



HAL
open science

Towards simultaneous reduction of both input and output spaces for interactive simulation-based structural design

Balaji Raghavan, Liang Xia, Piotr Breitkopf, Pierre Villon

► To cite this version:

Balaji Raghavan, Liang Xia, Piotr Breitkopf, Pierre Villon. Towards simultaneous reduction of both input and output spaces for interactive simulation-based structural design. *Computer Methods in Applied Mechanics and Engineering*, 2013, 265, pp.174-185. 10.1016/j.cma.2013.06.010 . hal-00977903

HAL Id: hal-00977903

<https://hal.science/hal-00977903>

Submitted on 14 Apr 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards simultaneous reduction of both input and output spaces for interactive simulation-based structural design

Balaji Raghavan, Liang Xia, Piotr Breitkopf*, Pierre Villon

Laboratoire Roberval UMR 7337 UTC-CNRS, Université de Technologie de Compiègne, Labex MS2T

* Corresponding author : piotr.breitkopf@utc.fr

Abstract

Engineering design problems generally involve a high-dimensional input space of design variables yielding an output space by means of costly high-fidelity evaluations. In order to decrease the overall cost, reduced-order models for the output space such as Proper Orthogonal Decomposition (POD) and proper Generalized Decomposition (PGD) are an active area of research. However, little research has been conducted into alleviating the problems associated with a high-dimensional input space. In addition to higher dimensionality being an impediment to efficient design by itself, complex shapes involve a high number of explicit/implicit constraints restricting the design space. Geometric parameterization methods in traditional CAD present difficulties in expressing these constraints leading to a high failure rate and the generation of inadmissible shapes. In this paper, we propose a simultaneous meta-modeling protocol for both input and output spaces. We perform a reparametrization of the input space using constrained shape interpolation by introducing the concept of an α -manifold of admissible meshed shapes. The output space is reduced using constrained Proper Orthogonal Decomposition. By simultaneously using meta-modeling for both spaces, we facilitate *interactive* design space exploration for the purpose of design. The proposed approach is applied to the industrial problem of designing a car engine intake port.

Keywords: Input space reduction, model reduction, finite elements, design, diffuse approximation.

1. Introduction, literature reviewed and motivation for research

When "high fidelity" computer simulations (finite elements, finite volumes, etc.) are used for calculating the objective functions and nonlinear constraints in the process of optimizing mechanical systems, the CPU time frequently becomes disproportionately large. This is due to the cost of individual simulations, the number of simulations or function evaluations needed, and the various additional aspects like license management of the legacy codes and the stability of a complex simulation chain involving multiple codes. All of these brought about a need for the development and validation of efficient surrogate-based methods in design optimization. Some of the recent advances in surrogate-based design methodology have been thoroughly discussed in [1]. In physical modeling, several techniques have been used to replace a complicated numerical model by a lower-order meta-model, like polynomial response surface methodology (RSM), kriging, least-squares regression and Moving Least Squares [2]. Surrogate functions and reduced-order meta-models have also been used in control systems to reduce the order of the overall transfer function [3]. [4] proposed a goal-oriented, model-constrained optimization framework. A popular physics-based meta-modeling technique consists of carrying out the approximation on the full vector fields using PCA and Galerkin projection [5] in CFD [6] as well as in structural analysis [7]. This approach has been successfully applied to a number of areas such as flow modeling [8, 9] optimal flow control [10], aerodynamics design optimization [11] or structural mechanics [12].

However, this requires manipulation of the input variables \bar{V} . Now for complex shapes the dimensionality d can be very high [13] and can greatly exceed the *intrinsic dimensionality* of the design problem. Another far more serious implication is the generation of inadmissible/infeasible structural shapes [14], which could

eventually lead to crashes of either the mesh generator or the solver. This was pointed out in [15] where *none* of the POD solutions, and only a few of the constrained POD solutions could be validated, since nearly ALL of the optimal geometries obtained were inadmissible and generated errors in the CAD stage!

The phenomenon of CAD failure due to the generation of inadmissible shapes is due to the difficulties in expressing all the technological and common sense constraints needed to convert a set of geometric parameters to an admissible shape. This issue is frequently encountered but remains relatively under-discussed in the context of design and optimization, and is the motivation for our work. While traditional shape morphing ([18, 19]) is popular, it requires a single reference shape and an arbitrary definition of the morphing boxes and control points and does not always yield exploitable CAD models.

A final inconvenience is that gradients/sensitivities need to be calculated using either finite differences or the Adjoint method [16]. All in all, geometric parameterization is an impediment to truly non-intrusive [17] optimization using a clearly separated *offline/online* approach that would allow for interactive design using, for example, a tablet PC.

The authors have not observed much if any research into using decomposition-based surrogate models for reducing dimensionality of the design domain in design or shape optimization [20]. This area is promising considering the obvious advantages of having far fewer parameters describing the domain with **implicit** verification of technological/admissibility constraints: better applicability to gradient-based solvers due to reduced dimensionality, and from the implementation point of view, a separation between the CAD and the simulation phases in system design.

In this paper, we develop an approach that builds up a design space by *learning* using *shape interpolation* between shape/mesh instances given by a sequence of parameter values. The input space is reduced using first direct POD on a set of *admissible* structural shapes in the *local neighborhood* of the evaluation point, followed by analyzing the *inter-relationship* between the projection coefficients α 's in this local neighborhood ($\mu(\bar{\alpha}) = 0$), represented using a local parametric expression $\bar{\alpha} = \bar{\alpha}(\bar{t})$. For this we introduce the concept of the α -manifold. The output space (physics) is modeled using constrained Proper Orthogonal Decomposition [21] giving the smallest set of coefficients $\beta_1 \dots \beta_m$ needed to conserve *linear* objective and constraint functions. The POD coefficients for the shape *and* physics are then analyzed *together* to get the local parametric expression for both α 's as well as β 's in the neighborhood of the evaluation point.

In addition, our approach gives an elegant, practical and straightforward method to compute the so-called "shape derivatives" of the performance objective(s), which are increasingly popular as used in "shape calculus" [34, 28, 29, 35].

The methodology and the overall algorithm are described in the next section with the help of a simple structural test case, and then applied to an industrial shape optimization problem in section 3 with a full set of numerical results and discussion before the concluding paragraph.

2. Meta-modeling for both the output and input spaces

On account of the computational cost of launching a number of "high fidelity" computer simulations and factors such as license management of the legacy codes and complex computing chains in engineering design optimization, research has been focussed on the development of efficient surrogate-based methods. One of the most popular among these is the method of Proper Orthogonal Decomposition (POD).

2.1. Output Space ROM

This Reduced Order Model (ROM) strategy for the output space approximates a physical field $\bar{v} \in \mathcal{R}^n$ in the vicinity of the current iteration point $\bar{V} \in \mathcal{R}^d$, where n is typically the size of a FE/CFD mesh/grid, and d is the design space dimensionality. The field vector $\bar{v}(\bar{V})$ may then be approximated around the nominal value \bar{v}_0 using a basis $\Psi = [\bar{\psi}_1 \dots \bar{\psi}_M]$ [5], typically obtained using a set of *a priori* computer experiments $\bar{V}^{(1)} \dots \bar{V}^{(2)}$ using Design of Experiments techniques [22], or other methods [23, 24]. This is followed by calculating the projection coefficients [17].

The reduced-order model is then built either by interpolation of Ψ e.g. [25] interpolated the basis Ψ vectors

onto the manifold of symmetric positive-definite matrices, or by interpolation of the β 's using kriging/Radial Basis Functions [21]/Diffuse Approximation [17, 26]. Assuming that only the projection coefficients β 's depend on the \bar{V} and that Ψ is constant for the design problem, followed by truncating the basis Ψ to a small number ($m \ll M$) of highly energetic modes, we get

$$\bar{v}(\bar{V}) \approx \tilde{v}(\bar{V}) = \bar{v}_0 + \sum_1^m \beta_i(\bar{V}) \bar{\psi}_i. \quad (1)$$

Since $\bar{\beta} = \bar{\beta}(\bar{V})$ we need to manipulate the vector of design variables \bar{V} either *after* optimization to verify the design or during the optimization/design to verify intermediate solutions. But as mentioned in the introduction working with \bar{V} can lead to issues like inadmissible geometries (infeasible \bar{V}) and thus CAD failures and frequently an elevated problem dimensionality, all of which are even more troublesome when using the meta-model in a non-intrusive procedure [15].

2.2. Input space ROM

We therefore seek to meta-model the *input space*, in other words the *structural shape* Ω itself in a POD-like manner in order to *implicitly* guarantee admissibility in β -space AND limit the dimensionality *while performing the design*. An ideal representation of Ω for this purpose (as will be explained in the next section) is by using the shape indicator function χ . The idea then is to perform POD on χ yielding

$$\chi = \chi_0 + \sum_i \alpha_i \bar{\phi}_i. \quad (2)$$

2.3. Combined input/output meta-modeling

The general approach to simultaneous reduction of both spaces of the design problem is shown in figure 1 with the input space meta-model on the left hand side, and the output space meta-model on the right.

The final step for the simultaneous meta-modeling is to obtain the relationship between the POD coefficients

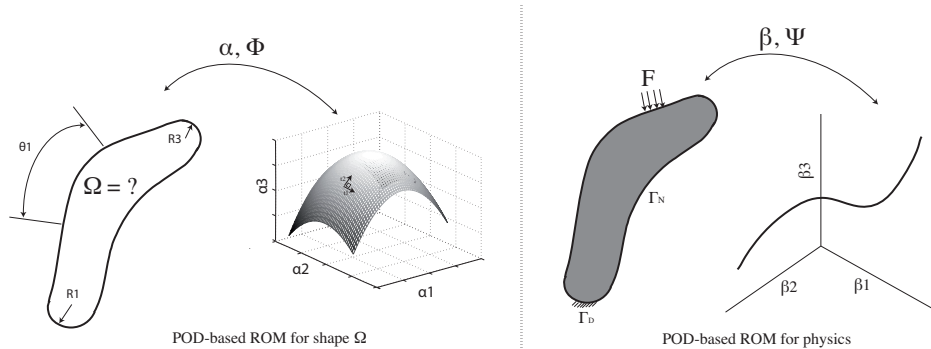


Figure 1: Basic idea of interactive design/optimization

for shape (α 's) and those for the physical fields (β 's). This can be easily obtained using a polynomial Response Surface:

$$\bar{\beta} = \bar{\beta}(\bar{\alpha}) = \bar{p}(\bar{\alpha})^T \bar{a} \quad (3)$$

where $\bar{p}(\bar{\alpha})$ is a series of monomials in $\bar{\alpha}$. In the next few sections, we will focus on the *input space*.

3. Analyzing the input space: the problem of admissibility and parameterization

The shape design problem in its fundamental form may be written as:

$$\begin{aligned}
 &\text{Find } \Omega^* = \text{Argmin}J(\Omega) \\
 &\quad \text{where } G(\Omega) \leq 0 \\
 &\text{div}(\sigma) = 0 \text{ on } \Omega, \sigma\bar{n} = \bar{F} \text{ on } \Gamma_N \\
 &\sigma\bar{n} = 0 \text{ on } \Gamma, \bar{u} = 0 \text{ on } \Gamma_D
 \end{aligned} \tag{4}$$

where $\Omega \subset \mathcal{R}^3$ is a representation of the structural shape, $\partial\Omega = \Gamma_D \cup \Gamma_N$ is the boundary and G is the full set of admissibility conditions on Ω . What we concern ourselves with here is the existence of G , so the details of stress and displacement boundary conditions though still present are not retained in the following discussion for the purpose of simplicity.

Now consider $\mathcal{A} \subset \mathcal{R}^3$, the *exhaustive* set of admissible shapes for the particular shape design problem. This means that if we limit the search for solutions to \mathcal{A}

$$\begin{aligned}
 &\text{Find } \Omega^* = \text{Argmin}J(\Omega) \\
 &\quad \text{where } \Omega \in \mathcal{A} \subset \mathcal{R}^3.
 \end{aligned} \tag{5}$$

The first step is then finding an adequate and convenient *representation* for the structural shape Ω . There are two possible approaches: Lagrangian and Eulerian.

3.1. Lagrangian representation and issues

The traditional CAD approach is a *local* approach to shape parameterization that attempts to express a complex structural shape as a set of geometric primitives (rectangles, circles/cylinders, splines, NURBS [?],etc), giving us a shape representation $\Omega = \Omega(\bar{V})$ using a bounded vector of geometric parameters $\bar{V} \in [\bar{V}_{min}, \bar{V}_{max}] \subset \mathcal{R}^d$.

This is convenient in allowing us to visualize a complex shape as a combination of simpler shapes. However, since this approach is local, it lacks a global comprehension of overall shape, topology and thus admissibility. In addition, there are redundancies possible in the final shape description.

The main issue is that it is difficult to express admissibility conditions (e.g. tangentiality in figure 2(a)) for complex shapes, and this can cause CAD failures despite respecting geometric bounds [14], in addition to possibly *overestimating* dimensionality for complex shapes [13], an impediment to optimization algorithms. Finally, the derivatives and Hessians are not easy to evaluate (if not available analytically) and frequently expensive (finite differences/adjoint method). For all these reasons, it is not always ideal for a non-intrusive design procedure.

The shape design problem is now posed in the familiar form:

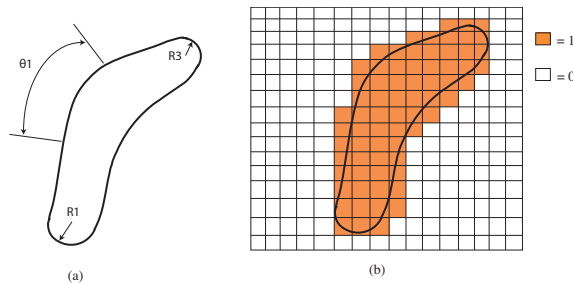


Figure 2: (a) Lagrangian representation, and (b) Eulerian representation of a structural shape Ω

$$\begin{aligned}
& \text{Find } \bar{V}^* = \text{Argmin} J(\Omega(\bar{V})) \\
& \bar{V} \in [\bar{V}_{min}, \bar{V}_{max}] \subset \mathcal{R}^N \\
& \text{subject to: } Y(\bar{u}(\bar{V})) = 0, G_L(\bar{V}) \leq 0
\end{aligned} \tag{6}$$

where $Y(u)$ represents the previously stated stress/displacement constraints, evaluated using FEA (for example) and G_L represents the eventually implicit *admissibility* conditions on \bar{V} such that:

$$\mathcal{A} = \{\Omega(\bar{V}) | G_L(\bar{V}) \leq 0\} \tag{7}$$

Since G_L is difficult to express, the design algorithm takes a more or less trial-and-error format and the entire chain can be thrown into disarray the instant a design is found to be infeasible. The key then is to attempt to find the subset of admissible shapes so that we can avoid this happening.

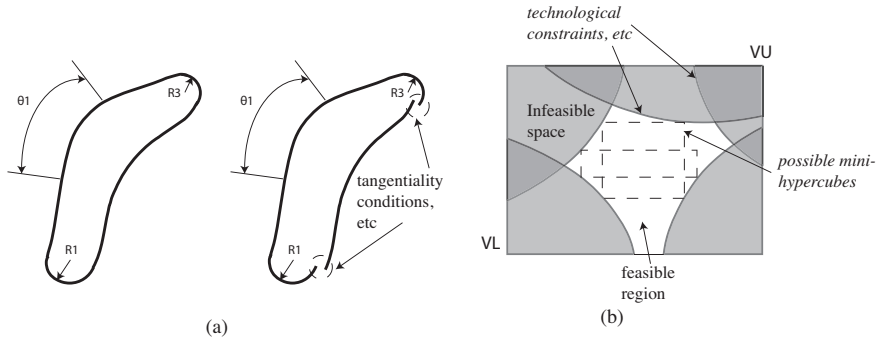


Figure 3: (a) Technological and admissibility constraints (b) Geometric space not a hypercube

Admissibility conditions need to be enforced on the geometric parameters (figure 3(a)) to ensure an admissible shape. By not explicitly enforcing these conditions - as is usually the case with traditional CAD, *ergo* the failures/crashes - we are incorrectly assuming that the design domain is a hypercube in \mathcal{R}^N , as illustrated in figure 3(b) thus limiting the design domain.

3.2. Eulerian representation

Eulerian representations [27] (figure 2(b)) use a *fixed* reference grid with a pre-determined resolution and fit the set of structural shapes to this grid.

In other words we represent a shape Ω by the *indicator function* $\chi(\bar{x})$: a unique, common and parameter-free representation of structural shape, where:

$$\begin{aligned}
\chi(\bar{x}) &= 1, \bar{x} \in \Omega \\
&= 0, \bar{x} \notin \Omega
\end{aligned} \tag{8}$$

Where again $\Omega \subset \mathcal{R}^2$ (\mathcal{R}^3 in 3D) is the set of all points in space contained in the shape χ which may be obtained in discrete form in several ways for negligible computational cost; for e.g. "voxellization" to create a binary array [31] $S^i \in \mathcal{R}^{N_c}, i = 1..M, N_c = \text{grid resolution}$, or the level sets approach [28, 30].

The Eulerian representation, if adopted directly as a parameterization, has some advantages over its Lagrangian counterpart: it gives us a *global, unique* description of shape and topology. Since it is basically an *image*, this representation is intrinsically linked with engineering and visualization (pixel/voxel maps, level sets). However, the Eulerian approach by itself does not permit us to *reliably* interpolate between admissible shapes.

When working with complex shapes, it would be desirable to work directly in "shape space" as we will explain in section 5.

4. Simultaneous meta-modeling using "shape space" and POD

The key points of our approach in this paper are as follows:

1. We build the parameterization scheme (shape space) directly by "learning" directly from example admissible **shapes** (regardless of the original parameterization scheme).
2. We perform POD directly on the structural shapes in *alocal neighborhood* of admissible shapes.
3. We do *not* truncate the basis, i.e. we retain ALL the modes without losing precision. Instead we look at the inter-relationship between the projection coefficients.
4. We calculate gradients, Hessian etc in this *local shape space* to get the so-called shape derivatives of the performance objectives.

To this end, consider a typical structural design problem in equation 9 (equivalent of (6)) with geometric dimensionality d with performance objective $J(\bar{V})$ (calculated using FEM):

$$\begin{aligned}
 \text{Find } \bar{V}^* = & \underset{\bar{V} \in [\bar{V}_{min}, \bar{V}_{max}] \subset \mathcal{R}^d}{\text{Argmin}} J(\bar{U}(\bar{V})) \\
 \text{such that } & \mathbf{K}(\bar{V})\bar{U}(\bar{V}) = \bar{F} \text{ (equation of equilibrium)} \\
 & \text{and } \bar{C}(\bar{V}) = \bar{0} \text{ (shape/admissibility constraint)} \\
 & \text{and } A^H(\bar{V}) \geq A_o \text{ (mass constraint)}
 \end{aligned} \tag{9}$$

where $\mathbf{K}(\bar{V})$ (stiffness matrix), $\bar{U} = \mathbf{K}^{-1}\bar{F}$, force \bar{F} and A_o are constants, $\bar{C} : [\bar{V}_{min}, \bar{V}_{max}] \rightarrow \mathcal{R}^{n_c}$ is the shape constraint function that is unknown/difficult to express analytically (n_c =size of constraint vector) where $\forall \bar{V} \in \mathcal{E}_e \subset [\bar{V}_{min}, \bar{V}_{max}]$, $\bar{C}(\bar{V}) = \bar{0}$, \mathcal{E}_e is the subset of admissible shapes.

So we need to restrict our search to the subset of admissible shapes i.e. \mathcal{E}_e in the original design domain.

NOTE: We need not concern ourselves with the mass constraint at this stage, although we could *further* restrict the design space to the admissible shapes that also satisfy the mass constraint if needed. So we treat the mass constraint separately from the shape admissibility constraint.

Now, the basic premise is to represent a design point (i.e. admissible shape) in a reduced (α, Φ) -space where:

- The basis Φ is calculated *locally* using sample admissible shapes
- $\bar{\alpha}$ is obtained by projecting a design point on to Φ .

We transform the problem from the original space to this reduced space in such a way as to *implicitly* satisfy the admissibility criterion i.e. $\bar{V}(\bar{\alpha}) \in \mathcal{E}_e$. Equation (9) would then reduce to a form:

$$\begin{aligned}
 \text{Find } \bar{\alpha}^* = & \underset{\Phi = \text{const}}{\text{Argmin}} \{J(\bar{\alpha}, \Phi)\}, \\
 \text{s.t. } \bar{\alpha} \in & [\bar{\alpha}_{min}, \bar{\alpha}_{max}] \subset \mathcal{R}^m, c(\bar{\alpha}) = 0, a^h(\bar{\alpha}) \leq 0,
 \end{aligned} \tag{10}$$

where $m \ll d$. The basis Φ is fixed in (10) since even though it will vary globally, (10) is applied locally. We achieve the transformation from equation 9 to 10 in a series of steps outlined in the next few subsections.

4.1. Design domain and shape indicator functions

We first study the range of admissible shapes (i.e. snapshots [32]) sweeping the neighborhood of the evaluation point in the physical design domain, typically in a Lagrangian description with a sampling of the geometry-based design variables within their range $\bar{V} \in [\bar{V}_{min}, \bar{V}_{max}] \subset \mathcal{R}^d$, (this could simply be the finite set of points describing the edges/boundaries of a series of CFD meshes/grid points) for an initial random sampling of M **admissible** designs $\bar{V}^1.. \bar{V}^M$ (i.e. satisfying $\bar{C}(\bar{V}) = \bar{0}$).

NOTE: No numerical analysis is performed for these M snapshots that are solely to analyze the overall "shape" variation and gauge the intrinsic dimensionality of the design domain.

We next convert each of these snapshots to the corresponding shape indicator functions $S^1...S^M$ in one of the ways described earlier. here *voxelization* has been used.

4.2. Principal Components Analysis of the Shapes

This is the first stage of the model reduction. We calculate the deviation and covariance matrices D_s and C_v for the snapshots $S^1..S^M$:

$$D_s = [S^1 - \bar{S} \quad S^2 - \bar{S} \quad \dots \quad S^M - \bar{S}] \quad \text{and} \quad C_v = D_s D_s^T \quad (11)$$

allowing us to express any S^j in terms of the eigenvectors $\bar{\phi}_i$ of C_v where $M \ll N_c = \text{number of snapshots}$, $S^i = i^{th}$ individual snapshot and \bar{S} is the mean snapshot.

$$S^j = \bar{S} + \sum_{i=1}^M \alpha_{ij} \bar{\phi}_i, \quad \alpha_{ij} = \bar{\phi}_i^T S^j \quad (12)$$

for the j th indicator function. (NOTE: $-1 \leq \bar{\phi}_i \leq 1$, $0 \leq \bar{S} \leq 1$ while S^j is binary). The usual reduction approach would be to limit the basis to the first $m \ll M$ most "energetic" modes:

$$\tilde{S}^j = \bar{S} + \sum_{i=1}^m \alpha_{ij} \bar{\phi}_i \quad \text{and} \quad \epsilon(m) = 1 - \frac{\sum_{i=1}^m \lambda_i}{\sum_{i=1}^M \lambda_i} \quad (13)$$

where $\epsilon(m)$ is the relative projection error, λ 's are the eigenvalues of C_v .

However, the last equation does not provide a sufficient basis for establishing the value of m since we need to specify a threshold value for ϵ . Since the α 's can not be directly interpreted as new design variables without taking into account the possible inter-relationships that exist between them so as to render feasible shapes. In addition, truncating the basis Φ would limit precision. Therefore, instead of truncating the basis, we instead analyze the inter-relationships between these α 's obtained from the M snapshots in order to detect the true dimensionality (p) of the design domain.

4.3. Concept of "Shape Space" using the α -manifold

We represent the inter-relationship between the α -coefficients by a manifold in \mathcal{R}^M space, which we call the α -manifold. For illustrative purposes, we will use some simple test-cases using a global sampling of 2000 points:

1. A Plate with a circular hole of varying radius, shown in figure 4.
2. Plate with two circular holes of independently varying radii, shown in figure 5.

In all cases, we consider the design domain of admissible shapes, bounded e.g $R_{min} \leq r_1, r_2 \leq R_{max}$ for the two holes plate, and each time, the approach is able to detect the feasible region and dimensionality from a set of *shape snapshots* (images) regardless of the geometric parameterization used. For example, in the first case, there is only one parameter (hole radius r) the dimensionality is clearly 1 and to detect this we generate a set of random snapshots by varying r followed by PCA on the S^i giving us a set of α 's. Figure 4(b) shows the one-dimensional α -manifolds clearly indicating that the design domain is parametrized by ONE single parameter t . Similarly, the α -manifolds for the second case are 2D in figure 5(b) (2 parameters) indicating a local parametric expression t_1, t_2

While 1D and 2D manifolds have been chosen since manifolds of higher order cannot be visualized in 3 dimensions, it is important to state that:

1. Any point outside the manifold will result in NON admissible shapes.
2. Any additional admissible point generated in \bar{V} space and then projected onto the existing basis ($\bar{\Phi}$) will always be on this manifold.

This illustrates that the α -manifold truly approaches the shape space for the given design problem and thus captures its intrinsic dimensionality (p). This hypothesis is important, since at no point do we explicitly *inform* the algorithm about the dimensionality or even a set of parameters (geometric or otherwise).

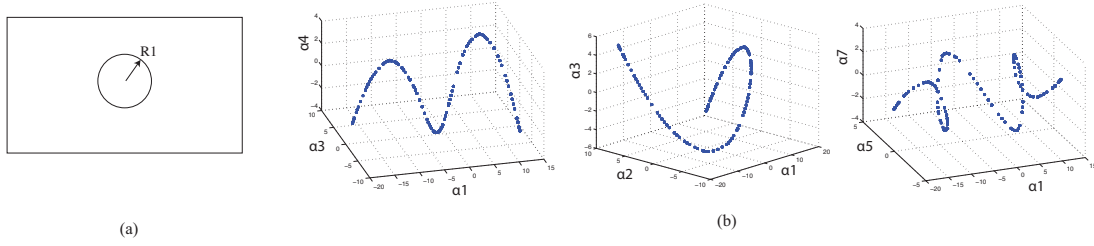


Figure 4: α -manifolds for plate with a circular hole of fixed center varying radius

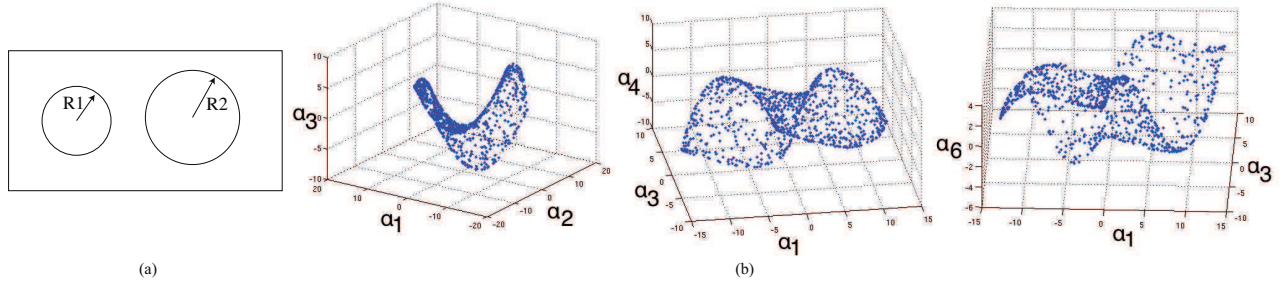


Figure 5: α -manifolds for plate with two circular holes - fixed centers, independently varying radii

4.4. Model reduction using the α -manifold instead of basis truncation

In the general case, we obtain a p -dimensional manifold with the local parametric expression $\bar{\alpha} = \bar{\alpha}(\bar{t})$, $\bar{t} \in R^p$, $p \leq d$. $t_1 \dots t_p$ are local parameters that allow us to move along the tangent plane to the manifold. For e.g., from the previous section - for a plate with an elliptical hole of varying radii, the α 's form a set of two-dimensional manifolds (figure 6) rather than a cloud of points in 3D space *regardless of the particular triplet of modes used*, clearly indicating that the design domain is parametrized by **two** local parameters t_1, t_2 . This means that $\alpha_1 = \alpha_1(t_1, t_2)$, $\alpha_2 = \alpha_2(t_1, t_2)$

Basically, points lying outside the manifolds always produce inadmissible shapes so the surfaces $\alpha_1, \alpha_2, \dots$ vs $t_1, t_2 \dots t_p$

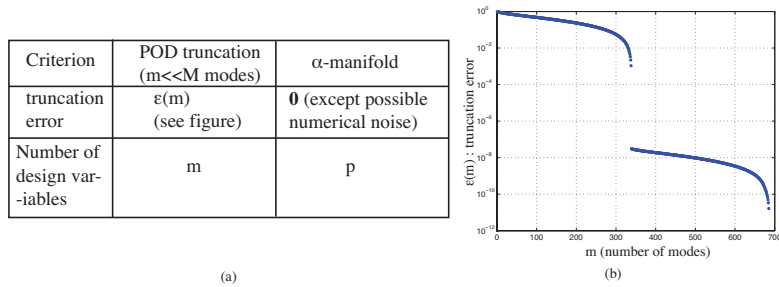


Figure 6: (a) Traditional POD truncation to m modes versus the α -manifold (feasible region of admissible shapes), (b) $\epsilon(m)$

may be interpreted as the set of all possible "constraints" (direct geometric constraints, technological constraints etc that are difficult to express mathematically) on the geometric parameters \bar{V} in the α -space. In other words, **the α -manifold represents the feasible region of admissible shapes**. In our work, we introduce this local parametric expression $\bar{\alpha} = \bar{\alpha}(t_1, \dots, t_p)$ using Diffuse Approximation [26].

4.5. Local manifold construction in the neighborhood of the design point

We present here a formal approach to *locally* identify the system dimensionality p from the α -manifolds. Consider the snapshots in α -space: $\bar{\alpha}^1, \dots, \bar{\alpha}^M \in \mathcal{R}^M$. We would like to implement an algorithm that:

1. Detects the "true"/inherent dimensionality of the design domain ($p \leq M$) from the local rank of the α -manifold in the vicinity of the evaluation point, so that the feasible region may (locally) be expressed as $\alpha_1 = \alpha_1(t_1..t_p), \dots, \alpha_M = \alpha_M(t_1..t_p)$.
2. Constrains the evaluation point ($\bar{\alpha}^{ev}$) to stay on the feasible region of admissible shapes, during the course of the design process.

To locally detect the dimensionality of the $\alpha_1 \dots \alpha_M$ hyper-surface in the neighborhood of $\bar{\alpha}^{ev}$, we extend the method of Fukunaga and Olsen [33]. First, we establish a sufficiently dense local neighborhood. Let $\bar{\beta}_1 \dots \bar{\beta}_{nbd}$ be nbd neighboring points in α -space, we next use a polynomial basis centered on $\bar{\alpha}^{ev}$

$$\mathbf{P}(\bar{\alpha}^{ev}) = \begin{bmatrix} 1 & \beta_1^1 - \alpha_1^{ev} & \beta_2^1 - \alpha_2^{ev} & \dots & \beta_M^1 - \alpha_M^{ev} & \dots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & \beta_1^{nbd} - \alpha_1^{ev} & \beta_2^{nbd} - \alpha_2^{ev} & \dots & \beta_M^{nbd} - \alpha_M^{ev} & \dots \end{bmatrix} \quad (14)$$

with an appropriate weighting function (e.g. Gaussian $w(d) = e^{-cd^2}$) and assemble the moment matrix $M_t = P^T W P$, where W is the diagonal matrix whose elements correspond to the weighted contributions of the nodes $\bar{\beta}_1 \dots \bar{\beta}_{nbd}$.

Next, we detect the local rank of the manifold by calculating the rank of the moment matrix from the number of singular values of M_t , this gives us the dimensionality $p \leq M$.

4.6. Diffuse predictor-corrector "walking" algorithm

This algorithm will bring the design point (in α -space) in the interactive design procedure back down to the α -manifold in subsequent iterations using a predictor-corrector scheme. The local surface tangent to the manifold is defined with respect to the tangent space iteratively updated. We "walk" along the manifold using a Diffuse Approximation-based scheme consisting of the following steps (figure 8).

1. Let Q_i be the current design point: $\bar{\alpha}_{ev}$ in α -space. In the predictor stage, using a single Quasi-Newton

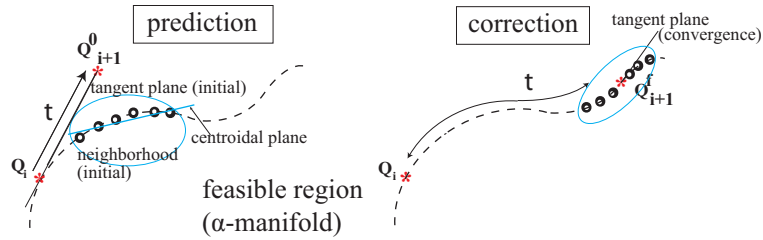


Figure 7: Overall shape interpolation algorithm for design space exploration

step from Q_i , let Q_{i+1}^0 be the new candidate point that needs to be brought back on to the manifold/feasible region (correction phase). We first establish the α -neighborhood $\bar{\mu}_1 \dots \bar{\mu}_{nbd}$ of Q_{i+1}^0 .

2. Calculate the centroid $\bar{\mu}_m = (\sum_{i=1}^{nbd} \bar{\beta}_i) / nbd$.
3. Find the centroidal plane for the neighborhood from the eigenvectors of the covariance matrix C_{nbd} , the first eigenvector representing the plane normal:

$$C_{nbd} = (1/nbd) \sum_{i=1}^{nbd} (\bar{\mu}_i - \bar{\mu}_m)(\bar{\mu}_i - \bar{\mu}_m)^T, (\bar{v}_1, \bar{v}_2 \dots) = \text{eigenvectors}(C_{nbd}) \quad (15)$$

4. Project the evaluation point as well as the neighborhood points in the local coordinate system $\bar{v}_1, \bar{v}_2 \dots$ (origin at centroid $\bar{\beta}_m$) to get the local co-ordinates $h, t_1 \dots t_p$ for a general $\bar{\alpha}$ where h is height over the centroidal plane using:

$$h = \bar{v}_1^T (\bar{\alpha} - \bar{\mu}_m), t_1 = \bar{v}_2^T \bar{\alpha}, \text{ etc} \quad (16)$$

5. We next obtain the shape (indicator function) and calculate J for the evaluation point. We recreate the structural shape for an arbitrary design point (\bar{t}) using the α coefficients obtained from \bar{t} (location on the α -manifold), and thus the indicator function \tilde{S} :

$$\tilde{S}(\bar{t}) = \bar{S} + \sum_{i=1}^m \alpha_i(\bar{t}) \bar{\phi}_i \quad (17)$$

To calculate the objective function $J(\tilde{S})$ we use a response surface between the output-space (previous section) and input space meta-models giving us the output space POD coefficients $\beta_1 \dots \beta_m$ as $\bar{\beta}(\bar{\alpha}(\bar{t}))$. We now get the objective function using $\bar{J}(\bar{v})$ where \bar{v} is the physical vector field reduced by the physics meta-model.

$$J \approx J(\bar{v}) = J(\bar{v}_0 + \Psi \bar{\beta}(\bar{\alpha}(\bar{t}))) \quad (18)$$

6. Do the Diffuse Approximation for $\bar{\alpha}^{ev}$ as well as the objective function J using the nb d neighboring points to get the local surface $h = \tilde{h}(t_1 \dots t_p)$ using a polynomial basis $\mathbf{P}_q(\bar{\alpha}^{ev})$ with weighting \mathbf{W} , and $J = \bar{J}(t_1 \dots t_p)$ in a similar fashion using a basis $\mathbf{P}_j(\bar{\alpha}^{ev})$.

$$[\tilde{h}(\bar{t}^{ev}), \frac{\partial \tilde{h}}{\partial t_1}(\bar{t}^{ev}), \frac{\partial \tilde{h}}{\partial t_2}(\bar{t}^{ev}), \dots]^T = (\mathbf{P}_q^T \mathbf{W} \mathbf{P}_q)^{-1} \mathbf{P}_q^T \mathbf{W} [h_1, \dots, h_{nb}]^T \quad (19)$$

$$[\bar{J}(\bar{t}^{ev}), \frac{\partial \bar{J}}{\partial t_1}(\bar{t}^{ev}), \frac{\partial \bar{J}}{\partial t_2}(\bar{t}^{ev}), \dots]^T = (\mathbf{P}_j^T \mathbf{W} \mathbf{P}_j)^{-1} \mathbf{P}_j^T \mathbf{W} [J_1, \dots, J_{nb}]^T \quad (20)$$

where the term in the LHS is the local tangent hyper-plane at Q_0 in the neighborhood $\bar{\beta}^1 \dots \bar{\beta}^{nb}$.

7. We then project the point Q_{i+1}^0 onto this tangent plane to get the adjusted evaluation point Q_{i+1}^1 , and then repeat by finding the new neighborhood, tangent plane and projection point, etc till the evaluation point stops changing Q_{i+1}^f .

In this, we "walk" the evaluation point Q_i along the surface of the α -manifold to ensure that we stay in the domain of feasible solutions.

4.7. Gradient and Hessian: Diffuse "Shape derivatives"

Our concept of shape derivatives ∇J (and Hessian H with additional derivatives) is simply that of the "diffuse" derivatives [26] of J with respect to \bar{t} (from the previous section), since \bar{t} is simply the *local description* of the shape manifold:

$$\nabla J = \left[\frac{\partial \bar{J}(\bar{t})}{\partial t_1}, \frac{\partial \bar{J}(\bar{t})}{\partial t_2}, \dots \right] \quad (21)$$

which is then evaluated using equation (20). However, the above may also be written as:

$$\nabla J = \Phi^T \frac{dJ}{dS} \quad (22)$$

which is simply the *projection* of the *Eulerian derivative* onto the POD basis Φ .

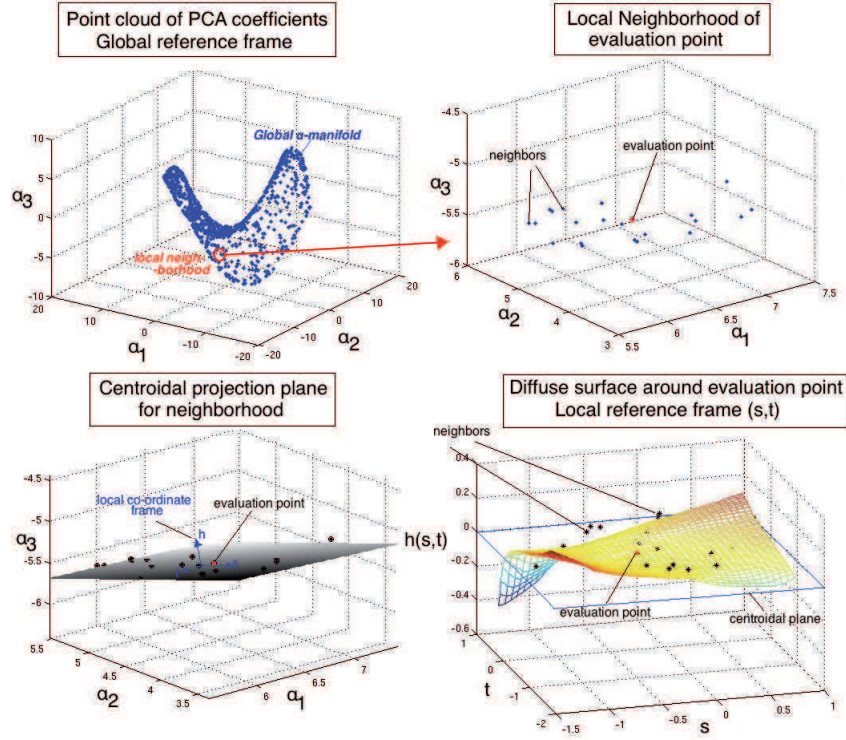


Figure 8: 2D interpretation of local Diffuse Approximation

4.8. Design problem in shape-space

The design problem in reduced-space now becomes:

$$\begin{aligned} & \text{Find } \bar{t}^* = \text{Argmin}_{\bar{t} \in \mathcal{R}^p} \{J(\bar{\alpha}(\bar{t}))\} \\ \text{s.t. } & c(\bar{\alpha}(\bar{t})) = 0, a^h(\bar{\alpha}(\bar{t})) \leq 0, \bar{\alpha}_{min} \leq \bar{\alpha}(\bar{t}) \leq \bar{\alpha}_{max} \end{aligned} \quad (23)$$

where c represents the admissibility constraint (i.e. the α -manifold), a^h represents the mass constraint, and $\bar{\alpha}_{min}$ and $\bar{\alpha}_{max}$ are the bounds on $\bar{\alpha}$.

In simple terms, we satisfy the admissibility constraint and bounds implicitly by constraining the evaluation point to stay on the manifold of admissible shapes by generating "neighbors" exclusively from admissible solutions close to the evaluation point, using tangent space construction by Diffuse Approximation. This is illustrated for a simple structural case in figure 9.

5. TEST-CASE: Design of 93-parameter engine intake port

5.1. Description of Test-Case, CAD issues and non-intrusive approach

This test-case is related to the design for performance improvement of an engine intake (figure 10(a)), originally proposed by Renault as a third benchmark test-case for the OMD2 project [36]. A sand model of this intake pipe is shown in figure 10(b). The intake pipe and port system has a very complex structural shape with physical details as shown in figure 10(c). Optimizing the combustion process within an engine block is central to the performance of many motorized vehicles. Associated with this process are two important performance objectives: the mass flow rate Q through the cylinder and tumble pattern of flow (vorticity τ), which optimize the mixing of fluid within each of an engine's cylinders. This variable 3D geometry is parameterized by using 93 bounded geometric design variables $\bar{V} \in [\bar{V}_L, \bar{V}_U] \subset \mathcal{R}^{93}$ in CATIA, involving an exorbitantly high CAD failure rate of over **60%**: tested at the beginning with a random sampling of

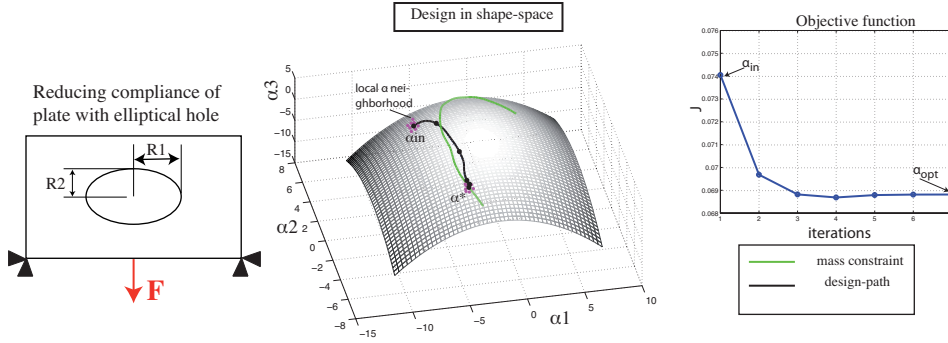


Figure 9: Design path in shape-space for a simple example

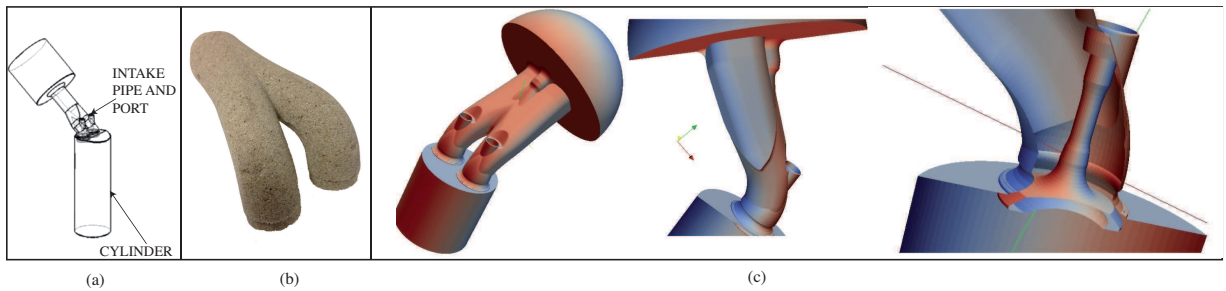


Figure 10: (a) Full system (b) Sand model of intake pipe (c) External & internal details of engine intake

1000 points in $[\bar{V}_L, \bar{V}_U]$ resulting in only 553 admissible shapes after regular CAD using CATIA. As in the previous test-case, this failure-rate of regular CAD was expectedly a serious impediment to standard search routines: gradient-based or otherwise, and also limiting the use of a standard POD to represent the physical solution fields alone since design points found using the ROM would frequently not be admissible.

5.2. Design of experiments and PCA

For the purpose of demonstrating the nature of the manifolds, 2300 design points were picked for CAD generation without meshing/CFD by a Latin Hypercube sampling between \bar{V}_{min} and \bar{V}_{max} , and these yielded a grand total of 1080 admissible designs.

These were then *voxelized* using a resolution of $300 \times 300 \times 300$ giving the indicator function snapshots $S^1 \dots S^{1080}$.

The 1080 snapshots were next decomposed using PCA/POD with a truncated basis of size 13, giving a set of $\alpha_1 \dots \alpha_{13}$ for each of the 1080 snapshots.

It is vital to understand that this procedure was performed *solely* for the purpose of demonstrating the manifolds, since only 7-8 snapshots in a *local neighborhood* are analyzed at any given stage.

5.3. Output Space meta-model

In this paper, we choose a simple linearly constrained POD approach [15] since the objective functions (flow rate Q and tumble T) are *linear* functions of the field variable vector compactly written in the form:

$$\bar{J} = [Q, T]^T = \mathbf{C}^T \bar{v} \quad (24)$$

This approach first modifies the basis $\Psi = [\bar{\psi}_1 \dots \bar{\psi}_m]$ itself in order to automatically conserve linear integral quantities *by design* followed by modifying the coefficients β_i to conserve quadratic objective functions/constraints if needed.

The first level involves QR decomposition of \mathbf{C}

$$\mathbf{C} = \mathbf{QR} = [Q_1 \quad Q_2] \begin{bmatrix} R_1 \\ 0 \end{bmatrix} \quad (25)$$

The relative snapshots are now projected on Q_2 :

$$\bar{v}_Q^{(k)} = Q_2^T (\bar{v}^{(k)} - \bar{v}_0) \quad (26)$$

where v_0 is the mean snapshot. The covariance matrix \mathbf{V}_Q is computed using

$$\mathbf{V}_Q = \sum \bar{v}_Q^{(k)} (\bar{v}_Q^{(k)})^T \quad (27)$$

The modes $\psi_1 \dots \psi_m$ are obtained from the formula

$$\Psi = \{Q_1, Q_2 \gamma\} \quad (28)$$

where γ are the first $m - p$ eigenvectors of \mathbf{V}_Q .

Finally, the *constrained* projector may be written as

$$\tilde{v}^{(k)} = \bar{v}_0 + \Psi \Psi^T (\bar{v}^{(k)} - \bar{v}_0) \quad (29)$$

or

$$\tilde{v}^{(k)} = \bar{v}_0 + \Psi \bar{\beta}^{(k)} \quad (30)$$

where the coefficients $\beta^{(k)}$ are given by projection of relative snapshots into the modal space

$$\alpha^{(k)} = \Psi^T (\tilde{v}^{(k)} - \bar{v}_0) \quad (31)$$

This basis Ψ automatically guarantees that $\mathbf{C}^T \tilde{v} = \mathbf{C}^T \bar{v}$ for a *linear* constraint matrix \mathbf{C} .

5.4. Results & Discussion

5.4.1. α -manifolds for the engine intake

The α -manifolds for the design problem of an engine intake are shown in figures 11. We obtain the same 3D extrusions regardless of the particular triplet of modes used for the calculation. This is very interesting since it indicates a dimensionality p **less than 3**, and this has been drawn using a set of snapshots corresponding to the set of admissible shapes generated using 93 geometric parameters.

Of course, this needs to be verified using the Fukunaga-Olsen algorithm [33] and this is shown in figure

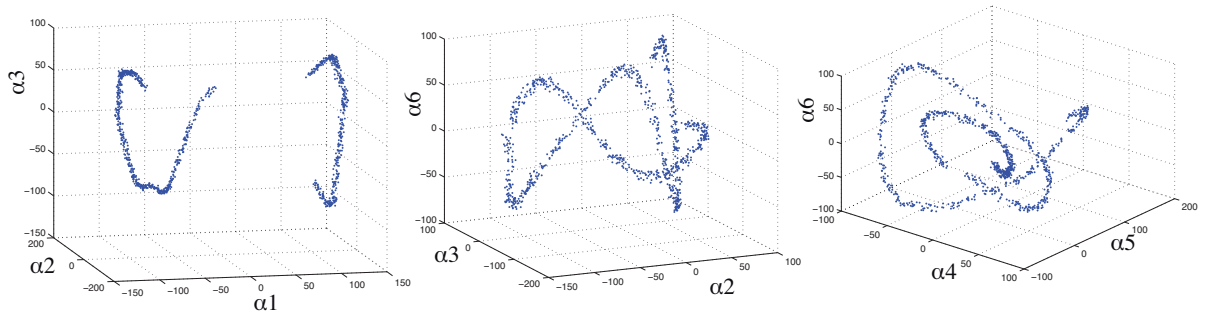


Figure 11: α -manifolds for the engine intake problem

12(a) using two Gaussian weighting functions for the neighborhood, and three values of modes retained (m). While mode truncation is NOT part of the proposed approach, using fewer modes in the *estimation* of p

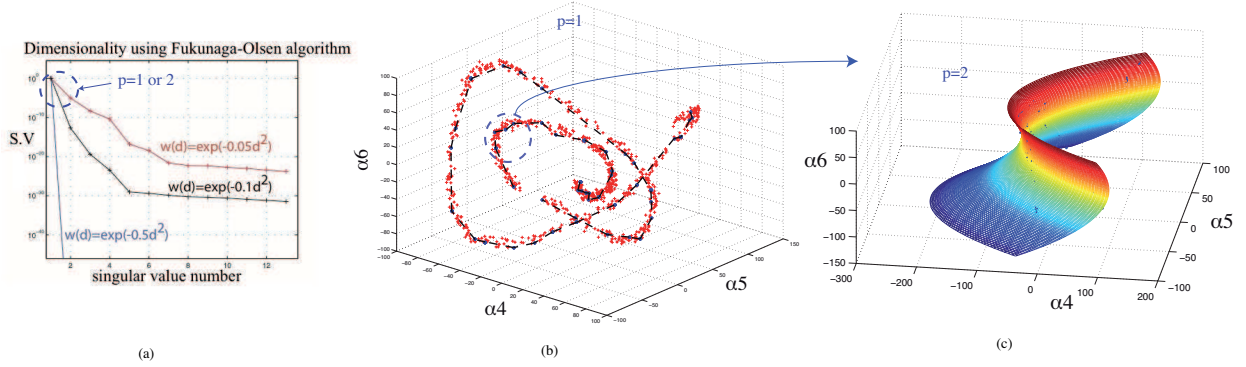


Figure 12: (a) Dimensionality p calculation, (b) Constrained shape interpolation using $p = 1$, (c) α -manifold approximation using $p = 2$ (normalized)

can definitely reduce the total computational effort without compromising the results. We can see that a $p = 1$ first-level approximation is possible (figure 12(b)).

It bears mentioning that the analysis of dimensionality from a cloud of points in ND -space is a multi-scale problem. At this level with the data points, we can build a $p = 1$ system, but if we zoom on a certain portion of the extrusion/curve and then resample the geometric space to generate 1000 points but in a smaller neighborhood, p might creep up to a value of 2 as illustrated in figure 12(c) showing the 2D α -manifolds (adjusted/normalized) for the same problem.

For the first stage, we will perform single-parameter shape morphing using a $p = 1$ approximation of the manifold, and for the second, we will use the zoomed-in approximation ($p = 2$) to build a 2D representation that we link the with CPOD meta-model for interactive design.

5.4.2. Single-parameter shape morphing for 93-parameter engine intake

Keeping the previous discussion in mind, we can safely build a single parameter ($p = 1$) approximation/representation of the engine intake design space. So from the set of α -manifolds, we generate a set of 58 points on the axis of the manifold, yet again using the Diffuse Approximation [26]. These are shown in figure 12(b).

We reconstruct the 58 admissible shapes using CATIA to generate smooth transitions in structural shape with a single t parameter. In the interest of providing an adequate representation of the great number of shape features changing while still being captured with a single parameter, we have selected 8 of these points and shown these in figure 13. In short, we are able to vary 93 geometric parameters using a single parameter (position along the axis of the manifold), giving us $\bar{V} = \bar{V}(\alpha_1(t), \alpha_2(t) \dots \alpha_M(t))$, while constantly staying in the feasible region. This will greatly simplify the task of optimizing the intake shape i.e. \bar{V}_{opt} for a given objective.

5.4.3. Design space exploration using 2 parameters and interactive design using meta-model for both spaces

Figure 14 shows 4 frames of the interactive design for the engine intake problem, using 2D α -manifolds in MATLAB. We see:

1. The flow streamlines within the zone of interest of the engine intake.
2. The physics of the design problem evolving with the CPOD coefficients (β_1, β_2) corresponding to the two objectives.
3. The α -manifold with the current design point (red) and 7-8 neighbors (green)

for 4 different locations on shape space, easily handled using even modest computing resources. Two points of "compromise" are obtained using the *predictor-corrector*

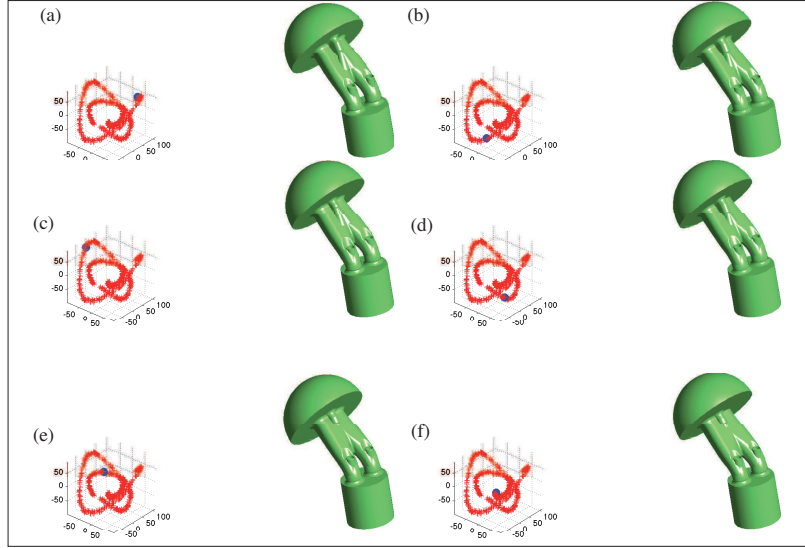


Figure 13: Shape morphing between admissible shapes of a 93-parameter engine intake, using a SINGLE parameter

No.	Q/Q_{max}	T/T_{max}
1	0.9482	0.4682
2	0.8981	0.7421

Table 1: Design points of compromise found

6. Computational effort and admissibility of CAD shapes generated

From the point of view of computational effort, there is significant saving in the calculation of shape derivatives of the objectives Q, T using the α -manifold in addition to the usual reduction in effort by using a constrained POD meta-model β_1, β_2 for the physical fields. This avoids the need for finite differencing/Adjoint equation and reduces the number of exact function evaluations needed.

From the point of view of optimization, at every stage, the optimization was performed with respect to 2 (local) variables instead of 6, and the total number of iterations from various starting points was usually between 5 and 10 (in the reduced $\alpha(\bar{t})$ space), but this can always be adjusted by controlling the step size of the **single** Quasi-Newton "leap" at every iteration. In addition to reducing the dimensionality of the problem, the presented predictor-corrector approach *consistently* produced admissible solutions from various initial points, despite the 60% failure rate of traditional CAD when optimizing/designing without the approach.

Conclusions

In this paper, the authors have introduced a concept of simultaneous POD on structural shapes as well as the outputs variables.

The overall interpolation technique is nonlinear, and is constrained to produce only shapes from an abstract manifold in shape space induced by learning. The non-varying zones used for boundary conditions are naturally preserved and additional constraints may be imposed using constrained versions of Proper Orthogonal Decomposition or simply further filtering the admissible solutions used for the "learning" phase. The method is intuitive and makes physical sense. Interpolation in *shape space* yields the minimum possible

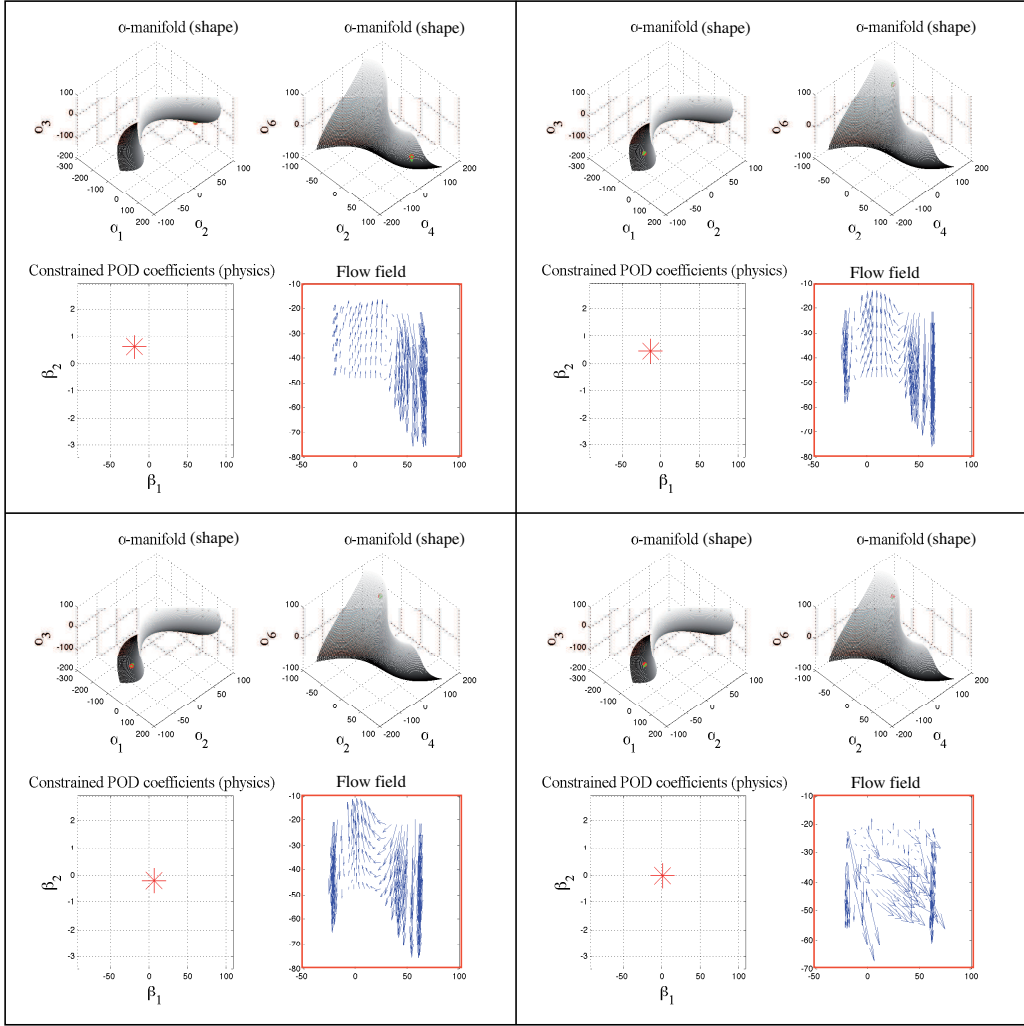


Figure 14: Interactive design using α -manifolds and CPOD coefficients for the 6-parameter intake problem

system dimensionality while guaranteeing solution admissibility. This allows us to develop a reparameterization procedure fine-tuned for the particular type of shape being designed. The main message here is that the feasible region is the α -manifold, and that moving along this manifold allows us to sweep the design space of admissible shapes.

The methodology definitely needs more evolution before it can be used as a robust protocol: testing the α -manifolds for continuity, branching, more complex test-cases etc.

Acknowledgements

This work has been supported by the French National Research Agency (ANR), through the COSINUS program (project OMD2 no. ANR-08-COSI-007). The authors acknowledge the Projet Pluri-Formations PILCAM2 at the Universite de Technologie de Compiègne (URL: <http://pilcam2.wikispaces.com>) for HPC resources. The second author acknowledges the support from the China Scholarship Council (CSC).

References

- [1] A. I. J. Forrester, A. J. Keane, Recent advances in surrogate-based optimization., *Progress in Aerospace Sciences* 45 (2009) 50–79.
- [2] P. Breitkopf, H. Naceur, A. Rassineux, P. Villon, Moving least squares response surface approximation: Formulation and metal forming applications, *Computers and Structures* 83 (2005) 1411–1428.
- [3] K. Willcox, J. Peraire, Balanced model reduction via the proper orthogonal decomposition, *AIAA Journal* 40 (2002) 2323–2330.
- [4] T. Bui-Thanh, K. Willcox, O. Ghattas, B. van Bloemen Waanders, Goal-oriented, model-constrained optimization for reduction of large-scale systems, *Journal of Computational Physics* 224 (2007) 880 – 896.
- [5] G. Berkooz, P. Holmes, J. L. Lumley, The proper orthogonal decomposition in the analysis of turbulent flows, *Annu Rev Fluid Mech* 25 (1993) 539–575.
- [6] C. Audouze, F. De Vuyst, P. Nair, Reduced-order modeling of parameterized pdes using time-space-parameter principal component analysis, *International Journal of Numerical Methods in Engineering* 80 (2009) 1025 – 1057.
- [7] L. Cordier, B. A. El Majd, J. Favier, Calibration of pod reduced order models using tikhonov regularization, *International Journal for Numerical Methods in Fluids* 63 (2010) 269–296.
- [8] R. A. Sahan, H. Gunes, A. Liakopoulos, A modeling approach to transitional channel flow, *Computers and Fluids* 27 (1998) 121–136.
- [9] M. Couplet, C. Basdevant, P. Sagaut, Calibrated reduced-order pod-galerkin system for fluid flow modeling, *Journal of Computational Physics* 207 (2005) 192–220.
- [10] S. S. Ravindran, A reduced-order approach for optimal control of fluids using proper orthogonal decomposition, *International Journal for Numerical Methods in Fluids* 34 (2000) 425–448.
- [11] P. A. LeGresley, J. J. Alonso, Airfoil design optimization using reduced order models based on proper orthogonal decomposition, *Fluids 2000 Conference and Exhibit*, Denver, CO (2000).
- [12] J.-L. Dulong, F. Druesne, P. Villon, A model reduction approach for real-time part deformation with nonlinear mechanical behavior, *International Journal on Interactive Design and Manufacturing* 1 (2007) 229–238.
- [13] W.-H. Zhang, P. Beckers, C. Fleury, Unified parametric design approach to structural shape optimization, *International Journal for Numerical Methods in Engineering* 38 (1995) 2283–2292.
- [14] A. Veiz, M. Egerland, Cad-parametric optimization with optislang-ansys workbench, *4th Weimar optimization and stochastic days* (2007).
- [15] B. Raghavan, M. Hamdaoui, M. Xiao, P. Breitkopf, P. Villon, A bi-level meta-modeling approach for structural optimization using modified pod bases and diffuse approximation, *Computers & Structures* (2012).
- [16] F. Van Keulen, R. Haftka, N. Kim, Review of options for structural design sensitivity analysis. part 1: Linear systems, *Computer Methods in Applied Mechanics and Engineering* 194 (2006) 3213–3243.
- [17] R. F. Coelho, P. Breitkopf, C. Knopf-Lenoir, Bi-level model reduction for coupled problems, *Int J Struc Multidisc Optim* 39 (2009) 401–418.
- [18] A. Y. N. Sofia, S. A. Meguid, K. T. Tan, Shape morphing of aircraft wing: Status and challenges, *Materials & Design* 31 (2010) 1284–1292.
- [19] R. Duvigneau, Adaptive parameterization using free-form deformation, *INRIA Research Report RR-5949* (2006).
- [20] D. Wang, W. Zhang, A bispase parameterization method for shape optimization of thin-walled curved shell structures with openings, *International Journal for Numerical Methods in Engineering* 90 (2012) 1598–1617.
- [21] M. Xiao, P. Breitkopf, R. F. Coelho, C. Knopf-Lenoir, M. Sidorkiewicz, P. Villon, Model reduction by cpod and kriging, *Int J Struc Multidisc Optim* 41 (2009) 555–574.
- [22] L. P. Swiler, G. D. Wyss, .a users guide to sandia’s latin hypercube sampling software: Lhs unix library standalone version, *Sandia National Laboratories* (2009).
- [23] C. Ghnatios, F. Chinesta, E. Cueto, Optimizing composites forming processes by applying the proper generalized decomposition, *International Conference on Advances in Materials and Processing Technologies*, Ctr Arts & Metiers ParisTech, Paris, FRANCE (2010).
- [24] R. F. Coelho, J. lebon, P. Bouillard, Hierarchical stochastic metamodells based on moving least squares and polynomial chaos expansion: Application to the multiobjective reliability-based optimization of space truss structures, *Int J Struc Multidisc Optim* 43 (2011) 707–729.
- [25] D. Amsallem, J. Cortial, K. Carlberg, C. Farhat, A method for interpolating on manifolds structural dynamics reduced-order models, *International Journal for Numerical Methods in Engineering* 80 (2009) 1241–1258.
- [26] B. Nayroles, G. Touzot, P. Villon, Generalizing the finite element method: diffuse approximation and diffuse elements, *Computational Mechanics* 10 (1992) 307–318.
- [27] N. H. Kim, Y. Chang, Eulerian shape design sensitivity analysis and optimization with a fixed grid, *Computer Methods in Applied Mechanics and Engineering* 194 (2005) 32913314.
- [28] G. Allaire, F. Jouve, A.-M. Toader, Structural optimization using sensitivity analysis and a level-set method, *Journal of Computational Physics* 194 (2004) 363–393.
- [29] F. Murat, J. Simon, Sur le controle par un domaine geometrique, *Pre-publication du Laboratoire d’Analyse Numerique*, no 76015, Universite de Paris 6 (1976).
- [30] S. Oshier, J. A. Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations, *Journal of Computational Physics* 79 (1988) 12–49.
- [31] A. Kaufman, D. Cohen, R. Yagel, Volume graphics, *IEEE Computer* 26 (1993) 51–64.

- [32] A. Chatterjee, An introduction to the proper orthogonal decomposition, *Current Science*, Special Section: Computational Science 78 (2005) 808–817.
- [33] K. Fukunaga, D. Olsen, An algorithm for finding intrinsic dimensionality of data, *IEEE Transactions on Computers* 20 (1971) 176–183.
- [34] S. Jan, J.-P. Zolesio, *Introduction to shape optimization: Shape sensitivity analysis*, springer verlag berlin (1992).
- [35] V. Schulz, *A riemannian view on shape optimization*, *Foundations of Computational Mathematics* (2012).
- [36] O. project home page, Last accessed feb 22, 2011, <http://omd2.scilab.org/> (2009).