu_p} \tilde{x}_{\mu_p}(\mu, t_{N_t})\|_F \leq \varepsilon,$$

(15)

for some prescribed admissible error level $\varepsilon > 0$. The snapshots used to generate the basis $U_{\mu_p}$ and reduced operators can be obtained with any parametric configuration $\mu_p \in P$.

Our methodology proposes to select a finite countable subset $I = \{\mu_{p_j}, j = 1, \ldots, M\} \subset P$ and for each $\mu_{p_j}$, a reduced order basis $U_{\mu_{p_j}}$ along with the reduced operators are constructed for $j = 1, \ldots, M$. We denote by $U$ the set of bases $U_{\mu_{p_j}}$, $j = 1, \ldots, M$. If for each parameter configuration $\mu_{p_j}$, there exists an open ball $B(\mu_{p_j}, r_j) \in P$ such that, for all parameters $\mu \in B(\mu_{p_j}, r_j)$, the reduced order solution $\tilde{x}_{\mu_p}(\mu, t)$ satisfies (15) for $\mu_p = \mu_{p_j}$ and the parametric domain $P$ is a subset of the union of all these open balls, we obtain the sought decomposition of the parametric domain.

Next we derive a condition that guarantees the actual reduced-order model error

$$\varepsilon^{HF}_{\mu, \mu_{p_j}, K_{POD}} = \|x(\mu, t_1) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_1) \cdot x(\mu, t_2) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_2) \cdot \cdots \cdot x(\mu, t_{N_t}) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_{N_t})\|_F,$$

(16)

depending on parameter configuration $\mu$, parameter configuration $\mu_{p_j}$ and basis dimension $K_{POD}$, satisfies the prescribed admissible threshold $\varepsilon^{HF}_{\mu, \mu_{p_j}, K_{POD}} \leq \varepsilon$, for any arbitrary parameter configuration $\mu$ inside of an open ball.

**Theorem 4.1.** If $\lim_{\mu \to \mu_{p_j}} \|x(\mu, t_1) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_1) \cdot x(\mu, t_2) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_2) \cdot \cdots \cdot x(\mu, t_{N_t}) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_{N_t})\|_F = \varepsilon^*$, and $\varepsilon^* \leq \varepsilon$, then there exists $r_j > 0$ such that $\varepsilon^{HF}_{\mu, \mu_{p_j}, K_{POD}} \leq \varepsilon$ is satisfied for all parameters $\mu$ inside the ball $B(\mu_{p_j}, r_j)$.

**Proof.** From $\lim_{\mu \to \mu_{p_j}} \|x(\mu, t_1) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_1) \cdot x(\mu, t_2) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_2) \cdot \cdots \cdot x(\mu, t_{N_t}) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_{N_t})\|_F = \varepsilon^*$, using the limit definition, we have that for all $\varepsilon > 0$, there exists another real number $\delta > 0$ such that

$$\|x(\mu, t_1) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_1) \cdot x(\mu, t_2) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_2) \cdot \cdots \cdot x(\mu, t_{N_t}) - U_{\mu_{p_j}} \tilde{x}_{\mu_{p_j}}(\mu, t_{N_t})\|_F - \varepsilon^* \leq \varepsilon,$$

for all $\mu$ satisfying $d(\mu, \mu_p) < \delta$. By taking $\varepsilon = \varepsilon^*$ and $\delta = r_j$ we obtain that

$$\varepsilon^{HF}_{\mu, \mu_{p_j}, K_{POD}} < 2\varepsilon^* \leq \varepsilon, \forall \mu \in B(\mu_{p_j}, r_j),$$

which completes the proof. □

The theoretical result allows to compute the reduced-order model error at $\mu = \mu_p$ at certain parametric configurations in the proximity of $\mu_p$. Moreover, one may be able to make statements about the degree of smoothness of the solution in the parametric space, therefore placing a lower limit on $\varepsilon$. A particular case is when the reduced solution error is monotonically decreasing with smaller distances $d(\mu, \mu_{p_j})$. A small radius $r_j > 0$ then can be simply obtained by sampling and computing the residuals of the high-fidelity model using the projected reduced-order model solution.
One can also test for linear behavior of the high-fidelity solution \( x(\mu, t) \) in a small neighborhood of \( \mu_{p_j} \). Another possible screening test consists in checking the derivatives \( \frac{\partial x}{\partial \mu}(\mu, t) \) at equally distributed parameter values across the neighborhood of \( \mu_{p_j} \). If the derivatives are small, then the high-fidelity solutions do not vary much inside the open ball and they can be well approximated in the reduced manifold spanned by \( U_{\mu_{p_j}} \).

The decomposition construction process ends as soon as the entire parameter domain \( \mathcal{P} \) is covered with a union of overlapping balls \( B(\mu_{p_j}, r_j), \ j = 1, \ldots, M \), corresponding to different reduced order bases and local models

\[
\mathcal{P} \subset \bigcup_{j=1}^{M} B(\mu_{p_j}, r_j),
\]

such that for each \( j = 1, 2, \ldots, M \) and \( \forall \mu \in B(\mu_{p_j}, r_j) \cap \mathcal{P}, \) the error of the reduced-order model solution (2) satisfies \( \varepsilon_{\mu, \mu_{p_j}}^{HF}, K_{POD} \leq \bar{\varepsilon} \).

The number of balls \( M \) is finite only if the space of all high-fidelity solution over the entire parametric domain can be approximated with a finite number of low-dimensional linear subspaces. This extends the concept of a single global low-dimensional manifold [70, 71]. The cardinality of \( \mathcal{I} \) depends on the high-fidelity solution variability along the parameter space. In theory, less variability should lead to smaller values of \( M \). For our reduced-order models to satisfy the accuracy threshold for the entire parametric domain, we have to select thresholds \( \bar{\varepsilon} \) not smaller than the Kolmogorov \( K_{POD} \)-width [108] of the so-called solution manifold \( \{x(\mu, t), \ \mu \in \mathcal{P}, \ t \in \{t_i\}_{i=1}^{N_t}\} \).

Different open balls may be associated with different admissible thresholds since the Kolmogorov \( K_{POD} \)-width of sub-regions of the solution manifold can vary significantly.

This approach is inspired from the construction of unsteady local reduced-order models where the time domain is split in multiple regions [46, 49]. In this way the reduced basis dimension is kept small allowing for fast on-line simulations. The cardinality of \( \mathcal{I} \) is inversely proportional with the prescribed level of accuracy \( \bar{\varepsilon} \). As the desired error threshold \( \bar{\varepsilon} \) decreases, the decomposition changes since usually the radii \( r_j \) are expected to become smaller, and more balls are required to cover the parametric domain; i.e., \( M \) is increased.

The construction of the parametric domain decomposition (17) using the local parametric reduced-order models requires the following ingredients

1. The ability to probe the vicinity of \( \mu_{p_j} \in \mathcal{P} \) and to efficiently estimate the level of error \( \varepsilon_{\mu, \mu_{p_j}}, K_{POD} \) (16).
2. The ability to find \( r_j > 0 \) such that \( \varepsilon_{\mu, \mu_{p_j}}, K_{POD} \leq \bar{\varepsilon} \), for all \( \mu \in B(\mu_{p_j}, r_j) \cap \mathcal{P} \).
3. The ability to identify the location of a new \( \mu_{p_j}^{*} \) (for the construction of a new local reduced-order model) given the locations of the previous local parameters \( \mu_{p_j}, \ j = 1, \ldots, \ell - 1 \), so that

\[
B(\mu_{p_j}^{*}, r_\ell) \subseteq \bigcup_{i=1}^{\ell-1} B(\mu_{p_i}, r_i), \quad B(\mu_{p_j}^{*}, r_\ell) \cap \bigcap_{i=1}^{\ell-1} B(\mu_{p_i}, r_i) \neq \emptyset .
\]

The second condition in (18) assures that the decomposition will have no coverage gap; i.e., equation (17) is satisfied.

The implementation of the first ingredient does not employ a-posteriori error estimation formulas [75]. An approximated MP-LROM error model introduced in Section 4 is used to sample the vicinity of \( \mu_{p_j} \) and predicts the error for each sample parameter value \( \mu \). Based on these error predictions, we construct the ball \( B(\mu_{p_j}, r_j) \), or perhaps a larger set called a \( \mu_{p_j} \)-feasible region, where the local reduced-order model is accurate to within the prescribed threshold \( \bar{\varepsilon} \) according to the MP-LROM models. Since the approximated MP-LROM model has errors in its predictions, the precision is guaranteed only if the sum of the reduced-order model error and MP-LROM model error is smaller than \( \bar{\varepsilon} \). For a one dimensional parametric domain, a greedy algorithm to be described in Subsection 6.3.1.2 is applied to identify the location of a new parametric configuration \( \mu_{p_j}^{*} \) (for the construction of a new basis) depending on the locations of the previous \( \mu_{p_i} \), \( i = 1, \ldots, \ell - 1 \). We seek to impose (18), so the entire parametric domain \( \mathcal{P} \) satisfies (17) after the decomposition construction is finished. Again this is not necessarily guaranteed since we employ an approximated MP-LROM error model for this task.
For the parametric area situated at the intersection of different feasible regions we can assign a new reduced-order model based on the information required to construct the initial feasible regions. This could be achieved by interpolation or concatenation of the underlying reduced bases or interpolation of the available high-fidelity solutions, as described in detail in Section 5.

4.2. Constructing a hierarchy of accurate local bases, local reduced-order and high-fidelity models

Another application of the MP-LROM error model consists in constructing a hierarchy of local bases, local reduced-order and high-fidelity models. More precisely, we can employ the MP-LROM error model to solve the following problem.

**Problem 2** (Selection of best bases). *For a parameter configuration $\mu^*$ construct a hierarchy of the available local bases, local reduced-order and high-fidelity models based on their accuracy.*

The approximated MP-LROM error model can be used to estimate the error $\varepsilon^{HF}_{\mu^*, \mu_{pj}, K_{POD}}$ defined in (16), for all available bases $U_{\mu_{pj}}, j = 1, 2, \ldots, M$ and associated reduced models in the database. The predicted reduced-order models errors are sorted and the associated bases, reduced operators and high-fidelity trajectories are ranked accordingly. Numerical experiments are discussed in Section 6.3.2. The top ranked bases can be used to generate a new reduced basis and operators that may increase the accuracy of the reduced order solution for the parameter configuration $\mu^*$ using interpolation or concatenation.

5. Combining available information for accurate local ROMs at arbitrary parametric configurations

For the parametric domain situated at the intersection of different feasible regions introduced in Section 4.1, we may improve the accuracy of the reduced solution by assigning a new reduced-order model based on the already existing local bases or high-fidelity simulations. Moreover, if a hierarchy of local reduced bases, local reduced-order and high-fidelity models discussed in Section 4.2 is available for a parametric configuration $\mu^*$, we can employ the top ranked bases and models to generate a reduced-order model whose accuracy may be increased. This can be achieved by interpolation or concatenation of the underlying reduced bases or interpolation of the available high-fidelity solutions.

The POD method produces an orthogonal basis that approximately spans the state solution space of the model for a specific parameter configuration. Moving away from the initial parametric configuration may require the construction of new bases and reduced operators since the initial reduced-order model may not be accurate anymore. However, if states depend continuously on parameters, the POD basis constructed for one parameter configuration may approximately span the solution space at different parametric settings in a local vicinity.

Several methods to combine the available information to generate more accurate reduced-order models for arbitrary parameter configurations $\mu^*$ have been proposed in the literature. One is the interpolation of the available reduced-order bases $U_{\mu_{pj}}, j = 1, \ldots, M$. The parametric dependence of the bases has been modeled with various linear and nonlinear spatially-dependent interpolants.

Here we discuss different strategies that involve Lagrange interpolation of bases in the matrix space and in the tangent space of the Grassmann manifold. In addition we propose to concatenate the available reduced bases followed by an orthogonalization process, and to interpolate the solutions of the high-fidelity model as means to derive the reduced-order basis for a parameter configuration $\mu^*$.

5.1. Basis interpolation

**Lagrange interpolation of bases.** Assuming the reduced manifold $U : \mathcal{P} \to \mathbb{R}^{N_{\text{state}} \times K_{POD}}$ poses a continuous and linear dependency with respect to the parametric space, and if $M$ discrete bases $U_{\mu_{pj}} = U(\mu_{pj})$ have been already constructed for various parametric configurations $\mu_{pj}, j = 1, 2, \ldots, M$, then a basis corresponding to the new configuration $\mu^*$ can be obtained using Lagrange’s interpolation.
formula

\[
U_{\mu^*} = \sum_{j=1}^{M} U_{\mu_p_j} L_j(\mu^*), \quad L_j(\mu^*) = \prod_{i \neq j} \frac{\mu^* - \mu_{p_i}}{\mu_{p_j} - \mu_{p_i}}.
\]  \hspace{1cm} (19)

It is worth mentioning that the resulting interpolated basis vectors are not orthogonal. One drawback of this approach is the lack of linear variation in the angles between pairs of reduced subspaces \[58\] spanned by the reduced bases \(U_{\mu_p_j}\). Differential geometry results can be employed to alleviate these deficiencies.

Grassmann manifold. In the study proposed by Amsallem and Farhat \[59\], basis (matrix) interpolation was performed in the tangent space of the Grassmann manifold \(G\) at a carefully selected point \(S\) representing a subspace spanned by one of the available reduced bases. It has been shown that Grassmann manifold can be endowed with a differentiable structure \[109, 110\], i.e., at each point \(S\) of the manifold a tangent space exists. The mapping from the manifold to the tangent space is called the logarithmic mapping, while the backward projection is referred to as exponential mapping \[111\]. According to \[59\], a new subspace \(S_{\mu^*}\), and its subsequent basis \(U_{\mu^*}\), associated with a new parameter \(\mu^*\), can be obtained by interpolating the reduced subspaces projections into the tangent space of the Grassmann manifold and then projecting back using the exponential mapping. The reduced subspaces \(\{S_{\mu_p_i}\}_{i=1}^{M}\) are spanned by the already computed reduced bases \(U_{\mu_p_i}, i = 1, \ldots, M\). The steps required by this methodology \[59\] are described in the Algorithm 2. A graphical description of the method is provided in Figure 1.

Algorithm 2 Interpolation in a tangent space to a Grassmann manifold \[59\]

1: Select a point \(S_{\mu_p_M}\) of the manifold to represent the origin point for the interpolation spanned by the basis \(U_{\mu_p_M}\).
2: The tangent space \(T_{S_{\mu_p_M}}\) and the subspaces \(\{S_{\mu_p_i}\}_{i=1}^{m}\) are considered, with \(m \leq M - 1\). Each point \(S_{\mu_p_i}\) sufficiently close to \(S_{\mu_p_M}\) is mapped to a point of \(T_{S_{\mu_p_M}}\), using the logarithm map \(\log_{S_{\mu_p_M}}\) \[111\]. The bases spanning the tangent space points \(\log_{S_{\mu_p_M}}(S_{\mu_p_i})\) are computed by

\[
(I - U_{\mu_p_M} U_{\mu_p_M}^T)U_{\mu_p_i} (U_{\mu_p_M} U_{\mu_p_i})^{-1} = R_i \Lambda_i Q_i^T \quad \text{(SVD factorization)},
\]

\[
\Gamma_{\mu_p_i} = R_i \tan^{-1}(\Lambda_i) Q_i^T.
\]

3: Each entry of the matrix \(\Gamma_{\mu^*}\) associated with the target parameter \(\mu^*\) is computed by interpolating the corresponding entries of the matrices \(\Gamma_{\mu_p_i} \in \mathbb{R}^{N_{state} \times K_{POD}}\) associated with the parameter points \(\mu_p_i, i = 1, \ldots, m\). A univariate or multivariate Lagrange interpolation may be chosen similar with the one introduced in \[19\].
4: The matrix \(\Gamma_{\mu^*}\) representing a point in the tangent space \(T_{S_{\mu_p_M}}\) is mapped to a subspace \(S_{\mu^*}\) on the Grassmann manifold spanned by a matrix \(U_{\mu^*}\) using the exponential map \[111\]

\[
\Gamma_{\mu^*} = R^* \Lambda^* Q^{T*} \quad \text{(SVD factorization)},
\]

\[
U_{\mu^*} = U_{\mu_p_M} Q^* \cos(\Lambda^*) + R^* \sin(\Lambda^*).
\]

According to \[59\], the subspace angle interpolation \[58, 112\] is identical to the interpolation in a tangent space to the Grassmann manifold of two reduced-order bases. Consequently the latter methodology can be viewed as a generalization of the former approach.
5.2. Basis concatenation

Basis concatenation idea was introduced in [60] and emerged from the notion of a global basis [70, 71]. In the global strategy, the existing high-fidelity snapshots corresponding to various parameter configurations are collected in a single snapshot matrix and then a matrix factorization is performed to extract the most energetic singular vectors. This global basis is then used to build reduced-order models for parameter values not included in the initial snapshots set.

Assuming \( X_{\mu_{p1}}, X_{\mu_{p2}}, \ldots, X_{\mu_{pM}} \in \mathbb{R}^{N_{\text{state}} \times N_t} \) are the snapshots corresponding to \( M \) high-fidelity model trajectories, the following error estimate holds [55, Proposition 2.16]:

\[
\bar{X} = [X_{\mu_{p1}} \cdots X_{\mu_{pM}}] = \tilde{U} \tilde{A} \tilde{V}^T \text{ (SVD factorization)},
\]

\[
\| \bar{X} - \bar{U}^d \bar{X} \|_F^2 = \sum_{i=K_{POD}+1}^{N_t} \lambda_i^2 = O(\lambda_{K_{POD}+1}^2), \quad \bar{U}_{ij}^d = \bar{U}_{ij}, \quad i = 1, 2, \ldots, N_{\text{state}}, \quad j = 1, 2, \ldots, K_{POD},
\]

where \( \lambda_i \) is the \( i \)-th singular value of \( \bar{X} \), and \( \bar{X} = [\bar{U}^d]^T \bar{X} \in \mathbb{R}^{K_{POD} \times (MN_t)} \). In practice, usually \( N_t < N_{\text{state}} \), so the snapshot matrices are rank deficient. In this case and when the reduced-order bases \( U_{\mu_{p1}}, \ldots, U_{\mu_{pM}} \), corresponding to the trajectories \( \mu_{p1}, \ldots, \mu_{pM} \), are available, we can construct a basis \( \tilde{U} \) by simply concatenating columns of \( U_{\mu_{p1}}, \ldots, U_{\mu_{pM}} \) such that the accuracy level in (20) is preserved.

**Proposition 5.1.** Consider the following SVD of snapshots matrices \( X_{\mu_{p1}}, \ldots, X_{\mu_{pM}} \in \mathbb{R}^{N_{\text{state}} \times N_t} \):

\[
X_{\mu_{pj}} = U_{\mu_{pj}} \Lambda_j V_{\mu_{pj}}^T, \quad j = 1, 2, \ldots, M,
\]

with \( N_t < N_{\text{state}} \) and \( \text{rank}(X_{\mu_{pj}}) < N_t, \ i = 1, 2, \ldots, M \). Then, there exist positive integers \( K_{POD}^1, \ldots, K_{POD}^M \), and \( \bar{X} \in \mathbb{R}^{(\sum_{i=1}^{M} K_{POD}^i) \times (MN_t)} \), such that \( \bar{X} \) defined in (20) satisfies

\[
\| \bar{X} - \bar{U} \bar{X} \|_F^2 \leq O(\lambda_{K_{POD}+1}^2),
\]

where \( \lambda_{K_{POD}+1} \) is the \( (K_{POD}+1) \)-th singular value of snapshots matrix \( \bar{X} \), and \( \bar{U} = [U_{\mu_{p1}}^d \cdots U_{\mu_{pM}}^d] \in \mathbb{R}^{N_{\text{state}} \times (\sum_{i=1}^{M} K_{POD}^i)}, \quad [U_{\mu_{pi}}^d]_{ij} = U_{\mu_{pi}}[i,j], \quad i = 1, 2, \ldots, N_{\text{state}}, \quad j = 1, 2, \ldots, K_{POD}^i, \ l = 1, 2, \ldots, M. \)

**Proof.** Since \( X_{\mu_{p1}}, \ldots, X_{\mu_{pM}} \in \mathbb{R}^{N_{\text{state}} \times N_t} \) are rank deficient matrices, there exist at least \( M \) positive integers \( K_{POD}^1, \ldots, K_{POD}^M. \)
such that the singular values associated with $X_{\mu_{p_1}}, \ldots, X_{\mu_{p_M}}$ satisfy

$$[\lambda_{K_{PFD}+1}^1]^2, \ldots, [\lambda_{K_{PFD}+1}^M]^2 \leq [\lambda_{K_{PFD}+1}^1]^2, \forall K_{PFD} = 0, \ldots, N_t-1. \quad (23)$$

From [55, Proposition 2.16] and (21) we have the following estimates

$$\|X_{\mu_{p_j}} - U_{\mu_{p_j}}^d X_{\mu_{p_j}}\|_F^2 = \sum_{i=K_{PFD}+1}^{N_t} \|\lambda_i\|^2 = O([\lambda_{K_{PFD}+1}^I]^2) \leq O([\lambda_{K_{PFD}+1}^I]^2), \quad (24)$$

where $\lambda_i^j$ is the $i^{th}$ singular value of $X_{\mu_{p_j}}$ and $\hat{X}_{\mu_{p_j}} = [U_{\mu_{p_j}}^d X_{\mu_{p_j}}] \in \mathbb{R}^{K_{PFD} \times N_t}$, for $j = 1, \ldots, M$.

By denoting

$$\hat{X} = \begin{bmatrix} X_{\mu_{p_1}} & 0 & \cdots & 0 \\ 0 & X_{\mu_{p_2}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & X_{\mu_{p_M}} \end{bmatrix},$$

where the null matrix $0_j$ belongs to $\mathbb{R}^{K_{PFD} \times N_t}$, $j = 1, \ldots, M$, we have

$$\|\hat{X} - \hat{U} \hat{X}\|_F^2 = \|X_{\mu_{p_1}} \cdots X_{\mu_{p_M}} - [U_{\mu_{p_1}}^d X_{\mu_{p_1}} \cdots U_{\mu_{p_M}}^d X_{\mu_{p_M}}]\|_F^2 \leq$$

$$\left( \sum_{i=1}^{M} \|X_{\mu_{p_i}} - U_{\mu_{p_i}}^d X_{\mu_{p_i}}\|_F^2 \right)^2 = \sum_{i=1}^{M} \sum_{j=1}^{M} \|X_{\mu_{p_i}} - U_{\mu_{p_i}}^d X_{\mu_{p_i}}\|_F^2 \leq O([\lambda_{K_{PFD}+1}^I]^2).$$

By assuming that $\text{rank}(X_{\mu_{p_i}}) < N_t$, for all $j = 1, \ldots, M$, imposes $N_t$ as an upper limit for all the positive integers $K_{PFD}$, $j = 1, \ldots, M$. This assumption is not unrealistic since typically the snapshots matrix stores correlated data and therefore contains linearly dependent columns. For linearly independent matrices, the precision is controlled by the spectra of snapshots matrices $X_{\mu_{p_i}}, i = 1, \ldots, M$, but there is no guarantee that the expansion based on the concatenated basis $\hat{U}$ can provide similar accuracy precision as the expansion based on $U$ [20] for all $K_{PFD} = 1, 2, \ldots, N_t$.

In practice usually $\sum_{i=1}^{M} K_{PFD}$ is larger than $K_{PFD}$, thus more bases functions are required to form $\hat{U}$ to achieve a similar level of precision as in [20] where $\hat{U}$ is built using a global singular value decomposition. According to [113], the faster approach to compute the left singular vectors and singular values only is to apply a QR factorization followed by a R-bidiagonalization [114]. The R-SVD decomposition of a matrix of dimension $N_{\text{state}} \times N_t$ has a computational complexity of $O(4N_{\text{state}}^2 N_t + 13N_t^3)$. As such, the decomposition of matrix $\hat{X}$ requires $O(4MN_{\text{state}}^2 N_t + 13MN_t^3)$ operations, whereas all combined singular value decompositions of matrices $X_{\mu_{p_i}}, i = 1, \ldots, M$, have a computational complexity of $O(4MN_{\text{state}}^2 N_t + 13MN_t^3)$. This estimation suggests that the matrix factorization of $\hat{X}$ is more computationally demanding. However, the first term $4MN_{\text{state}}^2 N_t$ often dominates as $N_t \ll N_{\text{state}}$ in practice, leading to almost similar computational times. The off-line stage of the concatenation method may also include the application of a Gram-Schmidt-type algorithm to orthogonalize the overall set of vectors in $\hat{U}$.

While the Lagrange interpolation of bases mixes the different energetic singular vectors in an order dictated by the singular values magnitude, this strategy concatenates the dominant singular vectors for each case and preserves their structure.
5.3. Interpolation of high-fidelity model solutions

The method discussed herein assumes that the model solution depends continuously on the parameters. Thus it is natural to consider constructing the basis for a parameter configuration $\mu^*$ by interpolating the existing high-fidelity model solutions associated with various parameter settings, and then performing the SVD of the interpolated results. For example, the Lagrange solution interpolant is given by

$$X_{\mu^*} = \sum_{j=1}^{M} X_{\mu_{p_j}} L_j(\mu^*),$$

(25)

where $X_{\mu_{p_j}} \in \mathbb{R}^{N_{\text{state}} \times N_t}$ is the model solution corresponding to parameter $\mu_{p_j}$ and the interpolation polynomials are defined in (19).

A new basis is constructed from the interpolated model solution (25) for the new parametric configuration $\mu^*$. From computational point of view the complexity of the off-line stage of the solution interpolation method (25) is smaller than in the case of the bases concatenation and interpolation approaches. Only one singular value decomposition is required in contrast with the multiple factorizations needed in the latter two strategies where the involved matrices have the same size $N_{\text{state}} \times N_t$. Having only $N_t$ snapshots the size of the outcome basis should be smaller than in the case of basis concatenation approach.

6. Numerical experiments

We illustrate the application of the proposed MP-LROM models to predict the error and dimension of the local reduced-order models for a one-dimensional Burgers model and their subsequent applications. The 1D-Burgers model proposed herein is characterized by two scalar parameters, but for the majority of the experiments we considered only variation in the viscosity coefficient space only. To assess the performance of the MP-LROM models constructed using Gaussian Process and Artificial Neural Network, we employ various cross-validation tests. The dimensions of the training and testing data sets are chosen empirically based on the number of samples. For Artificial Neural Network models the number of hidden layers and neurons in each hidden layer vary for each type of problems under study. The squared-exponential-covariance kernel (10) is used for Gaussian Process models.

The approximated MP-LROM error models are compared against the ROMES and multi-fidelity correction models, whereas the MP-LROM models that predict the dimension of the reduced-order models are verified against the standard approach based on the spectrum of snapshots matrix. For each of the applications introduced in Section 4 (constructing the decomposition of the parametric domain and a hierarchy of accurate local bases, local reduced-order and high-fidelity models), we present in detail the proposed solution approaches and the corresponding numerical results. We also combine the available bases and the high-fidelity model trajectories of the constructed hierarchy in an attempt to enhance the accuracy of the reduced-order model for a parametric configuration $\mu^*$. Only for this experiment we allow the advection term coefficient to vary.

6.1. One-dimensional Burgers’ equation

Burgers’ equation is an important partial differential equation from fluid mechanics [115]. The evolution of the velocity $u$ of a fluid evolves according to

$$\frac{\partial u}{\partial t} + \nu u \frac{\partial u}{\partial x} = \mu \frac{\partial^2 u}{\partial x^2}, \quad x \in [0, L], \quad t \in (0, t_f],$$

(26)

with $t_f = 1$ and $L = 1$. Here $\mu$ is the viscosity coefficient. The parameter $\nu$ has no physical significance and it is used to control the non-linear effect of the advection term. For the majority of the experiments $\nu$ is set to 1, except for those in Section 6.4 where $\nu$ is allowed to vary between $5 \cdot 10^{-4}$ and $5 \cdot 10^1$. Consequently, we test the bases concatenation and interpolation methods as well as the high-fidelity solutions interpolation technique for linear and non-linear regimes of Burgers model.
The model has homogeneous Dirichlet boundary conditions \( u(0, t) = u(L, t) = 0, \ t \in (0, t_f] \). For the initial conditions, we used a seventh order polynomial constructed using the least-square method and the data set \{(0, 0); (0.2, 1); (0.4, 0.5); (0.6, 1); (0.8, 0.2); (0.9, 0.1); (0.95, 0.05); (1, 0)\}. We employed the polyfit function in Matlab and the polynomial is shown in Figure 2.

The discretization uses a spatial mesh of \( N_s \) equidistant points on \([0, L]\), with \( \Delta x = L/(N_s - 1) \). A uniform temporal mesh with \( N_t \) points covers the interval \([0, t_f]\), with \( \Delta t = t_f/(N_t - 1) \). The discrete velocity vector is \( u(t_j) \approx [u(x_i, t_j)]_{i=1, 2, \ldots, N_{\text{state}}}, j = 1, 2, \ldots, N_t \), where \( N_{\text{state}} = N_s - 2 \) (the known boundaries are removed). The semi-discrete version of the model (26) is

\[
\dot{u} = -\nu \odot A_x u + \mu A_{xx} u, \tag{27}
\]

where \( \dot{u} \) is the time derivative of \( u \), and \( A_x, A_{xx} \in \mathbb{R}^{N_{\text{state}} \times N_{\text{state}}} \) are the central difference first-order and second-order space derivative operators, respectively, which take into account the boundary conditions, too. The model is implemented in Matlab and the backward Euler method is employed for time discretization. The nonlinear algebraic systems are solved using the Newton-Raphson method and the allowed number of Newton iterations per each time step is set to 50. The solution is considered to have converged when the Euclidean norm of the residual is less than \( 10^{-10} \).

The viscosity parameter space \( \mathcal{P} \) is set to the interval \([0.01, 1]\). Smaller values of \( \mu \) correspond to sharper gradients in the solution, and lead to dynamics more difficult to accurately approximate using reduced-order models.

![Figure 2: Seventh order polynomial used as initial conditions for 1D Burgers model.](image)

The reduced-order models are constructed using POD method whereas the quadratic nonlinearities are computed via tensorial POD [68] for efficiency. A floating point operations analysis of tensorial POD, POD and POD/DEIM for \( p \)th order polynomial nonlinearities is available in [68]. The computational efficiency of the tensorial POD 1D Burgers model can be noticed in Figure 3. Both on-line and off-line computational costs are shown. Here we selected \( \mu = \mu_p = 0.7, \nu = 1, N_t = 301, \) POD dimension \( K_{POD} = 9 \), and we let the number of space points \( N_s \) to vary. For \( N_s = 201 \) and 701, the tensorial POD model is 5.17 \( \times \) and 61.12 \( \times \) times faster than the high-fidelity version. The rest of our numerical experiments uses \( N_s = 201 \) and \( N_t = 301 \).

### 6.2. Multivariate prediction of local reduced-order models characteristics (MP-LROM) using regression machine learning methods

#### 6.2.1. Error estimation of local ROM solutions

Here, we will use GP and ANN to approximate the MP-LROM error model introduced in [5]. The approximated models have the following form

\[
\phi_{MP-LROM}^e : \{\mu, \mu_p, K_{POD}\} \rightarrow \hat{\log}^e_{HF, \mu, \mu_p, K_{POD}}, \tag{28}
\]
where the input features include a viscosity parameter value $\mu$, a parameter value $\mu_p$ associated with the full model run that generated the basis $U_{\mu_p}$, and the dimension of the reduced manifold $K_{POD}$. The target is the estimated logarithm of error of the reduced-order model solution at $\mu$ using the basis $U_{\mu_p}$ and the corresponding reduced operators computed using the Frobenius norm

$$\log \varepsilon_{HF, K_{POD}} = \log \left( \| x(\mu, t_1) - U_{\mu_p} \tilde{x}_{\mu_p}(\mu, t_1) \| + \| x(\mu, t_2) - U_{\mu_p} \tilde{x}_{\mu_p}(\mu, t_2) \| + \cdots + \| x(\mu, t_N) - U_{\mu_p} \tilde{x}_{\mu_p}(\mu, t_N) \|_F \right) .$$

The probabilistic models described generically in equation (28) are just approximations of the MP-LROM model (5) and have errors. The data set includes 10 and 100 equally distributed values of $\mu_p$ and $\mu$ over the entire parameter region; i.e., $\mu_p \in \{0.1, 0.2, \ldots, 1\}$ and $\mu \in \{0.01, \ldots, 1\}$, 12 reduced basis dimensions $K_{POD}$ spanning the interval $\{4, 5, \ldots, 14, 15\}$ and the reduced-order model logarithm of errors $\log \varepsilon_{HF, K_{POD}}$.

The entire data set contains 12000 samples, and for each 12 samples a high-fidelity model solution is calculated. Only one high-fidelity model simulation is required for computing the reduced solutions errors for the parametric configuration $\mu$ using reduced-order models of various $K_{POD}$ and constructed based on a single high-fidelity trajectory at parameter $\mu_p$. As such, 1000 high-fidelity simulations were needed to construct the entire data set. Whereas we do not construct new bases and reduced-order models for parameters $\mu$, we chose to compute high-fidelity simulations to accurately calculate the errors associated with the existing reduced-order models for these parametric configurations.

Figure 4 shows isocontours of the error $\varepsilon_{HF, K_{POD}}$ and log $\varepsilon_{HF, K_{POD}}$ of the reduced-order model solution for various viscosity parameter values $\mu$ and POD basis dimensions. The design of the reduced-order models relies on the high-fidelity trajectory for $\mu_p = 0.8$. The target values $\varepsilon_{HF, K_{POD}}$ vary over a wide range (from 300 to $10^{-6}$) motivating the choice of implementing models that target $\log \varepsilon_{HF, K_{POD}}$ to decrease the variance of the predicted results.

A more detailed analysis, comparing models

$$\phi^{\varepsilon_{MP-LROM}} : \{\mu, \mu_p, K_{POD}\} \mapsto \varepsilon_{HF, K_{POD}}$$

that target scaled data (28) and no scaled data, is given in the following.

The approximated MP-LROM models for estimating the local parametric reduced-order model errors are constructed using a Gaussian Process with a squared-exponential covariance kernel [10] and a neural network with six hidden layers and hyperbolic tangent sigmoid activation function in each layer. Tables 1 and 2 show the averages and variances of errors in prediction of MP-LROM models.
for different sample sizes. Every subset of samples is selected randomly from a shuffled original data set. The misfit is computed using the same formulas presented in (8a) to evaluate the prediction errors. Table 1 shows the prediction errors of (30) computed via equation (8a) with $y = \varepsilon^{HF}_{\mu, \mu, p, KPOD}$ and $\hat{y} = \hat{\varepsilon}^{HF}_{\mu, \mu, p, KPOD}$; i.e., no data scaling; the predictions have a large variance and a low accuracy. Scaling the data and targeting $\log \varepsilon^{HF}_{\mu, \mu, p, KPOD}$ results using (28), reduce the variance of the predictions, and increase the accuracy, as shown in Table 2. The same formula (8a) with $y = \log \varepsilon^{HF}_{\mu, \mu, p, KPOD}$ and $\hat{y} = \log \hat{\varepsilon}^{HF}_{\mu, \mu, p, KPOD}$ was applied. We notice that, for increasing small sample sizes and for scaled data, the variances of GP and ANN predictions are not necessarily decreasing. This behavior changes and the variances of both regression models decrease for increasing sample sizes larger than 700 as seen in Table 2. The performance of the ANN and GP is highly dependent on the number of samples in the data set. As the number of data points grows, the accuracy increases and the variance decreases. The results show that GP outperforms ANN for small numbers of samples $\leq 1000$ whereas, for larger data sets, ANN is more accurate than GP.

<table>
<thead>
<tr>
<th>Sample size</th>
<th>GP MP-LROM</th>
<th>ANN MP-LROM</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>13.4519</td>
<td>12.5189</td>
</tr>
<tr>
<td>400</td>
<td>6.8003</td>
<td>6.9210</td>
</tr>
<tr>
<td>700</td>
<td>5.6273</td>
<td>7.2325</td>
</tr>
<tr>
<td>1000</td>
<td>3.7148</td>
<td>5.6067</td>
</tr>
<tr>
<td>3000</td>
<td>0.5468</td>
<td>1.2858</td>
</tr>
<tr>
<td>5000</td>
<td>0.0565</td>
<td>3.8819</td>
</tr>
</tbody>
</table>

Table 1: Average and variance of error in predictions of MP-LROM models (30) constructed via ANN and GP using errors $\varepsilon^{HF}_{\mu, \mu, p, KPOD}$ in training data for different sample sizes.
Table 2: Average and variance of error in predictions of MP-LROM models constructed via ANN and GP using logarithms of errors \( \log \varepsilon_{HF,\mu,\mu_p,K_{POD}} \) in training data for different sample sizes.

Scaling the data and targeting \( \log \varepsilon_{HF,\mu,\mu_p,K_{POD}} \) errors clearly improve the performance of the MP-LROM models. Consequently for the rest of the manuscript we will only use model (28). To assess the quality of the MP-LROM models a five-fold cross-validation process...
Figure 6: Histogram of errors in prediction using GP MP-LROM.

is also used. The results computed using formula (8b) are shown in Table 3. ANN outperforms the GP and estimates the errors more accurately. It also has less variance than the Gaussian Process which indicates it has more stable predictions.

```
<table>
<thead>
<tr>
<th></th>
<th>E</th>
<th>VAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN</td>
<td>0.004004</td>
<td>2.16 × 10⁻⁶</td>
</tr>
<tr>
<td>GP</td>
<td>0.092352</td>
<td>1.32 × 10⁻⁵</td>
</tr>
</tbody>
</table>
```

Table 3: MP-LROM statistical results over five-fold cross-validation.

Figure 7 illustrates the average of errors in prediction of five different errors models computed using ANN and GP regression methods. The error models were constructed using a training set formed by 80% randomly selected data of the entire data set. The predictions were made using a fixed test set randomly selected from the entire data set and contains various values of $\mu$, $K_{POD}$ and $\mu_p$ shown in the x-axes of Figure 7. Building different GP and ANN MP-LROM error models, each trained on different part of the data set and then testing them with the same fixed test set, reduces the bias in prediction.

We also compared the MP-LROM models with those obtained by implementing ROMES method [52] and multi-fidelity correction.
technique [51]. The ROMES method constructs univariate models

$$\phi_{ROMES} : \log \rho(\mu) \mapsto \log \varepsilon_{HF}^{\mu},$$  \hspace{1cm} (31)

where the input $\rho(\mu)$ consists of error indicators. Examples of indicators include residual norms, dual-weighted residuals and other error bounds. The multi-fidelity correction method (MFC) implements input-output models

$$\phi_{MFC} : \mu \mapsto \log \varepsilon_{HF}^{\mu},$$  \hspace{1cm} (32)

where the input of error models is the viscosity parameter $\mu$. Both ROMES and multi-fidelity correction methods use a global reduced-order model with a fixed dimension in contrast to our method that employ local reduced-order models with various dimensions. ROMES and multi-fidelity correction models are univariate whereas the MP-LROM models are multivariate.

To accommodate our data set to the requirements of the ROMES and multi-fidelity correction methods, we separated the data set into multiple subsets. Each of these subsets has 100 samples corresponding to a single $\mu_p$ and $K_{POD}$ and 100 values of parameter $\mu \in \{0.01, 0.02, \ldots, 1\}$, so 100 high-fidelity simulations are required. For each subset we constructed ANN and GP models to approximate the input-output models defined in (31) and (32) using the same training set. In the case of ROMES method we employed the logarithms of residuals norms as inputs, so we first computed the corresponding reduced-order solution and then the associated logarithm of residual norm by using the projected reduced order solution into the high-fidelity model for parameter $\mu$. The output of the ROMES and multi-fidelity correction models approximate the logarithm of the Frobenius norm of the reduced-order-model errors.

Figures 8-11 shows the isocontours of the $E_{fold}$ and $VAR_{fold}$ computed using (3a) for different $K_{POD}$ and $\mu_p$ using ROMES, multi-fidelity correction, and MP-LROM models constructed using GP and ANN methods. In total there are $12 \times 10$ configurations corresponding to different $K_{POD}$ and $\mu_p$ and as many ROMES and multi-fidelity models. The MP-LROM models are global in nature and the training set is the whole original data set. The testing set is the same for all the compared models and differs from the training sets. We can see that MP-LROM models are more accurate than those obtained via ROMES and multi-fidelity correction models, however in our method we employed more samples due to the method nature. We also trained and tested all the models using five-fold cross-validation. The average error and variance of all 120 $E_{fold}$s and $VAR_{fold}$s are compared against those obtained using MP-LROM.
error models and are summarized in tables 4 and 5. This shows that the MFC models outperform the ROMES ones, for our experiment, and the MP-LROM models are the most accurate. The MP-LROM models perform better since they employ more features and samples than the other models which help the error models tune the parameters better. We also notice the efficiency of the MFC models from accuracy point of view considering that they use very few samples. In the case of large parametric domains the MP-LROM error models may require a very large data set with a lot of features. By using only subsets of the whole data set near the vicinity of the parameters of interest and applying the active subset method [92] can help prevent the potential curse of dimensionality problem that MP-LROM might suffer.

![Isocontours for the E\text{fold} using GP method.](image)

![Isocontours for the E\text{fold} using ANN method.](image)

<table>
<thead>
<tr>
<th></th>
<th>ROMES</th>
<th>MFC</th>
<th>MP-LROM</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN</td>
<td>0.3844</td>
<td>0.0605</td>
<td>8.8468 \times 10^{-4}</td>
</tr>
<tr>
<td>GP</td>
<td>0.2289</td>
<td>0.0865</td>
<td>0.0362</td>
</tr>
</tbody>
</table>

Table 4: Average error of all 120 E\text{fold}s for three methods.

<table>
<thead>
<tr>
<th></th>
<th>ROMES</th>
<th>MFC</th>
<th>MP-LROM</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN</td>
<td>0.0541</td>
<td>0.0213</td>
<td>4.9808 \times 10^{-4}</td>
</tr>
<tr>
<td>GP</td>
<td>0.0051</td>
<td>0.0049</td>
<td>5.4818 \times 10^{-4}</td>
</tr>
</tbody>
</table>

Table 5: Average variance of all 120 VAR\text{fold}s for three methods.
Finally we compare and show the error in prediction of ROMES, MFC, and MP-LROM models for one of the subsets corresponding to $K_{POD} = 10$ and $\mu_p = 1$. The testing set is randomly selected from the samples and is not included in the training sets. The training set for both ROMES and multi-fidelity correction models are the same. In order to prevent the bias in prediction, each time the error models are trained on randomly selected $80\%$ of the training sets and tested with the fixed test set. We repeated this five times and the average of error in prediction is obtained. Figure 12 shows the average of error in prediction for all models implemented using GP and ANN methods.

### 6.2.2. Selecting the dimension of reduced-order model

Here we construct MP-LROM models to predict the reduced basis dimension that accounts for specified accuracy levels in the reduced-order model solution. The models are constructed using GP and ANN methods and have the following form

$$
\phi_{MP-LROM}^{d} : \{\mu_p, \log \varepsilon_{HF}^{\mu_p, \mu_p, K_{POD}}\} \rightarrow \hat{K}_{POD}.
$$

The input features of this model consist of the viscosity parameter $\mu_p \in [0.01, 1]$ and the log of the Frobenius norm of the error between the high-fidelity and reduced-order models; i.e. $\log \varepsilon_{HF}^{\mu_p, \mu_p, K_{POD}}$ [29]. The searched output $\hat{K}_{POD}$ is the estimation of the dimension of the reduced manifold $K_{POD}$. The data set contains equally distributed values of $\mu_p$ over the entire parametric domain $\mu_p \in \{0.01, 0.0113, 0.0126, \ldots, 0.9956\}$, reduced basis dimensions $K_{POD}$ spanning the set $\{4, 5, \ldots, 14, 15\}$ and the logarithm of the reduced-order model error $\log \varepsilon_{HF}^{\mu_p, \mu_p, K_{POD}}$. We use GP and ANN methods to construct two MP-LROM models to predict the dimension...
of local reduced-order models given a prescribed accuracy level.

During the training phase, the MP-LROM models will learn the dimensions of reduced-order basis $K_{POD}$ associated with the parameter $\mu_p$ and the corresponding error $\log \epsilon_{HF,\mu_p,K_{POD}}^{HF}$. Later they will be able to estimate the proper dimension of reduced basis by providing it the specific viscosity parameter $\mu_p$ and the desired precision $\log \bar{e}$. The computational cost is low once the models are constructed. The output indicates the dimension of the reduced manifold for which the ROM solution satisfies the corresponding error threshold. Thus we do not need to compute the entire spectrum of the snapshots matrix in advance which for large spatial discretization meshes translates into important computational costs reduction. Figure 13 illustrates the contours of the log of reduced-order model error over all the values of the viscosity parameter $\mu_p \in \{0.01, 0.0113, 0.0126 \ldots 1\}$ and various POD dimensions $K_{POD} = \{4, 5, \ldots, 14, 15\}$.

A neural network with 5 hidden layers and hyperbolic tangent sigmoid activation function in each layer is used while for the Gaussian process we have used the squared-exponential-covariance kernel (10). For both MP-LROM models, the results were rounded such as to generate natural numbers. Table 6 shows the average and variance of error in GP and ANN predictions using different sample sizes. The ANN outperforms the GP and as the number of data points grows, the accuracy increases and the variance decreases. The results are obtained using a conventional validation with 80% of the sample size dedicated for training data and the other 20% for the test data. The employed formula is described in equation (8a).

<table>
<thead>
<tr>
<th>Sample Size</th>
<th>MP-LROM GP</th>
<th>MP-LROM ANN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$E_{fold}$</td>
<td>$VAR_{fold}$</td>
</tr>
<tr>
<td>100</td>
<td>0.2801</td>
<td>0.0901</td>
</tr>
<tr>
<td>1000</td>
<td>0.1489</td>
<td>0.0408</td>
</tr>
<tr>
<td>3000</td>
<td>0.1013</td>
<td>0.0194</td>
</tr>
<tr>
<td>5000</td>
<td>0.0884</td>
<td>0.0174</td>
</tr>
</tbody>
</table>

Table 6: Average and variance of errors in prediction of reduced basis dimension using MP-LROM models for different sample sizes.

Figures 14 and 15 show the prediction errors using 100 and 1000 training samples for both MP-LROM models constructed via ANN and GP models. The histograms shown in Figure 15, as stated before, can assess the validity of GP assumptions. Once the number of samples is increased, the data set distribution shape is closer to a Gaussian profile than in the case of the data set distribution shown in Figure 6 used for generation of MP-LROM models for the prediction of local reduced-order model errors. To assess the accuracy of the MP-LROM models, the data set is randomly partitioned into five equal size sub-samples, and five-fold
cross-validation test is implemented. The five results from the folds are averaged and they are presented in Table 7. The ANN model correctly estimated the dimension of the reduced manifold in 87% cases. GP correctly estimates the POD dimension 53% of the times. The variance results shows that the GP model has more stable predictions indicating a higher bias in the data.

<table>
<thead>
<tr>
<th>Dimension discrepancies</th>
<th>zero</th>
<th>one</th>
<th>two</th>
<th>three</th>
<th>four &gt; four</th>
<th>VAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN MP-LROM</td>
<td>87%</td>
<td>11%</td>
<td>2%</td>
<td>0</td>
<td>0</td>
<td>2.779 × 10⁻³</td>
</tr>
<tr>
<td>GP MP-LROM</td>
<td>53%</td>
<td>23%</td>
<td>15%</td>
<td>5%</td>
<td>3%</td>
<td>4.575 × 10⁻⁴</td>
</tr>
</tbody>
</table>

Table 7: POD basis dimension discrepancies between the MP-LROM predictions and true values over five-fold cross-validation. The errors variance is also computed.

In Figure 16 we compare the output of the MP-LROM models against the singular values estimation on a set of randomly selected test data. The singular values estimation is the standard method for selecting the reduced manifold dimension when a prescribed level of accuracy of the reduced solution is desired. Here the desired accuracy \( \bar{\varepsilon} \) was set to \( 10^{-3} \). The mismatches between the predicted and true dimensions are depicted in Figure 16. The predicted values are the averages over five different models where each MP-LROM model constructed using ANN and GP is trained on random 80% split of data set and tested on the fixed selected 20% test data. We
notice that the snapshots matrix spectrum underestimates the true dimension of the manifold as expected since the ‘in-plane’ errors are not accounted. The ANN predictions were extremely accurate for most of the samples while the GP usually overestimated the reduced manifold dimensions.

6.3. Applications of MP-LROM error prediction models

6.3.1. Designing the decomposition of the parametric domain

We seek to build a decomposition of the viscosity domain $[0.01, 1]$ for the 1D-Burgers model using the MP-LROM models introduced in Section 6.2.1. The non-physical parameter $\nu$ is set to 1. As discussed in Section 6.2.1, we take the following steps. First we identify “$\mu_p$-feasible” intervals $[d_\ell, d_r]$ in the parameter space such that local reduced-order model depending only on the high-fidelity trajectory...
at $\mu_p$ is accurate to within the prescribed threshold for any $\mu \in [d_l, d_r]$. Second, a greedy algorithm generates the decomposition

$$[0.01, 1] \subset \bigcup_{i=1}^{M} \left[ d_{li}, d_{ri} \right], \quad (34)$$

by covering the parameter space with a union of $\mu_p$-feasible intervals, where each $\mu_p$-feasible interval is characterized by an error threshold $\tilde{\varepsilon}$ (which can vary from one interval to another). This relaxation is suggested since for intervals associated with small parameters $\mu_p$, it is difficult to achieve small reduced-order models errors similar to those obtained for larger parametric configurations.

For the existing reduced basis methods a global reduced-order model depending on multiple high-fidelity trajectories is usually constructed. In contrast, our approach uses the MP-LROM models to decompose the parameter space into smaller regions where the local reduced-order model solutions are accurate to within some tolerance levels. Since the local bases required for the construction of the local reduced-order models depend on only a single full simulation, the dimension of the POD subspace is small, leading to lower on-line computational complexity.

6.3.1.1. Construction of a $\mu_p$-feasible interval. We noticed in the previous subsection that MP-LROM models can accurately estimate $\log \varepsilon_{HF, \mu_p, K_{POD}}$ associated with reduced-order models. Thus we can employ them to establish a range of viscosity parameters around $\mu_p$ such that the reduced-order solutions depending on $U_{\mu_p}$ satisfy some desired accuracy level. More precisely, starting from parameter $\mu_p$, a fixed POD basis dimension and a tolerance error $\log \tilde{\varepsilon}$, we are searching for an interval $[d_l, d_r]$ such that the estimated prediction $\log \varepsilon_{HF, \mu_p, K_{POD}}$ of the true error $\log \varepsilon_{HF, \mu_p, K_{POD}}$ meets the requirement

$$\log \varepsilon_{HF, \mu_p, K_{POD}} < \log \tilde{\varepsilon}, \forall \mu \in [d_l, d_r]. \quad (35)$$

Our proposed strategy makes use of a simply incremental approach by sampling the vicinity of $\mu_p$ to account for the estimated errors $\log \varepsilon_{HF, \mu_p, K_{POD}}$ forecasted by the MP-LROM models defined before. A grid of new parameters $\mu$ is built around $\mu_p$ and the error models predict the errors outward of $\mu_p$. Once the error models outputs are larger than the prescribed error $\log \tilde{\varepsilon}$, the previous $\mu$ satisfying the constraint (35) is set as $d_l$, for $\mu < \mu_p$ or $d_r$ for $\mu > \mu_p$.

Figure 17 illustrates the range of parameters predicted by the MP-LROM models via ANN and GP against the true feasible interval and the results show good agreement. For this experiment we set $\mu_p = 0.7$, dimension of POD subspace $K_{POD} = 9$ and $\varepsilon = 10^{-2}$. Values of $\mu = \mu_p \pm 0.001 \cdot i, i = 1, 2, \ldots$ are passed to the MP-LROM models. The average range of parameters obtained over five different configurations with ANN is $[0.650, 0.780]$ while in the case of GP we obtained $[0.655, 0.780]$. In each configuration, we train the model with 80% random split of the data set and test it over the fixed test set of Figure 17. For this design, the true range of parameters is $[0.650, 0.785]$ underlying the predicting potential of MP-LROM models built using the regression machine learning techniques.
6.3.1.2. The decomposition of the parametric domain as a union of $\mu_p$–feasible intervals. A union of different $\mu_p$-feasible intervals can be designed to cover a general entire 1D-parametric domain $[A, B]$. Once such construction is available, it will allow for reduced-order simulations with a-priori error quantification for any value of viscosity parameter $\mu \in [A, B]$.

A greedy strategy based on the MP-LROM error models constructed in Section 6.2.1 is described in Algorithm 3 and its output is a collection of feasible intervals $\bigcup_{k=1}^{n}[d^k_l, d^k_r] \supset [A, B]$. After each iteration $k$ of the algorithm, a $\mu_p$-feasible interval $[d^k_l, d^k_r]$ is constructed. Each interval is associated with some accuracy threshold $\varepsilon_k$. For small viscous parametric values we found out that designing $\mu_p$–feasible intervals associated with higher precision levels (very small thresholds $\varepsilon_k$) is impossible since the dynamics of parametric 1D-Burgers model solutions change dramatically with smaller viscosity parameters. In consequence we decided to let $\varepsilon_k$ vary along the parametric domain to accommodate the solution physical behaviour. Thus a small threshold $\varepsilon_0$ will be initially set and as we will advance charting the parameter domain $[A, B]$ from right to left, the threshold $\varepsilon_k$ will be increased.

The algorithm starts by selecting the first centered parameter $\mu_{p_0}$ responsible for basis generation. It can be set to $\mu_{p_0} = B$ but may take any value in the proximity of $B$, $\mu_{p_0} \leq B$. This choice depends on the variability of parametric solutions in this domain region and by selecting $\mu_{p_0}$ to differ from the right edge of the domain, the number $n$ of the feasible intervals should decrease.

The next step is to set the threshold $\varepsilon_0$ along with the maximum permitted size of the initial feasible interval to be constructed. This is set to $2 \cdot r_0$, thus $r_0$ can be referred as the interval radius. Along with the radius, the parameter $\Delta r$ will decide the maximum number of MP-LROM model calls employed for the construction of the first $\mu_{p_0}$-feasible interval. While the radius is allowed to vary during the algorithm iterations, $\Delta r$ is kept constant. Finally the dimension of POD basis has to be selected together with three parameters $\beta_1$, $\beta_2$ and $\beta_3$ responsible for changes in the threshold and radius and selecting a new parameter location $\mu_{p_k}$ encountered during the procedure.

Next the algorithm starts the construction of the $\mu_{p_0}$ feasible interval. The process is described in the top part of Figure 18(a). Then we are sampling the vicinity of $\mu_{p_0}$ for equally distributed parameters $\mu$ and compute the MP-LROM model predictions. The sampling process and the comparison between the predicted errors $\log \varepsilon_{\mu,\mu_{p_0},K_{POD}}$ and $\log \varepsilon_0$ are depicted in Figure 18(a). A green vertical segment indicates that the estimated error satisfies the threshold; i.e., $\log \varepsilon_{\mu,\mu_{p_0},K_{POD}} < \log \varepsilon_0$, whereas the red segment indicates the opposite. The left limit of the $\mu_{p_0}$–feasible interval is obtained when either $\mu > \mu_{p_0} - r_0$ or $\log \varepsilon_{\mu,\mu_{p_0},K_{POD}} > \log \varepsilon_0$. The left limit $d^0_l$, denoted with a green dashed line in Figure 18(a), is set equal to the last parameter $\mu$ such that $\log \varepsilon_{\mu,\mu_{p_0},K_{POD}} \leq \log \varepsilon_0$.

The next step searches for a centered parameter $\mu_{p_{k+1}}$, and this process is described at the bottom of the Figure 18(a) for $k = 0$. The
centered parameter $\mu_{pk+1}$ is first proposed based on an empirical formula described in line 25 of Algorithm 3. This formula depends on the current centered parameter $\mu_{pk}$, the number of tested parameters $\mu$ during the construction of $\mu_{pk}$—feasible interval, parameters $\Delta r$ and $\beta_3$. Next, the algorithm checks if the following constraint is satisfied

$$[d_{l}^{k+1}, d_{r}^{k+1}] \bigcap \left( \bigcup_{i=1}^{k} [d_{l}^{i}, d_{r}^{i}] \right) \neq \emptyset,$$

(36)

without taking into account the MP-LROM model error. This is achieved by comparing the error model prediction $\log \hat{\epsilon}_{k+1}$ (see instruction 27 and bottom of Figure 18(a) for $k = 0$). If the predicted error is smaller than the current threshold, assuming a monotonically increasing error with larger distances $d(\mu, \mu_{pk+1})$, the reduced-order model solutions should satisfy the accuracy threshold for all $\mu \in [\mu_{pk+1}, d_{l}^{k}]$. In consequence the equation (36) will be satisfied for the current $\mu_{pk+1}$, if we set $r_{k+1} = \mu_{pk+1} - d_{l}^{k}$ (see instruction 30). In the case the error estimate is larger than the present threshold, the centered parameter $\mu_{pk+1}$ is updated to the middle point between old $\mu_{pk+1}$ and $d_{l}^{k}$ (see also the bottom of Figure 18(a)). For the situation where the monotonic property of the error does not hold in practice, a simply safety net is used at instruction 12.

The instructions between lines 5 and 21 generate the $\mu_{pk}$—feasible interval, for the case when the current centered parameter $\mu_{pk} \neq d_{l}^{k+1}$ (see top part of Figure 18(b) for $k = 1$). Here by int we refer to the integer part of a real number. We used the Matlab command int for the implementation. For situation when $\mu_{pk} = d_{l}^{k+1}$ (see bottom of Figure 18(b) for $k = 2$), the threshold has to be increased (by setting $\hat{\epsilon}_{k} = \beta_{1} \hat{\epsilon}_{k}$ at line 23), since the reduced-order model solutions can not satisfy the desired precision according to the predicted errors. In consequence, $\beta_{1}$ has to be selected larger than 1. The need for relaxing the threshold suggests that the greedy search is currently operating into a parametric region where only a slight change in the parameter $\mu$ away from $\mu_{pk}$ leads to predicted ROM errors larger than the current threshold. Relaxing the threshold and decreasing the radius size (select $\beta_{2} < 1$ in line 23 of Algorithm 3) can be used as a strategy to identify a feasible region for the current centered parameter $\mu_{pk}$. Similarly, relaxing the threshold and expanding the search ($\beta_{2} > 1$) could also represent a viable strategy. However, expanding the search in a parametric regime with large changes in the model dynamics, even if the threshold was relaxed, may lead to useless evaluations of the expressions in lines 7 and 18 of the Algorithm 3. Thus $\beta_{2}$ should be selected smaller than 1. Once the feasible region is obtained, the radius $r_{k}$ is reset to the initial value $r_{0}$ (see line 25 of the Algorithm 3). By selecting $\beta_{3} > 1$, the computational complexity of Algorithm 3 is decreased since the first proposal of the new centered parameter $\mu_{pk+1}$ will always be smaller than the left limit $d_{l}^{k}$ of the current feasible interval. The entire algorithm stops when $\mu_{pk+1} \leq A$.

For our experiments we employed the ANN MP-LROM model, and we set $A = 0.01$, $B = 1$, $\varepsilon_{0} = 10^{-2}$, $\Delta r = 5 \times 10^{-3}$, $r_{0} = 0.5$, $K_{\text{POD}} = 9$, $\beta_{1} = 1.2$, $\beta_{2} = 0.9$ and $\beta_{3} = 1.4$. We initiate the algorithm by setting $\mu_{p0} = 0.87$, and the first feasible interval $[0.7700, 1]$ is obtained. Next the algorithm selects $\mu_{l} = 0.73$ with the associated range of $[0.6700, 0.8250]$ using the same initial threshold level. As we cover the parametric domain from right to left; i.e., selecting smaller and smaller parameters $\mu_{pk}$, the algorithm enlarges the current threshold $\hat{\epsilon}_{k}$, otherwise the error model predictions would not satisfy the initial precision. We continue this process until we get the threshold 6.25 with $\beta_{32} = 0.021$ and the corresponding feasible interval $[0.01, 0.039]$. The generated decomposition is depicted in Figure 19 where the associated threshold varies with the parameter change.
Algorithm 3 Generation of 1D-parametric domain decomposition for reduced-order models usage. Extension to multi-dimensional parametric space is subject to future research.

1: Select $\mu_{p_0}$ as the right edge of the parameter interval, i.e. $\mu_{p_0} = B$.
2: Set error threshold $\bar{\varepsilon}_0$, step size $\Delta r$ for selection of sampling parameters $\mu$, the maximum search radius $r_0$, dimension of POD basis $K_{POD}$ and $\beta_1$, $\beta_2$ and $\beta_3$.
3: Set $k = 0$.
4: WHILE $\mu_{p_k} \geq A$ DO
5: FOR $i = 1$ to $\text{int}(\frac{r_{k}}{\Delta r}) + 1$
6: Set $\mu = \mu_{p_k} + i\Delta r$
7: IF ($\phi(\mu, \mu_{p_k}, K_{POD}) > \log \bar{\varepsilon}_k \text{ OR } \mu > B$) THEN
8: Set $d_{k}^{i} = \mu_{p_k} + (i - 1)\Delta r$. EXIT.
9: END IF
10: END FOR
11: IF $k > 0$ THEN
12: IF $d_{k}^{i} < d_{k}^{i-1}$ THEN
13: $\mu_{p_k} = \frac{\mu_{p_k} + d_{k}^{i-1}}{2}$. GOTO 5.
14: END IF
15: END IF
16: FOR $j = 1$ to $\text{int}(\frac{r_{k}}{\Delta r}) + 1$
17: Set $\mu = \mu_{p_k} - j\Delta r$
18: IF ($\phi(\mu, \mu_{p_k}, K_{POD}) > \log \bar{\varepsilon}_k \text{ OR } \mu < A$) THEN
19: Set $d_{k}^{j} = \mu_{p_k} - (j - 1)\Delta r$. EXIT.
20: END IF
21: END FOR
22: IF ($i = 1$) OR ($j = 1$) THEN
23: Set $\bar{\varepsilon}_k = \beta_1 \cdot \bar{\varepsilon}_k$; $r_k = \beta_2 \cdot r_k$; GOTO 5.
24: ELSE
25: $\mu_{p_{k+1}} = \mu_{p_k} - \beta_3(j - 1)\Delta r$; $\bar{\varepsilon}_{k+1} = \bar{\varepsilon}_k$; $r_{k+1} = r_0$.
26: END IF
27: WHILE $\phi(d_{k}^{j}; \mu_{p_{k+1}}, K_{POD}) > \log \bar{\varepsilon}_{k+1}$ DO
28: $\mu_{p_{k+1}} = \frac{\mu_{p_{k+1}} + d_{k}^{j}}{2}$.
29: END WHILE
30: Set $r_{k+1} = \mu_{p_{k+1}} - d_{k}^{j}$.
31: $k = k + 1$.
32: END WHILE
6.3.2. Construct a hierarchy of accurate local bases, local reduced-order and high-fidelity models for a parameter value $\mu^*$

Here, we propose to solve another practical problem. Giving a collection of local reduced bases and operators and high-fidelity trajectories computed at various locations in the parameter space, construct a hierarchy of the available bases and models producing the most accurate local parametric reduced-order solutions for an arbitrary viscosity parameter $\mu^*$. We will rely on similar MP-LROM models built in Subsection 6.2.1. The input features for the MP-LROM models are the parameter $\mu^*$, another viscosity parameter $\mu_p$, whose corresponding trajectory is used to generate the basis $U_{\mu_p}$ and the dimension $K_{POD}$ of the reduced manifold. The output variable $\hat{y} = \log \varepsilon_{HF, \mu^*, K_{POD}}$ is the estimated log of error of the reduced-order model solution at $\mu^*$ using the basis $U_{\mu_p}$ and the associated reduced operators.

The data set constructed for this problem is similar to the one employed for designing the errors models in section 6.2.1. However we thought that by considering additional parameters $\mu_p$ in the vicinity of parameters $\mu$ inside the data set will be benefic and will increase the precision of the error models. To diminish the number of required high-fidelity simulations we decreased the number of configurations...
\( \mu \) selected inside the training set. As such, for the current problem, the data set includes 100 parameters \( \mu_p \in \{0.01, \ldots, 1\} \), 10 parameters \( \mu^* \in \{0.1, 0.2, \ldots, 1\} \), 12 reduced basis dimensions \( K_{POD} \) spanning the set \( \{4, 5, \ldots, 14, 15\} \) and the associated log of the Frobenius norm of true ROM errors \( \log \varepsilon_{HF, \mu_p, K_{POD}} \) [29]. The data set is randomly partitioned into five equal size sub-samples and a five-fold cross-validation process is used to assess the quality of the MP-LROM models. A neural network with 6 hidden layers and hyperbolic tangent sigmoid activation function in each layer is used and for the Gaussian Process we have employed the squared-exponential-covariance kernel [10].

Table 8 shows the averages and variances of errors in prediction of the MP-LROM models constructed via GP and ANN methods. As the number of samples increases the accuracy increases and the variance decreases. In contrast with the results obtained in Subsection 6.2.1, GP always outperforms ANN. Different data sets lead to different error models.

<table>
<thead>
<tr>
<th>Sample size</th>
<th>GP MP-LROM</th>
<th>ANN MP-LROM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( E_{fold} )</td>
<td>( \text{VAR}_{fold} )</td>
</tr>
<tr>
<td>100</td>
<td>0.5539</td>
<td>0.0356</td>
</tr>
<tr>
<td>400</td>
<td>0.3541</td>
<td>0.0012</td>
</tr>
<tr>
<td>700</td>
<td>0.2942</td>
<td>4.4623 \times 10^{-4}</td>
</tr>
<tr>
<td>1000</td>
<td>0.2243</td>
<td>8.5555 \times 10^{-4}</td>
</tr>
<tr>
<td>3000</td>
<td>0.1196</td>
<td>4.8735 \times 10^{-4}</td>
</tr>
<tr>
<td>5000</td>
<td>0.0773</td>
<td>5.7075 \times 10^{-6}</td>
</tr>
</tbody>
</table>

Table 8: Average and variance of error in predictions of (28) for MP-LROM models using logarithms of errors (\( \log(\varepsilon) \)) in training data for different sample sizes.

Table 9 shows numerical results obtained with ANN and GP, over five folds using (8b). The GP outperforms the ANN and has less variance which indicates the GP is more accurate and stable for this specific problem than ANN. The performances of the MP-LROM models, discussed in this subsection and subsection 6.2.1 constructed using two different data sets, can be compared in Tables 3 and 9. The five-fold cross-validation test shows that increasing the number of parameters \( \mu_p \) and decreasing the number of parameters \( \mu \) inside the database lead to a higher-accuracy of the MP-LROM error models.

<table>
<thead>
<tr>
<th></th>
<th>E</th>
<th>VAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN MP-LROM</td>
<td>0.002839</td>
<td>1.5496 \times 10^{-5}</td>
</tr>
<tr>
<td>GP MP-LROM</td>
<td>0.001135</td>
<td>2.5922 \times 10^{-8}</td>
</tr>
</tbody>
</table>

Table 9: Statistical results of error in predictions of MP-LROM models over five-fold cross-validation.

The results in Figure 20 illustrate the mean of the errors in prediction of the MP-LROM models for two viscosity parameters \( \mu^* = 0.35 \) and \( \mu^* = 0.65 \) and various bases. Thus we constructed a hierarchy of the available local bases and local reduced-order models producing the most accurate solutions for \( \mu^* = 0.35 \) and \( \mu^* = 0.65 \). We also ranked the corresponding high-fidelity solutions in accordance to the hierarchy of local reduced-order models to be used in the next sub-section experiments to enhance the accuracy of the local parametric reduced-order model for configurations \( \mu^* \).

Moreover we can notice that the estimation curves are crossing initially close to \( \mu_p = 0.45 \). Consequently, a reduced-order model constructed for \( \mu_p = 0.45 \) computes reduced-order solutions for \( \mu^* = 0.35 \) and \( \mu^* = 0.65 \) with similar accuracy levels \( \varepsilon_{HF, \mu_p, K_{POD}} \).

6.4. Combining available information for accurate local ROMs at arbitrary parametric configurations

Experiments in Subsection 6.3.2 revealed the potential of MP-LROM models to select a hierarchy of local bases and local reduced-order and high-fidelity models that produces higher accurate solutions for specific parametric configurations. Figure 20 depicts the accuracy of reduced-order models for \( \mu^* = 0.35 \) and \( \mu^* = 0.65 \) using POD basis computed at various locations \( \mu_p \) in the parameter domain. Among a subset of the existing POD reduced-order models constructed for parameters equally distributed along the interval
involved bases. This strategy was selected based on the spectra of the snapshots matrices associated with having the same dimension. Only even basis dimensions are utilized and the concatenation method combines half of the modes of each
simply described as "Lagrange bases" and "Lagrange sol" in the legends of Figures 21 and 22. All the methods employ reduced bases followed by orthogonalization is referred as Gram-Schmidt whereas the Lagrange interpolation of bases and high-fidelity solutions are
ments: variation in the final time $t_f$, in the non-linear advection coefficient $\nu$ and POD basis dimension. The bases concatenation method followed by orthogonalization is referred as Gram-Schmidt whereas the Lagrange interpolation of bases and high-fidelity solutions are simply described as "Lagrange bases" and "Lagrange sol" in the legends of Figures 21 and 22. All the methods employ reduced bases having the same dimension. Only even basis dimensions are utilized and the concatenation method combines half of the modes of each involved bases. This strategy was selected based on the spectra of the snapshots matrices associated with $\mu_{p1} = 0.3$ and $\mu_{p2} = 0.4$. More precisely, we selected the first $\frac{K_{POD}}{2}$ singular vectors from each of the bases since for all our experiments $\lambda_{K_{POD}}^{\mu_{p1}} > \lambda_{K_{POD}+1}^{\mu_{p2}}$ and $\lambda_{K_{POD}}^{\mu_{p1}} > \lambda_{K_{POD}+1}^{\mu_{p2}}$, where $\lambda_{K_{POD}}^{\mu_{p1}}$ denotes the singular value corresponding to the first singular vector associated with trajectory $\mu_{p1}$, not taken into account in the POD expansion. Of course this choice is not optimal. The optimal solution can be obtained by solving a combinatorial optimization problem, where the searched space has the size of $2K_{POD}$ chose $K_{POD}$.

The first two experiments scale the time and space and modify the linear and nonlinear characteristics of the model. For example, in the case of a tiny small final time and advection coefficient, the diffusion linear part represents the main dynamical engine of the model thus it behaves linearly. The results are compared against reduced-order models constructed using $U_{\mu_{p1}}$ and $U_{\mu_{p2}}$ respectively.

Figure 21 illustrates the Frobenius norm error between the high fidelity and reduced-order model solutions for the final time $t_f = 0.01$. Panel (a) presents the accuracy results as a function of the advection coefficient $\nu$. Interpolating the high-fidelity solutions leads to the most accurate reduced-order model. For large advection coefficients all of the methods suffer accuracy losses. Among the potential explanations we include the constant dimension of the POD basis and the linear dependence on the viscosity parameter assumed by all of the methods in various forms. Figure 24 shows that the basis dimension must be increased as the advection coefficient decreases to maintain constant error.

Since the Grassmann manifold approach is a generalization of the subspace angle interpolation method we decide to show only the results corresponding to the former method. While Lagrangian interpolation of the bases is performed in both matrix space and tangent space of the Grassmann manifold (shown in cyan and green), the later approach performs better in this scenario. The concatenation
of bases using Gram-Schmidt algorithm was successful only for larger advection coefficients (red curve in Figure 21(a)), for a POD dimension set to 14.

Increasing the dimension of the basis enhances the so called Gram-Schmidt reduced-order model solution accuracy for $\nu = 1$ (see Figure 21(b)). For this case Lagrange interpolation in the matrix space shows better performances in comparison with the output of the Grassmann manifold approach.

Next we increase the nonlinearity characteristics of the model by setting the final time to $t_f = 1$ and Figure 22 illustrates the Frobenius norm errors as a function of the advection coefficient $\nu$ and POD dimension. The errors produced by the reduced-order model derived via Grassmann manifold method are similar with the ones obtained by the surrogate model relying on a POD basis computed via the Lagrange interpolation of the high-fidelity model solutions.

The Lagrange interpolation of bases in the matrix space is not successful as seen in both panels of Figure 22. Increasing the POD dimension to 20, the Gram-Schmidt approach enhances the accuracy of the solution (see Figure 22(b)), for $\nu = 1$. 

Figure 21: Strategies comparison for generation of accurate ROMs for a new viscosity parameter $\mu^* = 0.35$ and $t_f = 0.01$. In panel (a) $K_{POD}$ is set to 14, whereas in panel (b) $\nu$ is set to 1.

Figure 22: Strategies comparison for generation of accurate ROMs for a new viscosity parameter $\mu^* = 0.35$ and $t_f = 1$. In panel (a) $K_{POD}$ is set to 14, whereas in panel (b) $\nu$ is set to 1.
7. Conclusions

In this study, we introduced new multivariate input-output models (MP-LROM) to predict the errors and dimensions of local parametric reduced-order models. Approximation of these mappings were built using Gaussian Process and Artificial Neural Network.

Initially, we compared our MP-LROM error models against those constructed with multi-fidelity correction technique and ROMES method. Since global bases are used by the multi-fidelity correction and ROMES methods, we implemented corresponding local error models using only small subsets of the data utilized to generate our MP-LROM models. In contrast, the MP-LROM models are global and rely on a global database. Moreover, our MP-LROM models differ from the ROMES models [52] and multi-fidelity correction models [51], having more additional features such as reduced subspace dimension and are specially projected for accurate predictions of local reduced-order models errors. As such, the MP-LROM models require significantly more and different data than multi-fidelity correction models. The numerical experiments revealed that our MP-LROM error models are more accurate than the error models constructed with multi-fidelity correction technique and ROMES method for estimating the errors of local parametric reduced-order 1D-Burgers models with a single parameter. In the case of large parametric domains, the MP-LROM error models may require a very large data set due to the large number of input features. In the future we plan to use only subsets of the global data set near the vicinity of the parameters of interest and combine our technique with the active subspace method [92] to prevent the potential curse of dimensionality that the MP-LROM models might suffer.

Next we addressed the problem of selecting the dimension of a local reduced-order model when its solution must satisfy a desired level of accuracy. The approximated MP-LROM models based on learning better estimated the ROM basis dimension in comparison with the results obtained by truncating the spectrum of the snapshots matrix.

Later we considered applications of the MP-LROM error models. First we demonstrated the value of our proposed MP-LROM models to guide the construction of parametric space partitioning for the usage of efficient and accurate local reduced-order models. While the current methodologies are defined in the sense of Voronoi tessellation [116] and rely on K-means algorithms [117–119], our approach delimitates sub-regions of the parametric space by making use of the Gaussian Processing and Artificial Neural Networks MP-LROM error models and parametric domain sampling. For each sub-region, an associated local reduced-order basis and operators are constructed depending on a single representative high-fidelity trajectory and, the corresponding local reduced-order models solutions have known precision levels. Under specific conditions, we also derived a theoretical lower limit for the accuracy level of a local parametric reduced-order model associated with a parametric sub-region. The novel methodology is applied for a 1D-Burgers model, and a decomposition of the viscosity domain with parametric sub-intervals and associated errors thresholds was designed.

For designing the parametric domain decomposition, we were not able to maintain the initial level of accuracy for smaller viscosity parameters. Even if we increased the local basis dimension, the initial threshold could not be maintained. This indicates that the high-fidelity solutions associated with new parameter configurations lie outside the space generated by the local basis. Enriching the current local bases with additional high-fidelity snapshots associated with other parameter configurations represents a potential solutions for preserving the initial threshold. In the present configuration, the parametric domain decomposition algorithm is limited to one-dimensional parameter problems. Current research investigates extension to small multi-dimensional parametric domains based on multivariate sampling.

Second, we employed machine learning MP-LROM models to select a hierarchy of local bases, local reduced-order and high-fidelity models producing the most accurate solutions for an arbitrary parameter configuration. Based on this hierarchy, three already existing methods involving bases interpolation and concatenation and high-fidelity model solutions interpolation were applied to enhance the quality of the associated reduced-order model solutions. Several experiments were performed by scaling the time and space and modifying the nonlinear characteristics of the 1D-Burgers model. In most cases, interpolating the already existing high-fidelity trajectories generated the most accurate reduced-order models for a new viscous parameter revealing that the solution behavior over the parametric
region under study can be linearly approximated. Lagrange interpolation of bases in the tangent space of the Grassmann manifold and concatenation of bases for larger reduced subspaces showed also good performances.

In the future we seek to decrease the computational complexity of the MP-LROM error models. Currently the training data required by the machine learning MP-LROM models relies on many high-fidelity simulations. By employing error bounds, residual norms and a-posteriori error estimation results, this dependency could be much decreased.

In addition, we plan to construct machine learning MP-LROM models to estimate the errors in quantities of interest computed with reduced-order models. The predictions of such error models can then be used to speed up the current trust-region reduced-order framework by eliminating the need of high-fidelity simulations for the quality evaluation of the updated controls.

Acknowledgements

This work was supported in part and by the award NSF CCF 1218454 and by the Computational Science Laboratory at Virginia Tech. Răzvan Ştefănescu thanks Dr. Dan Ştefănescu for his valuable suggestions on the machine learning topic.

References


[38] H. Hotelling, Analysis of a complex of statistical variables with principal components, Journal of Educational Psychology 24 (1933) 417–441.


