Interpolation among reduced-order matrices to obtain parameterized models for design, optimization and probabilistic analysis

Joris Degroote[1,2], Jan Vierendeels[1] and Karen Willcox[2,*,†]

[1]*Department of Flow, Heat and Combustion Mechanics, Ghent University, Sint-Pietersnieuwstraat 41, B-9000 Ghent, Belgium*
[2]*Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 77 Massachusetts Ave, Room 37-447, Cambridge, MA 02139, U.S.A.*

SUMMARY

Model reduction has significant potential in design, optimization and probabilistic analysis applications, but including the parameter dependence in the reduced-order model (ROM) remains challenging. In this work, interpolation among reduced-order matrices is proposed as a means to obtain parameterized ROMs. These ROMs are fast to evaluate and solve, and can be constructed without reference to the original full-order model. Spline interpolation of the reduced-order system matrices in the original space and in the space tangent to the Riemannian manifold is compared with Kriging interpolation of the predicted outputs. A heuristic criterion to select the most appropriate interpolation space is proposed. The interpolation approach is applied to a steady-state thermal design problem and probabilistic analysis via Monte Carlo simulation of an unsteady contaminant transport problem. Copyright © 2009 John Wiley & Sons, Ltd.

1. INTRODUCTION

High-fidelity numerical simulation tools have become indispensable for the analysis of complex engineering systems, but remain too time-consuming for many design and optimization applications—settings that require a large number of model evaluations. Since output quantities of interest are generally restricted to a small number of solution functionals (such as lift and

---

*Correspondence to: Karen Willcox, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 77 Massachusetts Ave, Room 37-447, Cambridge, MA 02139, U.S.A.
†E-mail: kwillcox@mit.edu

drag, or displacement and stress at a point), a reduced-order model (ROM) can often be employed to provide accurate estimates of output quantities of interest at greatly reduced computational cost. ROMs are derived from high-fidelity models using a generalizable, systematic procedure (as opposed to reduced-physics models that are created manually for each problem); the challenge in the design/optimization setting is to include parameter dependence in the ROM in a way that leads to accurate results and computationally efficient models. This paper presents an interpolation approach to achieve this goal. In this work, we consider both steady and unsteady models. A general steady model is written as

$$\mathbf{A}(\mathbf{z})\mathbf{x} = \mathbf{b}(\mathbf{z}) \tag{1a}$$

$$\mathbf{y} = \mathbf{C}(\mathbf{z})\mathbf{x} \tag{1b}$$

and an unsteady model as

$$\dot{\mathbf{x}}(t) = \mathbf{A}(\mathbf{z})\mathbf{x}(t) + \mathbf{B}(\mathbf{z})\mathbf{u}(t) \tag{2a}$$

$$\mathbf{y}(t) = \mathbf{C}(\mathbf{z})\mathbf{x}(t) \tag{2b}$$

with t the time and a dot indicating a time derivative. The state vector is indicated by $\mathbf{x} \in \mathbb{R}^n$, the input and output of the model are $\mathbf{u} \in \mathbb{R}^p$ and $\mathbf{y} \in \mathbb{R}^q$, respectively, and $\mathbf{z} \in \mathbb{R}^m$ denotes the parameter vector. While the systems (1) and (2) have a general form, we are particularly interested in those systems that arise from spatial discretization of partial differential equations. In this case, the system matrices $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times p}$, and $\mathbf{C} \in \mathbb{R}^{q \times n}$, and vector $\mathbf{b} \in \mathbb{R}^n$ arise from the particular spatial discretization scheme and the boundary conditions, and the dimension of the system, n, is very large. In this paper, all models considered are linear in the state but have general nonlinear dependence on the parameter vector $\mathbf{z}$. This assumption of state linearity does not limit our approaches, but rather permits us to focus on addressing the question of nonlinear parametric dependencies. Our interpolation approach is extensible to fully nonlinear systems, although it would need to be combined with an efficient method for evaluating the terms involving nonlinear state dependence (such as the Empirical Interpolation Method of [1, 2]).

Projection-based model reduction methods derive a ROM projecting the governing equations onto the subspace spanned by a set of basis vectors [3]. The basis vectors can be calculated with several techniques, including the proper orthogonal decomposition (POD) [4, 5], truncated balanced realization [6], and Krylov subspace methods [7, 8]. To be effective, ROMs for a design or optimization problem must be both accurate over the entire parameter space of interest to the designer or optimizer and computationally efficient to solve. One challenge is to derive a set of basis vectors that span the parameter space of interest. A second challenge is to incorporate the parametric dependence in the ROM in such a way that solutions can be generated in a computationally efficient manner. As pointed out by Farhat and Amsallem [9] and others, for design and optimization the computational cost to generate a ROM (if parameter dependence is not handled carefully) can outweigh the time gained by using ROMs, in spite of their fast evaluation. For example, if evaluation of the ROM at a new parameter point requires evaluating the full-order system matrices, projecting onto the reduced subspace, and subsequently solving the ROM, then the computational cost may still be significant compared with the original full-order model.

Several approaches have been proposed for deriving a basis that spans the parameter space, including POD and Krylov-based sampling methods [10–12], reduced basis methods combined with the use of error estimates and adaptivity [13], and optimization-based approaches [14]. In most

cases, these approaches exploit the specific structure of the parametric dependence to derive the ROM; however, when the dependence of the system matrices on the parameters is a general (possibly unknown) nonlinear function, a further challenge is efficient in the computation and evaluation of the ROM. In the next section we provide an overview of approaches to address this challenge in the context of a projection-based model reduction framework. Our proposed interpolation techniques are then described in Section 3, and a heuristic algorithm to select an appropriate interpolation method is proposed in Section 4. In Section 5, we present results for steady and unsteady problems and compare the methods. Finally, Section 6 concludes the paper.

2. PARAMETERIZED ROMs

In this section, the projection-based model reduction approach is described. An overview of existing methods for incorporating parametric dependence in ROMs is given, with a brief discussion of the relative advantages and disadvantages of the various approaches.

In this work, the basis is computed using the POD method of snapshots [5]. The full-order system (1) or (2) is solved for n_z different parameter vectors, $\mathbf{z}^i$, $i = 1, \ldots, n_z$. The resulting state solutions (referred to as snapshots) are collected in the columns of the matrix $\mathbf{X} \in \mathbb{R}^{n \times n_s}$

$$\mathbf{X} = [\mathbf{x}^1 \ \mathbf{x}^2 \ \ldots \ \mathbf{x}^{n_s}] \tag{3}$$

where $\mathbf{x}^i$ is the ith snapshot and n_s is the total number of snapshots. For the steady case, we obtain one snapshot for each parameter sample, $n_s = n_z$, while for the unsteady case we collect $n_s = n_t n_z$ snapshots, where n_t is the number of snapshots for each unsteady simulation. The POD specifies the basis to be the left singular vectors of the matrix $\mathbf{X}$ corresponding to the largest singular values. A basis of dimension, n_r, is contained in the matrix $\mathbf{V} \in \mathbb{R}^{n \times n_r}$

$$\mathbf{V} = [\mathbf{v}^1 \ \mathbf{v}^2 \ \ldots \ \mathbf{v}^{n_r}] \tag{4}$$

where $\mathbf{v}^i$ is the ith basis vector.

To derive the ROM, the full-order state is approximated as a linear combination of the $n_r \ll n$ basis vectors $\mathbf{v}^i$

$$\mathbf{x} \approx \mathbf{V} \mathbf{x}_r \tag{5}$$

The reduced state, $\mathbf{x}_r$, thus represents the coordinates of the full-order state with respect to the basis consisting of the columns of $\mathbf{V}$. The ROMs are created by projecting Equation (1a) or (2a) on the subspace spanned by the columns of a left projection matrix $\mathbf{W} \in \mathbb{R}^{n \times n_r}$. This yields the ROM for the general steady model (1) as

$$\mathbf{A}_r(\mathbf{z}) \mathbf{x}_r = \mathbf{b}_r(z) \tag{6a}$$

$$\mathbf{y}_r = \mathbf{C}_r(\mathbf{z}) \mathbf{x}_r \tag{6b}$$

where $\mathbf{A}_r(\mathbf{z}) = \mathbf{W}^T \mathbf{A}(\mathbf{z}) \mathbf{V}$, $\mathbf{b}_r(\mathbf{z}) = \mathbf{W}^T \mathbf{b}(\mathbf{z})$ and $\mathbf{C}_r(\mathbf{z}) = \mathbf{C}(\mathbf{z}) \mathbf{V}$. The ROM of the unsteady model (2) is given by

$$\dot{\mathbf{x}}_r = \mathbf{A}_r(\mathbf{z}) \mathbf{x}_r + \mathbf{B}_r(\mathbf{z}) \mathbf{u} \tag{7a}$$

$$\mathbf{y}_r = \mathbf{C}_r(\mathbf{z}) \mathbf{x}_r \tag{7b}$$

where $\mathbf{B}_r(\mathbf{z}) = \mathbf{W}^T\mathbf{B}(\mathbf{z})$, $\mathbf{A}_r$ and $\mathbf{C}_r$ are defined as for the steady case, and $\mathbf{W}$ is chosen so that $\mathbf{W}^T\mathbf{V} = \mathbf{I}$.

The challenge of efficient construction and solution of the ROM over the parametric input space can be seen in the definitions of the reduced matrices in (6) and (7). Unless special treatment is given to the parametric dependence, at each new parameter point $\mathbf{z}$, solution of the ROM requires evaluating the full-order system matrices, projecting those matrices onto the reduced subspace to compute the reduced-order matrices, and then solving the resulting ROM. Since many elements of these computations depend on n, the dimension of the full-order system, in general this process will not be computationally efficient. Ly and Tran propose interpolation of the reduced state directly using cubic spline interpolation, and show good results for predicting the steady-state temperature distribution of flow in a square cavity as the Rayleigh number is varied [15]; however, it is not clear that this approach can be extended to unsteady problems.

In the particular case that the system matrices can be expressed as an affine combination of constant matrices with weighting coefficients that depend on the parameters, then each term of the weighted sum can be projected *a priori* onto the set of basis vectors. For example, if the system matrix $\mathbf{A}$ can be written as a decomposition of n_a terms,

$$\mathbf{A} = \sum_{j=1}^{n_a} \mathbf{A}^j \theta_j(\mathbf{z}) \tag{8}$$

where the matrices $\mathbf{A}^j \in \mathbb{R}^{n \times n}$ do not depend on the parameter and the coefficients θ_j are general (known) functions of $\mathbf{z}$, then the corresponding reduced-order matrix is

$$\mathbf{A}_r = \sum_{j=1}^{n_a} \mathbf{A}_r^j \theta_j(\mathbf{z}) \tag{9}$$

where $\mathbf{A}_r^j = \mathbf{W}^T\mathbf{A}^j\mathbf{V}$. Since the reduced-order component matrices $\mathbf{A}_r^j$ can be computed *a priori*, construction and solution of the ROM for a new parameter point will be fast. In some cases, the problem structure admits such a decomposition of the system matrices; otherwise, it can be obtained from the low-order terms of a Taylor series expansion [14, 16, 17]. This approach can also be paired with Krylov moment matching if all weighting coefficients that are not linear in just one parameter are replaced by additional parameters [10, 18].

The Taylor series expansion to obtain a decomposition of the form (8) becomes too expensive for a large number of parameters, or when parameter dependence is sufficiently nonlinear so that higher-order terms of the expansion are required. An alternative can be found in the trajectory piecewise-linear method of Rewienski and White [19], which was developed to create ROMs of nonlinear systems. Instead of creating a high-order expansion around a single state, several low-order expansions at different states are generated and combined with appropriate weighting functions. This idea has been combined with parameterized moment matching by Bond and Daniel [20], who have also addressed the effect of the selection of states around which the model is linearized [21].

The previously mentioned techniques use some linearization of the parameter dependence of the system matrices, which results in an approximation of the system matrices if the parameter dependence is nonlinear. A more general approach is to employ the coefficient-function approximation of [1, 2], which replaces nonlinear parametric dependencies with a reduced basis expansion and then uses interpolation to efficiently compute the coefficients of that expansion for new parameter values. Interpolation can also be used to calibrate the ROM, as in [22], where a ROM is considered

for two-dimensional vortex shedding past a confined square cylinder at different Reynolds numbers. In that work, a POD basis is derived using a collection of snapshots computed over a range of Reynolds numbers. Lagrange interpolation is used to approximate the dependence of the projected pressure term on Reynolds number, giving a way to calibrate the model so that long-term dynamics are accurately captured for Reynolds numbers not included in the snapshot set.

Another class of approaches calculates the full-order system matrices at a new value of the parameters, but approximates the variation of the projection basis as a function of the parameters. For example, Taylor expansion of the projection basis as a function of the parameters has been applied to an eigenmode reduction of a structural system for which the derivatives of the eigen-modes with respect to the parameters are calculated analytically [23]. Similarly, the combined approximation technique [24, 25] estimates the eigenmodes for a new parameter value as a linear combination of some reduced basis, which results in a smaller eigenvalue problem than for the calculation of the actual eigenmodes. While the combined approximation does not generally require knowledge of the sensitivities of the eigenmodes with respect to the parameters, in [23] these sensi-tivities were used as the reduced basis. These so-called extended ROM techniques were applied to POD by Weickum *et al.* [26], who used the combined approximation on both the snapshots and the POD basis vectors, with and without their approximate sensitivities with respect to the parameters. The extended ROM techniques result in a parameter-dependent projection basis as opposed to the spanning ROM approaches, which create a single projection basis from snapshots generated with different values of the parameters [26]. Creating a spanning ROM by projecting on a single POD basis extracted from snapshots corresponding to distinct parameter values has also been called global POD [27, 28].

An alternative for the expansion of the basis is interpolation. Lieu and Lesoinne [29] demon-strated that a direct interpolation between two orthonormal POD bases does not result in a new orthonormal basis and applied the result from Björck and Golub [30], who showed that interpolation of the angles between the subspaces does maintain orthonormality. This subspace angle interpo-lation has been used for parameterized ROMs for large-scale aeroelastic simulations [29, 31–33], but proved difficult to extend for variation of more than one parameter and is limited to inter-polation between two sets of basis vectors at a time [34]. To interpolate among more than two orthonormal sets of basis vectors, each corresponding to different parameter values, Amsallem *et al.* [35] use interpolation in the space tangent to the Grassmannian manifold of the bases which also preserves orthonormality. Consequently, a ROM for a parameter value not included in the sampling set is obtained by creating a new full-order model for those parameter values and by projecting it on the orthonormal set of basis vectors obtained by the interpolation. That approach circumvents the calculation of a new set of basis vectors for each value of the parameters, which would require a simulation with the full-order model to generate POD snapshots; however, it still requires evaluation of the full-order matrices at the new parameter value.

In this paper, we introduce interpolation among the system matrices of the reduced models to efficiently handle nonlinear parametric dependencies. Our approach is efficient because, once an initial set of ROMs has been derived, obtaining a ROM solution for a new parameter value avoids any computations that depend on the dimension of the full-scale model. That is, at each new parameter point, we avoid the costs of creating new full-order model matrices, of interpolation among sets of basis vectors, and of projection onto this set of basis vectors. Our approach is summarized as follows. A set of POD basis vectors is first created by simulating the full-order system for several parameter values. The matrices for each of the parameter values are reduced by projection, which result in a set of reduced systems, each for a different value of the parameters.

Then a ROM for a new parameter value is obtained by interpolation among the known ROMs. To improve the effectiveness of the approach, we consider different spaces in which the interpolation among ROMs can be performed. These methods are described in detail in the following sections.

3. INTERPOLATION OF ROMs

We consider the general (steady or unsteady) ROM (6) or (7) derived using the projection framework described in the previous section. From the complete set of snapshots, $\mathbf{x}^1, \ldots, \mathbf{x}^{n_s}$, we compute a single basis $\mathbf{V}$ using the POD and choose an appropriate left projection basis $\mathbf{W}$. A ROM is evaluated for each parameter sampled in the snapshots, that is, for $\mathbf{z}^i$, $i = 1, \ldots, n_z$. Since we choose to create each ROM using the same basis $\mathbf{V}$, then the reduced state $\mathbf{x}_r$ has the same physical significance in every model. A ROM for a new parameter vector $\hat{\mathbf{z}}$ can thus be obtained by interpolating among the system matrices of the known ROMs. In the methods presented here, the interpolation among the reduced-order system matrices is performed element wise, thus standard techniques for interpolation of scalar values can be applied. Cubic spline interpolants [36, 37] are used in this work. Our specific approaches are described in the following subsections for a general reduced-order matrix $\mathbf{M}$. For example, to obtain the approximation of $\mathbf{A}_r$ at the new parameter point, set $\mathbf{M} = \mathbf{A}_r$; to obtain the approximation of $\mathbf{B}_r$, set $\mathbf{M} = \mathbf{B}_r$, etc.

3.1. Spline interpolation

Spline interpolation among matrices as a function of the parameters $\mathbf{z}$ is performed with cubic splines [36, 37]. The set of n_z reduced-order system matrices $\mathbf{M}^i$, $i = 1, \ldots, n_z$, where $\mathbf{M}^i$ corresponds to the parameter vector $\mathbf{z}^i$, is interpolated element wise to calculate the matrix $\hat{\mathbf{M}}$ which corresponds to an arbitrary parameter point $\hat{\mathbf{z}}$. We first describe how to approximate the variation of the element in the gth row and hth column of $\mathbf{M}$, M_{gh}, as a function of one parameter z_s. The samples z_s^i are ordered in monotonically increasing order and the spline interpolant f^i for $M_{gh}(z_s)$ in the interval $z_s^i \leqslant z_s \leqslant z_s^{i+1}$ is given by

$$f^i(z_s) = \sum_{k=0}^{3} \alpha_k^i \left(\frac{z_s - z_s^i}{z_s^{i+1} - z_s^i} \right)^k \tag{10}$$

where the superscript k is an exponent. The coefficients α_k^i are calculated by imposing the interpolation conditions

$$f^i(z_s^i) = M_{gh}^i \tag{11a}$$

$$f^{i-1}(z_s^i) = M_{gh}^i \tag{11b}$$

and equality of the first and second derivative evaluated at z_s^i

$$\frac{\mathrm{d}f^{i-1}}{\mathrm{d}z_s}(z_s^i) = \frac{\mathrm{d}f^i}{\mathrm{d}z_s}(z_s^i) \tag{12a}$$

$$\frac{\mathrm{d}^2 f^{i-1}}{\mathrm{d}z_s^2}(z_s^i) = \frac{\mathrm{d}^2 f^i}{\mathrm{d}z_s^2}(z_s^i) \tag{12b}$$

Natural end conditions $(\mathrm{d}^2 f / \mathrm{d}z_s^2 = 0)$ are imposed at $\mathbf{z}^1$ and $\mathbf{z}^{n_z}$.

The cubic spline interpolation can be readily extended to multiple parameters. For example, in the case of two parameters, z_s and z_t, with uniform sampling, the samples are ordered on the nodes of a square grid. In the cell of the grid with $z_s^i \leqslant z_s \leqslant z_s^j$ and $z_t^i \leqslant z_t \leqslant z_t^j$, the interpolation function f^{ij} for $M_{gh}(z_s, z_t)$ is

$$f^{ij}(z_s, z_t) = \sum_{k=0}^{3} \sum_{l=0}^{3} \alpha_{kl}^{ij} \left(\frac{z_s - z_s^i}{z_s^j - z_s^i} \right)^k \left(\frac{z_t - z_t^i}{z_t^j - z_t^i} \right)^l \tag{13}$$

For the determination of the coefficients α_{kl}^{ij} and for splines with more than two parameters and/or non-uniform sampling, the reader is referred to [38].

3.2. Spline interpolation in the space tangent to the manifold

Direct interpolation with splines among the system matrices of reduced models will be successful if the variation of the matrix elements as a function of the parameters is captured well by the interpolants. In case of general nonlinear dependence of the reduced-order matrix on the parameters, this variation will not necessarily be modeled well. It may then be more advantageous to map the matrices to a space where the variation of the matrix elements as a function of the parameter is more amenable to interpolation with cubic splines. After the interpolation is performed in the mapped space, the result of the interpolation is mapped back to the original space to obtain the desired reduced-order matrix.

Key to success of this idea is the identification of an appropriate mapping; that is, a mapping to a space in which variation of the matrix elements as a function of the parameters is smooth. Drawing on the mapping approach employed by Amsallem *et al.* [35] for interpolation of the POD basis vectors, we choose here to use the concept of a Riemannian manifold, originating from differential geometry [39–41]. A Riemannian manifold $\mathcal{M}$ is a differentiable manifold, which means that a tangent space $\mathcal{T}_{\mathbf{M}^i}$ exists for every matrix $\mathbf{M}^i$ of the manifold. The mapping from the manifold to the tangent space at $\mathbf{M}^i$ is called the logarithmic mapping $\mathrm{Log}_{\mathbf{M}^i}$ and the mapping back to the manifold from that same tangent space is the so-called exponential mapping $\mathrm{Exp}_{\mathbf{M}^i}$, which will both be defined below. Moreover, an inner product is defined in each tangent space of a Riemannian manifold such that angles and distances can be measured, and consequently interpolation in a space tangent to a Riemannian manifold can be performed [42].

The first step in the proposed interpolation approach is to select one of the matrices as a reference. The interpolation will be performed in the space tangent to the manifold at that reference. In the example shown in Figure 1, we depict interpolation of the reduced matrix $\mathbf{A}_r$, that is $\mathbf{M} = \mathbf{A}_r$. The reference in this example is chosen to be $\mathbf{A}_r^2$, the matrix at the second parameter point $\mathbf{z}^2$, and the interpolation is performed in the space $\mathcal{T}_{\mathbf{A}_r^2}$, tangent to the manifold $\mathcal{M}$ at $\mathbf{A}_r^2$. The influence of the chosen reference will be discussed in Section 5. Second, the matrices at other parameter points are mapped to the reference tangent space. In Figure 1, $\mathbf{A}_r^1$ and $\mathbf{A}_r^3$ on the manifold are mapped, respectively, to the matrices $\mathbf{\Theta}^1$ and $\mathbf{\Theta}^3$ in $\mathcal{T}_{\mathbf{A}_r^2}$ with the logarithmic mapping $\mathrm{Log}_{\mathbf{A}_r^2}$. The image of $\mathbf{A}_r^2$ is the origin of the tangent space $\mathcal{T}_{\mathbf{A}_r^2}$. Third, the interpolation of all images $\mathbf{\Theta}^i$ is performed in the tangent space with cubic spline interpolation (as explained in Section 3.1). Finally, the result is mapped back to the original space. In Figure 1, $\hat{\mathbf{\Theta}}$ is mapped to $\hat{\mathbf{A}}_r$ on the manifold with the exponential mapping $\mathrm{Exp}_{\mathbf{A}_r^2}$. As can be seen in Figure 1, the interpolation between points on the manifold does not necessarily lie on the manifold but if the interpolation is performed in the space
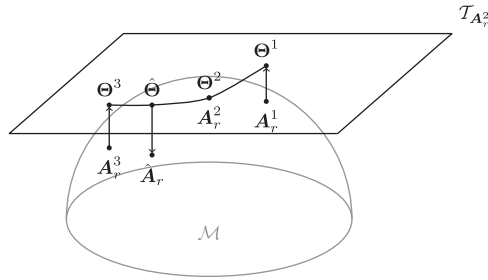
Figure 1. Representation of the Riemannian manifold $\mathcal{M}$ and the space $\mathcal{T}_{\mathbf{A}_r^2}$, tangent to the manifold at $\mathbf{A}_r^2$. The matrices $\mathbf{A}_r^1$ and $\mathbf{A}_r^3$ are mapped to $\mathbf{\Theta}^1$ and $\mathbf{\Theta}^3$ in $\mathcal{T}_{\mathbf{A}_r^2}$ with the logarithmic mapping $\mathrm{Log}_{\mathbf{A}_r^2}$. Subsequently, the interpolation of their images $\mathbf{\Theta}^i$ is performed in the tangent space and the result $\hat{\mathbf{\Theta}}$ is mapped to $\hat{\mathbf{A}}_r$ on the manifold with the exponential mapping $\mathrm{Exp}_{\mathbf{A}_r^2}$. Adapted from [42].

tangent to the manifold, then the mapping from the tangent space to the manifold makes sure that the result of the interpolation lies on the manifold.

The Log and Exp mappings are now described for elements of the general linear group $\mathcal{GL}(n)$, the set of invertible matrices in $\mathbb{R}^{n \times n}$ together with the operation of matrix multiplication [42]. In the example of Figure 1, the matrix $\mathbf{A}_r^1$ is mapped to a matrix $\mathbf{\Theta}^1$ in the tangent space of $\mathbf{A}_r^2$ with the logarithmic mapping

$$\mathbf{\Theta}^1 = \mathrm{Log}_{\mathbf{A}_r^2}(\mathbf{A}_r^1) = \log[\mathbf{A}_r^1(\mathbf{A}_r^2)^{-1}] \tag{14}$$

This mapping employs the matrix logarithm (log) of $\mathbf{A}_r^1(\mathbf{A}_r^2)^{-1}$, which is calculated with the algorithms of Davies and Higham [43]. The mapping of the matrix $\hat{\mathbf{\Theta}}$ from the tangent space of $\mathbf{A}_r^2$ to the matrix $\hat{\mathbf{A}}_r$ in the original space is given by the exponential mapping

$$\hat{\mathbf{A}}_r = \mathrm{Exp}_{\mathbf{A}_r^2}(\hat{\mathbf{\Theta}}) = \exp(\hat{\mathbf{\Theta}})\mathbf{A}_r^2 \tag{15}$$

which is obtained by calculating the matrix exponential (exp) of $\hat{\mathbf{\Theta}}$ using a Padé approximation with scaling and squaring [44]. If the system matrices are symmetric positive definite or if they have other special properties, then other mappings that preserve those properties as listed by Rahman *et al.* should be used [42]. Here, we apply the manifold interpolation only to the square matrices $\mathbf{A}_r$; the other system matrices $\mathbf{B}_r$ and $\mathbf{C}_r$ are interpolated directly with splines. The procedure is summarized in Algorithm 1.

Although the mapping to and from the tangent space in Algorithm 1 requires computation of matrix exponentials and matrix logarithms, these operations are performed only on reduced-order matrices, with no computations that depend on the size of the original full-order system. Thus, this approach will in general be significantly cheaper than existing approaches, which requires the construction of a new full-order matrix, possibly interpolation among basis vectors, and projection of the full-order model onto the basis vectors. A second advantage of our approach is that it is relatively non-intrusive; that is, derivation of ROMs at new parameter points can be carried out without any recourse to the original large-scale simulation code. Once the initial set of ROMs has been computed and stored, the ROM for a new parameter value is formed using only those precomputed ROMs. The effectiveness of this approach depends on how well the chosen

---

Algorithm 1 Spline interpolation in the space tangent to the manifold.

1. Select a reference parameter point $\mathbf{z}^t$ with corresponding matrix $\mathbf{A}_r^t$ to create the tangent space in which the interpolation will be performed
2. Set $\boldsymbol{\Theta}^t = \mathbf{0}$
3. **for** $i = 1, \ldots, n_z, \ i \neq t$ **do**
4.   Map $\mathbf{A}_r^i$ to $\boldsymbol{\Theta}^i = \mathrm{Log}_{\mathbf{A}_r^t}(\mathbf{A}_r^i)$
5. **end for**
6. Perform element-wise spline interpolation among the $\boldsymbol{\Theta}^i, i = 1, \ldots, n_z$, giving $\hat{\boldsymbol{\Theta}}$ as the interpolated matrix at parameter point $\hat{\mathbf{z}}$
7. Map $\hat{\boldsymbol{\Theta}}$ to $\hat{\mathbf{A}}_r = \mathrm{Exp}_{\mathbf{A}_r^t}(\hat{\boldsymbol{\Theta}})$, giving $\hat{\mathbf{A}}_r$ as the interpolated approximation of $\mathbf{A}_r(\hat{\mathbf{z}})$

---

interpolation scheme represents the variation of matrix elements. Thus, selection of an appropriate interpolation space is an important consideration, which is addressed in the next section.

4. SELECTION OF THE INTERPOLATION SPACE

In the previous section, two possibilities were presented for interpolation of the ROMs: interpolation of the matrix entries in their original coordinate system, or interpolation in a mapped space. In order to make the interpolation as effective as possible (i.e. maximizing ROM accuracy at interpolated parameter points while minimizing the number of samples), the interpolation should be carried out in the space where the variation of the matrix elements as a function of the parameters is most amenable to interpolation (here using cubic splines). The most appropriate space is clearly problem dependent; in this section we propose a heuristic criterion to give *a priori* guidance on the selection of the interpolation space.

For each element of the matrix to be interpolated, we perform a linear regression of that element's variation with the parameters. A normalized measure of the residual is then used as an indicator to select the interpolation space. For the dependence of element M_{gh} on the parameter vector $\mathbf{z} = [z_1, \ldots, z_m]^{\mathrm{T}}$, an approximate linear model is written as

$$M_{gh}(\mathbf{z}) \approx \alpha_0^{gh} + \sum_{j=1}^{m} \alpha_j^{gh} z_j \tag{16}$$

The coefficients $\alpha_j^{gh}, \ j = 0, \ldots, m$ are calculated by performing a linear regression over the matrices at the n_z parameter samples, $M_{gh}^1, \ldots, M_{gh}^{n_z}$. The 2-norm of the remaining least-squares residuals, normalized by the difference between the maximal and minimal value of M_{gh}, is an indicator of the nonlinearity of the variation of that particular matrix element as a function of the parameters. This indicator is given by

$$\gamma_{gh} = \frac{\sqrt{\sum_{i=1}^{n_z}(\alpha_0^{gh} + \sum_{j=1}^{m} \alpha_j^{gh} z_j^i - M_{gh}^i)^2}}{\max_i M_{gh}^i - \min_i M_{gh}^i} \tag{17}$$

The maximum of this normalized residual over all matrix elements,

$$\gamma = \max_{g,h} \gamma_{gh} \tag{18}$$

is used as the selection criterion.

An analogous criterion is defined for the matrices in the mapped space. That is, we consider fitting the linear model

$$\Theta_{gh}(\mathbf{z}) \approx \beta_0^{gh} + \sum_{j=1}^{m} \beta_j^{gh} z_j \tag{19}$$

where the coefficients β_j^{gh}, $j = 0, \ldots, m$ are calculated by performing a linear regression over the matrices at the n_z parameter samples, $\Theta_{gh}^1, \ldots, \Theta_{gh}^{n_z}$. The normalized residual is computed as

$$\lambda_{gh} = \frac{\sqrt{\sum_{i=1}^{n_z} \left(\beta_0^{gh} + \sum_{j=1}^{m} \beta_j^{gh} z_j^i - \Theta_{gh}^i \right)^2}}{\max_i \Theta_{gh}^i - \min_i \Theta_{gh}^i} \tag{20}$$

The selection criterion is again defined to be the maximum normalized residual, given by

$$\lambda = \max_{g,h} \lambda_{gh} \tag{21}$$

A comparison of γ and λ provides heuristic guidance to select the appropriate interpolation space. If the maximum normalized residual is lower for the $\mathbf{M}^i$ matrices than for the $\mathbf{\Theta}^i$ matrices (i.e. if $\gamma < \lambda$), then interpolation in the original space is recommended as the appropriate choice. Interpolation in the tangent space to the Riemannian manifold should be used if the maximum normalized residual is lower for the $\mathbf{\Theta}^i$ matrices. This idea could be easily extended to apply to other mappings, and to situations where more than two mapping options are employed. The procedure for the selection of the most appropriate space is summarized in Algorithm 2.

Algorithm 2 Heuristic criterion for selection of the interpolation space.

---

 1. **for all** M_{gh} **do**
 2.   Calculate regression parameters α_j^{gh}, $j = 0, \ldots, m$ in Equation (16)
 3.   Calculate γ_{gh} using Equation (17)
 4. **end for**
 5. The maximum normalized residual in the original coordinate system is $\gamma = \max_{g,h} \gamma_{gh}$
 6. **for all** Θ_{gh} **do**
 7.   Calculate regression parameters β_j^{gh}, $j = 0, \ldots, m$ in Equation (19)
 8.   Calculate λ_{gh}, using Equation (20)
 9. **end for**
10. The maximum normalized residual in the mapped space is $\lambda = \max_{g,h} \lambda_{gh}$
11. **if** $\gamma > \lambda$ **then**
12.   use interpolation in the tangent space
13. **else**
14.   use interpolation in the original space
15. **end if**

---

5. RESULTS

In this section, results are presented to demonstrate our approaches. In addition to presenting results for our proposed ROM interpolation approach, we provide a comparison with results obtained using Kriging interpolation applied directly to the outputs. In some situations it may be more effective to create a data-fit surrogate model—that is, to directly model the output as a function of the input parameters—rather than create a ROM. In this section we first briefly discuss some of the advantages and disadvantages of data-fit versus ROM strategies, and then present results for a steady thermal design problem and an unsteady contaminant transport application.

5.1. Data-fit models versus ROMs

Surrogate models can be categorized into three different classes: data-fit models, ROMs, and hierarchical models [45]. Data-fit models are generated using interpolation or regression of simulation data from the input/output relationships in the high-fidelity model [45, 46]. A significant advantage of this approach is that the process of deriving the data-fit model is completely non-intrusive. That is, the full-order model can be run in 'black-box' mode: specify a set of input parameters and generate the corresponding outputs. The process of deriving a ROM on the other hand is fairly intrusive, in particular requiring projections of full-order operators onto the reduced subspace. However, deriving a ROM endows the surrogate model with considerable flexibility by retaining the underlying structure of the model. For example, a projection-based ROM retains the concept of state in the surrogate model, which is not the case for a data-fit model that approximates directly the input/output map. This can be an important distinction, particularly for unsteady problems. For example, a ROM could be run for many different initial conditions or different temporal variations of boundary conditions, while a statistical data-fit model would be applicable only to the specific conditions under which it was derived. However, it is important to note that, unless rigorous error guarantees are available, all surrogate models must be used with care for values of parameters other than those over which they were derived.

Here, we employ Kriging interpolation of the output as a function of the parameters to create a data-fit model for comparative purposes. We follow a standard approach to create a Kriging model for every element of the output separately. Results are obtained using the MATLAB Kriging toolbox DACE [47]. The variation of an output element y_g as a function of the parameters $\mathbf{z}$ is modeled as the sum of a regression function, here taken to be a constant, and a Gaussian function. The Gaussian correlation model used is

$$R(\boldsymbol{\rho}, \mathbf{z}^i, \mathbf{z}^j) = \prod_{k=1}^{m} \exp[-\rho_k(z_k^i - z_k^j)^2] \tag{22}$$

where z_k^i is the kth element of the ith parameter sample and the $\boldsymbol{\rho}$ are scaling factors that are determined using a maximum likelihood estimator. A complete description of Kriging-based approximation models can be found in [48, 49].

5.2. Thermal fin example

The first example is a steady model of a thermal fin with parameterized geometry and material properties such that the model can be used for optimization [14]. For a ROM to facilitate such a design optimization task, it must be accurate and rapid to solve over a range of geometric and material parameters. The geometry of the fin is shown in Figure 2, with the subdomains Ω_j and
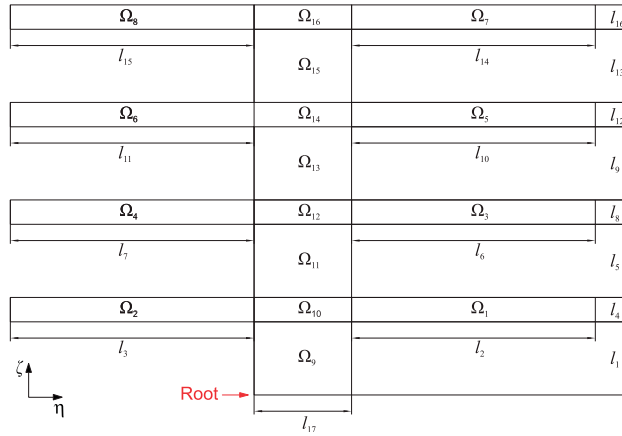
Figure 2. Geometry of the thermal fin with indication of the subdomains Ω_j
and dimension parameters l_j [14].

the fin dimension parameters l_j indicated. The temperature T in the subdomain Ω_j is described
by the steady heat equation

$$-\kappa_j \nabla^2 T = 0 \tag{23}$$

where κ_j is the constant thermal conductivity in Ω_j. At the root of the fin, a Neumann boundary
condition is prescribed to specify the heat flux,

$$-\kappa_9 (\nabla T \cdot \mathbf{n}) = -1 \tag{24}$$

where $\mathbf{n}$ is the unit normal vector pointing out of the domain. Convection to the air that surrounds
the fin is imposed on the remainder of the outer boundary with a Robin boundary condition

$$-\kappa_j (\nabla T \cdot \mathbf{n}) = Bi \cdot T \tag{25}$$

where Bi is the Biot number. The output of interest, y, is the area-weighted average temperature
of the fin

$$y = \frac{\sum_{j=1}^{16} \int_{\Omega_j} T \, d\Omega}{\sum_{j=1}^{16} \int_{\Omega_j} d\Omega} \tag{26}$$

For numerical simulation, the domain is discretized with linear, triangular finite elements using
17 899 degrees of freedom, leading to a system of equations of the form (1). In this example,
the full-order system matrices can be written as in Equation (8); thus, all elements of the system
matrices vary in the same (nonlinear) way as a function of $\mathbf{z}$. In practice for this example, the
nonlinearity of the parameter dependence in the functions $\theta_j(\mathbf{z})$ is known and is mainly $1/z$.
However, in order to demonstrate the general applicability of our approach, we do not use the
decomposition equation (8), and rather assume an unknown form for $\mathbf{A}(\mathbf{z})$. Results are presented
for three cases, described below.

In Case I, the thermal conductivities κ_j in subdomains Ω_1–Ω_8 are varied simultaneously as one
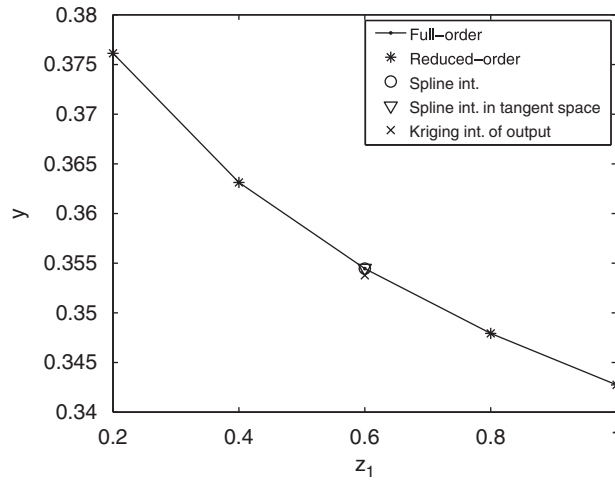parameter z_1, which is sampled at [0.2 0.4 0.8 1.0]. ROMs with dimension $n_r = 4$ are constructed

Figure 3. Area-weighted average temperature of the thermal fin as a function of the thermal conductivity $z_1 = \kappa_j$ in subdomains Ω_1–Ω_8 (Case I).

Table I. Relative error in the prediction of the output of interest (area-weighted average temperature of the thermal fin) for the different interpolation approaches applied to the three different cases.

Algorithm	Case I	Case II	Case III
Mapping space selected by indicator	Original	Tangent	Tangent
Spline interpolation in original space	3.40e−6	3.27e−2	5.22e−2
Spline interpolation in tangent space	1.43e−4	3.43e−3	5.25e−3
Kriging of the output	1.90e−3	3.19e−4	1.20e−3

at each of these parameter points using a Petrov–Galerkin projection of the form $\mathbf{W} = \mathbf{AV}$ [14]. The maximal relative error of the output of the ROMs compared with the output of the full-order models is $1.19e-14$. An interpolated ROM is then constructed for $z_1 = 0.6$ using the approaches described in Section 3. For interpolation in the space tangent to the manifold, the tangent space is created at $z_1 = 0.4$. The selection criterion in Algorithm 2 yields $\gamma = 0.44$ for spline interpolation and $\lambda = 0.61$ for interpolation in the space tangent to the manifold; thus, the criterion indicates that interpolation in the original space is preferred. Figure 3 shows the output as a function of the parameter, from which it can be seen that both interpolated ROMs yield accurate predictions.

Table I shows the relative errors between a full-order simulation with $z_1 = 0.6$ and the predicted output using the interpolation approaches. These results show that the error using spline interpolation in the original space is two orders of magnitude lower than the error incurred when using spline interpolation in the space tangent to the manifold; thus confirming the validity of the indicator used to select the interpolation space. For comparison, a Kriging model of the output is also used to predict the output at $z_1 = 0.6$. Table I and Figure 3 show that the Kriging model performs similarly to the interpolated ROMs, although the prediction accuracy is lower for this case.

In Case II, the geometric dimensions l_1, l_5, l_9, and l_{13} are varied simultaneously, modeled using one parameter z_1 that is sampled at [0.50 0.65 0.85 1.00]. The maximal relative error in
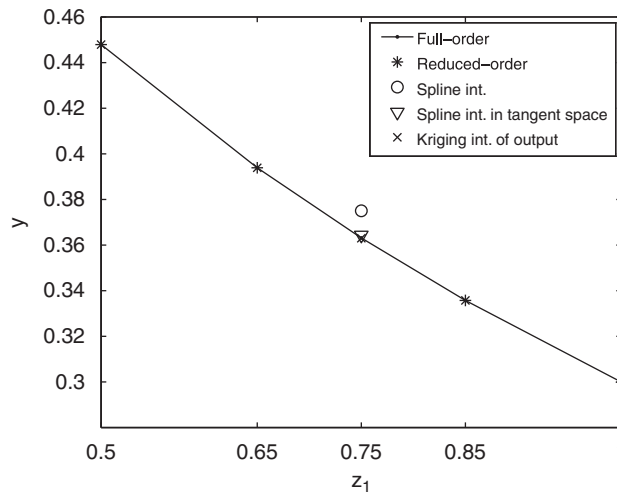
Figure 4. Area-weighted average temperature of the thermal fin as a function of the fin height dimensions l_1, l_5, l_9, and l_{13}, which are varied simultaneously as one parameter z_1 (Case II).

the output of the ROMs computed at each of these parameter points compared with the full-order models is $5.09\mathrm{e}-14$. The interpolated ROM for $z_1=0.75$ is generated and the tangent space is created at $z_1=0.65$. The selection criterion for interpolation in the original space yields $\gamma=0.79$ compared with $\lambda=0.65$ for interpolation in the tangent space, thus the criterion indicates that interpolation with mapping to the tangent space is preferred. This is confirmed by Figure 4, which shows the output as a function of the parameter. The relative errors in output prediction between the interpolation approaches and a full-order simulation with $z_1=0.65$ are shown in Table I. Again, the results obtained using a Kriging model of the output are similar to the interpolated ROM results, although in this case Kriging performs slightly better than the ROMs.

Our third case, Case III, considers interpolation with two parameters. The fin height dimensions l_1, l_5, l_9, and l_{13} are varied simultaneously as a first parameter z_1. The widths of the fins l_2, l_3, l_6, l_7, l_{10}, l_{11}, l_{14}, and l_{15} are varied as the second parameter z_2. The parameter space is sampled on a grid with the values for z_2 of $[1.5\ 2.0\ 2.5\ 3.0\ 3.5]$ and the values for z_1 as in Case II, resulting in a total of 20 parameter samples. The selection criteria for interpolation in the original and tangent spaces space yield $\gamma=2.84$ and $\lambda=1.74$, respectively. This indicates that interpolation in the tangent space is preferred. The results in Table I again confirm this result, showing that the error in the output for an interpolated model at $z_1=0.75$, and $z_2=2.75$ is an order of magnitude smaller for interpolation in the tangent space versus interpolation in the original space.

The influence of different choices for the origin of the tangent space is illustrated in Table II. The recommended approach is to select the origin of the tangent space to be that parameter point closest to the parameter value for which interpolation is desired. In Cases I and II above, this strategy leads to choices of $z_1=0.4$ or $z_1=0.6$ and $z_1=0.65$ or $z_1=0.85$, respectively (in each case two parameter points are equal distance from the desired point). As Table II shows, for each case these choices result in both a similar value of λ and a similar relative interpolation error. If however the tangent space is constructed using one of the other parameter values further from the desired point, both λ and the relative error are larger. We also note that the heuristic criterion is not

Table II. Indicator value and relative error in the prediction of the output of interest (area-weighted average temperature of the thermal fin) for spline interpolation in the tangent space as a function of the parameter value z_1 at the origin of the tangent space for (a) Case I and (b) Case II.

z_1	λ	Error
(a)		
0.2	0.64	1.76e−4
0.4	0.61	1.43e−4
0.8	0.60	1.00e−4
1.0	0.75	8.16e−4
(b)		
0.5	0.86	7.17e−3
0.65	0.65	3.43e−3
0.85	0.67	4.19e−3
1.0	0.77	8.80e−3

as reliable in these cases: for Case II, constructing the tangent space at $z_1 = 0.5$ leads to $\lambda = 0.86$ and $\gamma = 0.79$, yet the relative error is lower for interpolation in the tangent space.

5.3. Contaminant transport example

A second setting for which ROMs can provide valuable information is a real-time prediction and decision making. For example, the characterization of hazardous events and subsequent prediction of the evolution of the hazard requires simulation tools that can be solved rapidly in the field. Here, we consider the simple example of contaminant transport governed by the convection–diffusion equation. In [50], it was shown that given a specified velocity field in a domain, a ROM can be used to accurately predict the contaminant transport over a wide range of initial conditions. However, in practice, the velocity field would not be specified *a priori*, but would be estimated using local weather models and/or inference of sparse measurements. This leads to two challenges for ROM real-time prediction in this setting: first, the ROM should be able to accept as input an arbitrary velocity field (which may also vary in time); second, the uncertainty associated with the specified velocity will impact the corresponding predictions of contaminant transport. To support effective decision making, these prediction uncertainties must be quantified. Our interpolation methodology provides a mechanism to address the first challenge—incorporating velocity dependence in such a way that the ROM can be quickly regenerated for different velocity fields, with no need to run addition large-scale simulations. To address the second challenge, we show how our ROMs facilitate Monte Carlo simulations that can be used rapidly to characterize uncertainties in the ROM predictions.

Our specific example considers contaminant transport in the two-dimensional rectangular domain, Ω, shown in Figure 5. The x-coordinate is denoted as η and the y-coordinate is as denoted ζ. The physical process is modeled by the convection–diffusion equation

$$\frac{\partial C}{\partial t} + \mathbf{v} \cdot \nabla C - \mu \nabla^2 C = 0 \quad \text{in } \Omega \times (0, t_f) \tag{27}$$

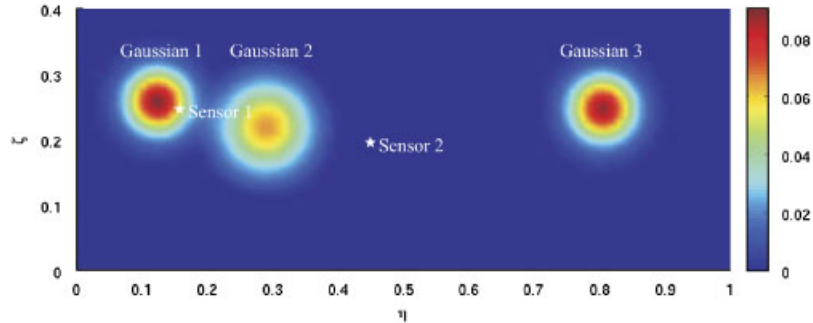$$C = 0 \quad \text{on } \Gamma_D \times (0, t_f) \tag{28}$$

Figure 5. Initial concentration of the contaminant in the unsteady contaminant transport problem, which is a superposition of three Gaussian functions. The sensor locations are also indicated.

$$\frac{\partial C}{\partial n} = 0 \quad \text{on } \Gamma_N \times (0, t_f) \tag{29}$$

$$C = C_0 \quad \text{in } \Omega \text{ for } t = 0 \tag{30}$$

where C is the contaminant concentration (which varies in time and over the domain Ω), $\mathbf{v}$ is the velocity vector field, μ is the diffusivity, t_f is the time horizon of interest, and C_0 is the given initial condition. Homogeneous Dirichlet boundary conditions are applied on the inflow boundary Γ_D, whereas homogeneous Neumann conditions are applied on the other boundaries Γ_N. For the domain shown in Figure 5, the inflow boundary, Γ_D, is defined by $\eta = 0$, $0 \leqslant \zeta \leqslant 0.4$; the remaining boundaries comprise Γ_N. The Péclet number based on the mean velocity and the width of the domain is 0.50.

The governing equations are discretized in time using the backward Euler method with 100 time steps over the time horizon $t = 0$ to $t_f = 0.5$. The spatial discretization employs the streamline upwind/Petrov–Galerkin finite element method [51]. The rectangular domain is discretized using linear triangular elements with a total of 4005 nodes. The initial concentration of the contaminant is given by the superposition of three Gaussian functions as depicted in Figure 5.

The velocity field is specified using two parameters z_1 and z_2, which are the coordinates of a velocity source point within the rectangular domain. The velocity vector $\mathbf{v}$ has a length inversely proportional to the distance from the source and is directed away from the source according to

$$\mathbf{v}(\eta, \zeta) = 0.001 \frac{(\eta - z_1)\mathbf{e}_\eta + (\zeta - z_2)\mathbf{e}_\zeta}{(\eta - z_1)^2 + (\zeta - z_2)^2} \tag{31}$$

where $\mathbf{e}_\eta$ and $\mathbf{e}_\zeta$ are the unit vectors in the η and ζ directions, respectively. The norm of the velocity is set to zero at the center of the velocity source ($\eta = z_1, \zeta = z_2$) and drops below 0.01 outside a circle with radius 0.1. In this example, the elements of the system matrices all have a different nonlinear dependence on the velocity source parameters.

We define the output of interest to be the accumulated contaminant concentration at two sensor locations in the domain. The accumulated contaminant concentration at a spatial location (η, ζ) is defined as

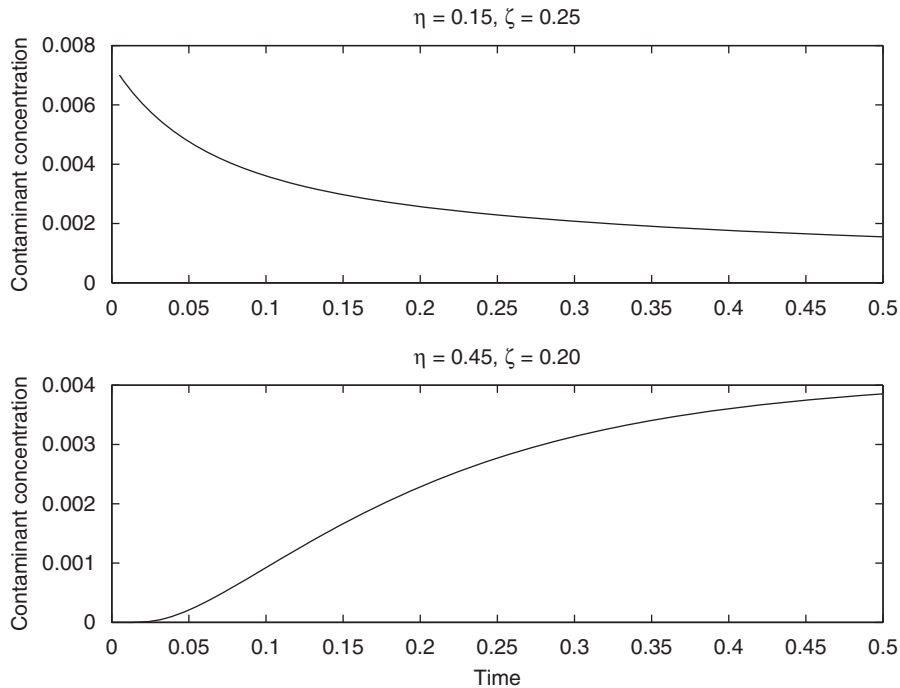$$\bar{C}(\eta, \zeta) = \int_0^{t_f} C(\eta, \zeta, t) \, \mathrm{d}t \tag{32}$$

Figure 6. Evolution of the contaminant concentration at Sensor 1 ($\eta=0.15, \zeta=0.25$) and Sensor 2 ($\eta=0.45, \zeta=0.20$) as a function of time for a velocity source at $z_1=0.50, z_2=0.20$ in the unsteady contaminant transport problem.

and is calculated using the trapezoidal rule to approximate the integral. For the cases considered here, Sensor 1 is located at $\eta=0.15, \zeta=0.25$, and Sensor 2 at $\eta=0.45, \zeta=0.20$ as indicated in Figure 5. When there is no velocity source present, the contaminant is transported only through the diffusion process (there is no convection). In that case, the accumulated contaminant over the time horizon of interest, $t_f=0.5$, is $\bar{C}=1.3503e-2$ for Sensor 1 and $\bar{C}=1.2535e-3$ for Sensor 2. For the case of a velocity source in the middle of the domain ($z_1=0.50, z_2=0.20$), the evolution in time of the concentration at the two sensor locations is shown in Figure 6.

In practice, the velocity field may be unknown or there may be significant uncertainties associated with the prescribed velocity field (which may be, for example, estimated from sparse measurements taken over the domain). These uncertainties in turn impact the prediction of the contaminant concentration and thus the subsequent decisions that are made. To demonstrate the ability of the interpolated ROMs to capture this variation, we consider the parameters describing the velocity field to be uncertain. Each parameter is assigned a uniform distribution with $0.1 \leqslant z_1 \leqslant 0.9$ and $0.1 \leqslant z_2 \leqslant 0.3$. The distributions of the accumulated contaminant concentration at each of the sensor locations are then estimated using Monte Carlo simulation over this parameter space. The top left plots in Figures 7 and 8 show the resulting distributions computed using the full-order model with 160 000 random samples.

It can be seen from Figures 7 and 8 that the output histograms have a rather unusual shape. With the relatively low Péclet number, the process is dominated by diffusion; hence the reasonably
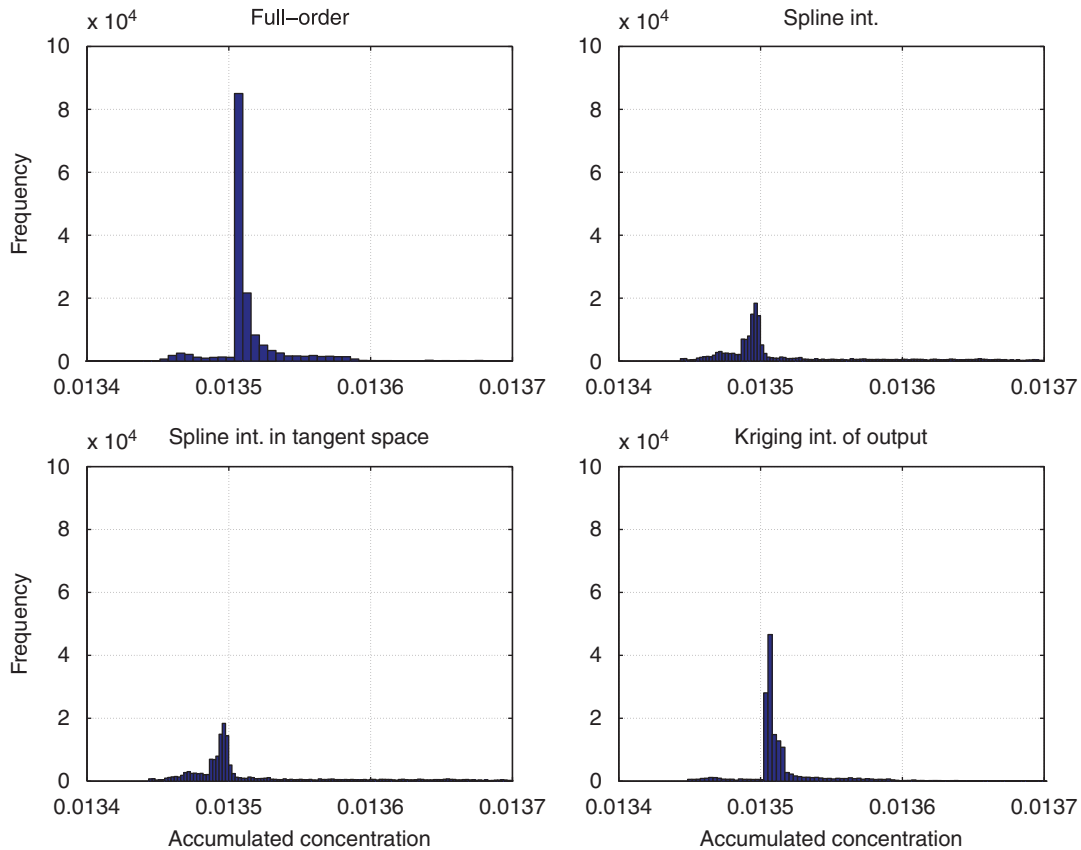
Figure 7. Estimated distribution of the accumulated contaminant concentration at Sensor 1 ($\eta = 0.15, \zeta = 0.25$) corresponding to a uniform distribution of the parameters z_1 and z_2 in the unsteady contaminant transport problem.

tight range of output values. The distribution of the accumulated contaminant concentration at Sensor 1 consists of a single peak for $1.352\mathrm{e}-2$ and very low probabilities for other values. This behavior is due to the fact that only when the velocity source is near the sensor does it influence the contaminant concentration at that point significantly. As Figure 9 shows, other parameter values result in an accumulated concentration of $\bar{C} = 1.352\mathrm{e}-2$, i.e. that due to diffusion. The accumulated concentration at Sensor 2 as a function of z_1 and z_2 is depicted in Figure 10, which shows that only a small range of the parameters results in an accumulated concentration between $1.24\mathrm{e}-3$ and $1.28\mathrm{e}-3$, as can be seen from the two distinct peaks in Figure 8. Physically, the shape of the distribution in Figure 8 can be explained since most of the contaminant reaching Sensor 2 originates from Gaussian 2 in the initial condition shown in Figure 5. If the velocity source is positioned far away from Sensor 2 ($z_1 < 0.3$ or $z_1 > 0.6$), the source's influence decreases and the accumulated contaminant at Sensor 2 is due mostly to diffusion. If the velocity source is positioned closer and to the left of the sensor ($0.3 < z_1 < 0.45$), some of the contaminant initially contained within Gaussian 2 is convected toward Sensor 2, increasing the accumulated contaminant. If the source is
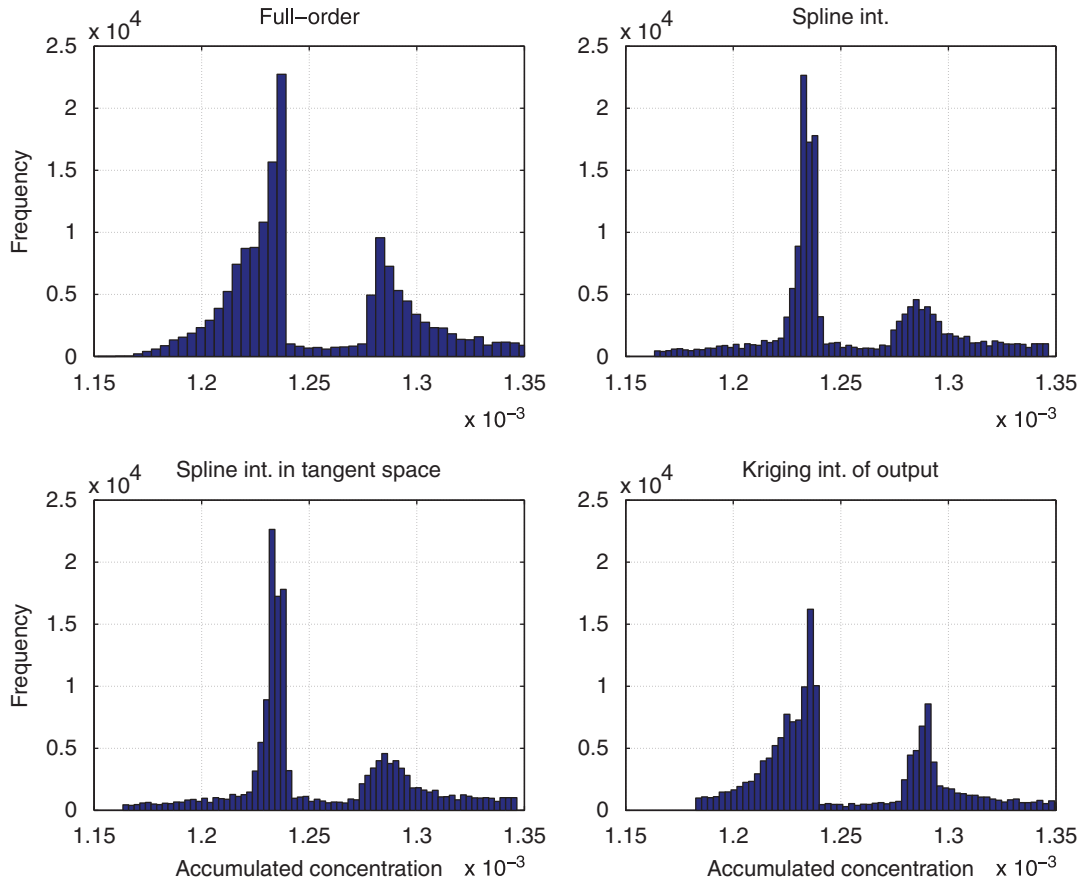
Figure 8. Estimated distribution of the accumulated contaminant concentration at Sensor 2 ($\eta=0.45, \zeta=0.20$) corresponding to a uniform distribution of the parameters z_1 and z_2 in the unsteady contaminant transport problem.

positioned close and to the right of Sensor 2 ($0.45 < z_1 < 0.6$), the contaminant initially contained within Gaussian 2 is convected away from the sensor, decreasing the accumulated contaminant. The contaminant from Gaussian 2 is hence convected either toward or away from Sensor 2, which explains the two peaks in the distribution.

This example highlights how output predictions can be substantially affected by uncertainties in input parameters. To achieve Monte Carlo simulations such as those shown in Figures 7 and 8 in real time—critical for supporting decision making in the field—we require ROMs that can be evaluated and solved rapidly over the possible range of velocity conditions. To implement our methodology, 121 ROMs are constructed over the parameter range specified above, using a uniform grid with 11 sample points in each direction. For the mapping, the space tangent to the manifold is constructed at $z_1=0.50$ and $z_2=0.20$. The Monte Carlo simulation results using ROMs interpolated in the original space and in the tangent space are shown in Figures 7 and 8 for Sensor 1 and Sensor 2, respectively. The figures show that representing the details of these
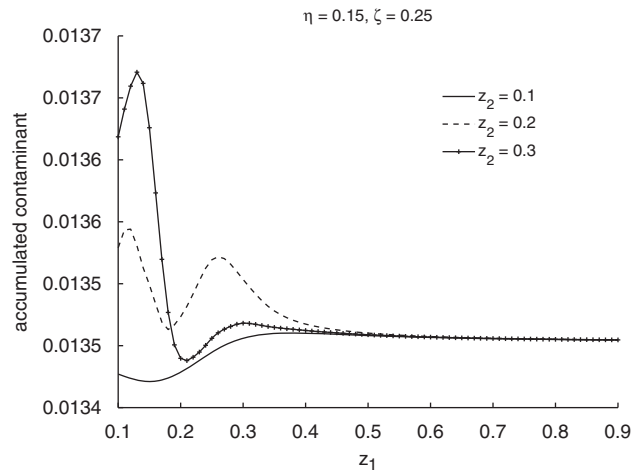
Figure 9. Accumulated contaminant concentration at Sensor 1 ($\eta = 0.15$, $\zeta = 0.25$) as a function of velocity source location described by z_1 and z_2.
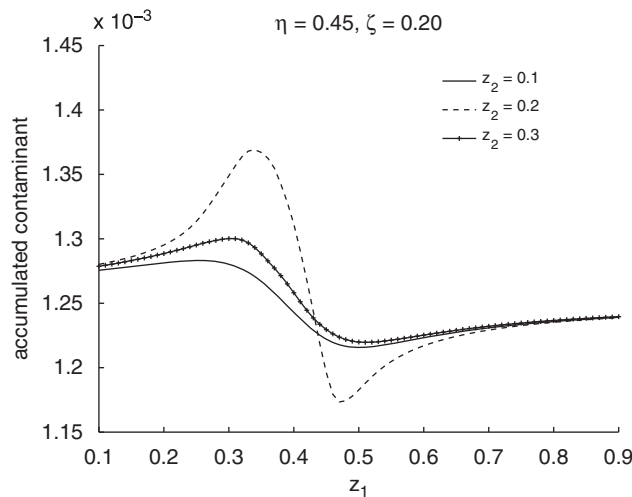

Figure 10. Accumulated contaminant concentration at Sensor 2 ($\eta = 0.45$, $\zeta = 0.20$) as a function of velocity source location described by z_1 and z_2.

unusually shaped distributions is rather difficult with an approximate model; however, it can be seen that the general character of the distributions is captured. In particular, the ROMs are able to successfully identify the bimodal nature of the distribution for Sensor 2. The figures also show the results using Kriging interpolation of the outputs. While Kriging of the output better approximates the distributions, the Kriging model cannot be used for different initial concentrations; thus its usefulness in a practical setting is limited. The ROMs on the other hand have the flexibility to be

Table III. Mean, standard deviation, and skewness of the accumulated contaminant concentration at (a) $\eta=0.15, \zeta=0.25$ and (b) $\eta=0.45, \zeta=0.20$, together with the relative error with respect to the full-order simulation.

Algorithm	Mean		Standard deviation		Skewness	
	Value	Error	Value	Error	Value	Error
(a)						
Full order	1.3526e−2	—	7.4901e−5	—	4.8630e+0	—
Spline interpolation	1.3522e−2	3.44e−4	5.2517e−5	2.99e−1	1.5854e+0	6.74e−1
Spline int. in the tangent space	1.3523e−2	2.70e−4	5.2518e−5	2.19e−1	1.8875e+0	6.12e−1
Kriging of the output	1.3520e−2	4.79e−4	4.5148e−5	3.97e−1	3.8232e+0	2.14e−1
(b)						
Full order	1.2538e−3	—	4.2499e−5	—	6.9435e−1	—
Spline interpolation	1.2559e−3	1.65e−3	1.2399e−5	7.08e−1	4.8405e−1	3.03e−1
Spline int. in the tangent space	1.2549e−3	8.54e−4	1.3899e−5	6.73e−1	5.2605e−1	2.42e−1
Kriging of the output	1.2541e−3	2.15e−4	4.2147e−5	8.28e−3	7.3012e−1	5.15e−2

Table IV. Computational time taken to complete the Monte Carlo simulation with 160 000 samples for the unsteady contaminant transport problem, relative to the fastest calculation (Kriging of the output).

Algorithm	Duration
Full order	78.36
Spline interpolation	6.04
Spline interpolation in the tangent space	7.33
Kriging of the output	1.00

applied to many different initial conditions, provided the basis is sufficiently rich to capture the important dynamics.

For further comparison, the mean, standard deviation, and skewness of the output distributions are estimated from the Monte Carlo simulation results and shown in Table III. All interpolation techniques estimate the mean value well, but have a significant error in the standard deviation and skewness. However, one should be careful in using only moment information for decision making given the irregular shapes of the distributions.

Table IV lists the relative CPU time taken to perform the Monte Carlo simulations with the different techniques. The computational results were obtained on a workstation with two quad-core 2.66 GHz Xeon processors. Both the full-order and reduced-order calculation take advantage of parallelization across multiple cores in the same way. The time to construct the 121 ROMs is included in the two ROM spline interpolation approaches. For the Kriging interpolation, the time to perform the 121 full-order simulations is included. While Kriging interpolation of the output is faster than the ROM interpolation methods, its usefulness is limited for unsteady problems, as mentioned above. The approximation methods show approximately an order of magnitude reduction in computational time over the full-order simulation.

6. CONCLUSION

Parametrized ROMs can be obtained by interpolating among the system matrices of a set of precomputed ROMs. This approach offers the advantage of being simple to implement, while yielding ROMs that are fast to evaluate and solve. It is important to select the most appropriate space in which to perform the interpolation, especially if the variation of the system matrices as a function of the parameters is nonlinear. For any problem, it is important to carefully weigh the relative advantages and disadvantages of different surrogate modeling methods. Interpolation among a set of outputs with Kriging is fast, can be implemented as a black-box strategy, and results in a good approximation of the output for many problems. However, the general class of data-fit surrogate models does not retain system structure, thus limiting their applicability for parameters and input values other than those over which they were derived.

REFERENCES

1. Barrault M, Maday Y, Nguyen N, Patera A. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathéematique*, *Académie des Sciences*, *Paris* 2004; **I**:339–667.
2. Grepl M, Maday Y, Nguyen N, Patera A. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *Mathematical Modelling and Numerical Analysis* 2007; **41**(3):575–605.
3. Antoulas A, Sorensen D, Gugercin S. A survey of model reduction methods for large-scale systems. *Contemporary Mathematics*, *Volume*: *Structured Matrices in Operator Theory*, *Numerical Analysis*, *Control*, *Signal and Image Processing*, Chapter 280. American Mathematical Society: Providence, RI, 2001; 193–219.
4. Berkooz G, Holmes P, Lumley J. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics* 1993; **25**(1):539–575.
5. Sirovich L. Turbulence and the dynamics of coherent structures, part 1: coherent structures. *Quarterly of Applied Mathematics* 1987; **45**(3):561–571.
6. Moore B. Principal component analysis in linear systems: controllability, observability, and model reduction. *IEEE Transactions on Automatic Control* 1981; **AC-26**(1):17–32.
7. Feldmann P, Freund R. Efficient linear circuit analysis by Padé-approximation via the Lanczos process. *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems* 1995; **14**(5):639–649.
8. Gallivan K, Grimme E, Van Dooren P. Padé approximation of large-scale dynamic systems with Lanczos methods. *Proceedings of the 33rd IEEE Conference on Decision and Control*, Lake Buena Vista, FL, December 1994.
9. Farhat C, Amsallem D. Reduced-order modeling, differential geometry and physics-based near-real-time predictions. *Eighth World Congress on Computational Mechanics*, *5th European Congress on Computational Methods in Applied Sciences and Engineering*, Venice, Italy, 2008; 1.
10. Daniel L, Ong C, Low S, Lee K, White J. A multiparameter moment matching model reduction approach for generating geometrically parameterized interconnect performance models. *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems* 2004; **23**(5):678–693.
11. Hinze M, Volkwein S. Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: error estimates and suboptimal control. In *Dimension Reduction of Large-Scale Systems*, Benner P, Mehrmann V, Sorensen D (eds). Lecture Notes in Computational and Applied Mathematics. Springer-Verlag: Berlin, Heidelberg, 2005; 261–306.

12. Kunisch K, Volkwein S. Control of Burgers' equation by reduced order approach using proper orthogonal decomposition. *Journal of Optimization Theory and Applications* 1999; **102**:345–371.
13. Prud'homme C, Rovas D, Veroy K, Maday Y, Patera A, Turinici G. Reliable real-time solution of parameterized partial differential equations: reduced-basis output bound methods. *Journal of Fluids Engineering* 2002; **124**:70–80.
14. Bui-Thanh T, Willcox K, Ghattas O. Model reduction for large-scale systems with high-dimensional parametric input space. *SIAM Journal on Scientific Computing* 2008; **30**(6):3270–3288.
15. Ly H, Tran H. Modeling and control of physical processes using proper orthogonal decomposition. *Journal of Mathematical and Computer Modeling* 2001; **33**:223–236.
16. Veroy K, Prudhomme C, Rovas D, Patera A. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. *The 16th AIAA Computational Fluid Dynamics Conference*, Orlando, FL, U.S.A., 2003; 1–18. *AIAA 2003-3847*.
17. Grepl M, Patera A. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *Mathematical Modelling and Numerical Analysis* 2005; **39**(1):157–181.
18. Daniel L, White J. Automatic generation of geometrically parameterized reduced order models for integrated spiral RF inductors. *Proceedings of the 2003 IEEE International Workshop on Behavioral Modeling and Simulation*, San Jose, CA, U.S.A., 2003; 18–23.
19. Rewienski M, White J. A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micro machined devices. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 2003; **22**(2):155–170.
20. Bond B, Daniel L. Parameterized model order reduction of nonlinear dynamical systems. *Proceedings of the IEEE Conference on Computer-Aided Design*, San Jose, CA, U.S.A., 2005; 1–8.
21. Bond B, Daniel L. A piecewise-linear moment-matching approach to parameterized model-order reduction for highly nonlinear systems. *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems* 2007; **26**(12):2116–2129.
22. Galletti B, Bruneau C, Zannetti L, Iollo A. Low-order modelling of laminar flow regimes past a confined square cylinder. *Journal of Fluid Mechanics* 2004; **503**:161–170.
23. Allen M, Weickum G, Maute K. Application of reduced order models for the stochastic design optimization of dynamic systems. *Tenth AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, NY, U.S.A., 2004; 1–19. *AIAA 2004-4614*.
24. Kirsch U. Combined approximations—a general reanalysis approach for structural optimization. *Structural and Multidisciplinary Optimization* 2000; **20**(2):97–106.
25. Keerti A, Nikolaidis E, Ghiocel D, Kirsch U. Combined approximations for efficient probabilistic analysis of structures. *AIAA Journal* 2004; **42**(7):1321–1330.
26. Weickum G, Eldred M, Maute K. Multi-point extended reduced order modeling for design optimization and uncertainty analysis. *Structures*, *Structural Dynamics and Materials Conference*, AIAA/ASME/ASCE/AHS/ASC, Newport, RI, U.S.A., 2006.
27. Taylor J. Dynamics of large scale structures in turbulent shear layers. *Ph.D. Thesis*, Clarkson University, 2001.
28. Taylor J, Glauser M. Towards practical flow sensing and control via POD and LSE based low-dimensional tools. *Journal of Fluids Engineering* 2004; **126**(3):337–345.
29. Lieu T, Lesoinne M. Parameter adaptation of reduced order models to three-dimensional flutter analysis. *The 42nd AIAA Aerospace Sciences Meeting and Exhibit*, vol. AIAA 2004-0888, Reno, NV, U.S.A., 2004.
30. Bjorck A, Golub G. Numerical methods for computing angles between linear subspaces. *Mathematics of Computation* 1973; **27**(123):579–594.
31. Lieu T, Farhat C. Adaptation of POD-based aeroelastic ROMs for varying mach number and angle of attack: application to a complete F-16 configuration. *AIAA Paper 2005-7666*, 2005.
32. Lieu T, Farhat C, Lesoinne M. POD-based aeroelastic analysis of a complete F-16 configuration: ROM adaptation and demonstration. *The 46th AIAA/ASME/ASCE/AHS/ASC Structures*, *Structural Dynamics and Materials Conference*, vol. AIAA 2005-2295, Austin, TX, U.S.A., 2005.
33. Lieu T, Farhat C, Lesoinne M. Reduced-order fluid/structure modeling of a complete aircraft configuration. *Computer Methods in Applied Mechanics and Engineering* 2006; **195**:5730–5742.
34. Lieu T, Farhat C. Adaptation of aeroelastic reduced-order models and application to an F-16 configuration. *AIAA Journal* 2007; **45**(6):1244–1257.
35. Amsallem D, Farhat C, Lieu T. Aeroelastic analysis of F-16 and F-18/A configurations using adapted CFD-based reduced order models. *The 48th AIAA/ASME/ASCE/AHS/ASC Structures*, *Structural Dynamics*, *and Materials Conference*, vol. AIAA 2007-2364, Honolulu, Hawaii, 2007; 1–20.

36. Carlson R, Fritsch F. Monotone piecewise bicubic interpolation. *SIAM Journal on Numerical Analysis* 1985; **22**(2):386–400.
37. De Boor C. *A Practical Guide to Splines*. Springer: New York, NY, U.S.A., 2001.
38. Chui C, Wang RH. On smooth multivariate spline functions. *Mathematics of Computation* 1983; **41**(163):131–142.
39. Spivak M. *A Comprehensive Introduction to Differential Geometry* (3rd edn), vol. 5. Publish or Perish: Houston, TX, U.S.A., 1999.
40. O'Neill B. *Elementary Differential Geometry* (Revised 2nd edn). Elsevier: Amsterdam, 2006.
41. Begelfor E, Werman M. Affine invariance revisited. *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, NY, U.S.A., 2006.
42. Rahman I, Drori I, Stodden V, Donoho D, Schröder P. Multiscale representation for manifold-valued data. *Multiscale Modeling and Simulation* 2005; **4**(4):1201–1232.
43. Davies P, Higham N. A Schur–Parlett algorithm for computing matrix functions. *SIAM Journal on Matrix Analysis and Applications* 2003; **25**(2):464–485.
44. Higham N. The scaling and squaring method for the matrix exponential revisited. *SIAM Journal on Matrix Analysis and Applications* 2005; **26**(4):1179–1193.
45. Eldred M, Giunta S, Collis S. Second-order corrections for surrogate-based optimization with model hierarchies. *Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, NY, 2004.
46. Venter G, Haftka R, Starnes J. Construction of response surface approximations for design optimization. *AIAA Journal* 1998; **36**(12):2242–2249.
47. Lophaven S, Nielsen H, Sündergaard J. *Aspects of the Matlab Toolbox DACE*. Informatics and Mathematical Modelling, DTU, Report imm-rep-2002-13 edn, 2002. Available from: http://www.imm.dtu.dk/hbn/publ/TR0213.ps.
48. Sacks J, Welch W, Mitchell T, Wynn H. Design and analysis of computer experiments. *Statistical Science* 1989; **4**(4):409–435.
49. Giunta A, Watson L. A comparison of approximation modeling techniques: polynomial versus interpolating models. *Seventh AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, MO, U.S.A., 1998; 1–13. *AIAA 98-4758*.
50. Bashir O, Willcox K, Ghattas O, van Bloemen Waanders B, Hill J. Hessian-based model reduction for large-scale systems with initial condition inputs. *International Journal for Numerical Methods in Engineering* 2008; **73**(6):844–868. DOI:10.1002/nme.2100.
51. Brooks A, Hughes T. Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1982; **32**:199–259.