



HAL
open science

Low-complexity moment-based robust design and uncertainty back-propagation

Radia Bouabdallah, Bijan Mohammadi, Robert Rapadamnaba

► **To cite this version:**

Radia Bouabdallah, Bijan Mohammadi, Robert Rapadamnaba. Low-complexity moment-based robust design and uncertainty back-propagation. 2021. hal-03132488v1

HAL Id: hal-03132488

<https://hal.science/hal-03132488v1>

Preprint submitted on 5 Feb 2021 (v1), last revised 9 Sep 2021 (v2)

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.

1. Introduction

The performance of a system designed for given functioning conditions often seriously degrades when these conditions are modified. Today's industrial robust design mainly relies on reduced-order modeling and intelligent sampling [21, 2, 22, 25] which either does not use high-fidelity simulations during design or uses lower accuracy than what would be affordable in a single-point optimization.

By robust design we mean a proposition which ensures similar performances over a given operation range. We will discuss the implication of this requirement on the moments of the performance functional. Our aim is to propose a methodology which permits to design a system having similar performances over a given range of its operating condition or functioning parameters. From a practical point of view, we would like this to be achieved modifying as less as possible an existing single-point optimization platform. In particular, we would like to avoid replacing the high-fidelity ingredients of the platform by low-complexity solvers. Finally, we would like the parallel time-to-solution to remain comparable to the single-point situation. We will show that our direct optimization algorithm permits to achieve such a task.

We address the solution of robust moment-based optimization problems after a multi-point reformulation. The first four moments are considered (i.e. mean, variance, skewness and kurtosis) going beyond classical engineering optimization based on the control of the mean and variance. In particular, the impact on the design of a control of the third and fourth moments is discussed. The multi-point formulation leads to discrete expressions for the moments. linking moment-based and multi-point optimizations. The linearity of the sums in the discrete moments permits an easy evaluation of their gradients with respect to the design variables. Optimal sampling issues are analyzed and a procedure is proposed to quantify the confidence level on the robustness of the design. The proposed formulation is fully parallel and the parallel time-to-solution is comparable to single-point situations.

The literature on multi-point optimization is vast and exhaustive referencing is out of our scope. This formulation has been used, in particular, to extend single-point optimization to account for additional operating conditions.

Forward and backward uncertainty propagation are obviously of great importance with a huge literature dedicated to uncertainty quantification (UQ) [1, 3, 12]. Forward propagation aims at defining, for instance, a probability density function (PDF) for a functional j knowing those of optimization variables [10, 19, 13]. 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) [7, 8, 25, 26]. Backward propagation aims at reducing models bias or calibrating models parameters knowing the PDF of j [4, 9, 24]. This can be seen as a minimization problem and Kalman filters [11] give, for instance, an elegant framework for this inversion assimilating the uncertainties on the observations.

We particularly discuss one aspect of UQ where the target state u^* used in an inverse problem is uncertain. This is the case, for instance, minimizing $j(u(\mathbf{x}), u^*) = \|u(\mathbf{x}) - u^*\|$ to reduce the distance between a model state $u(\mathbf{x})$ and observations or target u^* .

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 u^* is not necessary solution of the state equation making an exact or deterministic inversion pointless. Also, the approach permits to go beyond inversions based on least-squares minimization involving a mean state target.

The uncertainty in measurements is also an interesting way to account for the presence of variability in the state. More generally, as a model or numerical procedure is by nature imperfect and partial, we can consider this uncertainty as a representation and 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. Being able to handle uncertain targets in inverse design is therefore also useful to account for epistemic uncertainties related to possible model or solution procedures deficiencies.

The paper also addresses the issue of backward uncertainty propagation through the estimation of the covariance matrix of the optimization variables. Two procedures for the estimation of this matrix are presented taking advantage of the existence of local linear relations between the target state u^* and the optimization variables \mathbf{x} .

We present the different ingredients of the paper in the context of a linear problem for load distribution optimization with the objective of achieving a target deformation or displacement of a shape. The state equations we consider are those of linear elasticity and we would like to take advantage, as much as possible, of the linear relationship between the optimization variables (e.g loads) and the state variables (e.g the displacements) in the solution of the problem and the different uncertainty quantifications.

The paper starts with the description of our single-point optimization problem to illustrate the situation where the solution of the inverse problem can be explicitly expressed thanks to the mentioned linearity. Then we address robust parametric optimization and its solution through moment-based and multi-point formulations. The paper ends with two low-complexity approaches for backward uncertainty propagation and estimation of the covariance matrix of the optimization variables.

2. State equations and single-point optimization problem

Let us start with the single-point optimization problem of interest. We are interested by inverse problem where the functional typically measures a distance to a target solution.

We consider a generic state equation with a linear state equation and also linear dependency between the optimization variables \mathbf{x} (controlling in our case the loads distribution) and the state variable u : $F(u) = f(\mathbf{x})$, where u is therefore a function of \mathbf{x} .

Our single-point optimization problem reads:

$$\min_{\mathbf{x} \in \mathcal{O}_{ad}} j(u(\mathbf{x}), \alpha) \quad \text{under the constraint that} \quad F(u) = f(\mathbf{x}). \quad (1)$$

Where α represents the other independent variables of the problem such as geometry entities or operating conditions. Here, for simplicity we do not mentioned all dependencies in u . For instance, we do not indicate the spatial dependency and so forth.

In single-point optimization α is fixed in opposition to multi-point or robust optimizations where α components will no longer be fixed but given either through intervals or more sophisticated probability density functions.

The source term $f(\mathbf{x})$ contains N loads expressed as $x_i\delta(s_i)$ where $\delta(s_i)$ is a spatial localization function in space variable s . One would like to optimize:

$$f(\mathbf{x}) = \sum_{i=1}^N x_i\delta(s_i). \quad (2)$$

The optimization variable is then $\mathbf{x} = (x_1, \dots, x_N) \in (\mathbb{R}^+)^N$.

The inverse problem of interest aims at making the solution u as close as a target u^* over the calculation domain Ω , or some subset of it. Using a least-squares formulation over the domain with spatial integration, this reads:

$$j(u(\mathbf{x}), \alpha) = \frac{1}{2} \int_{\Omega} (u(\mathbf{x}) - u^*)^2 ds. \quad (3)$$

Taking advantage of the linear dependency between \mathbf{x} and the solution u and using the superposition principle, the solution can be expressed through projections over $N + 1$ elementary solutions:

$$u = u_0 + \sum_{i=1}^N x_i u_i,$$

In this decomposition, u_0 is the homogeneous solution (the zero load situation together with non-homogeneous boundary conditions) and the other elementary solutions $u_{i=1, \dots, N}$ are obtained respectively with $\mathbf{x} = (0, \dots, 0, 1, 0, \dots, 0)$ with 1 at the i^{th} position. The elementary solutions are obtained with homogeneous Dirichlet boundary conditions.

Once the homogeneous and elementary solutions known, the optimization variable \mathbf{x} is solution of a linear system:

$$A\mathbf{x} = b(\alpha), \quad (4)$$

of size N derived from first order optimality condition for j , $\nabla_x j(u(\mathbf{x}), \alpha) = 0$, with:

$$A_{i,j} = \langle u_i, u_j \rangle \quad \text{and} \quad b_i(\alpha) = \langle (u_0 - u^*), u_i \rangle, \quad \text{for } i, j = 1, \dots, N,$$

where $\langle v, w \rangle = \int_{\Omega} v(s) w(s) ds$.

A fundamental remark now is that the cost of this inversion is negligible compared to the cost of calculating the elementary solutions. Also, it is important to notice that the matrix A does not depend on α . Therefore, any numerical transformation necessary for its inversion is made only once. Uncertainty quantification is situation where the target solution u^* is uncertain becomes therefore feasible at low cost as it will not require any extra solution of the state equations. In the sequel we will discuss different ways to handle this problem.

2.1. 3D elasticity

We will illustrate the ingredients of the paper on a 3D problem where the state equation $F(u) = f(x)$ is the elasticity system whose solution is given using the Cast3M [5] industrial solver on tetrahedral meshes which we consider as a black-box. Something which is made possible thanks to the formulation of the optimization problem and this is one strength of the approach.

Briefly speaking, Cast3M is a finite element software for structural and fluid mechanics developed by the French Alternative Energies and Atomic Energy Commission (CEA). Cast3M provides a mesh generation tool, partial differential equations simulation modules coupled with visualization tools.

The 3D problem we consider consists of a geometry of an elastic piece represented in a $(10m \times 10m \times 1m)$ concrete slab with hollow bodies [23], supported by point supports at the four corners modeled by homogeneous Dirichlet boundary conditions for the displacement. The mechanical characteristics of the concrete considered are $E = 30GPa$ for Young's modulus and $\nu = 0.2$ for Poisson's ratio. This is a complex problem and a full 3D calculation is quite time consuming. The present solution based on the use of elementary 3D solution is therefore quite adapted for the optimization of load distributions for such a cases as the state equations calculations can be carried out independently from the optimization problem and in a fully parallel way. The hollow bodies involve complex mesh generation issues as shown in Figure 2.1. The 3D mesh has about 14400 grid points.

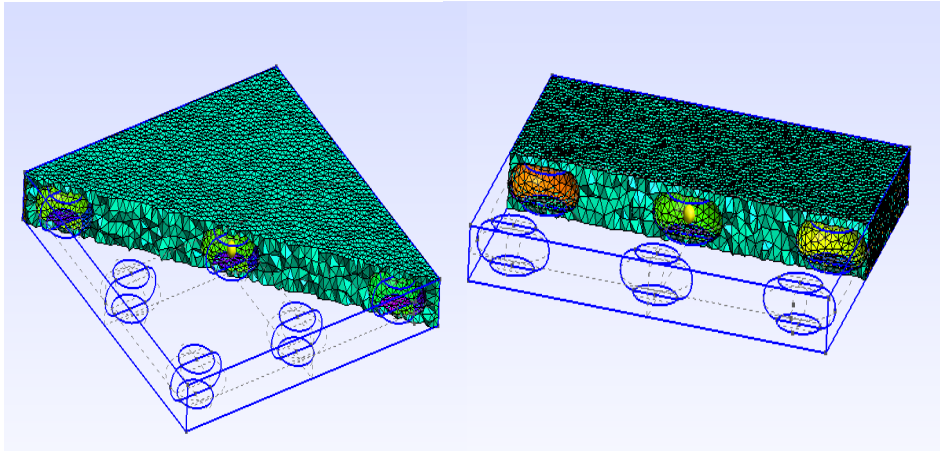


Figure 1. Example of a complex mesh generated by Gmsh [6] taking into account the presence of hollow bodies.

We assume negligible volume forces and consider a situation with a relatively small number of loads to ease the comparison between the different approaches. Considering higher number of loads does not pose any supplementary difficulty and only requires parallel calculations of additional independent elementary solutions.

2.2. Illustration using $N = 5$ optimization variables

We apply an arbitrary five point loads distribution (i.e. $N = 5$). Any other combination can be considered, five is suitable as enough large to illustrate the approach and enough low to show images of all elementary contributions.

We calculate the elementary displacements $u_{i=1,\dots,5}$ obtained after application of unit forces x_i . The total displacement is the sum of these elementary displacements $u = u_0 + \sum_{i=1}^{N=5} x_i u_i$ where u_0 is the displacement without any forces applied but in the presence of Dirichlet boundary conditions. In our problem we consider $u_0 = 0$.

Figure 2 shows the elementary and total structural displacements which will be considered as target deterministic displacement u^* . Uncertainty analysis will take place around this target displacement.

3. Robust parametric optimization

Consider a cost function involving the optimization variables \mathbf{x} in an admissible ensemble \mathbf{O}_{ad} and another parameter α , not considered as a design variable:

$$\min_{\mathbf{x} \in \mathbf{O}_{ad}} j(\mathbf{x}, \alpha), \quad \alpha \in \mathbf{I} \subset \mathbb{R}^n, \mathbf{O}_{ad} \subset \mathbb{R}^N. \quad (5)$$

As mentioned, α can be an operating condition or in an inverse problem [20, 24] the target solution u^* in a functional of the form $j(\mathbf{x}, \alpha = u^*) = \|u(x) - u^*\|$. Both the operating conditions and the target solution can be uncertain and given only through probability density functions. Handling the uncertainty on the operating conditions is necessary to achieve robust design and usually requires a multi-point formulation. The number of operating condition parameters is usually small compared to the size of the optimization problem. This is a very general context and we previously visited it to address robustness issues in optimization with respect to \mathbf{x} and α in general frameworks [17, 15, 16, 18]. In this work, we mainly focus on handling the uncertainty on $\alpha = u^*$ using the facility offered by the linear dependency between the optimization variables and the state variable.

3.1. Moment-based optimization

Multi-point optimization can be used to address such optimization problem [14]. The aim is to remove the dependency in α during optimization. This can be done, for instance, minimizing a functional $J(\mathbf{x})$ encapsulating this dependency under a constraint on the higher moments of j :

$$J = \mu = \mathbb{E}[j(\mathbf{x}, \alpha)], \quad \text{such that} \quad \sigma = \mathbb{E}[(j(\mathbf{x}, \alpha) - \mu)^2] \leq \sigma_0, \quad (6)$$

where $\mathbb{E}[\cdot]$ is the expectation operator.

We can go beyond the first two moments and use also the third and the fourth moments. 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 for the PDF of the solution of a simulation to remain uniform or Gaussian.

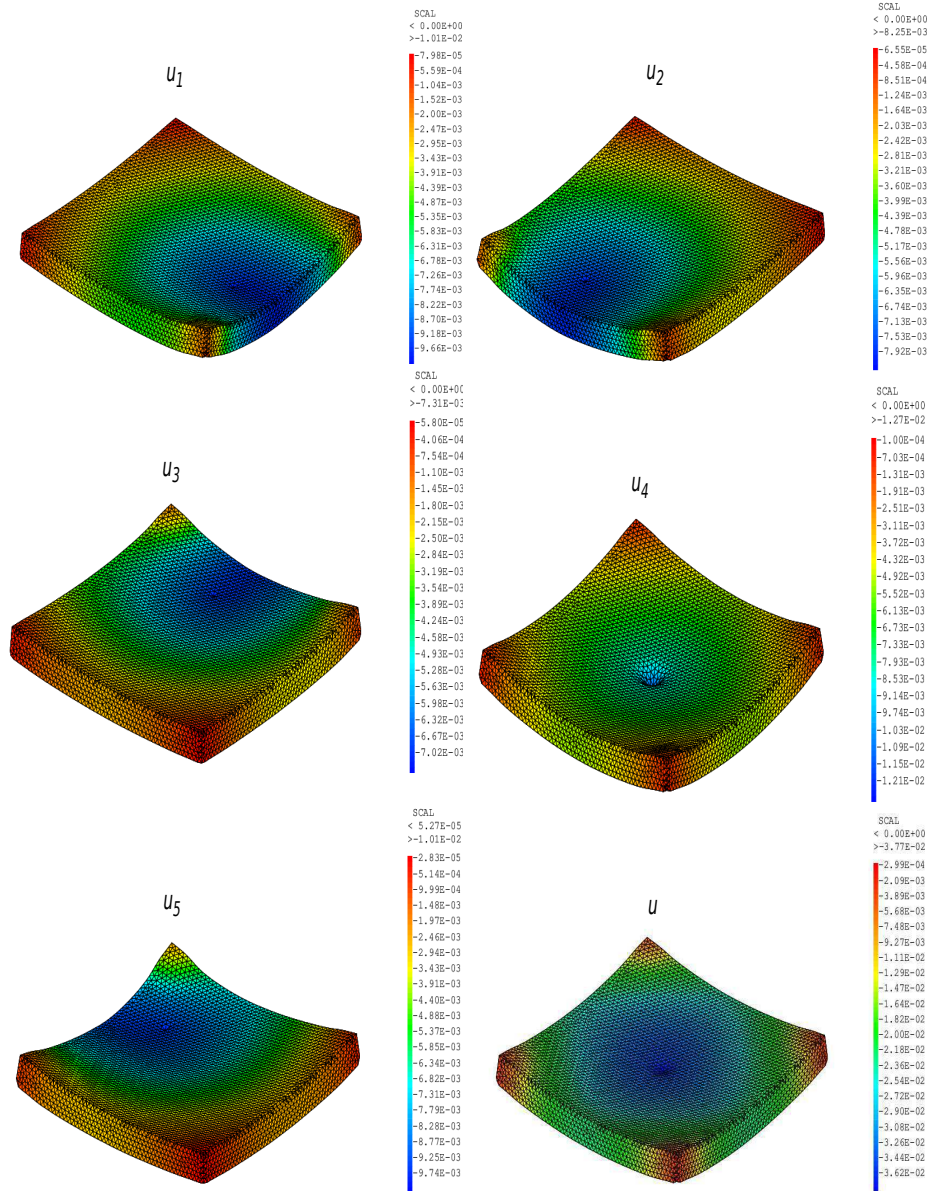


Figure 2. The 5 first pictures represent elementary displacements $u_{i=1,\dots,5}$ and the bottom right picture shows the total structural displacement $u = u_0 + \sum_{i=1}^{N=5} x_i u_i$, where $u_0 = 0$ in this case. The elementary displacements cannot be retrieved using symmetry considerations from each other.

The third and fourth moments, the skewness γ and the kurtosis κ , are defined as:

$$\gamma = \mathbb{E}\left[\left(\frac{j(x, \alpha) - \mu}{\sigma}\right)^3\right], \quad \kappa = \mathbb{E}\left[\left(\frac{j(x, \alpha) - \mu}{\sigma}\right)^4\right]. \quad (7)$$

The skewness is an indication of the deviation from symmetry of the PDF and the kurtosis permits to quantify if it is tall and skinny or short and squat.

One can consider that a robust design should favor symmetry in the distribution which means a lower absolute value of skewness. This is an assumption and other non-zero values can be targeted for the absolute value of the skewness. However, the assumption of symmetry appears reasonable. Indeed, when driving a car on a straight line, one expects the car to have the same behavior for small and symmetrical disruptions in holding the steering wheel by the driver. 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 either positive or negative. For an uni-modal PDF a reduction of the skewness comes when the mean and the mode of the distribution converge toward each other at given standard deviation.

Robust design means 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. We therefore consider that 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 .

To summarize, robust moment-based optimization can be seen through the following constrained minimization problem:

$$\begin{aligned} \min_{\mathbf{x} \in \mathbf{O}_{ad}} J(\mathbf{x}) = \mu(\mathbf{x}) \quad \text{such that} \\ \sigma(\mathbf{x}) - \sigma_0 \leq 0, \quad |\gamma(\mathbf{x})| - |\gamma_0| \leq 0, \quad \kappa(\mathbf{x}) - \kappa_0 \leq 0. \end{aligned} \quad (8)$$

Equality constraints on the moments are particular cases of these. In cases, higher kurtosis are targeted (see discussion above), the last constraint becomes $-\kappa(\mathbf{x}) + \kappa_0 \leq 0$.

3.2. Discrete expressions

Monte Carlo simulations permit to recover these moments with an error decreasing as σ/\sqrt{M} with M the number of functional evaluations and this with a convergence rate 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 the moments at a given accuracy. As we are interested by small values of N , this latter may therefore be preferred.

Both Monte Carlo trials and numerical integration lead to the introduction of weighted sums [14] over a M -point sampling \mathbf{I}_M as estimators of the previous moments (denoted with the same notation):

$$\mu = \sum_{\alpha_k \in \mathbf{I}_M} \omega_k j(\mathbf{x}, \alpha_k), \quad \sigma^2 = \sum_{\alpha_k \in \mathbf{I}_M} \omega_k (j(\mathbf{x}, \alpha_k) - \mu)^2, \quad (9)$$

$$\gamma = \sum_{\alpha_k \in \mathbf{I}_M} \omega_k (j(\mathbf{x}, \alpha_k) - \mu)^3, \quad \kappa = \sum_{\alpha_k \in \mathbf{I}_M} \omega_k (j(\mathbf{x}, \alpha_k) - \mu)^4. \quad (10)$$

A major difference between Monte Carlo and numerical integration is that in the former the sampling \mathbf{I}_M and the weights are chosen according to the PDF of α . In this paper we consider uniform PDF. We have therefore uniform sampling in each of the dimension of α and $\omega_k \sim 1/M$. Unbiased estimates use slightly different coefficients and also introduce corrections. For instance, one should consider $\omega_k = 1/(M-1)$ for σ , $\omega_k = M/((M-1)(M-2))$ for γ and κ . However, because we are interested by the gradients of these quantities with respect to \mathbf{x} and gradient-based minimization algorithms, these changes will only have slight impacts on the amplitude of the gradient with no real incidence on the optimization history, especially if optimal descent step sizes are used. This discussion also shows that moment-based and multipoint optimization can be seen as a whole. If different PDFs are targeted than uniform, we need to redefine the sampling \mathbf{I}_M and the weights ω_k accordingly. But, this is beyond the scope of this paper.

3.3. Gradients of the moments

The linearity in the sums permits to access to the gradients of the moments with respect to the optimization variables \mathbf{x} from the gradient of the functional at the sampling point α_k . We recall that we took $\sigma = 1$ in Equation (7) to ease this linearization. The different gradients read:

$$\begin{aligned} \nabla_{\mathbf{x}} \mu &= \sum_{\alpha_k \in \mathbf{I}_M} \omega_k \nabla_{\mathbf{x}} j(\mathbf{x}, \alpha_k), \\ \nabla_{\mathbf{x}} \sigma^2 &= \sum_{\alpha_k \in \mathbf{I}_M} 2\omega_k (j(\mathbf{x}, \alpha_k) - \mu) \nabla_{\mathbf{x}} (j(\mathbf{x}, \alpha_k) - \mu), \\ \nabla_{\mathbf{x}} \gamma &= \sum_{\alpha_k \in \mathbf{I}_M} 3\omega_k (j(\mathbf{x}, \alpha_k) - \mu)^2 \nabla_{\mathbf{x}} (j(\mathbf{x}, \alpha_k) - \mu), \\ \nabla_{\mathbf{x}} \kappa &= \sum_{\alpha_k \in \mathbf{I}_M} 4\omega_k (j(\mathbf{x}, \alpha_k) - \mu)^3 \nabla_{\mathbf{x}} (j(\mathbf{x}, \alpha_k) - \mu). \end{aligned} \quad (11)$$

Knowing the elementary gradients $\nabla_{\mathbf{x}} j(\mathbf{x}, \alpha_k)$ at sampling points α_k brings interesting information on the geometry of the problem. Indeed, they can be used to build a vector space and the analysis of this space provides important information on the complexity of the problem [14, 15, 16].

The natural presence of parallelism due to the M independent evaluations of the state, functional and its gradient is interesting to monitor computational complexity. In addition, as mentioned earlier for our problem, we notice that only the right-hand-sides $b_{i=1, \dots, N}$

need to be assembled after each perturbation of u^* . Therefore, the optimality condition mentioned in section 2 for j can be extended to μ writing:

$$\nabla_x \mu = \sum_{\alpha_k \in \mathbf{I}_M} \omega_k \nabla_x j(\mathbf{x}, \alpha_k) = 0,$$

which gives the following linear system to solve:

$$A\mathbf{x} - \sum_{\alpha_k \in \mathbf{I}_M} \omega_k b_{\alpha_k} = A\mathbf{x} - \sum_{\alpha_k \in \mathbf{I}_M} \omega_k < (u_0 - u^*), u > = 0,$$

with the matrix $A_{i,j} = < u_i, u_j >$ unchanged.

In the presence of constraints on moments, we can proceed in the same way to get the gradients of the moments, again using the nullity of the gradient of the first moment at the optimum:

$$\begin{aligned} \nabla_x \sigma^2 &= \sum_{\alpha_k \in \mathbf{I}_M} 2\omega_k (j(\mathbf{x}, \alpha_k) - \mu) \nabla_x j(\mathbf{x}, \alpha_k) \\ &= C_1(\mathbf{x}) A\mathbf{x} - \sum_{\alpha_k \in \mathbf{I}_M} 2\omega_k (j(\mathbf{x}, \alpha_k) - \mu) b_{\alpha_k}, \end{aligned}$$

$$\begin{aligned} \nabla_x \gamma &= \sum_{\alpha_k \in \mathbf{I}_M} 3\omega_k (j(\mathbf{x}, \alpha_k) - \mu)^2 \nabla_x j(\mathbf{x}, \alpha_k) \\ &= C_2(\mathbf{x}) A\mathbf{x} - \sum_{\alpha_k \in \mathbf{I}_M} 3\omega_k (j(\mathbf{x}, \alpha_k) - \mu)^2 b_{\alpha_k}, \end{aligned}$$

$$\begin{aligned} \nabla_x \kappa &= \sum_{\alpha_k \in \mathbf{I}_M} 4\omega_k (j(\mathbf{x}, \alpha_k) - \mu)^3 \nabla_x j(\mathbf{x}, \alpha_k) \\ &= C_3(\mathbf{x}) A\mathbf{x} - \sum_{\alpha_k \in \mathbf{I}_M} 4\omega_k (j(\mathbf{x}, \alpha_k) - \mu)^3 b_{\alpha_k}. \end{aligned}$$

We see that these are four vectors involving a same matrix A and elementary contributions b_{α_k} with different weightings and scaling C_1 , C_2 and C_3 which are defined by:

$$C_1(\mathbf{x}) = \sum_{\alpha_k \in \mathbf{I}_M} 2\omega_k (j(\mathbf{x}, \alpha_k) - \mu),$$

$$C_2(\mathbf{x}) = \sum_{\alpha_k \in \mathbf{I}_M} 3\omega_k (j(\mathbf{x}, \alpha_k) - \mu)^2,$$

$$C_3(\mathbf{x}) = \sum_{\alpha_k \in \mathbf{I}_M} 4\omega_k (j(\mathbf{x}, \alpha_k) - \mu)^3.$$

3.4. Controlling the moments

A classical approach to improve robust design in engineering is to control the first moment under the constraint of leaving unchanged or even reduce the second one. An easy way to adapt this concept to our situation is to consider the following descent direction orthogonal to the gradient of the variance:

$$d = - \left(\nabla \mu - \langle \nabla \mu, \frac{\nabla \sigma}{\|\nabla \sigma\|} \rangle \frac{\nabla \sigma}{\|\nabla \sigma\|} \right),$$

where $\nabla \sigma / \|\nabla \sigma\|$ is the unit vector along $\nabla \sigma$. Unfortunately, this formulation does not permit anymore to get the optimal solution directly inverting a linear system as it is the case in the single-point or even when minimizing solely the first moment. An iterative procedure is then necessary using some descent algorithm. We will see examples of these below.

One can generalize the previous construction to the case of more than two moments using a Gram-Schmidt orthonormalization procedure. Let us organize the four vectors in the following order:

$$(\nabla_{\mathbf{x}} \kappa, \nabla_{\mathbf{x}} \gamma, \nabla_{\mathbf{x}} \sigma^2, \nabla_{\mathbf{x}} \mu). \quad (12)$$

To reduce the impact of the solution of the first order optimality condition on the second, third and fourth moments, we require the descent direction to be orthogonal to the subspace generated by the the gradients of these moments. This can be easily achieved using a Gram-Schmidt orthonormalization which extracts from the set of the vectors above a new set of orthonormal vectors $v_{i=1,\dots,4}$:

$$(v_1, v_2, v_3, v_4).$$

The descent direction we use in minimization is defined by:

$$d = -\|\nabla_{\mathbf{x}} \mu\| v_4.$$

In this way, we make sure the descent direction is orthogonal to the subspace generated by moments 2, 3 and 4. The organization of the vectors as given in (12) is important to make sure the direction monitoring optimization is the one given by the first moment. Once this is achieved, this descent direction can be used in an iterative procedure which is again necessary as the constrained optimality condition cannot simply be cast into a single linear system.

3.5. Multi-point optimization for a quadratic functional

Let us illustrate our ingredients on a simple example. The functional involves a least-squares minimization:

$$j(\mathbf{x}, \alpha) = \frac{1}{2} \sum_{i=1}^N (x_i - \alpha)^2, \quad -0.5 \leq \alpha \leq 0.5, \quad -5 \leq x_{i=1,\dots,N} \leq 5, \quad N = 40. \quad (13)$$

Let us solve problem (8) with a gradient method. Here, α is a scalar and for a given α the optimality condition for $j(\mathbf{x}, \alpha)$ gives obviously $\mathbf{x}^* = (\alpha, \dots, \alpha) \in \mathbb{R}^N$. This would be the solution of a single-point optimization. Here α has a uniform PDF and we use

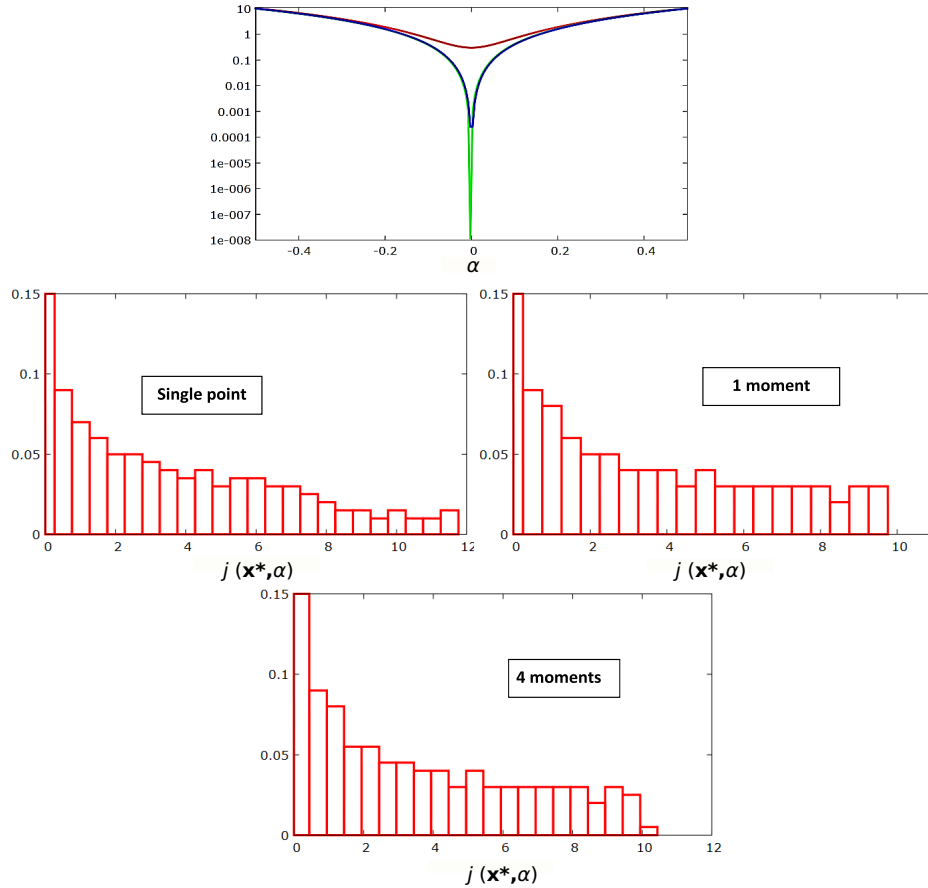


Figure 3. Upper: functional $j(\mathbf{x}^*, \alpha)$ given by Equation (13) vs. α for single-point optimization, mean-based and with constraints on moments two, three and four. Lower: histograms of $j(\mathbf{x}^*, \alpha_{k=1, \dots, M})$ for a single-point minimization and when controlling one and then all first four moments.

a uniform sampling \mathbf{I}_M with $M = 40$. Minimizing the mean μ without any constraint on the other moments still produces a non robust optimum as the performance has large variability over the range of the operating condition α . On the other hand, asking for the first moments to decrease under constraints on the other three produces a much flatter functional over the range of α . Figure 3 shows the histograms of $j(\mathbf{x}^*, \alpha_{k=1, \dots, M})$ for these minimization. In particular, one sees how controlling more moments affects the left peak and the distribution of the frequencies.

3.6. Application to our 3D elastic problem

We now illustrate the multi-point discussion with moments on our 3D elasticity problem presented in section 2.1 where target displacement u^* is assumed uncertain with its

Probability Density Function (PDF) known. The target is a large vector with a size given by the number of the 3D mesh points. We consider centered Gaussian perturbations of u^* with a standard deviation of 10%.

In our elasticity optimization problem, $\mathbf{O}_{ad} = (\mathbb{R}^+)^N$. This means that each of the optimization variable needs to be positive. This brings additional constraints to those expressed earlier on the moments. This also means that one cannot solve anymore the single-point optimization by a sole inversion of a linear system and that, for instance, a projected gradient algorithm is necessary. This can be easily included during the iterations when minimizing the first moment under constraints on the other three.

Figure 4 shows the distribution of the ratio of the functional $j(\mathbf{x}^*, \alpha_k)$ to the Young modulus for $M = 1000$. One sees that the uncertainty on u^* brings large variations in j . This variability shows the importance of a robust optimization approach for this problem. The impact of considering the variability during minimization through moment based functional can be easily demonstrated looking at the moment-based solutions of the minimization versus the solution of the single-point optimization for the target u^* which corresponds to the deterministic situation. Table 1 features the solution of the optimization variables for this comparison. We see that the results are clearly different. Also the histograms in Figure 4 show that controlling all four moments greatly reduces the variability of the functional and therefore improves the robustness of the design. This can be confirmed considering the 4 moments showing the variability of the functional induced by the variability of the target displacement, considered as uncertain, for the single-point optimal solution and for the different optimal solutions obtained when controlling respectively the first, second and eventually all four moments. The first column in Table 2 show the first moment for $j(\mathbf{x}, \alpha)/E$ and the other columns show the second, third and fourth moments for $(j(\mathbf{x}, \alpha) - \mu)/E$. By extension, we denote these $\mu(j(\mathbf{x}, \alpha)/E)$, $\sigma((j(\mathbf{x}, \alpha) - \mu)/E)$, $\gamma((j(\mathbf{x}, \alpha) - \mu)/E)$ and $\kappa((j(\mathbf{x}, \alpha) - \mu)/E)$. The first column indicates that the third first designs give about the same average behavior while controlling all four moments clearly provide a different, and actually better, solution. Robustness improvements can be seen from the evolution of the second and fourth moments where again controlling all four moments is recommended. Also, the final density functions are slightly less symmetric when controlling more moments. However, one regain symmetry when controlling 4 moments instead of only two. As we expressed in section 3.1, symmetric distributions should be preferred in robust designs.

	Single-point	1 moment	2 moments	4 moments
x_1 (KN)	5500	5496	5266	7035
x_2 (kN)	4000	4000	3917	9262
x_3 (KN)	6000	5994	5518	6834
x_4 (KN)	4500	4503	4330	0
x_5 (KN)	4999	5005	5594	1772

Table 1. Optimization variables for minimization using the deterministic target u^* (the single-point case) and when controlling 1, 2 and all 4 moments of j . The solutions are clearly different showing the impact of moment-based optimization. The robustness achieved with more moments accounted for in the design is well illustrated through the histograms in Figure 4.

	$\mu(j/E)$	$\sigma((j - \mu)/E)$	$\gamma((j - \mu)/E)$	$\kappa((j - \mu)/E)$
single point	74.980	0.938	-0.005	2.422
1 moment	74.940	0.865	-0.018	2.180
2 moments	74.144	0.840	0.094	2.146
4 moments	64.138	0.765	0.032	1.685

Table 2. Distributions of the different moments for the optimal solutions obtained after a single-point optimization and when controlling first, second and eventually all four moments. This final design is superior both in performance and robustness.

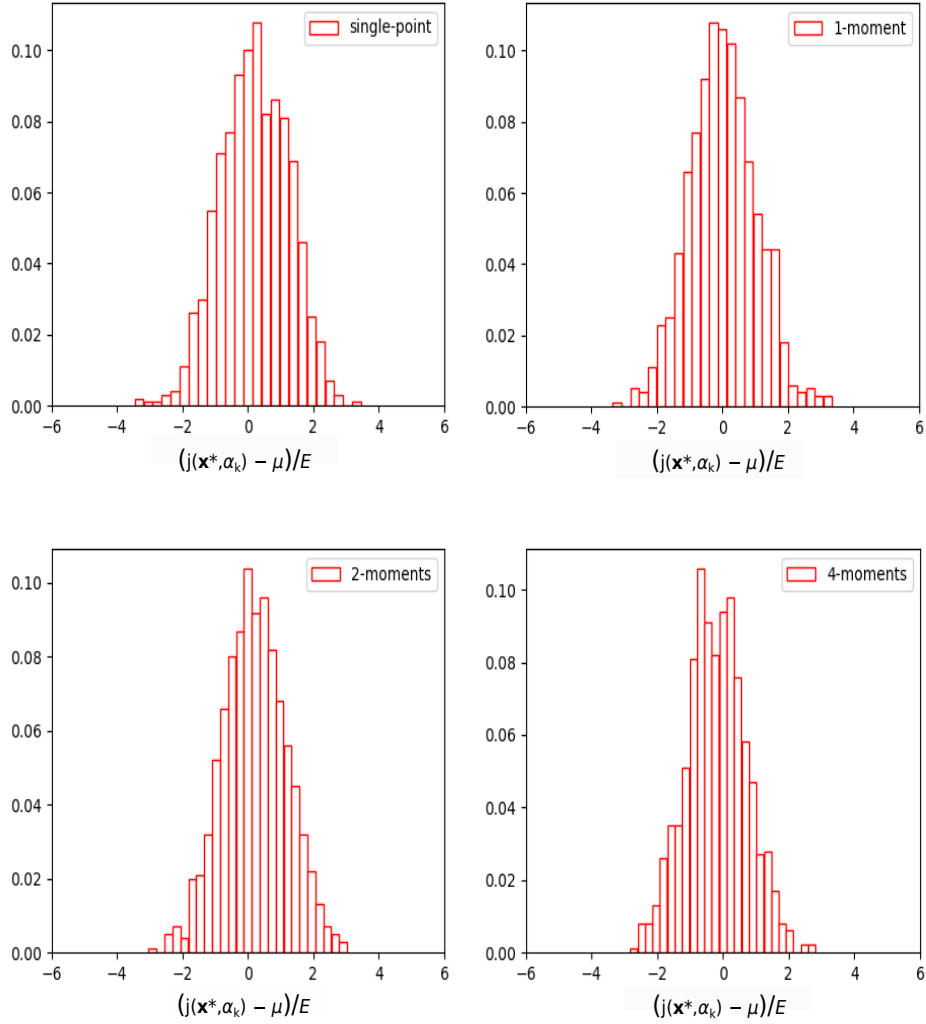


Figure 4. Relative frequency histogram of $(j(\mathbf{x}^*, \alpha_{k=1, \dots, M}) - \mu)/E$ with $M = 1000$. Controlling all four moments reduces the variability of the functional and therefore improves the robustness of the design.

4. Linear models and backward uncertainty propagation: covariance matrix of the optimization variables

We showed how to quantify the impact of target state uncertainties in inverse design through multi-point analysis and showed how it can be monitored through the moments of the functional. But the multi-point analysis requires a sampling of a large dimensional space. Something which is a burden when this dimension is large. To avoid a sampling we would like to see if an alternative low-complexity approach based on the existence of a local linear relationship between the target state u^* and the optimization variables \mathbf{x} can be applied to estimate the covariance matrix $Cov_{\mathbf{x}}^2$ of the optimization variables. We compare this construction to a reference $Cov_{\mathbf{x}}^1$ construction where a sampling is necessary.

As previously, we assume the PDF of the target solution u^* known and therefore its covariance matrix Cov_{u^*} . Also, we assume u^* is admissible and that there exists a set of optimization variables realizing the target.

As mentioned in (1), in our problem of interest the state equation is linear and to simplify the notation, and without any loss of generality, we consider the source term $f(\mathbf{x}) = \mathbf{x}$ considering that \mathbf{x} and u have a same dimension. In our implementation, however, we use spatial localization terms given in (2) in order to reduce the size of \mathbf{x} .

The construction can be applied in several useful situations. Let us describe two of them.

4.1. $Cov_{\mathbf{x}}^1$ when a linear model $Lu = \mathbf{x}$ is available

This is typically our case, for instance, with the elasticity equation and the optimization variables acting on the right-hand-side of the equation. In this case, we can express $Cov_{\mathbf{x}}^1$ with $u \sim u^*$, through:

$$\begin{aligned} Cov_{\mathbf{x}}^1 &= \mathbb{E} ((\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^\top) \\ &= L \mathbb{E} ((u^* - \bar{u}^*)(u^* - \bar{u}^*)^\top) L^\top \\ &= L Cov_{u^*} L^\top. \end{aligned}$$

But, in our case, the solver is a black-box and we do not have L in hand. In these cases, a linear model can be built through a machine learning approach using a database of simulation scenarios ($U - X$) generated by this code. This, actually, can also be done in a nonlinear situation. As the uncertainty analysis is aimed to take place around an optimal solution, it is enough for this surrogate model to have a local domain of validity. In this case, L is solution of the first-order optimality condition for a functional $J(L)$:

$$\nabla_L J(L) = 0 \Leftrightarrow UU^\top L^\top = UX^\top,$$

where U and X gather the scenarios of simulations available in the database for u and \mathbf{x} and we look for the best linear relation $X^\top = U^\top L^\top$ linking the scenarios in U to

those in X and, actually, vice versa. L^\top , and therefore L , can be found as solution of a least-squares problem:

$$J(L) = (X^\top - U^\top L^\top, X^\top - U^\top L^\top) = (X, X) - 2(XU^\top, L) + (L, L)_{UU^\top}.$$

The only requirement is for U to have full rank in order for UU^\top to be positive definite such that the scalar product $(L, L)_{UU^\top} \geq 0$. The formulation presented in section 2 presents, however, a way to avoid the previous development, taking advantage of the linearity to derive the expression of the form $\mathbf{x} = A^{-1}b(u^*)$. We recall that the components of A and b for $i, j = 1, \dots, N$ are given by:

$$A_{i,j} = \langle u_i, u_j \rangle \quad \text{and} \quad b_i(u^*) = \langle (u_0 - u^*), u_i \rangle.$$

Again, the covariance matrix of the optimization variables reads:

$$\begin{aligned} Cov_{\mathbf{x}}^1 &= \mathbb{E}((\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^\top) \\ &= A^{-1} \mathbb{E}((b(u^*) - b(\bar{u}^*))(b(u^*) - b(\bar{u}^*))^\top) A^{-\top} \\ &= A^{-1} Cov_{b(u^*)} A^{-\top}, \end{aligned} \tag{14}$$

where it is easy to estimate $Cov_{b(u^*)}$ knowing Cov_{u^*} . The elementary u_i and the homogeneous solutions are deterministic. The covariance matrix of $b(u^*)$ measures the covariance of the projection of the deviation between the uncertain target function u^* from the homogeneous solution u_0 on the elementary solutions u_i :

$$\begin{aligned} (Cov_{b(u^*)})_{ij} &= \mathbb{E}((b_i(u^*) - b_i(\bar{u}^*))(b_j(u^*) - b_j(\bar{u}^*))^\top) \\ &= \mathbb{E}(\langle (\bar{u}^* - u^*), u_i \rangle \langle (\bar{u}^* - u^*), u_j \rangle^\top). \end{aligned}$$

One interest of this formulation is that the size of this matrix is now five, the number of the optimization variables, instead of 14400, the number of the mesh grid points for Cov_{u^*} . Also, if the perturbation on u^* are centered, \bar{u}^* is the deterministic target solution.

4.2. $Cov_{\mathbf{x}}^2$ when a linear model $L\mathbf{x} = u$ is available

This is not directly the situation in this paper as the optimization variables intervene in the left-hand-side. However, this analysis is powerful when the state equation is linear, or if the Jacobian of the state with respect to the optimization can be estimated which is our situation where indeed the Jacobian is explicitly available knowing the elementary solutions as described in section (2).

After a deterministic optimization when the state is close to the target ($u \sim u^*$), we assume the covariance matrix of u close to the covariance matrix of u^* . Using the linear relationship $L\mathbf{x} = u$ we can write:

$$\begin{aligned} Cov_{u^*} &= \mathbb{E}((u^* - \bar{u}^*)(u^* - \bar{u}^*)^\top) \\ &= \mathbb{E}(L(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^\top L^\top) \\ &= L \mathbb{E}((\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^\top) L^\top \\ &= L Cov_{\mathbf{x}}^2 L^\top. \end{aligned}$$

Unlike the previous situation, to get $Cov_{\mathbf{x}}^2$ we need to invert this expression. In cases where exact inversion is impossible a least-squares formulation can be used looking for $Cov_{\mathbf{x}}^2$ minimizing:

$$\frac{1}{2} \langle L Cov_{\mathbf{x}}^2 L^\top, L Cov_{\mathbf{x}}^2 L^\top \rangle - \langle Cov_{u^*}, L Cov_{\mathbf{x}}^2 L^\top \rangle .$$

First-order optimality condition with respect to $Cov_{\mathbf{x}}$ gives:

$$L^\top L Cov_{\mathbf{x}}^2 L^\top L - L^\top Cov_{u^*} L = 0,$$

which leads to

$$Cov_{\mathbf{x}}^2 = (L^\top L)^{-1} L^\top Cov_u L (L^\top L)^{-1},$$

and eventually, to

$$Cov_{\mathbf{x}}^2 = L^{-1} Cov_{u^*} L^{-\top} = (L^\top Cov_{u^*}^{-1} L)^{-1}. \quad (15)$$

Therefore, knowing Cov_{u^*} one can access $Cov_{\mathbf{x}}^2$ if a linear relationship, even locally valid, is available between \mathbf{x} and u . In practice, Cov_{u^*} is diagonal as the target uncertainties are often assumed independent. The inversion of Cov_{u^*} is therefore straightforward. The second expression only requires the inversion of a $N \times N$ matrix in our case. This expression in (15) is also interesting because it involves the inversion of a square matrix giving a least-squares sense to the inversion of rectangular matrix. An interesting application of this construction is using the state sensitivity with respect to optimization variables or the Jacobian $\mathcal{J} = (\partial u / \partial \mathbf{x})$ in a first-order linear relationship $\mathcal{J} \delta \mathbf{x} = \delta u$. This permits to estimate the covariance matrix of the optimization variables perturbations even in the absence of a direct linear model linking \mathbf{x} and u . Actually, in situations where the state equation is linear (our case) we have $L = \mathcal{J}$. In situations where there are only a few optimization variables, the Jacobian can simply be estimated by a finite differences approach approximating the i^{th} line of the Jacobian by:

$$\mathcal{J}_{i=1, \dots, N} \approx \frac{(u(\mathbf{x}e_i + \varepsilon) - u(\mathbf{x}e_i - \varepsilon))}{2\varepsilon},$$

with $e_{i=1, \dots, N}$ the canonical basis of \mathbb{R}^N in our case and $0 < \varepsilon \ll 1$. But, we have seen in section 2 that u has a direct expression with respect to \mathbf{x} components x_i through:

$$u = u_0 + \sum_{i=1}^N x_i u_i.$$

Therefore, the finite difference formula reduces to:

$$\mathcal{J}_{i=1, \dots, N} \sim \frac{(u(\mathbf{x}e_i + \varepsilon) - u(\mathbf{x}e_i - \varepsilon))}{2\varepsilon} = u_i.$$

This is another advantage of our direct formulation for the inversion as it makes evaluating \mathcal{J} straightforward, not requiring any extra calculation. In addition, the Jacobian is exact in this case and is not affected by the finite differences approximation and the choice of ε . Actually, the Jacobian would be exact even using first order finite differences. This is because we recover the exact gradient expression in this case. Therefore, this second covariance estimation is well adapted to our situation and would be interesting if it gives

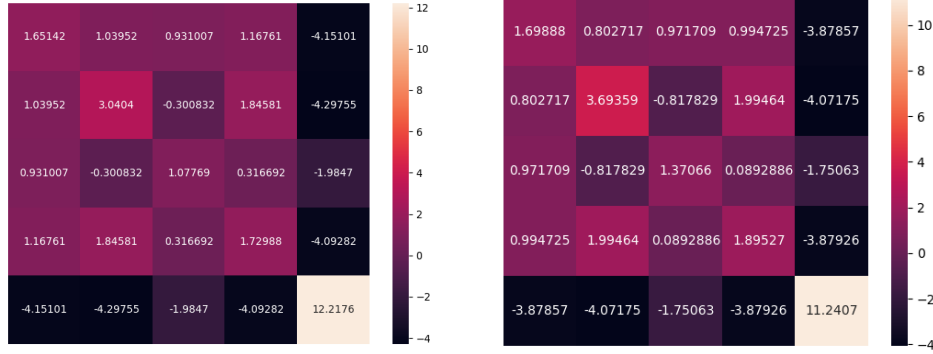


Figure 5. Covariance Matrix Cov_x^1 and Cov_x^2 (in $Newton^2$) Visualization, left: Cov_x^1 from (14), right: Cov_x^2 from (15).

nearly the same answer than the first construction where a sampling of the state variable variation domain is necessary.

Figure 5 shows the comparison of this two estimations of Cov_x . We see that the matrices are very close and especially the signs of all components agree.

Further comparisons are given in Table 3 through different distances between the two covariance matrices versus the distance between the covariance matrices and a diagonal matrix with the diagonal given by mean values of the covariance matrices components (i.e. $\frac{1}{2N^2} \sum_{i,j=1}^N (Cov_x^1 + Cov_x^2)_{i,j}$). This is to provide an estimation of how close the two covariance matrices are versus a third fixed one.

Distance d	$d(Cov_x^1, Cov_x^2)$	$d(Cov_x^1, CI_N)$	$d(Cov_x^2, CI_N)$
Kullback distance	0.334	165.589	166.682
Euclidean distance	1.674	17.020	16.123
Log Euclidean distance	0.737	7.193	7.469
Riemannian distance	0.811	7.193	7.469
Log-det distance	0.286	2.00	2.086
Wasserstein distance	0.389	4.098	4.097

Table 3. Different estimations of distances between the two covariance matrices. where $C = \frac{1}{2N^2} \sum_{i,j=1}^N (Cov_x^1 + Cov_x^2)_{i,j}$ is the mean value of the covariance matrices components and I_N is the N^{th} identity matrix.

5. Concluding remarks

Robust parametric optimization has been addressed through the control of the moments of the functional. Moments have been estimated using a multi-point formulation. This permits to address situations where operating parameters are not anymore single valued but defined through their PDF. Hence, the first four moments of the functional have been considered to go beyond the classical mean-variance based optimization. Subtleties of what should be the target for the skewness and the kurtosis of the probability density function of the performance of the system for a robust design have been discussed.

It has been shown that controlling second, third and fourth moments drastically improves the design while the time to the solution remains comparable to the single-point situation as all the extra calculations can be carried out in a fully parallel and independent manner. The cost is also reduced thanks to our construction of the optimal solution taking advantage of the presence of the linearity between the optimization variables and the solution of the state equation. As a consequence, the approach requires quite small additional coding and computational effort.

Low-complexity solutions for backward propagation of aleatory uncertain target data has also been presented. Derivation of the covariance matrix of the optimization parameters has been discussed through two formulations. These provide uncertainty quantification analysis for the inversion solution with confidence margins on the design parameters in very large design spaces. It has been shown that taking again advantage of existing linear relationships between the target state u^* and the optimization variables \mathbf{x} , samplings of the design and the target solution spaces can be avoided drastically reducing the computational cost of the approach.

6. References

- [1] AIAA. Guide for the verification and validation of computational fluid dynamics simulations. *AIAA*, G-077, 1998.
- [2] H.-J. Bungartz and M. Griebel. Sparse grids. *Acta Numerica*, 13:147–269, 2004.
- [3] J. Caers. *Modeling Uncertainty in the Earth Sciences*. Wiley-Blackwell, 2011.
- [4] G. Casella and R. Berger. *Statistical Inference, 2nd Ed.* Duxbury Press, London, 2001.
- [5] Le Fichoux E. Présentation et utilisation de cast3m. *ENSTA-LME*, 2011.
- [6] C. Geuzaine and J.-F. Remacle. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 2009.
- [7] 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.
- [8] R. Ghanem and P. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Springer Verlag, New York, 1991.
- [9] K. Ide P. Courtier M. Ghil and A. Lorenc. Unified notation for data assimilation: operational, sequential and variational. *Journal of the Meteorological Society of Japan*, 75/1B:181–189, 1997.
- [10] P.G. Hoel. *Introduction to Mathematical Statistics*. John Wiley, London, 1971.
- [11] R.E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering*, 82:35–45, 1960.
- [12] K.C. Kapur and L.R. Lamberson. *Reliability in Engineering Design*. John Wiley & Sons, New York, 1997.
- [13] H.R. Lindman. *Analysis of Variance in Complex Experimental Designs*. Freeman, New York, 1974.
- [14] B. Mohammadi. Reduced sampling and incomplete sensitivity for low-complexity robust parametric optimization. *Int. J. Num. Meth. Fluids*, 73/4:307–323, 2013.
- [15] B. Mohammadi. Principal angles between subspaces and reduced order modeling accuracy in optimization. *Structural and Multidisciplinary Optimization*, 50/2:237–252, 2014.
- [16] B. Mohammadi. Uncertainty quantification by geometric characterization of sensitivity spaces. *Compt. Meth. Appl. Mech. Eng.*, 280:197–221, 2014.
- [17] B. Mohammadi. Value at risk for confidence level quantifications in robust engineering optimization. *Optimal Control: Applications and Methods*, 35/2:179–190, 2014.
- [18] 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.
- [19] J. T. Spooner M. Maggiore R. O. Onez and K.M. Passino. *Stable Adaptive Control and Estimation for Nonlinear Systems: Neural and Fuzzy Approximator Techniques*. John Wiley, New York, 2002.
- [20] J. Chery B. Mohammadi; M. Peyret and C. Joulain. Plate rigidity inversion in southern california using interseismic gps velocity field. *Geophys. J. Int.*, 187/2:783–796, 2011.
- [21] A. Conn K. Scheinberg and L. Vicente. *Introduction to Derivative-Free Optimization*. SIAM, NY, 2002.

- [22] C. Schillings S. Schmidt and V. Schulz. Efficient shape optimization for certain and uncertain aerodynamic design. *Computers & Fluids*, 46(1):78–87, 2011.
- [23] M. Schnellenbach-Held and K. Pfeffer. Punching behavior of biaxial hollow slabs. *Elsevier Science*, 2002.
- [24] A. Tarantola. *Inverse problem theory and methods for model parameter estimation*. SIAM, N.Y., 1987.
- [25] X. Wan and G.E. Karniadakis. Multi-element generalized polynomial chaos for arbitrary probability measures. *SIAM J. Sci. Comput.*, 28/3:901–928, 2006.
- [26] D. Xiu. *Numerical Methods for Stochastic Computations: A Spectral Method Approach*. Princeton University Press, 2010.