Gradient-Enhanced Surrogate Modeling Based on Proper Orthogonal Decomposition

Zimmermann, Ralf

Published in:
Journal of Computational and Applied Mathematics

DOI:
10.1016/j.cam.2012.06.010

Publication date:
2013

Document version
Other version

Citation for published version (APA):
Abstract

A new method for enhanced surrogate modeling of complex systems by exploiting gradient information is presented. The technique combines the proper orthogonal decomposition (POD) and interpolation methods capable of fitting both sampled input values and sampled derivative information like Kriging (aka spatial Gaussian processes). In contrast to existing POD-based interpolation approaches, the gradient-enhanced method takes both snapshots and partial derivatives of snapshots of the associated full-order model (FOM) as an input. It is proved that the resulting predictor reproduces these inputs exactly up to the standard POD truncation error. Hence, the enhanced predictor can be considered as (approximately) first-order accurate at the snapshot locations. The technique applies to all fields of application, where derivative information can be obtained efficiently, for example via solving associated primal or adjoint equations. This includes, but is not limited to Computational Fluid Dynamics (CFD). The method is demonstrated for an academic test case exhibiting the main features of reduced-order modeling of partial differential equations.

Keywords: proper orthogonal decomposition, gradient-enhanced interpolation, surrogate model, snapshot derivatives

2000 MSC: 37M99, 65P99

1. Introduction

Even with today’s impressive high-performance computing (HPC) resources at hand, there remain many problems in science and engineering, especially in the context of industrial applications, where the use of high-order numerical methods is still out of reach. In Computational Fluid Dynamics (CFD), for example, high-order codes produce accurate solutions to the Navier-Stokes equations at very specific flow conditions but are often too costly to apply for optimization, uncertainty prediction or aero-data prediction over the entire flight envelope for full aircraft configurations. As a consequence, accurate but efficient reduced-order models (ROMs) are sought after.

Looking at things from a different perspective, one could argue that the need
for ROMs arises not despite today’s HPC capabilities but is motivated by them, since they permit tackling more challenging problems in the first place.

A powerful tool currently considered state-of-the-art for order reduction of nonlinear systems ([1, §3.8]) is proper orthogonal decomposition (POD), a technique which has been demonstrated in many fields of application, see e.g. [2, 3, 4, 5, 6, 7, 8] and is also subject to ongoing theoretical investigations, see e.g. [9, 10, 11, 12]. Following the POD approach, the high-fidelity system is projected onto a basis of low dimension, which is constructed relying on a finite number of solution snapshots of the corresponding full-order model (FOM) computed at preselected parameter conditions. No matter how sophisticated, all POD-based methods proposed in the literature eventually boil down to estimating the coefficients of the reduced-basis expansion of the approximate solution, the three main approaches for doing so being interpolation [3, 4, 6], solving a low-order optimization problem [5, 8] and solving a system of low-order PDEs via projection methods [2, 9, 10, 13, 14].

The original contribution of this paper is a method for constructing ROMs based on gradient-enhanced interpolation of POD coefficients, in the following referred to as GEPOD. In contrast to existing interpolation approaches, the GEPOD model takes both snapshots and partial derivatives of snapshots of the FOM in question as an input. It is proved that the resulting model reproduces this information exactly up to the standard POD truncation error. Hence, at the snapshot locations, the GEPOD model can be considered an order-1 accurate approximation of the FOM. Moreover, the method is shown to be simple, accurate and efficient.

In many application scenarios, including but not limited to CFD, partial derivatives of snapshot solutions can be obtained efficiently via the adjoint approach, see [15] and the extensive references therein, or via fast multiple right-hand side techniques, see [14, §4.4].

Related attempts to incorporate derivative information in building ROMs include [10], [14] and [16]. In [10], the snapshot input data is supplemented by finite difference approximations of the snapshot derivatives for the purpose of improved theoretical error estimates, while in [16], snapshots and their partial derivatives are directly used as a reduced basis in a finite-element-like framework, referred to as Hermite approach. In the recent work [14], a framework for constructing Petrov-Galerkin subspaces optimally tailored for output-oriented predictions is developed. The subspaces rely on both snapshot information and their partial derivatives, which are weighted according to their Taylor expansion coefficients and their spatial distance to the prediction point.

All the above papers share the idea of exploiting derivative information within the context of projection schemes, which is a key difference to the work presented here, where the derivative information is used to enhance the POD-based interpolation. Note that the simplicity of interpolation schemes makes them a well-established tool in industry.

The objective of building gradient-enhanced ROMs should not to be confused with a sensitivity analysis of the POD basis modes themselves, which is carried out, for example, to investigate their dependence on changes in the system pa-
rameters, see [17, 18].

The current work serves as a basic introduction to and analysis of the GEPOD method, while applications to real-life engineering problems (in the context of CFD) are the topic of upcoming work. The paper is organized as follows. In the following section, a brief review of POD in finite-dimensional non-euclidean vector spaces is given. All practical applications known to the author subordinate to this setting. In section 3, the new gradient-enhanced POD-based ROM approach is introduced and discussed theoretically. In section 4, the method is demonstrated on an academic test case featuring snapshots of similar shape as the solutions to the convection-diffusion problem in fluid dynamics [13]. Section 5 concludes the paper.

**Nomenclature**

\[ P^{n \times n} \in \mathbb{R}^{n \times n} \] : identity matrix

\[ e_i = (0, \ldots, 0, 1, 0, \ldots 0)^T \] : \(i\)th standard basis vector

\[ 1^n = (1, \ldots, 1)^T \in \mathbb{R}^n \] : vector with all entries equal to 1

\[ \langle \cdot, \cdot \rangle_2 \] : euclidean scalar product on \( \mathbb{R}^n \)

\[ \| \cdot \|_2 \] : euclidean norm induced by \( \langle \cdot, \cdot \rangle_2 \)

\[ \| \cdot \|_1 \] : 1-norm, \( \| X \|_1 = \sum |X_i| \)

\[ \| \cdot \|_F \] : Frobenius matrix norm

\[ \langle \cdot, \cdot \rangle_S, (X, Y)_S = X^T S^T S Y \] : scalar product on \( \mathbb{R}^n \) induced by regular matrix \( S \)

\[ \| \cdot \| S \] : norm induced by \( \langle \cdot, \cdot \rangle_S \)

\[ \perp_S \] : “orthogonal with respect to \( \langle \cdot, \cdot \rangle_S \)”

\[ L_2(D) \] : set of square integrable functions on \( D \)

\[ \langle \cdot, \cdot \rangle_{L_2} \] : standard scalar product on \( L_2(D) \)

\[ \| \cdot \|_{L_2} \] : norm induced by \( \langle \cdot, \cdot \rangle_{L_2} \)

**Abbreviations**

POD : Proper Orthogonal Decomposition

ROM : Reduced-Order Model

FOM : Full-Order Model

PDE : Partial Differential Equation

CFD : Computational Fluid Dynamics

SVD : Singular Value Decomposition

GEK : Gradient-Enhanced Kriging

#### 2. Theoretical Background

In this section, we review POD for non-euclidean scalar products, as they arise for example from the discretization of dynamical systems. While this is well-known to the experts, see e.g [10, §2.1] or [14], sketches of proofs are included here to demonstrate the simplicity of the POD and to make this paper self-contained. It is to mention that there is a close connection of POD to singular value decomposition (SVD), for the details the reader is referred to [10, §2] or [7, §2.5].

##### 2.1. POD with respect to non-euclidean scalar products

Let \( S \in \mathbb{R}^{n \times n} \) be regular. Then \( S^T S \) is symmetric positive definite and therefore induces a scalar product \( \langle \cdot, \cdot \rangle_S : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}, \langle V, W \rangle_S := V^T S^T S W \)
with associated norm \( \| \cdot \|_S = \sqrt{\langle \cdot, \cdot \rangle_S} \).

For a set of orthonormal vectors \( U^1, ..., U^m \in \mathbb{R}^n \), \( \|U^k\|_S = 1 \), \( \langle U^j, U^k \rangle_S = \delta_{jk} \), the orthogonal projection with respect to \( \langle \cdot, \cdot \rangle_S \) onto the \( m \)-dimensional subspace spanned by \( U^1, ..., U^m \) is given by

\[
\Pi : \mathbb{R}^n \to \mathbb{R}^n, \quad X \mapsto \left( \sum_{k=1}^m U^k (U^k)^T \right) S^T SX = \sum_{k=1}^m \langle U^k, X \rangle_S U^k.
\]

(1)

**Proof.** A straightforward computation shows that \( \Pi(\Pi(X)) = \Pi(X) \) for \( X \in \mathbb{R}^n \) and \( \Pi(X) = 0 \) if \( X \perp_S \text{span}\{U^1, ..., U^m\} \).

For a single normalized vector \( U \), we denote the corresponding orthogonal projection on the one-dimensional subspace spanned by \( U \) as \( \Pi_U = U U^T S^T S \).

Note that Pythagoras’ Theorem holds for \( \langle \cdot, \cdot \rangle_S \):

\[
\|W\|_S^2 = \|W - \Pi_U(W)\|_S^2 + \|\Pi_U(W)\|_S^2
\]

(2)

for \( W, U \in \mathbb{R}^n \), \( \|U\|_S = 1 \).

For \( Y = (W^1, ..., W^m) \in \mathbb{R}^{n \times m} \), the following matrix equation holds:

\[
Y Y^T = \sum_{k=1}^m W^k (W^k)^T \in \mathbb{R}^{n \times n}.
\]

(3)

Let \( \{W^1, ..., W^m\} \subset \mathbb{R}^n \) be a set of vectors, considered as given input data. For \( \tilde{m} \leq m \), proper orthogonal decomposition is concerned with finding the \( \tilde{m} \)-dimensional subspace closest to the input data with respect to the metric induced by \( S \) and determining an orthonormal ordered basis \( (U^1, ..., U^m) \) for this subspace.

Mathematically speaking, the objective is to determine the \( \tilde{m} \)-dimensional subspace minimizing the distance between the input data and their projection onto the subspace in question. As a starting point, we solve the problem for \( \tilde{m} = 1 \).

In this regard, we note the following lemma.

**Lemma 1.** The following optimization problems are equivalent.

\[
\min_U \sum_{k=1}^m \|W^k - \Pi_U(W^k)\|_S \quad \text{s.t.} \quad \|U\|_S = 1.
\]

(4)

\[
\max_U \sum_{k=1}^m \|\Pi_U(W^k)\|_S \quad \text{s.t.} \quad \|U\|_S = 1.
\]

(5)

\[
\max_U U^T (S^T S Y Y^T S^T S) U \quad \text{s.t.} \quad \|U\|_S = 1.
\]

(6)

**Proof.** (4) ⇔ (5) follows from (2). For (5) ⇔ (6) apply (3).

For solving (6), we use a Lagrangian framework. Let

\[
L(U) := U^T S^T S Y Y^T S^T S U + \lambda(\|U\|_S^2 - 1).
\]
Computing the stationary values of the gradient of $L$ leads to the following equation

$$
S^T SY Y^T S^T SU = \lambda S^T SU, \quad \|U\|_S = 1 \quad (\Leftrightarrow YY^T S^T SU = \lambda U, \quad \|U\|_S = 1).
$$

(7)

This is equivalent to the following euclidean eigenvalue problem

$$
(SYY^T S^T) \tilde{U} = \lambda \tilde{U}, \quad \|\tilde{U}\|_2 = 1.
$$

(8)

If $\tilde{U}$ is a solution to (8), then $U := S^{-1} \tilde{U}$ is a solution to (7).

The corresponding value of the objective function in (6) is

$$
U^T (S^T SYY^T S^T S) U = \lambda \|U\|^2_S = \lambda.
$$

Therefore, the solution maximizing the convex problem (6) is given by the eigenvector corresponding to the largest eigenvalue of $YY^T S^T S$.

As a consequence:

**Theorem 2.1.** Let $\{W^1, ..., W^m\} \subset \mathbb{R}^n$ be an input data set. The best order-$\tilde{m}$-approximation of span$\{W^1, ..., W^m\}$ in the sense of (4)-(6) is spanned the first $\tilde{m}$ eigenvectors $\{U^1, ..., U^{\tilde{m}}\}$ of $YY^T S^T S$.

These can be obtained by solving the euclidean eigenvalue problem

$$
SY Y^T S^T \tilde{U} = \lambda \tilde{U}, \quad \|\tilde{U}\|_2 = 1.
$$

and setting $U := S^{-1} \tilde{U}$.

**Proof.** Suppose the best order-$(\tilde{m} - 1)$-approximation is given by the set of pairwise orthonormal vectors $(U^1, ..., U^{\tilde{m}-1})$. By (7), (8), $U^1, ..., U^{\tilde{m}-1}$ are eigenvectors of $YY^T S^T S$. In order to obtain the best order-$\tilde{m}$-approximation, solve

$$
\max \sum_{k=1}^{m} \langle W(p^k), U^{\tilde{m}} \rangle_S \quad \text{s. t.} \quad \|U^{\tilde{m}}\|_S = 1, \langle U^{\tilde{m}}, U^j \rangle_S = 0, \forall j = 1, ..., \tilde{m}.
$$

(9)

Hence, $U^{\tilde{m}}$ corresponds to the $\tilde{m}$th eigenvector of $YY^T S^T S$, where the ordering is given by the ordering of the eigenvalues by size. The best order-$\tilde{m}$-approximation of span$\{W^1, ..., W^m\}$ is thus given by span$\{U^1, ..., U^{\tilde{m}}\}$.

**Definition 1.** In the context of reduced-order modeling, the input data vectors $W^i, i = 1, ..., m$ are called snapshots and the data matrix $Y = (W^1, ..., W^m) \in \mathbb{R}^{n \times m}$ is called snapshot matrix.

The eigenvectors $U^i, i = 1, ..., \tilde{m}$ spanning the best order-$\tilde{m}$-approximation of the input data are called POD eigenmodes, or in short POD modes, and span$\{U^1, ..., U^{\tilde{m}}\}$ is called POD subspace.

The relative information content of the $j$th mode is defined as the ratio $r_j = \frac{\lambda_j}{\sum_{j=1}^{\tilde{m}} \lambda_j}$ and the relative information content of the first $\tilde{m} \leq m$ POD modes is thus given by $\text{RIC}(\tilde{m}) = \sum_{j=1}^{\tilde{m}} r_j$.

If all non-zero eigenvalues are mutually distinct, then each corresponding eigenspace is spanned by a single normalized eigenvector, unique up to the sign. Hence, we speak of the eigenvector or the eigenmode corresponding to an eigenvalue, disregarding the ambiguity of the sign.
In practical applications, the dimension $\tilde{m}$ of the POD subspace is chosen such that $\text{RIC}(\tilde{m}) \geq 1 - \epsilon$, where $\epsilon \in [0, 1)$ is a user defined threshold.

2.2. POD basis computation in practice

In most practical applications, it holds $n \gg m$, so that a solution of the $n \times n$ eigenvalue problem (8) becomes unfeasible. The remedy is reducing the problem to a $n \times m$ eigenvalue problem. This technique has been introduced in [19] and is for historical reasons referred to as method of snapshots. Note that $SYY^T S^T$ is symmetric with rank($SYY^T S^T$) $\leq m$.

Let $\lambda_1 \geq \ldots \geq \lambda_m \geq \lambda_{m+1} = 0 = \ldots = \lambda_n = 0$ be the eigenvalues of $SYY^T S^T \in \mathbb{R}^{n \times n}$ and $Y^T SY \in \mathbb{R}^{m \times m}$ ordered by size. The non-zero eigenvalues of $SYY^T S^T \in \mathbb{R}^{n \times n}$ and $Y^T SY \in \mathbb{R}^{m \times m}$ coincide: Let $V \in \mathbb{R}^m$ be a solution to $Y^T SY V = \lambda V$, $\|V\|_2 = 1$.

Then $\tilde{U} = \frac{1}{\sqrt{\lambda}} SYV$ solves (8) and

$$U = S^{-1} \tilde{U} = \frac{1}{\sqrt{\lambda}} YV \quad (11)$$

is a solution to (4)-(6). An explicit inversion of the matrix $S$ is not necessary.

Using a terminology from the SVD context, the eigenvectors $V_1, \ldots, V_m$ of (10) will be referred to as right singular vectors.

Note that the same POD modes, orthogonal with respect to $\langle \cdot, \cdot \rangle_S$, are obtained via an SVD of the weighted matrix $\tilde{Y} := SY = \tilde{U} \Sigma V^T$ by setting

$$U = S^{-1} \tilde{U} = S^{-1} \tilde{Y} V \Sigma^{-1} = YV \Sigma^{-1}. \quad (12)$$

3. Gradient-Enhanced Surrogate Modeling via POD

This section features the original contribution of this paper. We propose a method for exploiting derivative information for POD-based surrogate modeling. The approach can be considered as an enhancement of the POD-based interpolation method [3, §3.3], [4, p. 2]. Let $W : \mathbb{R}^d \rightarrow \mathbb{R}^n, p = (p_1, \ldots, p_d) \mapsto W(p)$ be an at least first-order differentiable function, in this context considered as the exact full-order model of some complex (physical, biological,...) process. In standard applications, $W(p)$ is given indirectly by a PDE. Let $W(p_1), \ldots, W(p_m) \in \mathbb{R}^n$ be snapshots of $W$ at $m$ distinct parameter locations $p_i, i = 1, \ldots, m$ and let the partial derivatives of the snapshot vectors be denoted by $\partial_i W(p_1), \ldots, \partial_i W(p_m), \ldots, \partial_d W(p_1), \ldots, \partial_d W(p_m)$. For convenience, write $W^i := W(p_i)$ and $\partial_k W^i := \partial_k W(p_i)$.

Objective: Based on the input information, construct a surrogate model $W_r : p \mapsto W_r(p)$ that coincides with the full-order model $W : p \mapsto W(p)$ at the sample locations and features the same partial derivatives thereat, i.e.

$$W_r(p_i) = W(p_i) \quad \text{and} \quad \partial_k W_r(p_i) = \partial_k W(p_i) \quad \text{for} \quad i = 1, \ldots, m, k = 1, \ldots, d. \quad (12)$$

It is understood that the method can only be expected to lead to reasonable results, if the snapshots are differentiable with respect to the parameters of interest.
3.1. Algorithms

In order to achieve the objective (12), the following algorithms are proposed. The first algorithm constructs a reduced representation of the enhanced input data. This corresponds to standard POD, but for an augmented data set.

**Algorithm 1 (GEPOD: subspace construction). Input:**

- **Regular matrix**: $S \in \mathbb{R}^{n \times n}$ (inducing the underlying scalar product)
- **Snapshot vectors**: $W^1, ..., W^m \in \mathbb{R}^n$
- **Partial derivatives**: $\partial_1 W^1, ..., \partial_1 W^m, ..., \partial_d W^1, ..., \partial_d W^m$
- **Step size vector**: $H = (h_1, ..., h_d) \in \mathbb{R}^d_{>0}$
- **RIC threshold**: $\varepsilon \in [0, 1)$

1. Enhance the input data. For $i = 1, ..., m$ and for $k = 1, ..., d$ compute
   \[ W^{i,k} := W^i + h_k \partial_k W^i. \] (13)
2. Set $M := m(d + 1)$.
3. Set $Y = (W^1, ..., W^m, W^{1,1}, ..., W^{m,1}, ..., W^{1,d}, ..., W^{m,d}) \in \mathbb{R}^{n \times M}$.
4. Center data:
   \[ A := \frac{1}{M} \left( \sum_{i=1}^m W^i + \sum_{i=1}^m \sum_{k=1}^d W^{i,k} \right), \quad \bar{Y} := Y - A \cdot (1^M)^T \] (14)
5. Perform POD of $\bar{Y}$ w.r.t. the scalar product induced by $S$ according to section 2.2.
6. Determine $\tilde{M}$ such that $\text{RIC} (\tilde{M}) \geq 1 - \varepsilon$.

**Output:**

- Reduced dimension: $\tilde{M} \leq M - 1 = m(d + 1) - 1$
- Snapshot mean: $A \in \mathbb{R}^n$
- GEPOD eigenmodes: $U^1, ..., U^{\tilde{M}} \in \mathbb{R}^n$
- Eigenvalues: $\lambda_1, ..., \lambda_{\tilde{M}} > 0$
- Right singular vectors: $V^1, ..., V^{M} \in \mathbb{R}^M$

**Remark 1.**

1. The ordering of the columns of $Y$ in Alg. 1(iii) is arbitrary, yet entails a specific indexing of the POD coefficients. Hence, all indexing following below is to be understood with respect to this ordering.
2. By including the vectors $\{W^{i,k}\}_{i=1, ..., m, k=1, ..., d}$ in the snapshot matrix, it is ensured that the partial derivatives of the snapshots are contained in the GEPOD subspace. In fact, for the centered data, i.e. $\bar{W}^i := W^i - A, \bar{W}^{i,k} := W^{i,k} - A$ and $\tilde{M} = M - 1$, simple computations show that
   \[ W^i = \Pi(\bar{W}^i) = \sum_{j=1}^{M-1} \langle W^i, U^j \rangle U^j = \sum_{j=1}^{M-1} \sqrt{\lambda_j} V^j U^j, \] (15)
   \[ \bar{W}^{i,k} = \Pi(\bar{W}^{i,k}) = \sum_{j=1}^{M-1} \langle \bar{W}^{i,k}, U^j \rangle U^j = \sum_{j=1}^{M-1} \sqrt{\lambda_j} V^j_{i+km} U^j, \] (16)
where $\Pi$ denotes the projection onto the GEPOD subspace $\text{span}\{U^1, \ldots, U^{M-1}\}$ as introduced in (1). As a consequence

$$
\partial_k W(p_i) = \frac{1}{h_k}(\bar{W}_i^+ - \bar{W}_i^-) = \sum_{j=1}^{M-1} \sqrt{\lambda_j} \left( V^j_i - V^j_i \right) U^j_i = \Pi(\partial_k W(p_i)).
$$

(17)

3. The subspace spanned by the augmented snapshot set

$$
\text{span}\{W^i, W^i + h_k \partial_k W^i\} \quad i = 1, \ldots, m, k = 1, \ldots, d, h_k > 0
$$

is invariant under the choice of the step size vector $H = (h_1, \ldots, h_d)$ and is the same as $\text{span}\{W^i, \partial_k W^i\} \quad i = 1, \ldots, m, k = 1, \ldots, d$. Therefore, a non-compressed POD of the augmented snapshot data will lead to an orthogonal basis spanning this precise space no matter what choice of $H$ is applied, though the actual basis vectors and, more important, the weighting of the derivative information in the snapshot space may differ. A parametric study on the dependency of the GEPOD approximation on the chosen step size $H$ for a specific test case is performed in section 4.3. The influence of $H$ on the information content of the GEPOD eigenmodes is investigated in section 4.4.

4. Theoretically, it is possible to directly implement the snapshot derivatives in the snapshot matrix. Yet note that in most applications, the snapshots and their derivatives account for different physical quantities and will feature entries of different orders of magnitude, so that a combined orthogonal decomposition will be corrupted by the dominating data. This is avoided by shifting the derivatives to the space of solutions as introduced in Alg. 1(i).

5. A comparable approach of augmenting the input snapshot matrix has been proposed in [10, §2.2], but for the purpose of obtaining improved error estimates. In the aforementioned work, however, not exact derivatives but finite difference approximations of derivatives of input snapshots were utilized and a shifting as in Alg. 1(i) for compensating for the disagreement between primary data and derivative data was not considered.

6. The centering of the snapshot input is often motivated by applications to physical or engineering problems, see [10, §5.2], [7, Remark 1], [12, §2.1], [8, §2.2], [20, §2.3]. Think for example of processing physical quantities that are required to be strictly positive like, e.g. the density. The POD completely ignores this constraint. Hence, it is more feasible to apply the POD not to the density snapshots but to their fluctuations around the respective mean value, as the fluctuations may well take both positive and negative values.

Having the POD subspace representation at hand, we can use the model to compute reduced-order solutions at arbitrary parameter locations. At each parameter location $p^* \in \mathbb{R}^d$, the POD approximation of the full-order solution
\( W(p^*) \) is given by
\[
W_r(p^*) = \sum_{j=1}^{\hat{M}} a_j(p^*) U_j + A = \sum_{j=1}^{\hat{M}} (W(p^*), U_j) s U_j + A. \quad (18)
\]
Therefore, a reduced-order approximation to the unknown full-order solution \( W(p^*) \) can be obtained by estimating the coefficients \( a_j^* := a_j(p^*) \). Coefficient estimation is performed via the following

Algorithm 2 (GEPOD: Prediction).

**Input:**
- reduced dimension: \( \hat{M} \leq M - 1 = m(d + 1) - 1 \)
- GEPOD eigenmodes: \( U^1, ..., U^\hat{M} \in \mathbb{R}^n \)
- eigenvalues: \( \lambda_1, ..., \lambda_\hat{M} > 0 \)
- right singular vectors: \( V^1, ..., V^\hat{M} \in \mathbb{R}^M \)
- snapshot mean: \( A \in \mathbb{R}^n \)
- step size vector: \( H = (h_1, ..., h_d) \in \mathbb{R}^d \)
- sampled parameter locations: \( p^1, ..., p^m \in \mathbb{R}^d \)
- untried parameter location: \( p^* \in \mathbb{R}^d \)

(i) Compute sampled POD coefficients according to (15):

\[
\begin{pmatrix}
  a_1^j \\
  \vdots \\
  a_m^j
\end{pmatrix}
= \sqrt{\lambda_j}
\begin{pmatrix}
  V_1^j \\
  \vdots \\
  V_m^j
\end{pmatrix},
\quad (15)
\]

(ii) Compute derivatives of sampled POD coefficients according to (17):

\[
\partial_k a_i^j := \frac{\sqrt{\lambda_j}}{h_k} (V_{i+km}^j - V_i^j). \quad (19)
\]

(iii) Perform gradient-enhanced interpolation:

For \( j = 1, ..., \hat{M} \) do

1. fit the following data

\[
\begin{bmatrix}
  x & f(x) & \partial_1 f(x) & \ldots & \partial_d f(x) \\
p^1 & a_j^1 & \partial_1 a_j^1 & \ldots & \partial_d a_j^1 \\
\vdots & \vdots & \vdots & \ldots & \vdots \\
p^m & a_j^m & \partial_1 a_j^m & \ldots & \partial_d a_j^m \\
\end{bmatrix}
\quad (20)
\]

2. access the response surface at \( p^* \) to obtain an estimation for \( a_j^* = a_j(p^*) \).

(iv) Compute \( W^* := \sum_{j=1}^{\hat{M}} a_j^* U_j \in \mathbb{R}^n \).
Backshift by mean: $W^* = \bar{W}^* + A$

**Output:**
reduced-order approximation: $W^* \in \mathbb{R}^n$

**Remark 2.**
1. Obviously, steps (i), (ii), (iii.1) have to be performed only once, while steps (iii.2), (iv), (v) have to be performed for each new prediction. Hence, the computational effort of a surrogate prediction is given by the computational effort of evaluating $\tilde{M}$ response surface models, each based on an enhanced $m$-point data set, and subsequently computing the corresponding linear combination of POD modes.

2. For the examples presented in section 4, gradient-enhanced Kriging (GEK) was utilized in order to accomplish step (iii). An outline of this interpolation technique is beyond the scope of this work and the reader is referred to [21] for comprehensive background theory and to [22, 23], e.g., for applications in engineering. Note that Kriging models are also known as spatial Gaussian predictors or best linear unbiased predictors.

The next theorem shows that the surrogate model as constructed via the algorithms stated above features the desired properties.

**Theorem 3.1.** Let $W_r: p \mapsto W_r(p) = \sum_{j=1}^{\tilde{M}} a_j(p)U_j + A$ be the surrogate model as constructed in Algorithms 1, 2. For unreduced representation, that is $\tilde{M} = M - 1$, the objective (12) is achieved, i.e. it holds $W_r(p') = W(p')$, and $\partial_k W_r(p') = \partial_k W(p')$ for $i = 1, ..., m, k = 1, ..., d$. For reduced data, $\tilde{M} < M - 1$, it is achieved approximately, with an error estimate

$$\|W(p_i) - W_r(p')\|_S^2 \leq E, \quad \|\partial_k W(p_i) - \partial_k W_r(p')\|_S^2 \leq E,$$

where $E := \sum_{j=\tilde{M}+1}^{M-1} \lambda_j$ is the standard POD truncation error.

**Proof.** Keeping notations as introduced above, observe that rank($\bar{Y}$) $\leq M - 1$ for the centered snapshot matrix. Therefore, at most $M - 1$ eigenmodes correspond to non-zero eigenvalues, such that

$$\text{span}\{U^1, ..., U^{M-1}\} = \text{span}\{W^i, W^{i,k} | i = 1, ..., m, k = 1, ..., d\}.$$

Because of (17), it holds

$$\langle \partial_k W(p^i), U_j \rangle_S = \frac{\lambda_j}{h_k} \left(V^j_{i+k+1} - V^j_{i+k} \right). \tag{21}$$

Once computed, the POD modes $\{U_j\}_j$ do not depend on the parameter $p$. Hence

$$\partial_k W_r(p) = \sum_{j=1}^{M-1} \partial_k a_j(p)U_j.$$
By Taylor’s formula, we have for the full-order model
\[ W(p^i + te_h) = W(p^i) + t \partial_h W(p^i) + O(t^2) \in \mathbb{R}^n \]
for \( t \to 0 \). Since \( W_r(p^i + te_h) = \Pi(W(p^i + te_h)) \), it holds for \( t \to 0 \):
\[ a_j(p^i + te_h) = \langle W(p^i + te_h), U_j \rangle_S = \langle W(p^i), U_j \rangle_S + t \langle \partial_h W(p^i), U_j \rangle_S + O(t^2). \]

It follows
\[ \frac{1}{t} (a_j(p^i + te_h) - a_j(p^i)) = \frac{1}{t} (t \langle \partial_h W(p^i), U_j \rangle_S) + O(t) \]
\[ = \langle \partial_h W(p^i), U_j \rangle_S + O(t). \]

Taking the limit \( t \to 0 \) on both sides leads to
\[ \partial_h a_j(p^i) = \langle \partial_h W(p^i), U_j \rangle_S. \] (22)

By (21) and steps (ii), (iii) in Algorithm 2, the main claim follows. The error estimate is standard, cf. [10, eq. 22 & Prop. 1].

It is worth emphasizing that (22) does not depend explicitly on the step size vector \( H \), which has been used to implement the derivative information in the snapshot matrix. The POD modes \( \{U_j\} \), however, differ for different choices of \( H \), but not the subspace they are spanning.

3.2. Primary analysis

In the following simple observation, the theoretical error behavior of the POD- and the GEPOD-interpolation models are compared. Let \( W_r^{POD} \) and \( W_r^{GEPOD} \) denote the surrogate models obtained via POD coefficient interpolation and gradient-enhanced POD coefficient interpolation, respectively. Let \( q \in \mathbb{R}^d \) be a small displacement vector.

**Observation 1.** For \( W_r \in \{W_r^{POD}, W_r^{GEPOD}\} \) and \( i = 1, \ldots, m \), it holds
\[ \|W(p^i + q) - W_r(p^i + q)\| \leq \|W(p^i) - W_r(p^i)\| + \|D_W p^i q - (D_W)p^i q\| + O(\|q\|^2) \]
\[ \leq \begin{cases} E + O(\|q\|), & \text{if } W_r = W_r^{POD} \\ (1 + \|q\|_1)E + O(\|q\|^2), & \text{if } W_r = W_r^{GEPOD} \end{cases}, \]
where \( E \) is as introduced in Theorem 3.1, \( \|q\|_1 = \sum_{k=1}^d |q_k| \) and \( \| \cdot \| \) is an arbitrary vector norm. Note that \( E \) cancels out in both cases, if all POD modes corresponding to non-zero eigenvalues are kept. If \( \|q\| < \frac{1}{E} \) holds for the displacement, then \( \frac{1}{4} \|q\| > \|q\|^2 \), so that the error of the POD model is asymptotically larger than the error of the GEPOD model even at half the distance to the closest sample location.
Next light will be shed on the role of the step size vector \( H = (h_1, ..., h_d) \) introduced in Algorithms 1, 2. From Theorem 3.1, it is clear that \( H \) does not affect the accuracy of the derivatives of the GEPOD ROM unless (19) degenerates numerically.

Let \( \Psi := (W^1, ..., W^m) \in \mathbb{R}^{n \times m} \) be the matrix of primary snapshots only and let \( Y \) be as introduced in Algorithm 1. Writing \( \partial_k \Psi = (\partial_k W^1, ..., \partial_k W^m) \in \mathbb{R}^{n \times M} \), we have

\[
Y = (\Psi, ..., \Psi) + (0^{n \times m}, h_1 \partial_1 \Psi, 0^{n \times m}, ..., 0^{n \times m}) + ... + (0^{n \times m}, ..., 0^{n \times m}, h_d \partial_d \Psi),
\]

so that \( Y = (\Psi, ..., \Psi) + O(\|H\|_1) \in \mathbb{R}^{n \times M} \).

The mean value vector \( A \) of the enhanced data from (14) can be written as

\[
A = \frac{1}{M} \left( (d+1) \sum_{i=1}^m W^i + (DW_{p^1} + ... + DW_{p^m}) H \right) = A^m + O(\|H\|_1),
\]

where \( A^m = \frac{1}{M} \sum_{i=1}^m W^i \) denotes the mean of the primary snapshots. Let \( \overline{\Psi} = \Psi - A^m (1^M)^T = (W^1 - A^m, ..., W^m - A^m) \). Then, for \( \bar{Y} \) as in (14),

\[
\bar{Y} = (\overline{\Psi}, ..., \overline{\Psi}) + O(\|H\|_1) \in \mathbb{R}^{n \times M}.
\]  

(23)

It follows

\[
\bar{Y} \bar{Y}^T = (d+1) \overline{\Psi} \overline{\Psi}^T + O(\|H\|_1) \in \mathbb{R}^{n \times n}
\]  

(24)

and

\[
\bar{Y}^T \bar{Y} = \left( \begin{array}{ccc}
\overline{\Psi}^T \overline{\Psi} & \cdots & \overline{\Psi}^T \overline{\Psi} \\
\vdots & \ddots & \vdots \\
\overline{\Psi}^T \overline{\Psi} & \cdots & \overline{\Psi}^T \overline{\Psi}
\end{array} \right) + O(\|H\|_1) \in \mathbb{R}^{M \times M}.
\]  

(25)

If \( v^j \) denotes the \( j \)th normalized eigenvector of \( \Psi^T \overline{\Psi} \) corresponding to the eigenvalue \( \lambda_j \), then the \( j \)th normalized eigenvector of the matrix summand independent of \( H \) in (25) is obviously \( V^j = \sum_{k=1}^{d+1} (v^j)^T, ..., (v^j)^T \) \( \in \mathbb{R}^M \), its eigenvalue being \((d+1)\lambda_j\). Note that by the theorem of Wielandt-Hoffman [24, Theorem 8.1.4], the eigenvalues of a matrix \( M \) perturbed by a symmetric matrix \( \Sigma \) do not move more than \( \|\Sigma\| \), more precisely \( \sum_j (\lambda_j(M + \Sigma) - \lambda_j(M))^2 \leq \|\Sigma\|^2_F \). Bearing in mind (8) and (15), we deduce

**Observation 2.** For step size vectors \( H \) of small norm, the gradient-enhanced POD eigenvalue problem occurring in Algorithm 1(v) is but a perturbed version of the corresponding standard POD eigenvalue problem for the primary snapshot data. Therefore, the first \( m-1 \) POD eigenmodes as well as the associated set of sampled coefficients obtained through the GEPOD approach coincide with the eigenmodes and sampled coefficients of the standard POD approach up to the perturbation. The eigenvalues coincide up to the perturbation and a multiplicative factor of \( d+1 \). The level of perturbation is controlled by the step size vector \( H \) required as a user input to Algorithms 1.
The above observation’s predication is illustrated by Figure 3 and Figure 6, where the sample points (computed via the entries of the right singular vectors) virtually coincide. An experimental parametric study on the influence of the step size on the prediction is carried out in Section 4.3.

3.3. Computational Costs

Assuming \( n \gg m \), the computational complexity of the GEPOD method can be estimated as follows. When following the method of snapshots, the complexity of Algorithm 1 is dominated by solving the EVD problem (10) and computing the large POD basis modes via (11). The \( m(d+1) \times m(d+1) \) EVD problem is usually solved via iterative methods, which approximately take \( O(m(d+1)) \) flops [25]. (For a definition of a flop, see [24, §1.2.4].) For a diagonal weights matrix \( S \), computing \( Y^T S^T SY \) and (11) takes \( O(n(m(d+1))^2) \) flops. Hence the total computational complexity of Algorithm 1 is \( O(nm^2(d+1)^2) \).

In Algorithm 2, \( \tilde{M}m < m^2(d+1) \) flops are due in step (i) and \( 3\tilde{M}md < 3m^2(d^2+d) \) flops are due in step (ii). The complexity of step (iii) depends on the chosen response surface method. If the method of choice is GEK, the corresponding predictor function has to be trained depending on model hyperparameters [21]. For each iteration step in training a GEK model, an equation system of dimension \( m(d+1) \) must be solved [22], leading to \( O((m(d+1))^3) \) flops for building the GEK predictor for each POD coefficient \( a_j, j = 1, ..., \tilde{M} \), when assuming a constant upper bound on the total number of training iterations. For evaluating the GEK predictor at an untried location, essentially a standard scalar product of dimension \( m(d+1) \) has to be evaluated. Performing this step for each POD coefficient accumulates to \( O(m^3(d+1)^2) \) flops. Building the POD series and the backshifting in the two final steps (iv), (v) of Algorithm 2 take \( 3n\tilde{M} = O(nm(d+1)) \) flops.

On summarizing, we note

**Observation 3.** Let \( m \in \mathbb{N} \) snapshots of size \( n \in \mathbb{N} \) depending on \( d \in \mathbb{N} \) parameters be given. If \( n \gg m \), the computational effort of building the GEPOD ROM is \( O(nm^2(d+1)^2) + O((m(d+1))^3) \). Once built, evaluating the ROM is of complexity \( O(nm(d+1)) \). Hence, both model building and evaluating scale linearly in \( n \).

In most practical applications, the dimension \( n \) of the snapshots will be defined by the discretization of a continuous problem on a large computational grid (up to several million points for CFD), while the number of solution snapshots, \( m \) is in the range of tens to hundreds, so that the effort expressed in \( m \) and even in \( m(d+1) \) is insignificant.

For a comparison to the computational effort of standard POD-Galerkin ROMs the reader is referred to [12, §6].
4. An Academic but Non-Trivial Example

In order to demonstrate the GEPOD approach, we present an application to an analytic model function. Despite its academic nature, it mimics some relevant features of real-life applications of POD based reduced-order modeling to PDE systems. In particular, the model function is chosen such that the solution snapshots exhibit some similarities to solutions of the convection-diffusion problem in fluid dynamics [13].

4.1. Setting

Let \( f : [0,10] \times [0,10] \to \mathbb{R}, (x, \omega) \mapsto x^2 \sin (\frac{\pi}{10} \omega x) \). We define the full-order model (FOM) to be

\[
F : [0,10] \times [0,10] \to \mathbb{R}, (x, \omega) \mapsto \frac{f(x, w)}{\|f(\cdot, w)\|_{L^2}},
\]

where \( \| \cdot \|_{L^2} \) denotes the norm induced by the scalar product

\[
\langle g_1, g_2 \rangle_{L^2} = \int_0^{10} g_1(x) g_2(x) dx
\]
on \( L^2([0,10]) \). In particular \( \|f(\cdot, w)\|_{L^2} = \left( \int_0^{10} f(x, \omega)^2 dx \right)^{1/2} \). The function \( F \) is displayed in Figure 1. The partial derivative of \( F \) with respect to \( \omega \) is given by

\[
\partial_{\omega} F(x, \omega) = \frac{1}{\|f(\cdot, w)\|_{L^2}} \partial_{\omega} f(x, \omega) - \frac{\langle f(\cdot, \omega), \partial_{\omega} f(\cdot, \omega) \rangle_{L^2}}{\|f(\cdot, w)\|_{L^2}^3} f(x, w),
\]
where \( \partial_\omega f(x, \omega) = \frac{\pi}{10} x^3 \cos(\frac{\pi}{10} \omega x) \). Although given in this example, we emphasize that differentiability with respect to \( x \) is not required by the method.

The analogy to real-life applications is as follows: The variable \( x \in [0, 10] \) represents the spatial coordinate, while the variable \( \omega \in [0, 10] \) mimics a design variable or a parameter condition. The trajectory \( F(\cdot, w) \) corresponds to the solution of a PDE at conditions \( \omega \). In Computational Fluid Dynamics (CFD), for example, a numerical solutions to the Navier-Stokes Equations governing the flow around an aerodynamic body at certain flow conditions is sought after. In this case, \( x \) gives the spatial coordinate, while \( \omega = (\omega_1, ..., \omega_d) \) specifies flow conditions such as combinations of Mach number, angle of attack, Reynolds number etc..

Since POD corresponds to optimizing projection lengths, see (5), vectors of large norm will dominate in the POD. In real-life applications, however, the PDEs in question will most certainly represent physical conservation laws, such that snapshot solutions automatically undergo a normalization process. To mimic this effect, the normalization with respect to the spatial coordinate was introduced in (26). Yet, note that while \( \|F(\cdot, w)\|_{L^2([0,10])} = 1 \), generally \( \|F\|_{L^2([0,10]\times[0,10])} \neq 1 \).

The ROM is built relying on \( m = 6 \) snapshot solutions of the FOM at \((\omega_1, ..., \omega_m) = (1, 2, 3, 4, 5, 6), \) each snapshot being a 1D-function \( F_{\omega_j} := F(\cdot, \omega_j) : [0,10] \to \mathbb{R}, x \mapsto F(x, \omega_j), \ j = 1, ..., 6. \)

A snapshot can be compared to a trajectory of a PDE at \( \omega \). The snapshots are displayed in Figure 2. The shape and the number of oscillations of the trajectory function changes significantly when going from one snapshot to the next. Also note that since we sampled at integer values of \( \omega \) only, the resulting snapshots all attain 0 at the upper boundary of the spatial interval \([0,10]\), rendering a
prediction based on interpolation very difficult in the corresponding region. In this regard, the snapshots are ill-chosen, so that building a ROM upon this input is challenging. Optimal sample data selection is beyond the scope of this work.

4.2. From Continuous Problems to Discrete Solutions

![Figure 3: Eigenvalues corresponding to the centered GEPOD snapshot data and the centered POD snapshot data. For comparison purpose, the latter ones are multiplied by a factor of \((d + 1) = 2\), see Observation 2. For the GEPOD, a step size of \(h_1 = 0.001\) has been chosen in Algorithm 1.](image)

The POD method can be formulated for infinite dimensional function spaces, see [26] for a comprehensive treatise. However, practical applications of POD in the context of reduced-order modeling almost always start with snapshots of discretized solutions to PDEs.

To continue the example, we discretize the spatial dimension \([0.01, 10]\) by a grid featuring \(n = 1000\) equidistant points \(\{x_i = iv\mid i = 1, \ldots, 1000\}\), where \(v = 0.01\) is the grid cell volume. Hence, the interval \([0.01, 10]\) is identified with the vector \((x_1, \ldots, x_n) \in \mathbb{R}^n\) and for \(j = 1, \ldots, 6\), the function \(F_{\omega_j} : [0.01, 10] \to \mathbb{R}\) is identified with its discrete analogon

\[W^j = \left(W^j_1, \ldots, W^j_n\right)^T := (F_{\omega_j}(x_1), \ldots, F_{\omega_j}(x_1))^T \in \mathbb{R}^n.\]

In the same way, the discrete vector \(\partial_{x_i} W^k \in \mathbb{R}^n\) is defined based on (27).

We approximate the \(L_2\) scalar product by a Riemannian sum

\[\langle F_{\omega_j}, F_{\omega_i} \rangle_{L_2} \approx \sum_{i=1}^{n} vW^j_i W^i_i = \langle W^j, W^i \rangle_S,\]

where \(S = \sqrt{\Omega} I^{n \times n} \in \mathbb{R}^{n \times n}\). Moreover, we set \(h_1 = 10^{-3}\) and \(\epsilon = 0\). Now, we apply Algorithm 1 to the input data \(\{S, W^1, \ldots, W^n, \partial_{x_i} W^1, \ldots, \partial_{x_i} W^n, h_1, \epsilon\}\).

As a result, an order–\(M\) ROM of the order–\(n\) FOM is constructed, where \(M = 2m - 1 = 11\) and \(n = 1000\). Subsequently, we use Algorithm 2 to compute
Figure 4: Surface plot of the full-order model $F(x, \omega)$ for $(x, \omega) \in [0.01, 10] \times [1, 6]$ (FOM, middle) compared to the gradient-enhanced POD-based ROM approximation (FGEPOD, right) and the standard POD-based ROM approximation (FPOD, left) in the same region. Here, a step size of $h = 0.001$ was chosen in order to enhance the POD subspace by derivative information, see eq. (13).

A reduced-order representation of the FOM, denoted by $F_{\text{GEPOD}}$ and compare the results to the standard POD-based interpolation model, without derivative information, denoted by $F_{\text{POD}}$. The standard POD predictor features the maximum of 5 basis modes (plus the vector of averages). Due to the centering, the 6th mode of the POD basis and the 12th mode of the GEPOD basis, respectively, are associated to a singular value of exactly zero, cf. Fig. 3. For GEPOD the GEK response surface method and for POD the Kriging response surface method was applied. In both cases the cubic spline correlation functions and Hooke & Jeeves direct search model parameter training were used accordingly for data fitting, for details on Kriging and GEK see [22] and references therein.

Figure 3 shows the eigenvalues defining the relative information content associated with the centered snapshot sets in question. For creating the surface plots displayed in Figure 4, the functions were evaluated on a $1000 \times 1000$ grid in $[0.01, 10] \times [1, 6]$. The enhanced $F_{\text{GEPOD}}$ approximation, (Fig.4, right) matches the FOM better than the standard POD model $F_{\text{POD}}$, (Fig.4, left), especially in the regions close to the boundary $\omega = 6$, where $F_{\text{POD}}$ misses the trend of the FOM completely. For better judging the accuracy of the results, the ROM has been tried at $\omega_l = 3.565065$ and $\omega_r = 5.501401$ and the resulting trajectories are again compared to the standard POD-based interpolation model. The approximate functions are displayed in Figure 5. The approximation errors with respect to the exact full-order solution are given in Table 1. Depending on the chosen norm, the GEPOD approximation error is about two to five times lower than the POD approximation error. The differences between the gradient-enhanced and the standard predictor show clearly on the curves which were obtained by
Figure 5: FOM reference trajectory ($F_{\text{ref}}$) compared to the gradient-enhanced POD-based ROM prediction ($F_{\text{GEPOD}}$) and the standard POD-based ROM prediction ($F_{\text{POD}}$) at $\omega_l = 3.565065$ (left-hand side) and at $\omega_r = 5.501401$ (right-hand side). Here, a step size of $h_1 = 0.001$ was chosen in regard of eq. (13).

<table>
<thead>
<tr>
<th>Error</th>
<th>$| \cdot |_{L_2}$</th>
<th>$| \cdot |_{L_1}$</th>
<th>$| \cdot |<em>{L</em>{\infty}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F(\cdot, \omega_l) - F_{\text{GEPOD}}(\cdot, \omega_l)$</td>
<td>0.204</td>
<td>0.381</td>
<td>0.169</td>
</tr>
<tr>
<td>$F(\cdot, \omega_l) - F_{\text{POD}}(\cdot, \omega_l)$</td>
<td>0.638</td>
<td>0.814</td>
<td>0.922</td>
</tr>
<tr>
<td>Ratio:</td>
<td>3.3</td>
<td>2.3</td>
<td>5.6</td>
</tr>
<tr>
<td>$F(\cdot, \omega_r) - F_{\text{GEPOD}}(\cdot, \omega_r)$</td>
<td>0.172</td>
<td>0.305</td>
<td>0.198</td>
</tr>
<tr>
<td>$F(\cdot, \omega_r) - F_{\text{POD}}(\cdot, \omega_r)$</td>
<td>0.847</td>
<td>1.266</td>
<td>0.997</td>
</tr>
<tr>
<td>Ratio:</td>
<td>4.9</td>
<td>4.1</td>
<td>5.0</td>
</tr>
</tbody>
</table>

Table 1: Approximation errors of the GEPOD and POD ROM solutions displayed in Fig. 5 for $\omega_l = 3.565065$, $\omega_r = 5.501401$. The norms have been computed with respect to the discretization introduced in Section 4.2.
fitting the sample data for the 4th and 5th POD coefficient, see Figure 6.

### 4.3. Parametric Study in the Step Size

In order to shift the snapshot derivative vectors to the space of the snapshots, a step size vector has been introduced in Alg. 1(ii), eq. (13), mocking a finite difference approximation. In this section, the influence of this vector is investigated via a parametric study. For the one-parameter problem at hand, the analogue to (13) reads

\[ F_{\omega_i + 1} := F_{\omega_i} + h \partial_{\omega_i} F(x, \omega_i). \]

The experiment of predicting the trajectories at \( \omega = 3.565065 \) and \( \omega = 5.501401 \) has been repeated for the following choices of step sizes

\[ h \in \{10^k | k = 1, 0, -1, -2, -3, -4, -5, -6\}. \]

The resulting trajectories are displayed in Figure 7. The corresponding approximation errors are listed in Table 2.

**Observation 4.** While the step size \( h \) ranges over eight orders of magnitude, the GEPOD approximations are hardly distinguishable and hence only weakly dependent on the step size.

This observation is supported theoretically by the following facts:

1. When refraining from truncating the GEPOD basis, the GEPOD subspace is invariant under the choice of the step size, see Remark 1.3.
Figure 7: Comparison of the gradient-enhanced POD-based ROM approximations for different step sizes, see Alg. 1(i), with the FOM reference trajectory at $\omega_l = 3.565065$ (left-hand side) and at $\omega_r = 5.501401$ (right-hand side). Note that while the step size ranges over 8(!) orders of magnitude, the various approximation curves almost coincide showing that the method is only weakly sensitive to the choice of the step size.

2. The derivatives of the GEPOD coefficients to be interpolated do not depend on the step size, see (22). If the snapshot derivatives are exact, then so are the derivatives of the GEPOD coefficients with respect to the corresponding GEPOD basis. Even when the step sizes tend to zero, the exact derivative information is preserved, up to numerical error, as the reciprocal values tend to infinity, see (19).

Different step sizes lead to different weightings of the information in the snapshot subspace and therefore to different eigenmodes and right singular vectors, which define the sampled values for the GEPOD coefficients, see Algorithm 2, (i), (ii). In Figure 8, the interpolated curves obtained via a gradient-enhanced Kriging for the fourth GEPOD coefficients are displayed for various choices of the step size $h$. As can be seen from this figure, the actual values of the sampled coefficients and the shape of the interpolated curves may vary considerably for different choices of $h$, yet Figure 7 suggests that the linear expansion with the associated eigenmodes compensates for this effect.

4.4. Some remarks on truncating the GEPOD basis

In this section, the GEPOD predictor’s accuracy is investigated for different truncation levels of the GEPOD basis. More precisely, the predictor using the maximum of $M - 1 = 11$ basis modes is compared to the predictors obtained when truncating the basis to numbers of 5 and 6 modes respectively. The threshold of 5 modes is chosen, because it is precisely the maximum number of independent eigenmodes that the corresponding standard POD basis excluding
Figure 8: Examples of interpolation curves obtained for the fourth coefficient in the GEPOD basis expansion for various step sizes $h$. For step sizes $h = 0.01$ and smaller, the curves virtually coincide.

derivative information can feature at the most. Moreover, the comparison is performed in interaction with the step size parameter $h$ from eq. (13).

The graphs of the resulting functions are displayed in Fig. 9. As can be seen from this figure, for step sizes of $h = 0.1$ and $h = 0.001$, the predictors built on 6 modes and the maximum of 11 modes virtually coincide. This is confirmed by the corresponding $L_2$-errors shown in Table 3. When truncating the basis to a number of 5 modes, the corresponding predictor misses the trend of the reference function near the right boundary of the interval. In contrast, for a comparably large step size of $h = 10.0$ a number of 5 modes is sufficient to capture the trend in this region.

Remember that all snapshots feature a zero at the right boundary. Hence, the prediction accuracy closeby can only be improved by exploiting derivative information. Therefore, we deduce that for a step size of $h = 10.0$, the first 5 GEPOD modes already capture some of the information contained in the derivatives, while for the smaller step sizes $h = 0.1$ and $h = 0.001$, the derivative information is contained in higher-order modes. This observation does not come as a surprise but is actually in line with the theoretical investigations summarized in Observation 2: If the step size is small, the GEPOD eigenvalue problem is but a perturbed version of the corresponding POD eigenvalue problem. As a consequence, the first 5 GEPOD eigenmodes resemble the first 5 standard POD
Prediction at $\omega_l = 3.565065$

<table>
<thead>
<tr>
<th>Error</th>
<th>$10^1$</th>
<th>$10^0$</th>
<th>$10^{-1}$</th>
<th>$10^{-2}$</th>
<th>$10^{-3}$</th>
<th>$10^{-4}$</th>
<th>$10^{-5}$</th>
<th>$10^{-6}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$|\cdot|_{L_2}$</td>
<td>0.207</td>
<td>0.207</td>
<td>0.204</td>
<td>0.204</td>
<td>0.204</td>
<td>0.204</td>
<td>0.204</td>
<td>0.204</td>
</tr>
<tr>
<td>$|\cdot|_{L_1}$</td>
<td>0.432</td>
<td>0.409</td>
<td>0.377</td>
<td>0.381</td>
<td>0.381</td>
<td>0.381</td>
<td>0.381</td>
<td>0.381</td>
</tr>
<tr>
<td>$|\cdot|<em>{L</em>{\infty}}$</td>
<td>0.157</td>
<td>0.158</td>
<td>0.172</td>
<td>0.169</td>
<td>0.169</td>
<td>0.169</td>
<td>0.169</td>
<td>0.169</td>
</tr>
</tbody>
</table>

Prediction at $\omega_l = 5.501401$

<table>
<thead>
<tr>
<th>Error</th>
<th>$10^1$</th>
<th>$10^0$</th>
<th>$10^{-1}$</th>
<th>$10^{-2}$</th>
<th>$10^{-3}$</th>
<th>$10^{-4}$</th>
<th>$10^{-5}$</th>
<th>$10^{-6}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$|\cdot|_{L_2}$</td>
<td>0.147</td>
<td>0.152</td>
<td>0.174</td>
<td>0.172</td>
<td>0.172</td>
<td>0.172</td>
<td>0.172</td>
<td>0.172</td>
</tr>
<tr>
<td>$|\cdot|_{L_1}$</td>
<td>0.277</td>
<td>0.275</td>
<td>0.310</td>
<td>0.306</td>
<td>0.305</td>
<td>0.305</td>
<td>0.305</td>
<td>0.305</td>
</tr>
<tr>
<td>$|\cdot|<em>{L</em>{\infty}}$</td>
<td>0.143</td>
<td>0.155</td>
<td>0.195</td>
<td>0.198</td>
<td>0.198</td>
<td>0.198</td>
<td>0.198</td>
<td>0.198</td>
</tr>
</tbody>
</table>

Table 2: Approximation errors of the GEPOD approximations w.r.t. the FOM solution displayed in Fig. 4.3 for $\omega_l = 3.565065$, $\omega_r = 5.501401$. The norms have been computed with respect to the discretization introduced in Section 4.2.

Figure 9: FOM reference trajectory ($F_{\text{ref}}$) compared to the gradient-enhanced POD-based predictor ($F_{\text{GEPOD}}$) using all 11 modes, the GEPOD predictor truncated to a number of 5 basis modes ($F_{\text{GEPOD}_5}$) and the GEPOD predictor truncated to a number of 6 basis modes ($F_{\text{GEPOD}_6}$) at $\omega_l = 3.565065$. As step sizes for eq. (13) were applied $h = 10.0$ (left), $h = 0.1$ (middle) and $h = 0.001$ (right).
Table 3: Approximation errors at $\omega_0 = 3.565065$ of the GEPOD predictors using a maximum of 11 modes and a number of 5 and 6 modes respectively in interaction with different step sizes $h$ applied for eq. (13) in Algorithm 1. The corresponding graphs are displayed in Fig. 9.

eigenmodes and the derivative information is essentially contained in the higher-order GEPOD modes. In this regard, choosing larger step sizes as an input to Algorithm 1 may be beneficial. We emphasize again that, theoretically, there is no need to apply particularly small step sizes.

5. Summary and Final Remarks

A new POD-based approach for enhanced surrogate modeling by incorporating gradient information, termed GEPOD, has been developed. In contrast to the existing POD-based interpolation approach for non-linear reduced-order modeling, the method presented here leads to an approximation accuracy of order one at the sample locations, meaning that both the snapshot inputs and the derivatives of the snapshots of the FOM are reproduced by the GEPOD predictor up to the relative information content of the reduced-order basis. The GEPOD method has been demonstrated on an academic test case trying to mimic the main phenomena of applications of reduced-order modeling to systems of PDEs. The prediction quality was shown to be significantly higher when compared to the standard interpolation-based POD approach. The rather obvious conclusion is that, when derivative information is available, it should be made use of.

Based on a parametric study the GEPOD predictor has been demonstrated to be robust with respect to changes in the main input parameters. Given the required input data, the additional effort of performing a GEPOD prediction rather than a standard POD prediction is usually negligible. Considering real-life applications, the costs of computing snapshot derivatives have to be weighted against the costs of computing a finer snapshot set. This trade-off will be investigated in future work. Note that in the recent paper [14], the advantages of exploiting both snapshot data and partial derivatives at the given snapshot locations have been successfully demonstrated in the context of projection-based ROMs.

Acknowledgments

This research was sponsored in part by the European Regional Development Fund, Economic Development Fund of the Federal German State of Lower Saxony, contract/grant number: W3-80026826, and the German Federal Ministry of Economics and Technology (BMWi).
References


25