



HAL
open science

Data analytic UQ cascade

Bijan Mohammadi

► **To cite this version:**

Bijan Mohammadi. Data analytic UQ cascade. Frediani Aldo; Mohammadi Bijan; Pironneau Olivier; Cipolla Vittorio. Variational Analysis and Aerospace Engineering, 116, Springer, pp.305-331, 2016, Springer Optimization and Its Applications, 978-3-319-45680-5. 10.1007/978-3-319-45680-5 . hal-01362349

HAL Id: hal-01362349

<https://hal.science/hal-01362349>

Submitted on 8 Sep 2016

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.

Copyright

Data analytic UQ cascade

Bijan MOHAMMADI

Abstract This contribution gathers some of the ingredients presented at Erice during the third workshop on 'Variational Analysis and Aerospace Engineering'. It is a collection of several previous publications on how to set up an uncertainty quantification (UQ) cascade with ingredients of growing computational complexity for both forward and reverse uncertainty propagation. It uses data analysis ingredients in a context of existing deterministic simulation platforms. It starts with a complexity-based splitting of the independent variables and the definition of a parametric optimization problem. Geometric characterization of global sensitivity spaces through their dimensions and relative positions through principal angles between vector spaces bring a first set of information on the impact of uncertainties of the functioning parameters on the optimal solution. Joining the multi-point descent direction and Probability Density Function (PDF) quantiles of the optimization parameters permits to define the notion of Directional Extreme Scenarios (DES) without sampling of large dimension design spaces. One goes beyond DES with Ensemble Kalman Filters (EnKF) after the multi-point optimization algorithm is cast into an ensemble simulation environment. This formulation accounts for the variability in large dimension. The UQ cascade continues with the joint application of the EnKF and DES leading to the concept of Ensemble Directional Extreme Scenarios (EDES) which provides a more exhaustive description of the possible extreme scenarios. The different ingredients developed for this cascade also permits to quantify the impact of state uncertainties on the design and provide confidence bounds for the optimal solution. This is typical of inverse designs where the target should be assumed uncertain. Our proposal uses the previous DES strategy applied this time to the target data. We use these scenarios to define a matrix having the structure of the covariance matrix of the optimization parameters. We compare this construction to another one using available adjoint-based gradients of the functional. Eventually, we go beyond inverse design and apply the method to general optimization problems. The ingredients of the paper have been applied to constrained aerodynamic performance analysis problems.

1 Context

Our domain of interest is aerodynamic shape optimization. The questions of interest are:

- can we propose an aircraft shape designed to have similar performances over a given range of some functioning parameters (to be formulated through the moments of a functional) ?
- can we do that modifying as less as possible an existing mono-point optimization shape design loop ?
- is it possible for the time-to-solution cost of this parametric shape design to remain comparable to the mono-point situation ?

We consider a generic situation where the simulation aims at predicting a given quantity of interest $j(\mathbf{x}, \alpha)$ and there are a few functioning or operating parameters α and several design parameters \mathbf{x} involved. The ranges of the functioning parameters define the global operating/functioning conditions of a given design. This splitting of the independent variables in two sets is important for the sequel.

We propose a cascade of ingredients to account for uncertainties avoiding any sampling of large dimensional spaces. A sampling will be only necessary for the functioning parameters \mathbf{u} range leading to a multi-point optimization problem.

The literature on uncertainty quantification (UQ) is huge. In short, forward propagation aims at defining a probability density function for j knowing those of \mathbf{x} and α [20, 30, 23]. This can be done, for instance, through Monte Carlo simulations or a separation between deterministic and stochastic features using Karhunen-Loeve theory (polynomial chaos theory belongs to this class) [16, 21, 59, 17, 57]. Examples of shape optimization with polynomial chaos and surrogate models during optimization are given in [48, 8].

Backward propagation aims at reducing models bias or calibrating models parameters knowing the probability density function of j (or other constraints and observations) [54, 28, 6]. This can be seen as a minimization problem and Kalman filters [29] give, for instance, an elegant framework for this inversion assimilating the uncertainties on the observations.

Our aim is to propose a geometric framework to address the curse of dimensionality of existing approaches related to the explosion of their computational complexity due to the sampling necessary to access probabilistic information, even if this can be improved with intelligent sampling techniques [52, 5]. The different ingredients presented here can be applied with either high-fidelity or reduced order models, when available [50, 47, 49, 56]. Low-order models are often used instead of the full models to overcome the computational complexity of UQ.

After the splitting of the independent variables mentioned above, we define a multi-point formulation to account for the variability on α . This is feasible because the size of α is assumed small. We define a global sensitivity space using the sensitivities of j with respect to \mathbf{x} for the multi-point problem. Once this space built, we analyze its dimension. We previously showed how to perform this task and how to use this information for adaptive sampling [34, 45].

The next step is to analyze the impact of different modeling or discretizations on the results. Different models or solution procedures lead to different sensitivity spaces. Beyond their respective dimensions, principal angles between the respective sensitivity vector spaces permit to measure the deviation due such changes. The dimensions of the spaces and the angles are interesting measures for both the epistemic and aleatory uncertainties. Indeed, suppose that, for a given model the dimensions of the sensitivity spaces remain unchanged when enriching the sampling of the functioning parameter range. This stability would be a first indication of a low level of sensitivity of the simulations with respect to this parameter. Once this is established, principal angles between subspaces permit to analyze both the impact of a given evolution of the modeling on the sensitivity spaces or an enrichment of our sampling. Eventually, constant dimension and low angles will clearly indicate a situation of low uncertainty.

These ingredients can be used in a context of multi-point robust analysis of a system to define worst-case scenarios for its functioning. To this end we combine a multi-point sensitivity with the probabilistic features of the control parameters through their quantiles [37, 27] to define the concept of Directional Extreme Scenarios (DES) without a sampling of large dimension design spaces.

Ensemble Kalman filters (EnKF) [29, 2, 14, 15, 11, 12] permit to go beyond the directional uncertainty quantification concept when accounting for the uncertainties in large dimension. They also permit backward uncertainty propagation assimilating the uncertainty on the functional and constraints during the design. We cast our multi-point optimization problem into the ensemble formulation. Joint application of the EnKF and DES leads to the concept of Ensemble Directional Extreme Scenarios (EDES) which provides a more exhaustive description of possible extreme scenarios.

Despite these approaches avoid the sampling of a large dimensional space the computation cost remains high and the procedures difficult to simply explain in engineering environments. We propose a low-complexity approach for the inversion of uncertain data where the target state \mathbf{u}^* used in an inverse problem is uncertain. In this situation, we consider functional of the form $j(\mathbf{x}, \alpha, \mathbf{u}^*) = \|\mathbf{u}(\mathbf{x}, \alpha) - \mathbf{u}^*(\alpha)\|$ to reduce the distance between a model state $\mathbf{u}(\mathbf{x}, \alpha)$ and observations.

Targeting uncertain data is a realistic situation as the acquired data are usually uncertain. It is therefore interesting to be able to quantify the impact of this uncertainty on the inversion results. An important information will be the sensitivity of the design to a given level of uncertainty on the data at some location. Indeed, if this sensitivity is low, this would be an indication that a more accurate acquisition there is unnecessary.

Considering the target as uncertain is also interesting because we do not always have existence of a solution for an inversion problem as \mathbf{u}^* is not necessary solution of the state equation making an exact or

deterministic inversion inversion pointless. Also, the approach permits to go beyond inversions based on least square minimization involving a mean state target.

Finally, the uncertainty in measurements is also an interesting way to account for the presence of variability in the state (e.g. due to the presence of turbulence in the flow). More generally, as the model and numerical procedures are by nature imperfect and partial, we can consider this uncertainty as a representation or estimation of these imperfections. These imperfections are even more present in inverse problems where one cannot afford the same level of resolution than for a single simulation. We therefore need to be able to quantify the impact of these weaknesses on the design. The approach presented here is therefore also useful to account for epistemic uncertainties related to possible model or solution procedures deficiencies.

Concerning the computational cost of these analysis, one can say that, when using the same calculation ingredients than in a high-fidelity simulation (i.e. without calling for low-order models or cheaper discretizations), the best calculation complexity one might think of for a simulation under uncertainty is when its cost is comparable to the deterministic situation. This is clearly unreachable except if all the extra effort can be achieved in a fully parallel manner and parallel to the initial deterministic calculation in order for the time to solution to remain unchanged when accounting for the presence of uncertainties. This is the case with Monte Carlo approaches. But these are quite expensive and do not take advantage of available simulation environments. In particular, when an adjoint-based optimization environment exists. Our proposal consists of upgrading existing platforms without abandoning what has been built for the deterministic situations and with keeping the time to solution unchanged in the presence of uncertainties with two sources of parallelism coming from the multi-point formulation to account for the uncertainties on the functioning parameters and from the EnKF formulation for those on the optimization variables and observation data.

2 Parametric optimization

We are interested in a class of optimization problems where the cost function involves a functioning parameter α not considered as a design parameter:

$$\min_{\mathbf{x} \in \mathbf{O}_{ad}} j(\mathbf{x}, \alpha), \quad \alpha \in \mathbf{I} \subset \mathbb{R}^n, \mathbf{O}_{ad} \subset \mathbb{R}^N. \quad (1)$$

where \mathbf{x} is the design vector belonging to \mathbf{O}_{ad} the optimization admissible domain. Usually, the functioning parameters (or operating conditions) α are just a few. On the other hand, the size N of \mathbf{x} is usually large. Together, \mathbf{x} and α fully describe our system and we have $n \ll N$. This splitting between functioning parameters (or operating conditions) and design variables is central to our discussion.

In [34, 35] we showed how to use multi-point optimization to address such optimization problem. The aim is to remove, during optimization, the dependency in α . This is done minimizing a functional $J(\mathbf{x})$ encapsulating this dependency expressed through $\mathbf{A} = \{j(\mathbf{x}, \alpha_k), \alpha_k \in \mathbf{I}_M\}$ over \mathbf{I}_M a given sampling of \mathbf{I} :

$$J = \mathbf{J}(\mathbf{A}), \quad \text{such that} \quad \mathbf{G}(\mathbf{A}) \leq 0. \quad (2)$$

Several choices are possible for \mathbf{J} and \mathbf{G} to address the issue of robust design. For instance, following Taguchi's definition, one can look for minimal-variance design or only a given level for the variance. Indeed, a classical approach to extend single point design and improve off-design points is to control μ mean performance and σ variance of the functional [53] as in First-Order Second Moment (FOSM) methods [33]. One can also look for information about the tails of the distributions which can be linked to the variance in the Gaussian framework and we use this relationship in quantile-based extreme scenarios.

Often it might be interesting to go beyond the first two moments and in particular consider the first four moments of j during the design. Going beyond the first two moments is important when the PDF of j deviates from a pure Gaussian distribution. Indeed, even with interval-based (with uniform PDF) or Gaussian entries there is no reason the PDF of the solution of a simulation to remain uniform or Gaussian.

The third and fourth moments are the skewness γ and the kurtosis κ . One can consider that a robust design should favor symmetry in the distribution which means lower absolute value of skewness. For instance, in a Gaussian distribution we have $\gamma = 0$. Also, in a normal distribution the mean and median coincide and if a PDF is not too far from a normal distribution, the median will be near $\mu - \gamma\sigma/6$. Therefore, if $|\gamma| \rightarrow 0$ the PDF tends toward a normal distribution. This provides an inequality constraint on $|\gamma|$ as γ can be positive

or negative. For an unimodal PDF a reduction of the skewness comes when the mean and the mode of the distribution converge toward each other at given standard deviation.

Concerning the fourth moment, a robust design should favor higher density near the mean which means higher kurtosis, but this is more subtle. Indeed, despite higher kurtosis means concentration of the probability mass around the mean, it could also imply thicker tails in the PDF. This means that more of the variance is the result of infrequent extreme deviations. We need therefore to define what we mean by robust design: acceptance of frequent modest deviations or acceptance of infrequent extreme ones. If operational security is a major concern the latter should be definitely avoided. Hence, a reasonable requirement would be to have a design reducing the initial kurtosis value: $\kappa \leq \kappa_0$ together with a constraint on the variance σ .

3 Gradients, sensitivity spaces and admissible search directions

Monte Carlo simulations permit to recover these moments with an error decreasing as $\sigma M^{-1/2}$ with M the number of functional evaluations and this rate is independent of n . But, for small n , classical numerical integration over-performs Monte Carlo simulations in term of complexity based on the number of functional evaluations to recover at a given accuracy these moments. As we are interested by small values of n (typically $n = 2$ or 3 in our applications), this latter can therefore be preferred. Anyway, both Monte Carlo trials and numerical integration lead to the introduction of weighted sums over a M -point sampling \mathbf{I}_M of I as estimators of the previous moments.

The linearity in the sums permits to access the gradient of the moments with respect to the control parameters \mathbf{x} from the gradient of the functional at the sampling point α_k . These are four vectors in $S_M = \text{Span}\{\nabla_{\mathbf{x}} j(\mathbf{x}, \alpha_k), \alpha_k \in \mathbf{I}_M\} \subset \mathbb{R}^N$. In applications of interest N is large. However, we showed that often $\dim(S_M) \ll N$ [35, 34, 36]. This analysis also permits to a posteriori give confidence bounds on the choice of the sampling size M which should be clearly larger than $\dim(S_M)$. Figure 1 shows an example of this analysis during the optimization of the shape of an aircraft with $N = 5000$ and $M = 100$ showing that $\dim(S_M)$ always remains below 35 making 100 a safe choice and clearly smaller than the dimension of the optimization space. This is interesting as indeed, without other information and considering vector spaces theory, the size of the sampling should be larger than the dimension of the control space (i.e. $M = N + 1$).

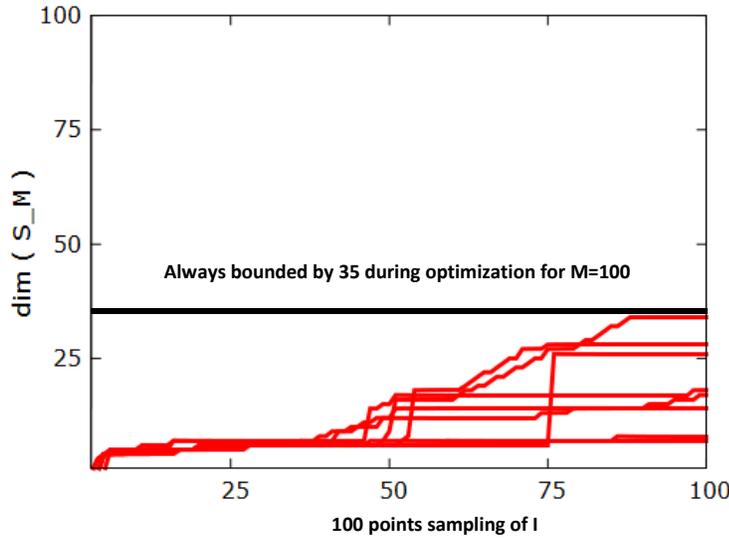


Fig. 1 Histories of Gram-Schmidt orthonormalization of $\{\nabla_{\mathbf{x}} j(\mathbf{x}, \alpha_k), \alpha_k \in \mathbf{I}_{100}\}$ during optimization. The dimension of the global search space S_M always remains below 35 which makes safe the choice of $M = 100$.

Let us denote by $C_{i=1,\dots,3}$ the three constraints on the second, third and fourth moments and let us consider the subspace $s_M = \text{Span}\{\nabla_{\mathbf{x}} C_{i=1,\dots,3}\} \subset \mathbb{R}^3 \subset \mathbb{R}^N$. Obviously $p = \dim(s_M) \leq 3$. Let us denote by $\{\mathbf{q}_{i=1,\dots,p}\}$

an orthonormal basis for s_M obtained, for instance, orthonormalizing the three gradient vectors by the Gram-Schmidt procedure. The gradients G of the constraints can therefore be expressed as linear combination of q_i : $G = (\nabla_{\mathbf{x}} C_{i=1,\dots,3}) = P^{-1}(\mathbf{q}_{i=1,\dots,p})$ with P being the matrix expressing the coordinates of \mathbf{q} in G .

With equality constraints, a descent direction d can be obtained writing the first order optimality condition stating that d needs to be orthogonal to s_M . Hence, using the local orthonormal basis $\{\mathbf{q}_{i=1,\dots,p}\}$, we consider d given by:

$$d = \nabla_{\mathbf{x}}\mu - \sum_{i=1}^p \langle \nabla_{\mathbf{x}}\mu, \mathbf{q}_i \rangle \mathbf{q}_i. \quad (3)$$

Denoting by Π the matrix of the projection operator $\langle \nabla_{\mathbf{x}}\mu, q \rangle$ we have:

$$d = \nabla_{\mathbf{x}}\mu - (\Pi P G)^t P G = \nabla_{\mathbf{x}}\mu - (G^t P^t \Pi P)^t G = \nabla_{\mathbf{x}}\mu + \Lambda^t G,$$

with $\Lambda^t = (\lambda_1, \lambda_2, \lambda_3) \in \mathbb{R}^3$. We have $d \rightarrow 0$ with the optimization iterations converging. Stationarity in d therefore realizes the first order optimality condition for the Lagrangian $L = J + \Lambda^t C$.

With inequality constraints, the solution of our minimization problem needs to verify the first order KKT conditions [46]. But, the optimality condition for the Lagrangian will involve only positive Lagrange multipliers: $\Lambda \in \mathbb{R}_+^3$ and $\nabla_{\mathbf{x}}L = \nabla_{\mathbf{x}}J + \Lambda^t \nabla_{\mathbf{x}}C = 0$ with the complementarity condition $\Lambda^t C = 0$ meaning that $\lambda_i = 0$ if $C_i \leq 0$ and $\lambda_i > 0$ if $C_i = 0$ (i.e. C_i is an active constraint). To define d we follow what put in place for the equality constraints, but only considering active constraints gradients in the definition of s_M which is not anymore a subspace but a convex cone:

$$s_M = \{\mathbf{x} \mid \mathbf{x} = \sum_{i=1}^3 \beta_i \nabla_{\mathbf{x}} C_i, \beta_i > 0 \mid C_i = 0\} \subset \mathbb{R}^3 \subset \mathbb{R}^N \quad (4)$$

At the solution, $\nabla_{\mathbf{x}}J$ is orthogonal to this cone. Before working on the cone, let us start defining a local orthonormal basis $\{\tilde{\mathbf{q}}_{i=1,\dots,p}\}$ for \tilde{s}_M from (4) both with $\beta_i \in \mathbb{R}$. This is therefore a subspace and the basis can be defined as previously with $p = \dim(s_M)$. Now, $\mathbf{q}_i = \pm \tilde{\mathbf{q}}_i$ and the sign chosen such that $\langle \mathbf{q}_{i=1,\dots,p}, \nabla_{\mathbf{x}} C_j \rangle \geq 0$, if $C_j = 0$ for $j = 1, \dots, 3$ (i.e. pointing inside the cone). Here, $\{\mathbf{q}_{i=1,\dots,p}\}$ are therefore the generators of the cone s_M deduced from a basis of \tilde{s}_M . If the generators cannot be defined, the problem is found having no solution as at least two of the constraints are incompatible with the gradients parallel and pointing in opposite directions. These generators permit to define the admissible search direction d from (3) but taking into account that we only remove the non admissible contribution:

$$d = \nabla_{\mathbf{x}}\mu - \sum_{i=1}^p \chi_i \langle \mathbf{q}_i, \nabla_{\mathbf{x}}\mu \rangle \mathbf{q}_i, \quad (5)$$

with $\chi_i = 0$ if $\langle \mathbf{q}_i, \nabla_{\mathbf{x}}\mu \rangle \geq 0$ and $\chi_i = 1$ if $\langle \mathbf{q}_i, \nabla_{\mathbf{x}}\mu \rangle < 0$.

4 A multi-point descent algorithm

Our aim is to use existing platforms. Hence, to compute the ingredients above we use an available single-point optimization environment which can be easily modified for parallel multi-point calculations. This platform involves a direct simulation chain linking the parameters (\mathbf{x}, α) to the state \mathbf{u} solution of a state equation $F(\mathbf{u}(q(\mathbf{x}, \alpha))) = 0$ and its adjoint \mathbf{v} and to a functional j :

- Given $\mathbf{x}_0, 0 < \rho, \mathbf{I}_M, p_{max}$,
- Optimization iterations $p = 1, \dots, p_{max}$
 - 1- M parallel state equation solutions $F(\mathbf{u}(q(\mathbf{x}_p), \alpha_k)) = 0, \alpha_k \in \mathbf{I}_M$,
 - 2- M parallel evaluations of $j(\mathbf{x}_p, \alpha_k), \alpha_k \in \mathbf{I}_M$,
 - 3- M parallel solutions of the adjoint state \mathbf{v} equation:
$$\mathbf{v}^t F_{\mathbf{u}}(\mathbf{u}(q(\mathbf{x}_p), \alpha_k)) = j_{\mathbf{u}}^t, \alpha_k \in \mathbf{I}_M,$$
 - 4- M parallel evaluations of $\nabla_{\mathbf{x}} j(\mathbf{x}_p, \alpha_k) = j_{\mathbf{x}} + (\mathbf{v}^t F_{\mathbf{x}})^t, \alpha_k \in \mathbf{I}_M$,
 - 5-define d the descent direction using (5),

- 6-minimization using d : (e.g. $\mathbf{x}_{p+1} = \mathbf{x}_p - \rho d$),
- Stop if a given stopping criteria is achieved.

In multi-criteria problems steps 2, 3 and 4 include the treatment of more than one functional inducing a different definition of the descent direction d to account for other constraints (mainly physical this time) than the moment-based ones mentioned above.

Despite the natural presence of parallelism due to the M independent evaluations of the state, functional and its gradient, computational complexity remains an issue. We have shown previously how to reduce this effort optimizing the sampling size [34] together with the use of incomplete sensitivity concept in the evaluation of the gradients which permits to avoid the solution of the M adjoint equations [43]. This is particularly suitable when using black-box state equation solvers not providing the adjoint of the state variables.

Such minimization problems have brought new interest to descent methods and this not only because of their lower computational complexity compared to gradient free methods [1, 24, 46]. Indeed, beyond minimization, we saw that gradients are useful to see what should actually be the search space in a context of robust multi-point design [35, 36]. Hence, beyond individual gradient accuracy (i.e. at each of the sampling point), what is important in multi-point problems is the global search space defined by the ensemble of the gradient vectors. This means that one might tolerate higher error levels in each of the gradient defined at the different sampling point than for a single-point optimization situation as what is important is for the global search space to remain nearly unchanged. An interesting mathematical concept which permits to measure the deviation between two subspaces is the principal angles between subspaces.

5 Angles between subspaces

We use the mathematical concept of 'principal angles' between subspaces in the Euclidean spaces (here \mathbb{R}^N) initially introduced by C. Jordan [26]. If the maximum principle angle between the two subspaces is small, the two are nearly linearly dependent. Geometrically, this is the angle between two hyperplanes embedded in a higher dimensional space.

Let us briefly recall the concept of principal angles and how to practically compute them [18, 25]. For simplicity, suppose A and B are two subspaces of dimension k of \mathbb{R}^N , $N \geq 2k$, although this is not a prerequisite to define the principal angles. The k principal angles $\{\theta_i, i = 1, \dots, k\}$ are recursively defined as:

$$\cos(\theta_i) = \frac{\langle a_i, b_i \rangle}{\|a_i\| \|b_i\|} = \max \left\{ \frac{\langle a, b \rangle}{\|a\| \|b\|} : a \perp a_m, b \perp b_m; m = 1, \dots, i-1 \right\},$$

where $a_j \in A$ and $b_j \in B$.

The principal angles θ_i are between 0 and $\pi/2$. This is an important point and will be used later to take advantage of the positivity of the cosine of the angles. The procedure finds unit vectors $a_1 \in A$ and $b_1 \in B$ minimizing the angle θ_1 between them. It then takes the orthogonal complement of a_1 in A and b_1 in B and iterates. This procedure is not useful in practice as computationally inadequate. We would like to be able to find the angles θ_i from the inner products $\langle a_i, b_j \rangle$ of the elements of two bases of A and B [51]. This would be interesting in our multi-point optimization context where we can exhibit an orthonormal basis of the global search space for the multi-point optimization problem using Gram-Schmidt orthonormalization. Now, let $\{a_i, i = 1, \dots, k\}$ and $\{b_i, i = 1, \dots, k\}$ be two arbitrary orthonormal bases for A and B . Orthonormal bases are easy to obtain through the Gram-Schmidt orthonormalization procedure. Consider M being the matrix of the projection operator Pr_A of B onto A defined by:

$$Pr_A(b_i) = \sum_{j=1}^k \langle b_i, a_j \rangle a_j, \quad M = (\langle b_i, a_j \rangle)_{i,j}.$$

The principal angles can be linked to this operator [51] through:

$$M = G \Sigma H^t,$$

where G and H are orthogonal matrices and $\Sigma = \text{diag}(\cos(\theta_i))$.

As G and H are orthogonal matrices, this is a Singular Vector Decomposition (SVD) of M . G and H are unknown at this point. But, we will show that we do not need them to get the θ_i . Otherwise, the approach will be again computationally useless.

We recall that the columns of G are the left-singular vectors of M and eigenvectors of MM^T and the columns of H are the right-singular vectors of M and eigenvectors of $M^T M$. And most important that $\cos^2(\theta_i)$ are the eigenvalues of $P r_A^T P r_A$ which writes in matrix form as: $M^T M = (G \Sigma H^T)^T (G \Sigma H^T) = H \Sigma^2 H^T$ with $\Sigma^2 = \text{diag}(\cos^2(\theta_i))$.

Therefore, to find the principal angles between subspaces A and B , knowing an orthonormal basis in each subspace, one should calculate M and find the eigenvalues of $M^T M$ and take the square root of them. This last operation is valid as the angles are between 0 and $\pi/2$, and their cosine therefore always positive.

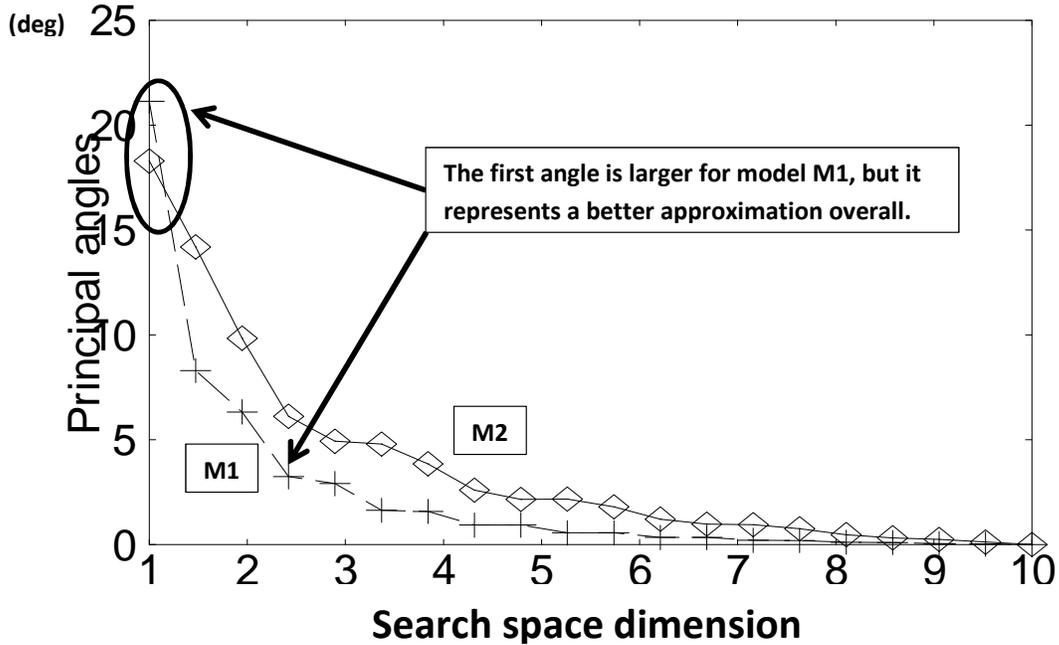


Fig. 2 Principal angles between the subspaces generated by an exact gradient calculation and the linearization of two reduced order models. This permits to quantify the pertinence of an approximation from the whole spectrum. Model M_1 is found to be a better approximation even with a first principal angle slightly larger than with M_2 .

We presented the approach for subspaces of the same dimension k , but it is not necessary for the two subspaces to be of the same size in order to find the angles between them. We need $N \geq 2k$ to be able to exhibit two orthogonal subspaces. If $N < 2k$, some principal angles necessarily vanish and for $N = k$ they all vanish. This procedure is still valid if the subspaces have different dimensions. The projection operator can be defined as well as its transpose and the eigenvalues of $M^T M$ are real as this is a symmetric square matrix.

In our optimization applications we always proceed first with a reduction in size of the search space using a sampling reduction size algorithm [34]. This makes the calculation of the whole set of eigenvalues feasible in terms of calculation complexity. However, if this is out of reach, one can evaluate the bounds on the angles to see the global pertinence of our reduced order models and gradient approximations. This can be done without an exact calculation of the all eigenvalues. It is sufficient to use the Gershgorin circle theorem to find these bounds as every eigenvalue of $M^T M$ lies within at least one of the Gershgorin discs $D((M^T M)_{ii}, R_i)$ centred on $(M^T M)_{ii}$ and with radius $R_i = \sum_{j \neq i} |(M^T M)_{ij}|$. And because $M^T M$ is symmetric, the eigenvalues being real, we only consider the intersection of the discs with the x -axis. Alternatively, the largest and smallest principal angles can be found using iterative power and inverse power methods applied to $M^T M$.

One should however be aware that these bounds might not be sufficiently sharp to discriminate between two reduced order models and decide, for instance, which one is more adequate for sensitivity analysis. Figure 2 shows a typical sketch. It represents principal angles calculated between a first subspace generated by the exact gradients of a transport model for a 10 points sampling of one of the functioning parameters of the model and two subspaces generated by the sensitivities derived from two approximations of this model for the same sampling. Details of the models can be found in [38]. But the modeling problem is not of a main concern here. What is important is that if one only considers the first and last principal angles, model M_2 is found being a better approximation to be used in a linearization procedure. However, with the whole spectrum in hand the picture is quite different and M_1 appears to be more suitable if one intends to use this reduced order model for sensitivity analysis.

Principal angles between multi-point search spaces are interesting to measure the pertinence of sensitivity definitions based on different models or numerics. Indeed, the design will be unaffected by a reduction in the model's complexity if the search subspaces, generated by the gradients at the sampling points of the functioning parameter interval and their approximations, remain the same. This is therefore an original quantification tool for epistemic uncertainties.

6 Inversion for uncertain data

Let us expand the class of problem introduced in section 2 to the following situation:

$$\min_{\mathbf{x} \in \mathbf{O}_{ad}} j(\mathbf{x}, \alpha, \mathbf{u}^*), \quad \mathbf{u}^* \in \mathbb{R}^p, \alpha \in \mathbf{I} \subset \mathbb{R}^n, \mathbf{O}_{ad} \subset \mathbb{R}^N. \quad (6)$$

\mathbf{u}^* represents either measurements or state estimations. It is a vector of random variables. We are interested in functionals j of the form:

$$j(\mathbf{x}, \alpha, \mathbf{u}^*) = \tilde{j}(\mathbf{x}, \alpha) + \frac{1}{2} \|\Pi \mathbf{u}(\mathbf{x}, \alpha) - \mathbf{u}^*(\alpha)\|^2. \quad (7)$$

The first term is what has been discussed up to now. Operator $\Pi : \mathbb{R}^N \rightarrow \mathbb{R}^p$ (typically a linear interpolation operator) makes the state available at data locations. Inverse problems are in this class [22, 54]. This formulation also permits to see the state as uncertain as a whole with Π the identity operator. One can also introduce zoning techniques (as shown in figure 6) to discriminate through the level of confidence one might have on the state evaluation following the variability one observes in practice (experimental or in flight). It is indeed well known that the flow distribution is quite stable in the cockpit and over the first and business class sitting area where the flow is nearly potential. On the other hand, flow variability increases spanwise (easy to see from wings tips motions) and also toward the tail of the aircraft (flying coach once makes this easy to understand). These are due, among others, to separated turbulent flows instabilities and fluid-structure interactions which are more difficult to predict and the state is therefore more 'uncertain' there.

To summarize, we assume the components of \mathbf{u}^* independent, uncertain and given by their Gaussian PDF, for instance, $\mathcal{N}(\mu_i, \sigma_i^2)$, $i = 1, \dots, p$ with mean μ_i and variance of σ_i^2 . $Cov_{\mathbf{u}^*}$ is therefore a diagonal matrix.

The simplest way to measure the effect of these uncertainties on the inversion result is again to proceed with Monte Carlo simulations. This implies a sampling of the variation domain of the data consistent with their PDF. This means we proceed with M independent inversions for M data sets defined by independent choices compatible with the PDF of \mathbf{u}^* given by:

$$\mathcal{N}(\mu_i, \sigma_i^2) \rightarrow (\mathbf{u}^*_i)^m, \quad i = 1, \dots, p, \quad m = 1, \dots, M.$$

These independent inversions will produce M optimal control parameters \mathbf{x}_{opt}^m , $m = 1, \dots, M$ from which statistical moments can be defined (typically the mean and variance) with again a rate of convergence in $M^{-1/2}$ independent of p . Such generation of scenarios is already very demanding when involving only a direct simulation chain. In our problem, each of the scenarios involves an inversion, each requiring several

solutions of the direct and adjoint problems. This complexity makes that this approach is clearly out of the table even if the calculations are independent and can be carried out in parallel.

6.1 Low-complexity uncertainty evaluation

In the sequel, we discuss two low-complexity constructions of $Cov_{\mathbf{x}}$ the covariance matrix of the control parameters from $Cov_{\mathbf{u}^*}$ the covariance matrix of the data. We want these constructions to have a cost comparable to a deterministic inversion and, again, we want to avoid any sampling of a large dimension space.

7 a -Quantile

Consider a random variable v with its PDF known (either analytic or tabulated). The tail of the PDF can be characterized defining for a given probability level ($0 < a < 1$) the following threshold value:

$$\text{VaR}_a = \inf\{l \in \mathbb{R} : P(v > l) \leq 1 - a\}.$$

Different a -quantile are available. One very well known is the Value at Risk (VaR) which has been widely used in financial engineering as a measure of risk of loss on a given asset [27]. We do not need the time dependency issue here but it is interesting as it permits to account for possible improvement of measurement accuracy as discussed in [37].

7.1 Bounding the uncertainty domain

We would like to use the concept of a -quantile (we call in the sequel VaR) to define a closed domain of variation for the uncertain data [37]. Given a threshold $0 \leq a < 1$, a data $\mathbf{u}^*_i, i = 1, \dots, p$ belongs to the interval $[\mu_i + \text{VaR}_a^-, \mu_i + \text{VaR}_a^+]$ with $\text{VaR}_a^- \leq 0 \leq \text{VaR}_a^+$ with probability a . As we consider Gaussian probability density functions we have $\text{VaR}_a^- = -\text{VaR}_a^+$ and the values at risk are explicitly known:

$$\text{VaR}_{0.99}(N(0, 1)) = 2.33, \quad \text{and} \quad \text{VaR}_{0.95}(N(0, 1)) = 1.65,$$

and $\text{VaR}_a(N(0, \sigma^2)) = \sigma^2 \text{VaR}_a(N(0, 1))$. We have therefore, with probability a , an uncertainty domain for the data given by:

$$B_a(\mu) = \prod_{i=1}^p [\mu_i - 1.65\sigma_i^2, \mu_i + 1.65\sigma_i^2] \subset \mathbb{R}^p$$

This is a large domain and we do not want to proceed with any sampling.

7.2 Directional Extreme Scenarios (DES)

However, using the sensitivity of the functional with respect to the data we can identify two directional extreme sets of data corresponding to the intersection of $B_a(\mu)$ and $d = \mu + t \partial j / \partial \mathbf{u}^*$, $t \in \mathbb{R}$. Let us call these two data sets $(\mathbf{u}^*)^\pm$ defined by:

$$(\mathbf{u}^*)^\pm = \mu \pm 1.65 \sigma_i^2 \left(\frac{\partial j / \partial \mathbf{u}^*}{\|\partial j / \partial \mathbf{u}^*\|} \right)_i. \quad (8)$$

To measure of the impact of this variability on the result of the inversion, we proceed with two minimizations with the target data given by $(\mathbf{u}^*)^\pm$ starting from $\mathbf{x}^* = \mathbf{x}_{opt}(\mathbf{u}^* = \mu)$. Let us call $(\mathbf{x}^*)^\pm$ the results of these inversions.

We assume monotonic behavior of the outcome of the inversion with respect to the data:

$$\|\mathbf{x}^*(\boldsymbol{\mu}) - \mathbf{x}^*(\boldsymbol{\nu})\| \nearrow \quad \text{if} \quad \|\boldsymbol{\mu} - \boldsymbol{\nu}\| \nearrow. \quad (9)$$

This assumption is reasonable and means that larger deviations in data sets bring larger variations in the outcome of the optimization. This also suggests that the maximum deviation for the results of the inversion due to the uncertainty on the data can be estimated through: $X^\pm = (\mathbf{x}^*)^+ - (\mathbf{x}^*)^-$. Hence, we introduce a first approximation to the covariance matrix $Cov_{\mathbf{x}^\pm}$ [58] for \mathbf{x} :

$$Cov_{\mathbf{x}^\pm} = \mathbb{E}((X^\pm)(X^\pm)^t) - \mathbb{E}(X^\pm)\mathbb{E}(X^\pm)^t \sim (X^\pm)(X^\pm)^t - (\overline{X^\pm})(\overline{X^\pm})^t, \quad (10)$$

with $\overline{X^\pm} = ((\mathbf{x}^*)^+ + (\mathbf{x}^*)^- - 2\mathbf{x}^*)/2$.

The monotonicity hypothesis can be a posteriori checked, at least partially, measuring the distance between $(\mathbf{x}^*)^\pm$ and $\mathbf{x}^* \pm \rho \nabla_{\mathbf{x}} j(\mathbf{x}^*, \mathbf{u}^*)$, $\rho > 0$. This expression permits to identify two bounds ρ^\pm and two intervals $[0, \rho^\pm]$ on which the monotonicity is verified. Larger values of parameters ρ^\pm a posteriori enforce the hypothesis.

If one looks at optimization from the view point of controllability for dynamical systems [3, 43], quantiles can be introduced in optimization algorithms [37]. The notion of over-solving appears then naturally where it becomes useless to solve accurately near an optimum when the variations in control parameters between two iterations of the optimizer fall below the uncertainties defined through a local uncertainty ball: all the points inside this ball being indeed equivalent in term of the confidence one can have on their performance.

We have presented the concept of Directional Extreme Scenarios in [35, 36] with applications to robust shape optimization in aeronautics, atmospheric dispersion and also to quantify the sensitivity of littoral erosion to uncertainties in bottom sand characteristics [42]. Directional Extreme Scenarios can be defined for \mathbf{x} as well, considering the components of the design vector as random variables. It is indeed interesting to account for uncertainties in large dimensional spaces. We have also extended the DES considering ensemble-based simulations after casting the multi-point optimization algorithm into the Ensemble Kalman Filters (EnKF) formalism (see [39] for the details). The joint application of the EnKF and DES leads to the concept of Ensemble Directional Extreme Scenarios (EDES) which provides more exhaustive possible extreme scenarios knowing the Probability Density Function of our optimization parameters. A sketch of these constructions is shown in figure 3.

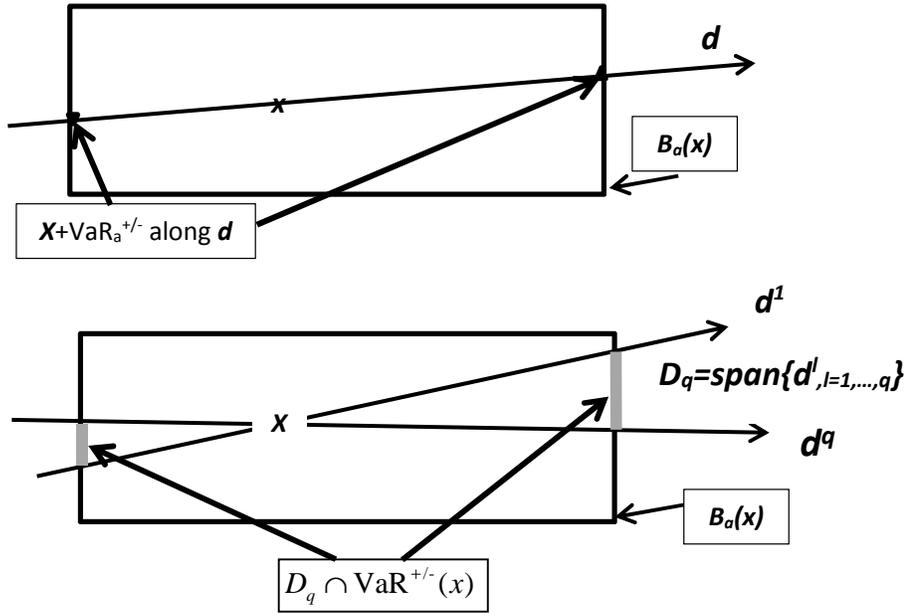


Fig. 3 Sketch of Directional Extreme Scenarios (DES) given by $\mathbf{x}^\pm = d \cap \partial \mathbf{B}_a(\mathbf{x})$ and Ensemble Directional Extreme Scenarios (EDES) $D_q \cap \partial \mathbf{B}_a(\bar{\mathbf{x}})$ for an ensemble of size q ($\bar{\mathbf{x}}$ being the ensemble mean) The grey zone is not necessary connected.

8 From the adjoint to the covariance matrix of the optimization parameters

Another construction of $Cov_{\mathbf{x}}$ takes advantage of our adjoint calculation leading to $\nabla_{\mathbf{x}}j$ the gradient of the functional with respect to the optimization parameters [41].

Let us recall the adjoint formulation for a generic state equation $F(\mathbf{u}(\mathbf{x}, \alpha)) = 0$. The gradient of j with respect to \mathbf{x} writes:

$$\nabla_{\mathbf{x}}j = \frac{\partial j}{\partial \mathbf{x}} + \left(\left(\frac{\partial j}{\partial \mathbf{u}} \right)^t \frac{\partial \mathbf{u}}{\partial \mathbf{x}} \right)^t = \frac{\partial j}{\partial \mathbf{x}} + \left(\left(\frac{\partial j}{\partial \mathbf{u}} \right)^t \left(\frac{\partial F}{\partial \mathbf{u}} \right)^{-1} \frac{\partial F}{\partial \mathbf{x}} \right)^t = \frac{\partial j}{\partial \mathbf{x}} + \left(\mathbf{v}^t \frac{\partial F}{\partial \mathbf{x}} \right)^t,$$

where we have introduced the adjoint variable \mathbf{v} solution of:

$$\mathbf{v}^t \frac{\partial F}{\partial \mathbf{u}} = \left(\frac{\partial j}{\partial \mathbf{u}} \right)^t, \quad (11)$$

and used in algorithm of section 4. In cases the governing equations are self adjoint (i.e. $\frac{\partial F}{\partial \mathbf{u}} = \left(\frac{\partial F}{\partial \mathbf{u}} \right)^t$), one can use the corresponding solver with $\frac{\partial j}{\partial \mathbf{u}}$ as the right-hand side and simply solve:

$$\frac{\partial F}{\partial \mathbf{u}} \mathbf{v} = \frac{\partial j}{\partial \mathbf{u}}.$$

Also, if F is linear, $\frac{\partial F}{\partial \mathbf{u}}$ is a constant operator independent of \mathbf{u} . The interest of the adjoint formulation is that the cost of getting $\nabla_{\mathbf{x}}j$ becomes independent of the size of \mathbf{x} . But, the problem with the adjoint approach is that, except for the two situations we mentioned (linear or self adjoint state equations), it needs the development (and maintenance) of a new code. This is why we use automatic differentiation when possible.

In multi-criteria problems like the one shown in section 9, where the functional j is minimized under equality or inequality constraints $c_{i=1, \dots, q}$, we need to solve an adjoint problem for the functional and each of the active constraints (needed to express the first order KKT conditions). This can be seen as a block diagonal matrix inversion with all blocks similar and the right-hand side given by $(\partial_{\mathbf{u}}j, \partial_{\mathbf{u}}c_1, \dots, \partial_{\mathbf{u}}c_q)^t$ if we have q active constraints. Automatic differentiation in reverse mode with multiple right-hand side capacity can be used to address this problem. Otherwise, deflation techniques for linear systems with multiple right-hand sides can be applied [55, 31] taking advantage of the fact that the blocks being the same the Krylov decomposition needs to be conducted only once.

j involves the least square deviation at data location between model and data. $\partial_{\mathbf{u}}j$ in the right-hand side of (11) can be obtained writing:

$$\begin{aligned} j(\mathbf{x}, \mathbf{u}^*) &= \tilde{j} + \frac{1}{2} \|\Pi \mathbf{u} - \mathbf{u}^*\|^2 = \tilde{j} + \frac{1}{2} \langle \Pi \mathbf{u} - \mathbf{u}^*, \Pi \mathbf{u} - \mathbf{u}^* \rangle \\ &= \tilde{j} + \frac{1}{2} \langle \Pi^t \Pi \mathbf{u}, \mathbf{u} \rangle - \langle \Pi^t \mathbf{u}^*, \mathbf{u} \rangle + \frac{1}{2} \langle \mathbf{u}^*, \mathbf{u}^* \rangle, \end{aligned}$$

and we have $\partial_{\mathbf{u}}j = \partial_{\mathbf{u}}\tilde{j} + \Pi^t \Pi \mathbf{u} - \Pi^t \mathbf{u}^*$. On the other hand, the sensitivity of j with respect to the data $\partial_{\mathbf{u}^*}j$ needed in (8) is given by $\partial j / \partial \mathbf{u}^* = -(\Pi \mathbf{u} - \mathbf{u}^*)$.

With $\nabla_{\mathbf{x}}j$ in hand, let us establish another expression for the covariance matrix of \mathbf{x} considered as a vector of zero-mean random variables. Denote, for simplicity, by \mathbf{u} the model solution (zero-mean valued: $\mathbf{u} \leftarrow \mathbf{u} - \mu$) at data location and suppose it is linked to the parameters through a linear model: $\mathbf{u} = L\mathbf{x}$. The covariance matrix for \mathbf{u} is therefore:

$$Cov_{\mathbf{u}} = \mathbb{E}(\mathbf{u}\mathbf{u}^t) = \mathbb{E}(L \mathbf{x}\mathbf{x}^t L^t) = L \mathbb{E}(\mathbf{x}\mathbf{x}^t) L^t = L Cov_{\mathbf{x}} L^t.$$

If the dependency of \mathbf{u} with respect to the parameter \mathbf{x} is nonlinear the analysis holds for the linearized model. Introducing $\mathcal{J} = \partial \mathbf{u} / \partial \mathbf{x}$ we have:

$$Cov_{\mathbf{u}} = \mathcal{J} Cov_{\mathbf{x}} \mathcal{J}^t.$$

To get $Cov_{\mathbf{x}}$ we need therefore to invert this expression and because the amount of data can be large and probably impossible to exactly fit, we proceed with a least-square formulation looking for $Cov_{\mathbf{x}}$ minimizing:

$$\frac{1}{2} \langle \mathcal{J} \text{Cov}_{\mathbf{x}} \mathcal{J}^t, \mathcal{J} \text{Cov}_{\mathbf{x}} \mathcal{J}^t \rangle - \langle \text{Cov}_{\mathbf{u}}, \mathcal{J} \text{Cov}_{\mathbf{x}} \mathcal{J}^t \rangle.$$

First order optimality condition with respect to $\text{Cov}_{\mathbf{x}}$ gives:

$$\mathcal{J}^t \mathcal{J} \text{Cov}_{\mathbf{x}} \mathcal{J}^t \mathcal{J} - \mathcal{J}^t \text{Cov}_{\mathbf{u}} \mathcal{J} = 0,$$

which leads to

$$\text{Cov}_{\mathbf{x}} = (\mathcal{J}^t \mathcal{J})^{-1} \mathcal{J}^t \text{Cov}_{\mathbf{u}} \mathcal{J} (\mathcal{J}^t \mathcal{J})^{-1},$$

and eventually, to

$$\text{Cov}_{\mathbf{x}} = \mathcal{J}^{-1} \text{Cov}_{\mathbf{u}} \mathcal{J}^{-t} = (\mathcal{J}^t \text{Cov}_{\mathbf{u}}^{-1} \mathcal{J})^{-1}. \quad (12)$$

To get $\text{Cov}_{\mathbf{x}}$ and knowing $\text{Cov}_{\mathbf{u}}$, it is therefore sufficient to evaluate $\mathcal{J} = \partial \mathbf{u} / \partial \mathbf{x}$. The second expression in (12) is interesting as it involves the inversion of a square matrix and gives a least square sense to the inversion of rectangular matrices. Also, if the optimization is successful and model u and data u^* close, we can use the fact that data are usually independent and use the covariance matrix of the observation instead of $\text{Cov}_{\mathbf{u}}$:

$$\text{Cov}_{\mathbf{u}} \sim \text{Cov}_{\mathbf{u}^*},$$

which is then diagonal and its inversion straightforward.

The question is, therefore, how to efficiently evaluate $\mathcal{J} = \partial \mathbf{u} / \partial \mathbf{x}$. The model at data locations $\Pi \mathbf{u}$ is obtained applying, for instance, a linear interpolation operator Π to the model solution \mathbf{u} on the mesh. Therefore, we have:

$$\mathcal{J} = \Pi \frac{\partial \mathbf{u}}{\partial \mathbf{x}}.$$

Now recall that $\nabla_{\mathbf{x}} j$ is available and has been computed with an adjoint approach. We now use it to access $\partial \mathbf{u} / \partial \mathbf{x}$ without extra calculation:

$$\nabla_{\mathbf{x}} j = \frac{\partial j}{\partial \mathbf{x}} + \left(\left(\frac{\partial j}{\partial \mathbf{u}} \right)^t \frac{\partial \mathbf{u}}{\partial \mathbf{x}} \right)^t = \frac{\partial j}{\partial \mathbf{x}} + \left(\left(\frac{\partial j}{\partial \mathbf{u}} \right)^t \Pi^{-1} \mathcal{J} \right)^t,$$

The first terms in the right-hand side is zero if there is no direct dependency on \mathbf{x} in j . It is non-zero, for instance, if a Tykhonov regularization term is introduced in the functional [54]. This leads to:

$$\left(\frac{\partial j}{\partial \mathbf{u}} \right)^t \Pi^{-1} \mathcal{J} = \left(\nabla_{\mathbf{x}} j - \frac{\partial j}{\partial \mathbf{x}} \right)^t,$$

and eventually to,

$$\mathcal{J} = \Pi \left(\frac{\partial j}{\partial \mathbf{u}} \right)^{-t} \left(\nabla_{\mathbf{x}} j - \frac{\partial j}{\partial \mathbf{x}} \right)^t. \quad (13)$$

the components of $\left(\frac{\partial j}{\partial \mathbf{u}} \right)^{-t}$ which is a line vector are given by the inverse of those of $\left(\frac{\partial j}{\partial \mathbf{u}} \right)$ and scaled by the inverse of its size in order to have $\left(\frac{\partial j}{\partial \mathbf{u}} \right)^{-t} \cdot \left(\frac{\partial j}{\partial \mathbf{u}} \right) = 1$. Alternatively, to avoid numerical difficulties with small components of $\left(\frac{\partial j}{\partial \mathbf{u}} \right)$, (13) can again be seen in a least square sense with the inverse of a normal matrix involved:

$$\mathcal{J} = \Pi \left(\left(\frac{\partial j}{\partial \mathbf{u}} \right) \left(\frac{\partial j}{\partial \mathbf{u}} \right)^t \right)^{-1} \frac{\partial j}{\partial \mathbf{u}} \left(\nabla_{\mathbf{x}} j - \frac{\partial j}{\partial \mathbf{x}} \right)^t. \quad (14)$$

This expression involves the inverse of the information matrix $\left(\left(\frac{\partial j}{\partial \mathbf{u}} \right) \left(\frac{\partial j}{\partial \mathbf{u}} \right)^t \right)$. One should be aware that the numerical condition of this matrix can be very poor. We do not discuss this issue here but typically the Bunch and Kaufman [4] algorithm should be used in order to account for this possible deficiency. In particular, if rank deficiency is detected the Moore-Penrose inverse should be used based on the eigenvalue decomposition of the information matrix [9].

Under the hypothesis of the validity of the physical model, this analysis gives indications on the level of backward sensitivity of the optimization parameters with respect to the model solution at data locations which is also the sensitivity with respect of the deviation between the model and data at the data locations (as the data are independent of the optimization parameters):

$$\frac{\partial \mathbf{u}}{\partial \mathbf{x}} = \frac{\partial(\mathbf{u} - \mathbf{u}^*)}{\partial \mathbf{x}}.$$

9 Applications to robust aircraft shape design

These ingredients have been applied to several aircraft shape designs in cruise conditions [34, 36, 35, 41]. Many sources of variability exist in these problems, for instance, due to a change in the weight of the aircraft during the flight because of fuel consumption or due to variability in the flight conditions. Two parameters should be particularly given consideration as our α : the Mach number and the sideslip incidence angle. The sideslip angle is important to account for situations where the aircraft cruises against transverse winds which are very common. A non zero sideslip angle induces fully 3D effects on the flow around the plane making necessary the consideration of a full aircraft during the design. Usually aircraft are designed for a range of angle of incidence. But, these designs are usually realized with the sideslip angle set to zero. It is therefore necessary to reduce the sensitivity of the design with respect to this parameter. However, because the airplane geometry is symmetric spanwise, it is not necessary to consider a symmetric range for the transverse wind but we need to consider the whole aircraft as there is no spanwise symmetry in the flow for non zero sideslip angles.

9.1 Single-point shape optimization platform

We work in the framework of an existing shape optimization platform. We use, in particular, several of its simulation codes for the shape parameterization and deformation, for the mesh deformation, for the flow calculations around the aircraft and for the shape adjoint sensitivity analysis of aerodynamic coefficients. This is a very standard and generic situation and one shall consider these as black-boxes.

Let us briefly recall our direct dependency chain linking independent variables (α, \mathbf{x}) to the dependent variables $(q(\mathbf{x}), U(\alpha, \mathbf{x}))$ describing geometrical entities and state variables and to the cost function (here the drag coefficient C_d) and to the constraints $c_{i=1,\dots,4}$:

$$(\alpha, \mathbf{x}) \rightarrow (q(\mathbf{x}), U(\alpha, q(\mathbf{x}))) \rightarrow (C_d, c_{i=1,\dots,4})(\alpha, \mathbf{x}, q(\mathbf{x}), U(\alpha, q(\mathbf{x}))). \quad (15)$$

It is important to identify all dependencies in order for the sensitivity analysis to be complete, especially when the operating conditions are not anymore single valued. The functional and constraints will be described in section 9.1.3.

9.1.1 Shape parameterization and geometrical entities

In (15) \mathbf{x} denotes a CAD-free parameterization [43, 44] which does not require a priori local regularity assumptions on the shape as it is implicitly the case in Computer Aided Design (CAD) based shape definitions. More precisely, \mathbf{x} represents shape deformations along the normal to the triangular faces of the surface mesh as shown in figure 4. For the problem discussed here this search space has a dimension N of several thousands. This parameterization receives different denominations and belongs to the same class as node-based or free-form shape definitions. In all these approaches the regularity of the deformation needs to be monitored [32, 43]. This parameterization is intermediate in term of generality between CAD definitions of shapes and fully free topological optimization choices where both the regularity and topology of the shape are free. Examples of shape deformation produced by our optimization procedure for different regularity requirements are shown in figure 5. Need for regularity control comes from the fact that, unlike with a CAD definition, the shape $\partial\Omega$ of an object Ω and a gradient-based deformation of $\partial\Omega$ do not belong to the same function space in terms of regularity and, actually, the second is always less regular [41, 43, 44].

This can be illustrated on a simple example with $J(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - b\|^2$ taking $\mathbf{x} \in H^1(\partial\Omega)$, $\mathbf{A}\mathbf{x}$ and b in $L^2(\partial\Omega)$. The gradient $J'_\mathbf{x} = 2\mathbf{A}^T(\mathbf{A}\mathbf{x} - b)$ belongs to $H^{-1}(\partial\Omega)$. Therefore, any variation along $J'_\mathbf{x}$ will have less regularity than \mathbf{x} : $\delta\mathbf{x} = -\rho J'_\mathbf{x} = -\rho(2(\mathbf{A}\mathbf{x} - b)\mathbf{A}) \in H^{-1}(\partial\Omega)$. We therefore need to project (or filter or

smooth) into $H^1(\partial\Omega)$. Now, suppose the shape is described in a finite dimensional parameter space, as for instance with a polynomial definition of a surface (this is like a CAD parameterization). When we consider as control parameters the coefficients of the polynomial, changes in those do not change the regularity as the new shape will always belong to the same polynomial space. Sobolev inclusions give the key for the choice of the regularity operator with the CAD-free parameterization [43]. In our case, because we are using a piecewise linear discretization, a second-order elliptic system with a local definition of the viscosity is sufficient.

This capacity to monitor the regularity of the shape is also interesting as often the optimal solution is not reachable by the current CAD parameterization of the shape. Hence, after an optimization with the CAD-free parameterization and using different level admissible regularity for the shape, one can decide which realization is more suitable and also whether it is interesting or not to enrich the current CAD definition of the shape.

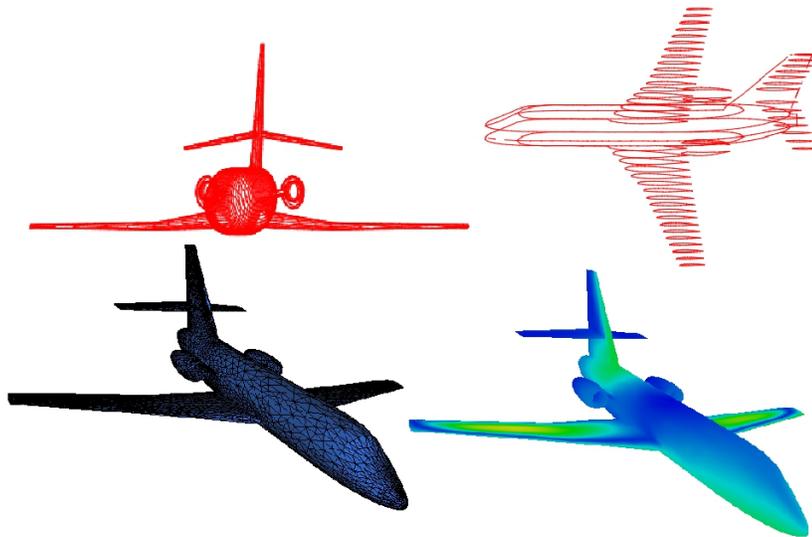


Fig. 4 CAD-free shape parameterization (lower-left) and by-section definitions (upper) of the shape for geometric constraints enforcement. Lower-right is one single $\nabla_x C_d - \langle \nabla_x C_d, \pi \rangle > \pi$ described in section 9.1.3 for this CAD-free parameterization.

$q(\mathbf{x})$ denotes the auxiliary unstructured mesh related geometrical quantities (surfaces, volumes, normals, etc). When the shape is modified, this change must be propagated through the mesh keeping it admissible and we need to recalculate all related geometrical quantities. Admissible and positive mesh deformation is achieved by a 3D torsional spring analogy method [13].

9.1.2 Flow solver

In (15) $U(\alpha, q(\mathbf{x})) = (\rho, \rho\mathbf{u}, \rho E)^t$ denotes the flow variables in conservation form solution of the Euler equations where, T being the temperature, the total energy is given by $E = C_v T + \|\mathbf{u}\|^2/2$ and the pressure by the state law $p = \rho RT$ with R the perfect gas constant.

The details of the implementation of the flow solver are available in [43]. It is based on a finite volume Galerkin method on unstructured tetrahedral meshes [10]. Of course, other choices are possible for the flow solver and the literature on numerical methods for compressible flows is huge. This is not central to our discussion. We target steady solutions and use time marching with local time steps to reach these. The time integration procedure is explicit and is based on a low-storage Runge-Kutta scheme. To improve computational efficiency we only use partial convergence for the state equations. In particular, the sufficient level of convergence retained is when the flow solver iterations only modify the third digits in the aerodynamic coefficients. This is achieved with about 100 RK iterations for this inviscid configurations starting from a

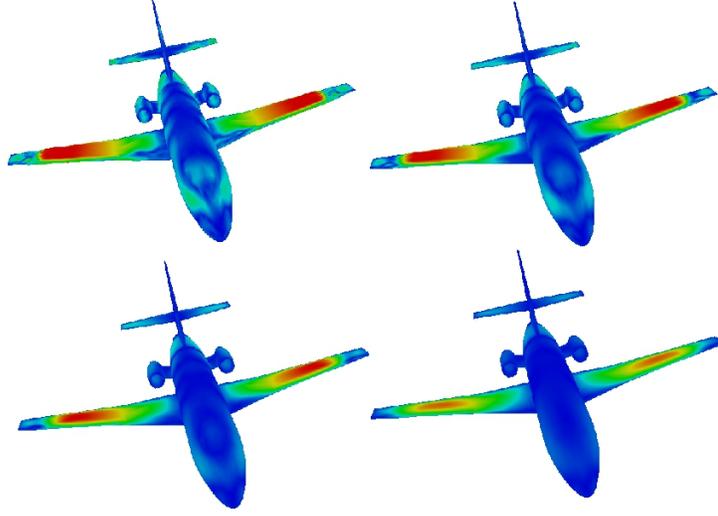


Fig. 5 Regularity control in CAD-free shape parameterization: examples of shape deformation produced by our optimization procedure for different regularity requirements.

uniform solution distribution. During optimization a new calculation for a new shape is always started from the previously available solution making us to proceed with typically only 20 RK new iterations [35, 36, 41].

Often in practice the mesh used for such optimization problems is insufficiently fine. It is however important that the approach uses the ingredients of a generic high fidelity platform and does not remove or simplify any of its ingredients as often it is the case in uncertainty quantification procedures using reduced order models. We should rather consider that in practice our modeling capability and our computational resources will always be limited. The backward uncertainty propagation procedures presented in section 6 permits to quantify the impact of this lack of resolution on the design as shown in figure 6.

9.1.3 Optimization problem

We consider a classical aerodynamic problem where two main quantities of interest are the drag C_d and lift C_l coefficients:

$$C_d(\mathbf{x}) = \frac{1}{2\rho_\infty \|\mathbf{u}_\infty\|^2} \int_{shape(\mathbf{x})} p(q(\mathbf{x})) (\mathbf{u}_\infty \cdot \mathbf{n}(q(\mathbf{x}))) d\gamma, \quad (16)$$

where superscript ∞ indicates inflow conditions. The lift coefficient is evaluated with formula (16) where u_∞ is replaced by \mathbf{u}_∞^\perp in the boundary integral. Aircraft performance analysis concerns its payload and range. These are directly linked to the aerodynamic coefficients of the aircraft called the lift (conditioning the payload) and drag (conditioning the fuel consumption) coefficients. The lift coefficient often appears through an inequality $C_l - C_l^{target} \geq 0$ or equality constraint $c_1 = |C_l^{target} - C_l(p(q(\mathbf{x})))|$ with C_l^{target} a target performance.

Structural efficiency and necessity of useful free volume also implies the consideration of geometric criteria such as a constraint on the volume V of the aircraft or its by-section definition. As for the lift coefficient, this gives a constraint of the form $c_2 = |V^{target} - V(q(\mathbf{x}))|$. The volume of an object Ω (here the aircraft) is expressed through the boundary integral formula: $V = \int_\Omega 1 = \int_\Omega \frac{1}{3} \nabla \cdot (\mathbf{X}) = \int_{\partial\Omega} \mathbf{X} \cdot \mathbf{n}$, where $\mathbf{X} = (x_1, x_2, x_3)^t$ is the local coordinate over the shape.

The last geometric constraint concerns the local wing by-section thickness which is prescribed. We define by-section definitions of the shape where the number of sections n_s is free and can be adapted to account for the complexity of the geometry. Each node in the parameterization is associated to a section Σ_i , and for each section, we define the maximum thickness Δ_i . This last operation requires the projection of the upper-surface nodes over the lower surface for each section. This constraint is expressed as: $c_3 = \sum_{i=1}^{n_s} |\Delta_i(q(\mathbf{x})) - \Delta_i^{target}|$.

An alternative solution which is much simpler to implement is to only enforce a local volume constraint in each section Σ_i using the volume formula above: $V(\Sigma_i) = \int_{\Sigma_i} 1 = \int_{\Sigma_i} \frac{1}{3} \nabla \cdot (\mathbf{X}) \chi_{\Sigma_i} =$

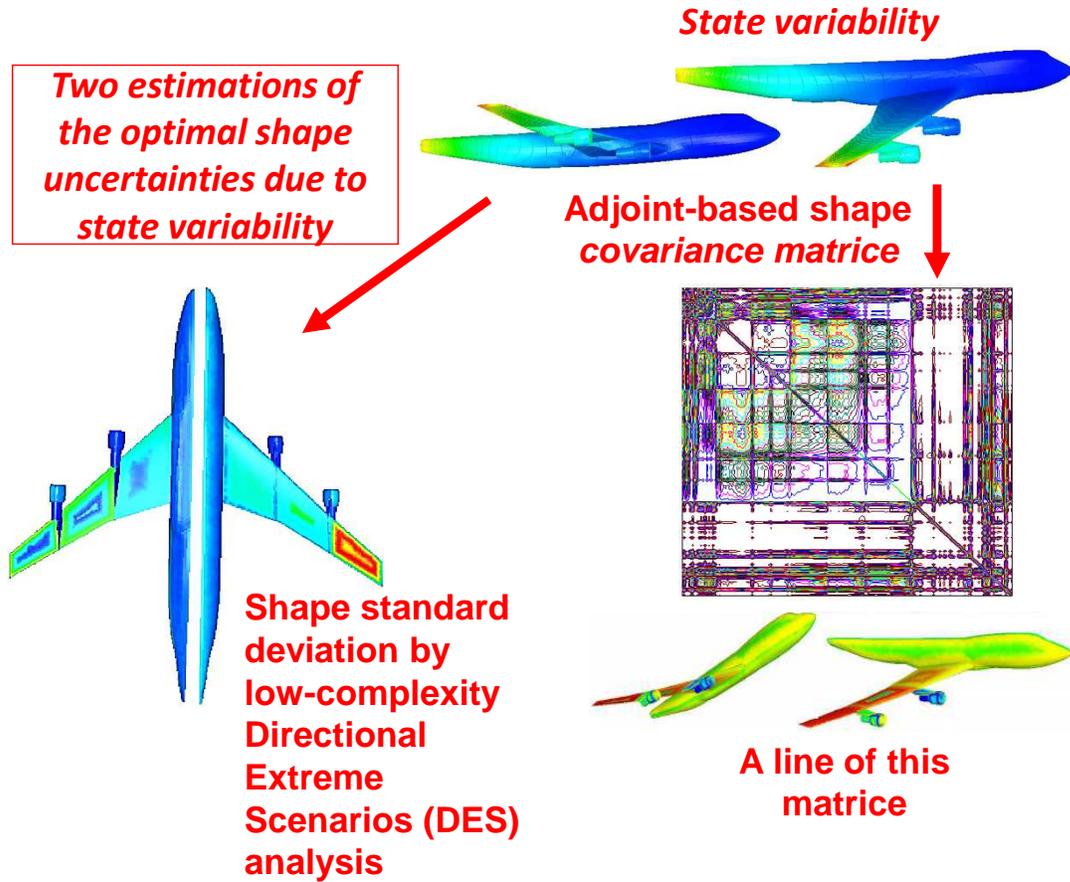


Fig. 6 Two approaches to build $\text{diag}(\text{Cov}_{\mathbf{x}})$ from $\text{diag}(\text{Cov}_{\mathbf{u}^*})$ indicating the variability of \mathbf{u}^* over the shape. The covariance distribution over the shape of the aircraft shows that for the design to be robust in variable flight conditions the engines pylon, fairing and air intakes should have different shapes following their position on the wing.

$\int_{\partial\Omega} \mathbf{X} \cdot \mathbf{n} \chi_{\partial\Sigma_i}$, where χ is an indicator function ($\chi = 1$ if the point is in section Σ_i and $\chi = 0$ otherwise). Testing if a point is in Σ_i is easy and only requires an interval-based coordinate check, spanwise in this situation.

Finally, a fourth term concerns the data assimilation criteria for the pressure over the shape as introduced in section 6: $c_4 = \frac{1}{2} \|\Pi p(\mathbf{x}) - p^*\|^2$, p^* is a vector of random variables for the pressure values on the shape and can be used to account for the impact on the design of the uncertainty on the pressure estimation by the Euler model.

During optimization, the constraints can be accounted for by introducing a penalty term in the cost function: $j = C_d + \sum_{i=1,4} a_i c_i$, $a_i \in \mathbb{R}^+$. But this should be avoided when possible. We use it, however, for the definition of the directional extreme scenarios [41].

One classical technique to recover the lift during optimization is to change the flow incidence taking advantage of the linear relationship between the incidence and lift away from stall conditions. Suppose, however, that we do not want to use either penalty or such approximations. An alternative would be to follow what presented in section 3 and consider a locally admissible gradient orthogonal to $\mathbf{S} = \text{Span}(\nabla_{\mathbf{x}} c_i, i = 1, \dots, 4)$ with $\dim(\mathbf{S}) \leq 4$. Let us denote by π an orthonormal basis of this subspace obtained by the Gram-Schmidt procedure applied to the gradients of the constraints. The admissible gradient is given by:

$$\delta_k = \delta(\mathbf{x}, \alpha_k) = \nabla_{\mathbf{x}} C_d - \langle \nabla_{\mathbf{x}} C_d, \boldsymbol{\pi} \rangle \boldsymbol{\pi}, \quad (17)$$

where \langle, \rangle indicates the scalar product over subspace \mathbf{S} . This is therefore similar to the construction given in (3) where $\boldsymbol{\pi} = \{\mathbf{q}_{i=1, \dots, 3}\}$ and with the constraints C_i replaced by c_i . In the presence of inequality constraints $c_i \leq 0$ instead of equality we build the admissible direction based on the KKT optimality conditions following (5). Once δ_k obtained, the developments of section 3 are followed with the gradient $\nabla_{\mathbf{x}} j$ replaced by the search direction δ_k .

To complete the picture we need to provide $\nabla_{\mathbf{x}} C_d$, $\nabla_{\mathbf{x}} C_l$, $\nabla_{\mathbf{x}} \| \Pi p(\mathbf{x}) - p^* \|^2$, $\nabla_{\mathbf{x}} V$ and $\nabla_{\mathbf{x}} \Delta_i$. The three former require the adjoint of the state equation and we take advantage of the capability for multi-right-hand-side adjoint calculation of `Tapenade` in reverse mode to access these gradients without the solution of three separate adjoint problems. Our direct Euler code uses time marching to the steady solution with local time steps. An optimization of the reverse mode of AD comes from the fact that, our situations of interest being stationary in time, there is no interest in storing the forward states for backward integration [7, 40, 43].

10 CONCLUDING REMARKS

In order to be easily integrated in engineering environments to quantify our confidence on optimal solutions without intensive sampling of large dimensional parameter spaces a cascade of geometric uncertainty quantification concepts has been presented. The cascade is based on the application of data analysis concepts together with existing deterministic simulation platforms.

The analysis starts with the geometric characterization of global sensitivity spaces through their dimensions and relative positions by the principal angles between global search subspaces. Then, joining a multi-point descent direction and extreme values information from the probability density functions of design variables the concept of Directional Extreme Scenarios (DES) has been introduced.

The construction goes beyond DES with Ensemble Kalman Filters (EnKF) after the multi-point optimization algorithm is cast into an ensemble simulation environment. This permits to account for the variability on the functioning parameters through the multi-point formulation and for the variability on the optimization parameters and observation data through the ensemble Kalman filter formulation.

The joint application of the EnKF and DES leads to the concept of Ensemble Directional Extreme Scenarios (EDES) which provides exhaustive possible extreme scenarios knowing the PDF of the optimization parameters and this without a sampling of the admissible space.

The UQ cascade ends with low-complexity solutions for reverse propagation of aleatory uncertain target data in inverse design with two approximations of the covariance matrix of the optimization parameters. These provide uncertainty quantification analysis for the inversion solution with confidence margins on the design parameters in very large design spaces. The constructions also permit to account for epistemic uncertainties considering a model or solution procedure as always imperfect. Hence, seeing the associated error as uncertainty these reverse propagation constructions provide a quantification of the impact of these weaknesses on the design.

Acknowledgements The author would like to thank M. Meaux and F. Gallard from Airbus, M. Montagnac from CERFACS and G. Rogé from Dassault Aviation for their feedback. The different multiple right-hand-side adjoint solvers used for the definition of descent directions in multi-point and constrained optimization problems have been obtained using `Tapenade` AD tool developed at INRIA-Sophia Antipolis by L. Hascoet and his team [19].

References

1. K. Scheinberg A. Conn and L. Vicente. *Introduction to Derivative-Free Optimization*. SIAM, NY, 2002.
2. B. Anderson and J. Moore. *Optimal Filtering*. Prentice-Hall, NY, 1979.
3. P. Redont B. Mohammadi. Improving the identification of general pareto fronts by global optimization. *C. R. Acad. Sci. Paris*, 347:327:331, 2009.
4. J. R. Bunch and L. Kaufman. Some stable methods for calculating inertia and solving symmetric linear systems. *Mathematics of Computation*, 31(137):163–179, 1997.

5. H.-J. Bungartz and M. Griebel. Sparse grids. *Acta Numerica*, 13:147–269, 2004.
6. G. Casella and R. Berger. *Statistical Inference*, 2nd Ed. Duxbury Press, London, 2001.
7. B. Christianson. Reverse accumulation and implicit functions. *Optimization Methods and Software*, 9/4:307–322, 1998.
8. P. Cinnella and S. J. Hercus. Robust optimization of dense gas flows under uncertain operating conditions. *Computers & Fluids*, 39:1893–1908, 2010.
9. P. Courrieu. Fast computation of moore-penrose inverse matrices. *CoRR*, abs/0804.4809, 2008.
10. A. Dervieux. Steady euler simulations using unstructured meshes. *VKI Lecture series, Revised version published in Partial Differential Equations of hyperbolic type and Applications*, World Scientific, 1985/04:23–64, 1985.
11. G. Evensen. Advanced data assimilation for strongly nonlinear dynamics. *Monthly Weather Review*, 125:1342–1354, 1997.
12. G. Evensen. *Sequential Data Assimilation for Nonlinear Dynamics: The Ensemble Kalman Filter In Ocean Forecasting: Conceptual basis and applications*. Springer-Verlag, Heidelberg, 2002.
13. C. Farhat and C. Degand. A three-dimensional torsional spring analogy method for unstructured dynamic meshes. *Computers & Structures*, 80/3:305–316, 2002.
14. A. Gelb. *Stochastic Processes and Filtering Theory*. Academic Press, NY, 1970.
15. A. Gelb. *Applied Optimal Estimation*. M.I.T Press, Boston, 1974.
16. R. Ghanem and A. Doostan. On the construction and analysis of stochastic models: characterization and propagation of the errors associated with limited data,. *J. of Comput. Phys.*, 217:63–81, 2006.
17. R. Ghanem and P. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Springer Verlag, New York, 1991.
18. F. Warner H. Gluck. Great circle fibrations of the three-sphere. *Duke Math. J.*, 50:107:132, 1983.
19. L. Hascoet and V. Pascual. Tapenade user’s guide. In *INRIA Technical report*, pages 1–31. INRIA, 2004.
20. P.G. Hoel. *Introduction to Mathematical Statistics*. John Wiley, London, 1971.
21. G. Iaccarino. *Quantification of Uncertainty in Flow Simulations Using Probabilistic Methods*. VKI, 2008.
22. M. Peyret J. Chery, B. Mohammadi and C. Joulain. Plate rigidity inversion in southern california using interseismic gps velocity field. *Geophys. J. Int.*, 187/2:783–796, 2011.
23. R. O. Onez J. T. Spooner, M. Maggiore and K.M. Passino. *Stable Adaptive Control and Estimation for Nonlinear Systems: Neural and Fuzzy Approximator Techniques*. John Wiley, New York, 2002.
24. J. Jahn. *Vector Optimization: Theory, Applications and Extensions*. Springer, Berlin, 2004.
25. S. Jiang. Angles between euclidean subspaces. *Geometricae Dedicata*, 36(2):113:121, 1996.
26. C. Jordan. Essay on geometry in n dimensions. *Bull. Soc. Math. France*, 3:103:174, 1875.
27. Ph. Jorion. *Value at Risk: The New Benchmark for Managing Financial Risk*. McGraw-Hill, New York, 2006.
28. M. Ghil K. Ide, P. Courtier and A. Lorenc. Unified notation for data assimilation: operational, sequential and variational. *Journal of the Meteorological Society of Japan*, 75/1B:181–189, 1997.
29. R.E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering*, 82:35–45, 160.
30. H.R. Lindman. *Analysis of Variance in Complex Experimental Designs*. Freeman, New York, 1974.
31. A. Dywayne A. Nicely M. Abdou, R.B. Morgan and W. Wilcox. Deflated and restarted symmetric lanczos methods for eigenvalues and linear equations with multiple right-hand sides. *SIAM Journal on Scientific Computing*, 32/1:129–149, 2010.
32. R. Wuchner M. Firl and K. Bletzinger. Regularization of shape optimization problems using fe-based parametrization. *Structural and Multidisciplinary Optimization*, 47/4:507–521, 2013.
33. R. E. Melchers. *Structural Reliability Analysis and Prediction*. John Wiley and Sons, Chichester, 1999.
34. B. Mohammadi. Reduced sampling and incomplete sensitivity for low-complexity robust parametric optimization. *Int. J. Num. Meth. Fluids*, 73/4:307–323, 2013.
35. B. Mohammadi. Principal angles between subspaces and reduced order modeling accuracy in optimization. *Structural and Multidisciplinary Optimization*, 50/2:237–252, 2014.
36. B. Mohammadi. Uncertainty quantification by geometric characterization of sensitivity spaces. *Compt. Meth. Appl. Mech. Eng.*, 280:197–221, 2014.
37. B. Mohammadi. Value at risk for confidence level quantifications in robust engineering optimization. *Optimal Control: Applications and Methods*, 35/2:179–190, 2014.
38. B. Mohammadi. Value at risk for confidence level quantifications in robust engineering optimization. *optimal Control: Applications and Methods*, 35/2:179–190, 2014.
39. B. Mohammadi. Ensemble kalman filters (enkf) and geometric characterization of sensitivity spaces for uncertainty quantification in optimization. *Computer Methods in Applied Mech. & Eng.*, 290:228–249, 2015.
40. B. Mohammadi. Parallel reverse time integration and reduced order models. *J. of Computational Mathematics*, 2:17–33, 2015.
41. B. Mohammadi. Backward uncertainty propagation in shape optimization. *Int. J. for Numerical Methods in Fluids*, 103(4)-DOI: 10.1002/fld.4077:307–323, 2016.
42. B. Mohammadi and F. Bouchette. Extreme scenarios for the evolution of a soft bed interacting with a fluid using the value at risk of the bed characteristics. *Computers & Fluids.*, 89:22–46, 2014.
43. B. Mohammadi and O.Pironneau. *Applied Shape Optimization for Fluids (2nd Edition)*. Oxford Univ. Press, Oxford, 2009.
44. B. Mohammadi and O. Pironneau. Shape optimization in fluid mechanics. *Annual Revue of Fluid Mechanics*, 36/1:255–279, 2004.
45. F. Gallard B. Mohammadi M. Montagnac and M. Meaux. An adaptive multipoint formulation for robust parametric optimization. *J. Opt. Theory & Appl.*, 165(1):DOI 10.1007/s10957–014–0595–6, 2014.
46. J. Nocedal and S. Wright. *Numerical Optimization*. Springer, NY, 2006.

47. G. Obinata and B. Anderson. *Model reduction for control system design*. Springer, Berlin, 2000.
48. C. Correa P.M. Congedo and J.-M. Martinez. Shape optimization of an airfoil in a bzt flow with multiple-source uncertainties. *CMAME*, 200:16–32, 2011.
49. Z. Qu. *Model Order Reduction Techniques with Applications in Finite Element Analysis*. Springer, Berlin, 2004.
50. W. Schilders, H. Van der Vorst, and J. Rommes. *Model order reduction: Theory, research aspects and applications*. 13. Springer Math in Industry series, Berlin, 2008.
51. C. Shonkwiler. *Poincare duality angles for Riemannian manifolds with boundary*. PhD thesis, Univ. Pennsylvania, 2009.
52. SA. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Dokl. Akad. Nauk SSSR* **148**: 1042-1043. *Russian, Engl. Transl.: Soviet Math. Dokl.*, 4:240–243, 1963.
53. Z. Tang and J. Periaux. Uncertainty based robust optimization method for drag minimization problems in aerodynamics. *CMAME*, 12(24):217–220, 2012.
54. A. Tarantola. *Inverse problem theory and methods for model parameter estimation*. SIAM, N.Y., 1987.
55. A. Curioni V. Kalantzis, C. Bekas and E. Gallopoulos. Accelerating data uncertainty quantification by solving linear systems with multiple right-hand sides. *Numerical Algorithms*, 62/2:637–653, 2014.
56. K. Veroy and A. Patera. Certified real-time solution of the parametrized steady incompressible navier-stokes equations: Rigorous reduced-basis a posteriori error bounds. *Int. J. Numer. Meth. Fluids*, 47(8):773–788, 2005.
57. X. Wan and G.E. Karniadakis. Multi-element generalized polynomial chaos for arbitrary probability measures. *SIAM J. Sci. Comput.*, 28/3:901–928, 2006.
58. L. Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. ISBN 0-387-40272-1. Springer, 2004.
59. D. Xiu. *Numerical Methods for Stochastic Computations: A Spectral Method Approach*. Princeton University Press, 2010.