

Implicit constraint handling for shape optimisation with POD-morphing

Balaji Raghavan, Manyu Xiao, Piotr Breitkopf* and Pierre Villon

Laboratoire Roberval, UMR 6253 UTC-CNRS, BP 20529, 60205 Compiègne, France

In the former paper, we have introduced an original morphing approach based on Proper Orthogonal Decomposition (POD) of shapes, designed to replace parametrized CAD models in structural optimization. Here, we expand the method to interpolate exclusively between admissible instances of structural shapes, thus permitting a global understanding of the design domain and also reducing the size of the optimisation problem. The result is a bi-level reparametrization approach for structural geometries based on Diffuse Approximation in a properly chosen locally linearized space, and the geometric parameters are replaced with the smallest set of variables needed to represent a manifold of admissible shapes for a chosen precision. We demonstrate the approach in a typical shape optimisation problem.

Notre article précédent a introduit le concept original de morphing basé sur la Décomposition Orthogonale aux valeurs Propres (POD) des formes, qui vise à remplacer la démarche classique, basée sur les modèles CAO paramétriques, par un méta-modèle géométrique permettant de diminuer le nombre de variables dans les problèmes d'optimisation de forme. Ici, on étend ce concept pour permettre d'interpoler entre les instances de la géométrie de manière à obtenir toujours des structures admissibles, et d'avoir une "compréhension" globale du domaine de conception tout en minimisant le nombre de paramètres. C'est une approche bi-niveau de réparamétrage basée sur l'Approximation Diffuse dans un espace linéarisé où les paramètres de conception sont remplacés par un nombre réduit de variables permettant de représenter la variété de formes admissibles pour une précision donnée. Nous illustrons l'approche proposée sur un cas typique d'optimisation de forme.

Keywords: model reduction; optimisation; diffuse approximation; POD

Mots-clés: modèles réduits; optimisation; l'approximation diffuse; POD

1. Introduction

Shape optimisation may be viewed as the task of combining a parameterised geometric model with a numerical simulation code in order to predict the geometric state that minimises a given cost function while respecting a set of equality/inequality constraints. In this paper, we consider the task of shape/mesh interpolation or hypothesising the structure, which occurs between shape/mesh instances given by a sequence of parameter values. The need for this arose during the development of multidisciplinary optimisation techniques, because CAD parameterised models involved in automatised computing chains suffered from excessive design space dimensionality eventually leading to crashes of either the mesh generator or the solver. This phenomenon 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) within existing parametrisation methods.

*Corresponding author. Email: piotr.breitkopf@utc.fr

Most current approaches to shape parametrisation require hand-constructed CAD models. We are interested in developing an alternative approach in which the interpolation system builds up structural shapes automatically by learning from existing examples. One of the central components of this kind of learning is the abstract problem of inducing a smooth nonlinear constraint manifold from a set of the examples, called “Manifold Learning” by Bregler and Omohundro (1995) who developed approaches closely related to neural networks for doing it. Carlberg and Farhat (2010) proposed a similar approach in the domain of reduced order modelling (ROM) for complex flow problems. In this paper, we apply manifold learning to the shape interpolation problem to develop a parametrisation scheme tailored to the structural optimisation problem (e.g. airplane wing, A/C duct, engine inlet, etc.).

Several techniques (Ravindran, 2000; Willcox & Peraire, 2002; Bui-Thanh, Willcox, Ghattas, and van Bloemen, 2007) have been used to replace a complicated numerical model by a lower order meta-model, usually based on polynomial response surface methodology (RSM), kriging, least squares regression and moving least squares (Breitkopf, Naceur, Rassineux, & Villon, 2005). Surrogate functions and reduced order meta-models have also been used in the field of control systems to reduce the order of the overall transfer function (Willcox and Peraire, 2002). A very popular physics-based meta-modelling technique consists of carrying out the approximation on the full vector fields using PCA and Galerkin projection (Berkooz, Holmes, & Lumley, 1993) in CFD (Raghavan and Breitkopf, 2012) as well as in structural analysis (Cordier, El Majd, & Favier, 2010) and has been successfully applied to a number of areas such as flow modelling (Couplet, Basdevant, & Sagaut, 2005), optimal flow control (Ravindran, 2000), aerodynamics design optimisation (Coelho, Breitkopf, & Knopf-Lenoir, 2009) or structural mechanics (Dulong, Druesne, & Villon, 2007). In Carlberg and Farhat (2008), a snapshot-weighting scheme introduced using vector sensitivities as system snapshots to compute a robust reduced order model well-suited to optimisation. Carlberg and Farhat (2010) also demonstrated a goal-oriented local POD approach that is computationally less expensive than using a global POD approach.

However, we have not observed much if any research into using decomposition-based surrogate models to reducing dimensionality of the design domain in shape optimisation, and for that matter, structural optimisation of any type. This area, we feel is promising considering the obvious advantages of having far fewer parameters describing the domain: easier visualisation, more flexibility in the choice of admissible shapes, better applicability to gradient-based solvers due to reduced dimensionality and thus a reduction in the overall size of the optimisation, and of course a separation between the CAD and the optimisation phases in system design by giving the optimisation group a protocol to reparametrise structural shapes for a given set of admissible shapes/meshes that can be generated by the CAD group, and using the presented algorithm (or a variant thereof) on these to get the new set of design variables.

In this paper, we extend the POD-morphing approach presented in Raghavan, Breitkopf, and Villon (2011) and present a manifold learning approach combining diffuse approximation and principal component analysis, whose performance is easily compared to that of simple linear interpolation, “classical” morphing (Sofia, Meguid, & Tan, 2010) and a posteriori mesh parametrisation (Chappuis, Rassineux, Breitkopf, & Villon, 2004).

We propose a four-step “a posteriori” reparametrisation approach to reduce the number of design variables needed while describing the shape of a structure:

- Discretisation of a shape to get the indicator function.
- Decomposition by principal components analysis.
- Two-level dimensionality reduction: in the first reduction phase, the snapshot “pixel arrays” (or “voxels” in 3D) are reduced to obtain a small number of dominant basis

vectors spanning the physical design domain and the vector of coefficients $\bar{\alpha}$ is then obtained by projecting a structural shape onto the basis Φ .

In the second reduction phase, diffuse approximation performed in the α -space gives the final minimal set of parameters t_1, \dots, t_p , $p \leq m$, thus a two-level model reduction. Since they have been obtained from an “a posteriori” sweep of the design domain followed by decomposition, these new variables can be directly used in an optimisation algorithm to obtain the optimal shape for a given performance objective.

The methodology is described in the next section with the overall algorithm and the optimisation test-case and results are presented in Section 3. We close with a discussion of possible future work.

2. A posteriori grid parametrisation methodology

This approach works by first studying a sampling of structural shapes sweeping the physical design domain, typically in a Lagrangian description with a sampling of the geometry-based design variables within their range, (this could simply be the finite set of points describing the edges/boundaries of a series of CFD meshes/grid points).

In order to work directly with the *shape* rather than just a set of geometric parameters, we use a unique, common and parameter-free representation of structural shape, the shape indicator function $\chi(\bar{x})$ where:

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

where $\Omega \subset \mathcal{R}^2$ (\mathcal{R}^3 in 3D) is the set of all points in space contained in the shape χ . The indicator function may be obtained in discrete form in several ways (voxelisation (Kaufman, Cohen, & Yagel 1993), topological level sets (Allaire, DeGournay, Jouve, & Toader, 2005) or “marching cubes” for negligible computational cost; but in all results presented in this article, we have used pixel/voxel maps where we simply map the edges/boundaries for each snapshot onto a reference grid to generate a binary array with a value 1 for every “penetrated” cell and 0 for every cell outside the boundary: $S^i \in \mathcal{R}^{N_c}$, $i = 1, \dots, M$, where N_c is the grid resolution.

2.1. Creation of “pixel map” snapshots

We build the parametrisation scheme after studying the full range of admissible shapes (i.e. snapshots (Chatterjee, 2005)) constituting the design domain. For structural optimisation problems of a fixed topology, these admissible shapes could be obtained in a Lagrangian description by a sampling of the geometry-based design variables within their feasible range $\bar{X} \in [\bar{L}_B, \bar{U}_B] \subset R^N$ or simply from the finite set of points describing the edges/boundaries of a series of CFD meshes/grid points for an initial random sampling of M designs. We next map the edges/boundaries for each snapshot onto a reference grid and store them as arrays ($S^i \in \mathcal{R}^{N_c}$, $i = 1, \dots, M$) of 1 s and 0 s (Figure 1), where N_c is the chosen resolution.

2.2. Principal components analysis

This is the first phase of model reduction. We first calculate the deviation matrix D_S and covariance matrix C_v for the snapshots using:

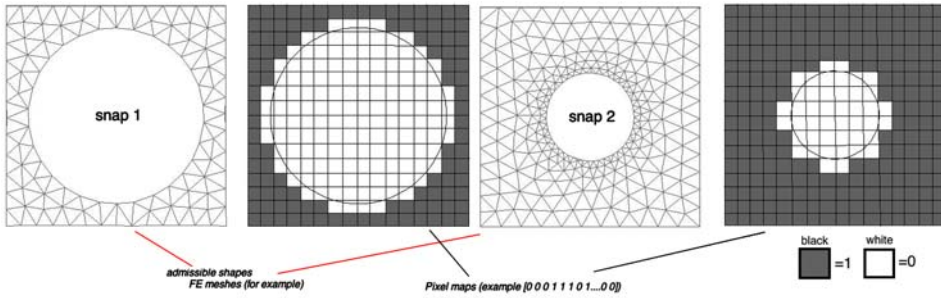


Figure 1. Reference grid mapping for pixel map: plate with circular hole.

$$D_S = [S^1 - \bar{S} \quad S^2 - \bar{S} \quad \dots \quad S^M - \bar{S}], \quad C_v = D_S \cdot D_S^T \quad (2)$$

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

$$S^j = \bar{S} + \sum_{i=1}^M \alpha_{ij} \bar{\phi}_i, \quad \alpha_{ij} = \bar{\phi}_i^T S^j \quad (3)$$

In the first reduction phase, we limit the basis to the first $m \ll M$ most “energetic” modes for a given reconstruction error $\epsilon(m)$ (Berkooz et al., 1993)

$$\tilde{S}^j = \bar{S} + \sum_{i=1}^m \alpha_{ij} \bar{\phi}_i \quad \text{and} \quad \epsilon(m) = 1 - \frac{\sum_{i=1}^m \lambda_i}{\sum_{i=1}^M \lambda_i} \quad (4)$$

2.3. Model reduction and design domain dimensionality

Equation (3) does not provide a sufficient basis for establishing the value of m as one needs to specify the threshold value for ϵ . Also, the α_{ij} may not be taken as design variables without taking into account the possible relationships between them so as to render feasible shapes.

Let us consider the same system (plate with circular hole $R_{\min} \leq r \leq R_{\max}$. Ignoring the fact that the dimensionality is 1, we construct 50 random snapshots by varying the radius r . The pixelisation and PCA are then performed in succession giving us a set of α 's corresponding to each snapshot. As illustrated in Figure 2, the α 's form a set of one-dimensional manifolds, clearly indicating that the design domain is parametrised by ONE single parameter t , which in this case happens to be the hole radius (in the general case we obtain a vector $\bar{t} \in R^p$, $p \leq m$), i.e. $\alpha_1 = \alpha_1(t)$, $\alpha_2 = \alpha_2(t)$, \dots . These manifolds are easily obtained by performing a diffuse approximation (Breitkopf et al., 2005, Nayroles, Touzot, & Villon, 1992) over all the $\alpha_1 \dots \alpha_M$ obtained from snapshots S_1 to S_M . Furthermore, the curves of $\alpha_1, \alpha_2, \dots$ vs t may be interpreted as possible “constraints” (direct geometric constraints, technological constraints, etc. that are difficult to express mathematically) on the geometric parameters \bar{X} (here simply the single hole radius r) in the α -space, since points lying outside the manifolds will produce inadmissible shapes as shown. Thus, in the second reduction phase, we locally introduce the parametric expression of the α -manifolds.

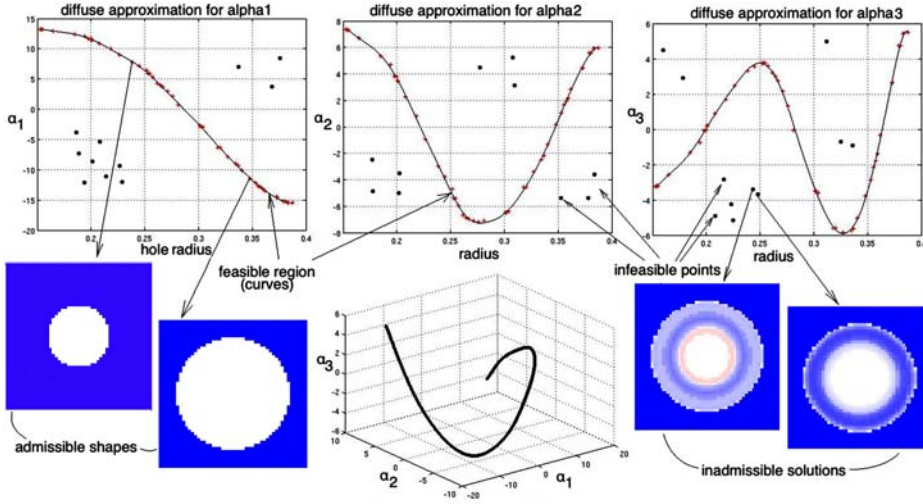


Figure 2. Feasible region for a plate with a circular hole of varying radius.

2.4. Manifold approximation and updating

We present here a formal approach to locally identify the system dimensionality from the α -manifolds. Consider a system of M pixel snapshots converted to the PCA-space retaining $m < M$ coefficients thus giving us a set of points $\bar{\alpha}^1 \dots \bar{\alpha}^M \in \mathcal{R}^m$. We would like to implement an algorithm that:

- (1) Detects the “true” dimensionality ($p \leq m$) 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 $\alpha_1 = \alpha_1(t_1 \dots t_p) \dots \alpha_m = \alpha_m(t_1 \dots t_p)$.
- (2) Constrains the evaluation point ($\bar{\alpha}^{ev}$) to stay on the feasible region of admissible shapes, during the course of the optimisation.

2.4.1. Local rank detection of α -manifold

To locally detect the dimensionality of the $\alpha_1 \dots \alpha_m$ hypersurface in the neighbourhood of $\bar{\alpha}^{ev}$, we first establish the local neighbourhood, this may be done in the original geometric space (if available) or, if the original parameters are unavailable which is what this approach is intended for, by using the α values if the neighbourhood is sufficiently dense. So, if $\bar{\beta}_1 \dots \bar{\beta}_{nbd}$ are neighbouring points in α -space, we next use a polynomial basis (here linear) centred around the evaluation point

$$P = \begin{bmatrix} 1 & \beta_1^1 - \alpha_1^{ev} & \beta_2^1 - \alpha_2^{ev} & \dots & \beta_m^1 - \alpha_m^{ev} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \beta_1^{nbd} - \alpha_1^{ev} & \beta_2^{nbd} - \alpha_2^{ev} & \dots & \beta_m^{nbd} - \alpha_m^{ev} \end{bmatrix} \quad (5)$$

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

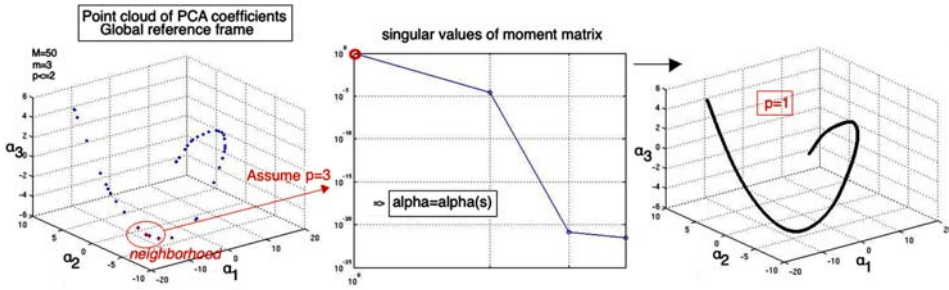


Figure 3. Detecting dimensionality for a plate with a circular hole of varying radius.

Next, we detect the local rank of the manifold by calculating the singular values of the moment matrix A for a chosen precision. This gives us the dimensionality $p \leq m$, as shown in Figure 3 for a plate with a circular hole of varying radius, giving us $p = 1$ allowing us to parametrise the curve with a single parameter $\alpha_1 = \alpha_1(t_1), \dots, \alpha_m = \alpha_m(t_1)$.

2.4.2. Tangent plane construction and diffuse “walking”

The idea is to bring the current design point given by the optimisation algorithm in subsequent iterations, down to the surface, which represents locally the manifold of admissible shapes. The local surface tangent to the manifold is defined with respect to the projection plane iteratively updated. To achieve this, we use a diffuse approximation-based manifold “walking” scheme consisting of the following steps shown in Figure 4.

- (1) Let P_i be the evaluation point (on the α -manifold) and P_{i+1}^0 be the new candidate point (that needs to be brought back onto the manifold/feasible region). We first establish the neighbourhood $\bar{\beta}_1 \dots \bar{\beta}_{nbd}$ of P_{i+1}^0 .
- (2) Calculate the centroid $\bar{\beta}_m = (\sum_{i=1}^{nbd} \bar{\beta}_i) / nbd$.
- (3) Find the centroidal hyperplane for the neighbourhood from the eigenvectors $\bar{v}_1, \bar{v}_2, \dots$ of the covariance matrix C_{nbd} , the first eigenvector representing the plane normal:

$$C_{nbd} = (1/nbd) \sum_{i=1}^{nbd} (\bar{\beta}_i - \bar{\beta}_m)(\bar{\beta}_i - \bar{\beta}_m)^T \tag{6}$$

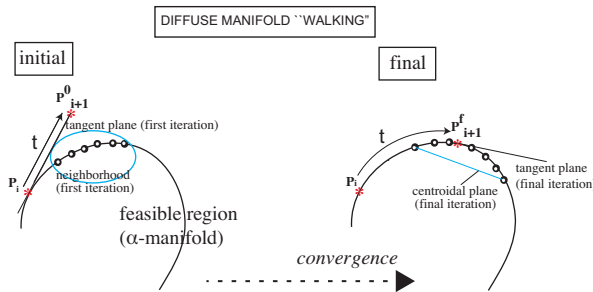


Figure 4. Walking the evaluation point along the α -manifold.

- (4) Project the evaluation point as well as the neighbourhood points in the local coordinate system $\bar{v}_1, \bar{v}_2, \dots$ (origin at centroid $\bar{\beta}_m$) to get the local coordinates h, t_1, t_2, \dots, t_p where h is the height over the centroidal plane using the equations (for a general point $\bar{\alpha}$).

$$h = \bar{v}_1^T (\bar{\alpha} - \bar{\beta}_m), \quad t_1 = \bar{v}_2^T \bar{\alpha}, \text{ etc.} \quad (7)$$

- (5) Perform a diffuse approximation for the nb points, to obtain the local surface $\tilde{h} = \tilde{h}(t_1 \dots t_p)$ using a polynomial basis P centred around $\bar{\alpha}^{ev}$, with a weighting matrix W .

$$\left[\tilde{h}(\bar{t}^{ev}), \frac{\partial \tilde{h}}{\partial t_1}(\bar{t}^{ev}), \frac{\partial \tilde{h}}{\partial t_2}(\bar{t}^{ev}), \dots \right]^T = (P^T W P)^{-1} P^T W [h_1, \dots, h_{nb}]^T \quad (8)$$

where $\left[\frac{\partial \tilde{h}}{\partial t_1}(\bar{t}^{ev}), \frac{\partial \tilde{h}}{\partial t_2}(\bar{t}^{ev}), \dots \right]^T$ is the local tangent hyperplane at P_0 in the neighbourhood $\bar{\beta}^1 \dots \bar{\beta}^{nb}$.

- (6) We then project the point P_{i+1}^0 onto this tangent plane to get the adjusted evaluation point P_{i+1}^1 and then repeat the process by finding the new neighbourhood and new tangent plane and new projection point P_2 till the evaluation point stops changing P_{i+1}^f .

In other words, we “walk” along the surface of the α -manifold to ensure that we stay in the domain of feasible solutions.

2.5. Shape interpolation and identifying the original geometric parameters

There are two ways to do this step in order to recreate the structural shape for an arbitrary design point (\bar{t}). In the first, we perform POD reconstruction to get the pixel map \tilde{S}

$$\tilde{S}(\bar{t}) = \bar{S} + \sum_{i=1}^m \alpha_i(\bar{t}) \bar{\phi}_i \quad (9)$$

followed by density filtering using Canny’s algorithm (Canny, 1986) to eliminate possible greyscale, then followed by locating the coordinates of the corner points/vertices of the boundary pixels using one of various possible methods (Breitkopf, 1998) and finally a local moving least square approximation using radial basis weighting functions (Breitkopf et al., 2005; Nayroles et al., 1992) to construct the smooth structural boundaries/edges from the vertices of the interpolated pixel map. The original geometric parameters \bar{X} may then be *directly* identified from the smooth reconstructed shape if they are needed. This method of shape construction could be difficult in case the original parameter set contains spline parameters (for example). The other issue is the possible loss of precision that would make it necessary to increase the resolution of the voxel maps in order to capture/identify very small geometric facets (like chamfers, etc.) that might affect the value of the objective function.

For this reason, we use “reverse-lookup” i.e. an RSM between the geometric parameters and the α -coefficients to get $\bar{X}(\bar{\alpha})$ if the manifold shape (and dimensionality) need a very fine pixel/voxel resolution for convergence or if direct identification of the \bar{X} is difficult or impossible. We then rerun the original CAD model with $\bar{X}(\bar{\alpha})$ to obtain a full-featured shape.

2.6. Objective function and gradients

The reconstructed shape is meshed and the numerical analysis is performed using a method chosen based on the disciplines involved in the analysis i.e. CFD/Navier–Stokes for incompressible flows (Larsson, Diez, & Huerta, 2010; Launder and Spalding, 1974), FEA for structural analysis, etc. The only difference is that instead of obtaining X_i^{opt} we attempt to find the final governing parameters \bar{t}^{opt} and thus the coefficients $\bar{\alpha}(\bar{t}^{\text{opt}})$ optimise the performance objective. An important phase here is remeshing the surfaces obtained. Chappuis et al. (2004) developed an approach of calculating principal curvatures from an existing mesh or shape using a secondary local model with diffuse interpolation and then, using these curvatures to identify shape primitives, such as cylinders, torus, etc., for the purpose of meshing.

The drawback here is that a *smooth* shape needs to be reconstructed at every iteration. An attractive alternative strategy is available where we only use the same set of 6–7 snapshots around the evaluation point that were used to construct the α -manifold in Subsection 2.4.2 and obtain the function values at these points. This is followed by a *second* diffuse approximation for the objective function $\tilde{J}(t_1, t_2, \dots)$ that will yield the function values as well as the gradient (sensitivities $\frac{\partial J}{\partial t_1}, \frac{\partial J}{\partial t_2}, \dots$) and Hessian in a single shot, rather than relying on a finite difference grid with a prohibitively large number of evaluations.

3. Optimisation test-case: air-conditioning duct

We return to the problem of the air-conditioning duct modelled and optimised in Raghavan et al. (2011) with 13 geometric parameters $X_1 \dots X_{13}$ shown in Figure 5.

The optimisation problem in the geometric space may be written as:

$$\text{Find } \bar{X}^{\text{opt}} = \text{Argmax } P_{\text{flow}}(X_1, \dots, X_{13}) \quad \text{s.t. } \bar{L}_B \leq \bar{X} \leq \bar{U}_B \quad (10)$$

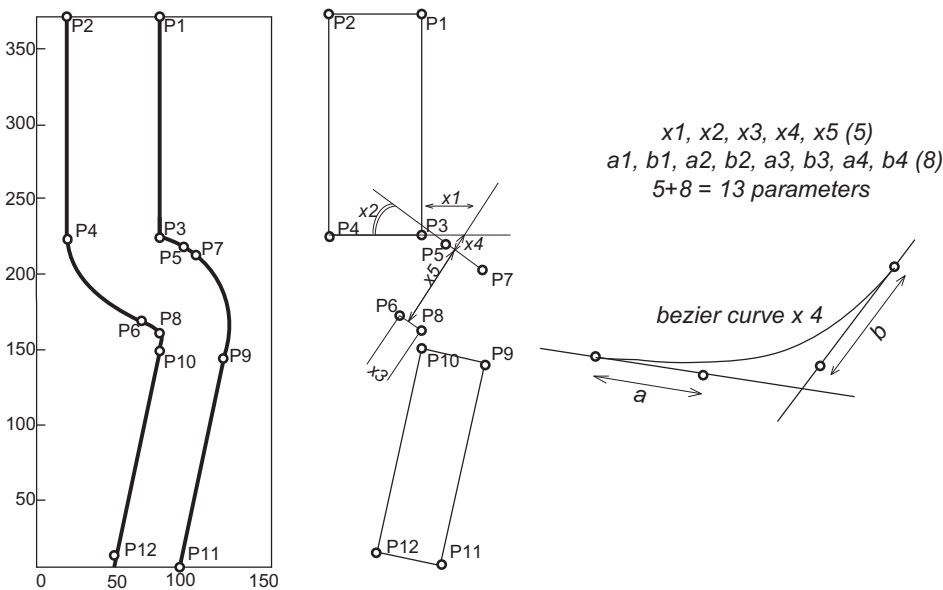


Figure 5. Duct geometry showing four different regions.

where P_{flow} = flow permeability = $1/(\text{pressure loss from inlet to outlet}) = 1/(P_{\text{inlet}} - P_{\text{outlet}})$ and \bar{U}_B and \bar{L}_B are the upper and lower bounds on the 13 design variables. Once we switch over to the reduced space, we can express the objective function as a function of the PCA coefficients (α_i) and hence the new design variables \bar{t} , the optimisation problem may be written as:

$$\text{Find } \bar{t}^{\text{opt}} = \text{Argmax } P_{\text{flow}}(\bar{\alpha}(t_1 \dots t_p)) \quad \text{s.t. } g_{\text{min}} \leq g_i(\bar{\alpha}(\bar{t})) \leq g_{\text{max}} \text{ and } h(\bar{\alpha}(\bar{t})) = 0 \quad (11)$$

where we now intend to find the pixelised shape $S(\bar{\alpha}(\bar{t}^{\text{opt}}))$. The constraints $g_i, i \in [1, N]$ are obtained by transferring bounds \bar{U}_B and \bar{L}_B on the α -space, while h represents the feasible region (set of manifolds in $\bar{\alpha}$ -space). Both g_i and h are taken into account implicitly with the diffuse approximation-based approach outlined earlier allowing the α 's to be expressed locally as functions of the final parameters t_1, \dots, t_p .

Since the mapping to the new space is highly non-linear, we need to ensure we stay in the feasible region during the course of the optimisation. As previously, we have used the same $M = 102$ snapshots and used the bi-level approach of model reduction using basis truncation and the α -space diffuse approximation.

Figure 6 shows the dimensionality deduction based on Section 2. After analysing the set of snapshots in α -space, it is clear from the set of 2D surfaces obtained that the behaviour of the various α 's is governed by just two parameters (say t_1 and t_2).

The feasible regions are represented by the α -manifolds, and as explained in Section 2, staying on the manifold ensures an admissible solution, even though we may need to invoke

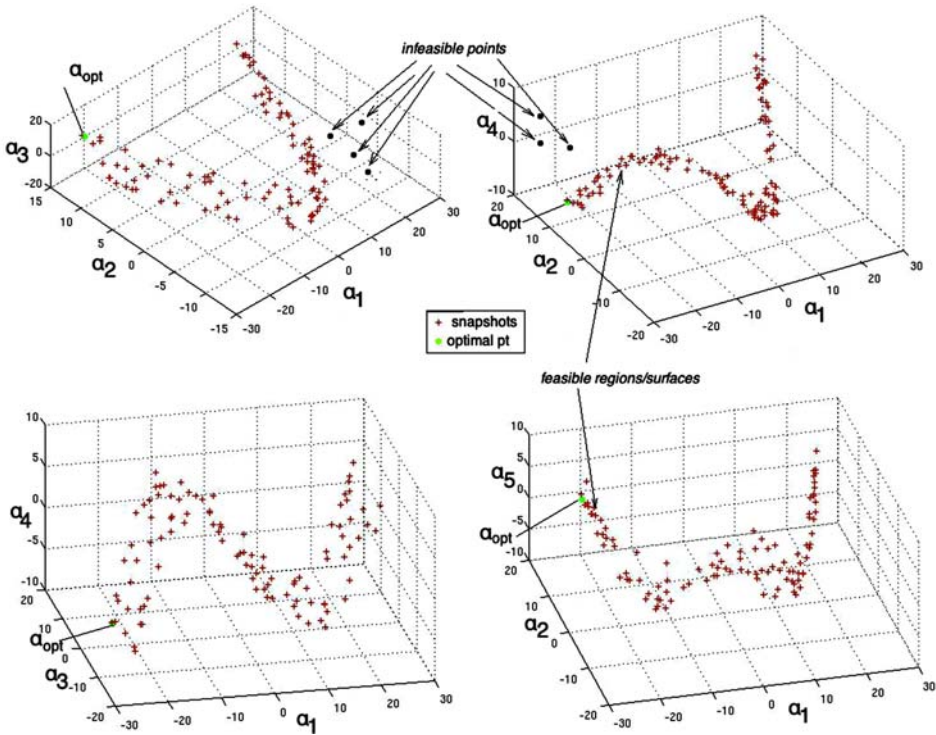


Figure 6. Dimensionality (=2) and model reduction for A/C duct.

Table 1. Reverse look-up: comparison between α^{opt} and α_{rev} for different m .

Modes (m)	$\bar{\alpha}^{\text{opt}}$	$\bar{\alpha}_{\text{rev}}(\mathcal{S}(\bar{\alpha}^{\text{opt}}))$
5	-22.6226, 13.4301, 5.8149, -9.4755, 3.1751	-21.8192, 13.7669, 6.6731, -8.8569, 3.2854
6	-22.4873, 14.2647, 6.2187, -9.3431, 3.0856, 3.7891	-22.0924, 13.274, 5.8194, -9.2671, 3.0162, 4.2320
7	-22.7028, 14.4708, 5.6493, -9.5091, 3.1863, 3.9603, 5.2279	-22.1531, 13.8087, 5.2442, -9.1170, 2.8065, 4.2241, 4.6575
8	-22.3745, 14.7165, 5.1094, -9.7710, 3.2653, 3.9784, 5.2527, -4.2108	-22.0974, 13.8794, 5.6017, -9.4491, 2.4682, 5.2825, 4.7886, -2.3602

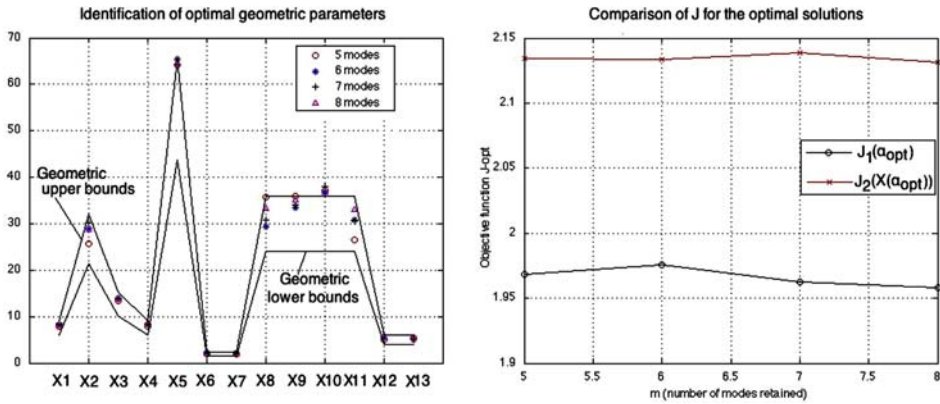


Figure 7. Optimisation in α -space and identification of geometric parameters \bar{X} .

the density filter from time to time during the optimisation. This also means that $\bar{\alpha} = [\alpha_1(t_1, t_2), \alpha_2(t_1, t_2) \dots \alpha_m(t_1, t_2)]$ if using a truncated basis of size m .

We first perform optimisation (using a standard Newton algorithm) in the reduced space getting \bar{t}_{opt} , and then calculating $\bar{\alpha}(\bar{t}_{\text{opt}})$, and next to estimate \bar{X}_{opt} (original geometric parameters) from the values of $\bar{\alpha}(\bar{t}_{\text{opt}})$ either by inspection of the optimised shape $\mathcal{S}(\bar{\alpha}(\bar{t}_{\text{opt}}))$ or using an RSM between the \bar{X}_i and α_i . The permeability P_{flow} for every possible design was calculated using the inverse of the total pressure drop across the duct length (inlet to outlet) and an optimal shape obtained using 5–8 modes, followed by identification by response surface methodology over the values of the original 13 geometric design parameters for each $\bar{\alpha}_{\text{opt}}$, i.e. getting $\bar{X}(\bar{\alpha}_{\text{opt}})$. The optimal solution obtained has been added to Figure 6 and as expected, it lies on the edge of the constraint/feasible region. This is followed by “reverse look-up” (Table 1) by projecting the pixel array obtained by shape generation/pixelisation, onto the truncated basis of m modes to get $\bar{\alpha}_{\text{rev}}$ from $\bar{X}(\bar{\alpha}^{\text{opt}})$, needed to account for the error introduced by the RSM. The values of $\bar{\alpha}^{\text{opt}}$ and $\bar{X}(\bar{\alpha}^{\text{opt}})$ are shown in Figure 7.

4. Conclusions

In this paper, the authors have introduced an “a posteriori” scheme with a two-level model reduction to replace the geometry-based variables with a more compact and normalised set of variables and replace the higher dimensional design space with a newer design space of lower dimension, implicitly taking into account constraints on feasible shapes.

The presented methodology has a few possible areas of improvement. The first is in resolving the difficulty in setting upper and lower bounds on the α -based design variables. The second area is in the treatment of possible degenerate cases for the structural shape.

Acknowledgements

This work has been supported by the French National Research Agency (ANR) through the COSINUS programme (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 providing HPC resources that have contributed to the research results reported, as well as Mr. V. Picheny, Ecole des Mines, France for contributing the CFD model. Mrs Xiao acknowledges the support of the NPU basic project JC20120241'.

References

- Allaire, G., DeGournay, F., Jouve, F., & Toader, A.M. (2005). Structural optimization using topological and shape sensitivity via a level set method. *Control & Cybernetics*, 34, 59–80.
- Berkooz, G., Holmes, P., & Lumley, J.L. (1993). The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25(1), 539–575.
- Bregler, C., & Omohundro, S.M. (1995). Nonlinear image interpolation using manifold learning. In G. Tesauro, D.S. Touretzky, and T.K. Leen (Eds.), *Advances in Neural Information Processing Systems 7* (pp. 973–980). Cambridge, MA: MIT Press.
- Breitkopf, P. (1998). An algorithm for construction of iso-valued surfaces for finite elements. *Engineering with Computers*, 14(2), 146–149.
- Breitkopf, P., Naceur, H., Rassineux, A., & Villon, P. (2005). Moving least squares response surface approximation: Formulation and metal forming applications. *Computers and Structures*, 83(17–18), 1411–1428.
- Bui-Thanh, T., Willcox, K., Ghattas, O., & van Bloemen, W.B. (2007). Goal-oriented, model-constrained optimization for reduction of large-scale systems. *Journal of Computational Physics*, 224(2), 880–896.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6), 679–698.
- Carlberg, K., & Farhat, C. (2010). A low-cost, goal-oriented compact proper orthogonal decomposition basis for model reduction of static systems. *International Journal for Numerical Methods in Engineering*, 86(3), 381–402.
- Carlberg K., & Farhat C. (2008). *A compact proper orthogonal decomposition basis for optimization-oriented reduced-order models*. 12th AIAA/ISSMO multidisciplinary analysis and optimization conference, Victoria, Canada, 10–12, 2008.
- Chappuis, C., Rassineux, A., Breitkopf, P., & Villon, P. (2004). Improving surface remeshing by feature recognition. *Engineering with Computers*, 20(3), 202–209.
- Chatterjee, A. (2005). An introduction to the proper orthogonal decomposition. *Current Science, Special Section: Computational Science*, 78(7), 808–817.
- Coelho, R.F., Breitkopf, P., & Knopf-Lenoir, C. (2009). Bi-level model reduction for coupled problems. *Structural and Multidisciplinary Optimization*, 39(4), 401–418.
- Cordier, L., El Majd, B.A., & Favier, J. (2010). Calibration of POD reduced order models using Tikhonov regularization. *International Journal for Numerical Methods in Fluids*, 63(2), 269–296.
- Couplet, M., Basdevant, C., & Sagaut, P. (2005). Calibrated reduced-order POD-Galerkin system for fluid flow modeling. *Journal of Computational Physics*, 207(1), 192–220.
- Dulong, J.-L., Druesne, F., & Villon, P. (2007). A model reduction approach for real-time part deformation with nonlinear mechanical behavior. *International Journal on Interactive Design and Manufacturing*, 1(4), 229–238.
- Kaufman, A., Cohen, D., & Yagel, R. (1993). Volume graphics. *IEEE, Computer*, 26(7), 51–64.
- Larsson, F., Diez, P., & Huerta, A. (2010). A flux-free a posteriori error estimator for the incompressible Stokes problem using a mixed FE formulation. *AIAA Journal*, 199(37–40), 2383–2402.
- Launder, B.E., & Spalding, D.B. (1974). The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 3(2), 269–289.

- Nayroles, B., Touzot, G., & Villon, P. (1992). Generalizing the finite element method: Diffuse approximation and diffuse elements. *Computational Mechanics*, 10(5), 307–318.
- Raghavan, B., & Breitkopf, P. (2012). Asynchronous evolutionary shape optimization based on high-quality surrogates: application to an air-conditioning duct. *Engineering with Computers*.
- Raghavan, B., Breitkopf, P., & Villon, P. (2011). POD-morphing: an a posteriori reparametrization approach for shape optimization. *European Journal of Computational Mechanics*, 19(5–7), 673–699.
- Ravindran, S.S. (2000). A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, 34(5), 425–448.
- Sofia, A.Y.N., Meguid, S.A., & Tan, K.T. (2010). Shape morphing of aircraft wing: Status and challenges. *Materials & Design*, 31(3), 1284–1292.
- Willcox, K., & Peraire, J. (2002). Balanced model reduction via the proper orthogonal decomposition. *AIAA Journal*, 40(11), 2323–2330.