ECCOMAS Congress 2016
VII European Congress on Computational Methods in Applied Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.)
Crete Island, Greece, 5–10 June 2016

ADJOINT-BASED ROBUST OPTIMIZATION USING POLYNOMIAL CHAOS EXPANSIONS

Joao Miranda, Dinesh Kumar, and Chris Lacor

Vrije Universiteit Brussel (VUB), Department of Mechanical Engineering, Fluid Mechanics and Thermodynamics Research Group
Pleinlaan 2, 1050 Brussels, Belgium
e-mail: Joao.Miranda@vub.ac.be, Dinesh.Kumar@vub.ac.be, Chris.Lacor@vub.ac.be

Keywords:

Abstract. *Adjoint methods are nowadays widely used to efficiently perform optimization for problems with a large number of design variables. However, in reality, the problem at hand might be subjected to uncertainties in the operational conditions or, in case of optimizing geometries, the design variables itself might be uncertain due to manufacturing tolerances. For such applications, the optimum obtained using deterministic methods might be very sensitive to small variations in the uncertainties, i.e. it lacks robustness. In a robust optimization, the uncertainties are taken directly into account during the optimization process by introducing, next to the mean objective, its variance as a second objective. This implies that, when using gradient based optimization methods, the gradients of both objectives (mean and variance) must be known. In this work the Polynomial Chaos Expansion (PCE) is used in combination with adjoint methods to efficiently obtain both gradients. A non intrusive, regression based PCE is used, requiring a new adjoint solution for each sampling point in order to build the PCE of the gradient. A PCE for the objective is also built (at no extra cost) in order to compute the gradient of the variance.*

A weighted average of both gradients is then used to find an optimum. By changing the weighting factors the solution can be found favouring either of the two objectives. The developed approach is applied to relevant engineering problems, such as geometrical optimization of pipe flows and flow over airfoils. The design variables are the shape coordinates and no parameterization is used. In this work, the design variables were considered deterministic with the uncertainties coming from operational conditions.

1 INTRODUCTION

With the increase in available computational power, CFD has evolved from an analysis tool towards a design tool, see [1]. In the design process, CFD simulations are used in order to find the value of the objective(s) for a given set of design variables. In order to find the best design configuration, an optimization algorithm is then used to drive the design variables to their optimal value, minimizing the objective(s), or, in case of conflicting objectives, leading to a Pareto front.

The optimization techniques can be classified as gradient free or gradient based. Gradient based approaches make use of the gradient (with respect to the design variables) of the objective function, which provides the optimal search direction in the design space, see [2] and [3]. The gradient free methods, essentially sample the design space and try to find the optimum solely based on the value of the objective(s), see [4].

Depending on the parameterization, optimization problems often have a large number of design variables, see [5] and [6], especially in the case of shape optimization. This makes traditional techniques for gradient calculation, such as finite differences, which require an extra flow simulation for every design variable, too expensive. An adjoint method offers a cheap alternative as it allows to calculate the gradient, at roughly the cost of an extra CFD simulation, independently of the number of design variables, see [7]. Adjoint solvers can be classified as discrete or continuous, depending on how the adjoint equations are derived, see [8] and [9]. Both approaches have their positive and negative aspects and the choice of one or the other method is usually related to a personal preference.

While advances in computational capabilities and numerical techniques lead to a widespread use of optimization algorithms to design engineering components, these are mainly employed on deterministic problems. This means that all the design conditions are well defined, immutable and are assumed to be an exact representation of the reality. However, in real world applications this is seldom the case as the operational conditions may significantly differ from the design conditions. Moreover, due to manufacturing tolerances, the finished component may be slightly different than the optimal design. Due to the high number of uncertainties in the manufacture and operating life of any engineering component, a deterministic optimization approach may result in an optimum that is very sensitive to all those small uncertainties, leading to a lower performance than desired. Consequently, an optimization methodology that takes into account the robustness of the design is of special interest for real world engineering applications.

In a robust optimization setting the objective is to reduce both the value of the mean objective and its sensitivity to operating and manufacture uncertainties. To this end, when using a gradient based optimization method, the gradients of both mean objective and of its variance must be known. These gradients are then combined, using weighted averaging, to find the best solution for each problem. In the case of conflicting objectives, e.g. the design corresponding to the minimum objective is not robust (has high variance), a Pareto front, or part of it, can be built by varying the weights given to each gradient. In such a case is the design engineer responsibility to chose a suitable compromise.

The success of the robust optimization algorithms is intrinsically connected to the performance of the chosen uncertainty quantification (UQ) method. In intrusive PC approaches, the PC expansion is directly embedded in the code leading to a less expensive methodology than the non-intrusive approaches. However, their implementation is cumbersome and prone to errors and involves deep knowledge of the CFD code. An example of an intrusive approach can be

found in [10]. On the other hand, non-intrusive methods such as Monte-Carlo and Non-Intrusive Polynomial Chaos (NIPC) have the need for an higher number of function evaluations, but the implementation is straightforward and independent of the problem being studied, see [11]. However, the number of required samples will increase exponentially with the number of uncertainties and the order of the polynomials. This is known as the curse of dimensionality.

In order to calculate the gradient of the mean objective and of its variance, PCEs of the objective and of its gradient are required. To this end, both the primal and adjoint solutions are needed in each sample point. As the samples are independent of each other they can be evaluated in parallel reducing the overall simulation time.

In order to compute the gradient of the variance, a PCE is also built for the objective. This is done at no extra cost since the objective samples are calculated form the primal solution at each sample.

In this work the uncertainties are restricted to finite physical quantities. A uniform distribution for the stochastic variables is chosen. Consequently the polynomials, used for building the PCE expansion, are the Legendre polynomials, see [12]. In order to find the polynomial coefficients an overdetermined system is solved using regression.

2 GRADIENT BASED ROBUST OPTIMIZATION

A general minimization problem can be written as:

$$\begin{aligned} &\text{minimize} \quad J(U,\alpha) \\ &\text{subject } to \quad R(U,\alpha) = 0 \end{aligned} \tag{1}$$

where J is a cost function to be minimized, R is the set of governing equations, U the state variables of the primal problem and α the design variables. In a gradient based optimization, the gradient of the cost function with respect to the design variables, i.e. $\frac{dJ}{d\alpha}$, has to be calculated.

$$\frac{dJ}{d\alpha} = \frac{\partial J}{\partial \alpha} + \frac{\partial J}{\partial U}\frac{\partial U}{\partial \alpha} \tag{2}$$

Where the term $\partial U/\partial \alpha$ depends on the primal solution, and so it is expensive to find. Taking into account that

$$\frac{dN}{d\alpha} = \frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha} + \frac{\partial R}{\partial \alpha} = 0 \tag{3}$$

eq. 2 becomes:

$$\frac{dJ}{d\alpha} = \frac{\partial J}{\partial \alpha} - \frac{\partial J}{\partial U}\left(\frac{\partial R}{\partial U}\right)^{-1}\frac{\partial R}{\partial \alpha} \tag{4}$$

Introducing λ, the vector of the adjoint variables:

$$\left(\frac{\partial R}{\partial U}\right)^{T}\lambda = \left(\frac{\partial J}{\partial U}\right)^{T} \tag{5}$$

eq. 4 can be rewritten as:

$$G = \frac{dJ}{d\alpha} = \frac{\partial J}{\partial \alpha} - \lambda^{T}\frac{\partial R}{\partial \alpha} \tag{6}$$

Eq. 6 does not depend on the primal solution. Hence the calculation of the gradient with respect to any design variable α is reduced to the calculation of simple partial derivatives.

2.1 Stochastic Adjoint Based Gradient

The stochastic gradient $G(\xi)$ can be written as PCE of order p for n random non-design variables $\xi \equiv \{\xi_i\}_{i=1}^{n}$ as:

$$G(\xi) = \sum_{i=0}^{P} \hat{G}_i \psi_i(\xi) \tag{7}$$

where the total number of terms on the expansion is given by $P+1 = \frac{(p+n)!}{p!n!}$ and G_i represents the i$^{\text{th}}$ random mode of $G(\xi)$.

In order to find the components of the PCE (G_i), a regression approach is chosen. An overdetermined system is built:

$$\begin{pmatrix} \psi_0(\xi^1) & \psi_1(\xi^1) & \cdots & \psi_P(\xi^1) \\ \psi_0(\xi^2) & \psi_1(\xi^2) & \cdots & \psi_P(\xi^2) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_0(\xi^{n_s}) & \psi_1(\xi^{n_s}) & \cdots & \psi_P(\xi^{n_s}) \end{pmatrix} \begin{pmatrix} \hat{G}_0 \\ \hat{G}_1) \\ \vdots \\ \hat{G}_P \end{pmatrix} = \begin{pmatrix} G(\xi^1) \\ G(\xi^2) \\ \vdots \\ G(\xi^{n_s}) \end{pmatrix} \tag{8}$$

where $G(\xi^1), ...G(\xi^{n_s})$ are the gradients calculated at each sample, $\hat{G}_0, ...\hat{G}_P$ are the components of the polynomial of order p and $\xi^0, ...\xi^{n_s}$ are the n_s samples mapped in the chosen polynomial domain.

When using regression to find the PCE components, the advised number of samples is two times the total number of terms in the PC expansion, $n_s = 2 * (P+1)$, [13].

2.2 Gradient of the Variance

Although both the mean ($\hat{G}_0$) and variance (σ_G) of the gradient can directly be calculated from eq.(7), in order to do robust optimization the gradient of the variance of the objective ($\nabla\sigma_J$) is also required.

From Eq. 6 and eq.(7) it comes directly:

$$G(\xi) = \frac{dJ}{d\alpha}(\xi) = \sum_{i=0}^{P} \widehat{G_i} \psi_i(\xi) \tag{9}$$

On the other hand, a PCE of the objective functional gives:

$$J(\xi) = \sum_{i=0}^{P} \widehat{J}_i \psi_i(\xi) \tag{10}$$

Taking the gradient one can write:

$$\frac{dJ(\xi)}{d\alpha} = \sum_{i=0}^{P} \frac{d\widehat{J}_i}{d\alpha} \psi_i(\xi) \tag{11}$$

Comparison with equation 9 gives:

$$\widehat{G_i} = \frac{d\widehat{J}_i}{d\alpha} \tag{12}$$

The first term in the PCE of eq.(10), $\widehat{J_0}$, is the mean objective. According to eq.(12), the gradient of the mean objective, follows from the PCE of the gradient:

$$\hat{G}_0 = \frac{d\widehat{J_0}}{d\alpha}. \tag{13}$$

Furthermore:

$$\hat{G}_i = \frac{d\widehat{J_i}}{d\alpha} \tag{14}$$

For uniform distribution the variance of the objective can be written as:

$$\sigma_J^2 = \frac{\hat{J}_1^2}{3} + \frac{\hat{J}_2^2}{5} + \cdots \tag{15}$$

Taking the gradient:

$$\nabla\sigma_J^2 = \frac{d}{d\alpha}\left(\frac{\hat{J}_1^2}{3} + \frac{\hat{J}_2^2}{5} + ...\right) = \left(\frac{1}{3}\frac{d\hat{J}_1}{d\alpha}\hat{J}_1 + \frac{1}{5}\frac{d\hat{J}_2}{d\alpha}\hat{J}_2 + ...\right) = 2\left(\frac{1}{3}\hat{G}_1\hat{J}_1 + \frac{1}{5}\hat{G}_2\hat{J}_2 + ...\right) \tag{16}$$

As an example, for a second order polynomial one obtains:

$$\nabla\sigma_J^2 = 2\left(\frac{1}{3}\hat{G}_1\hat{J}_1 + \frac{1}{5}\hat{G}_2\hat{J}_2\right) \tag{17}$$

2.3 Optimization

For the optimization process, a weighted combination of the two gradients is used as:

$$G_t = \omega\hat{G}_0 + (1-\omega)\nabla\sigma_J^2 \tag{18}$$

Where, G_t is the effective gradient to be used during the optimization and ω is a weighting coefficient to be chosen by the user.

$$\alpha^{n+1} = \alpha^n - \beta * \frac{dJ}{d\alpha^n} \tag{19}$$

where α is a vector containing the design variables and α^{n+1}, α^n are respectively the updated and the original values. β is a scalar that is chosen iteratively, i.e. it can change at each iteration, in order to guarantee a smooth optimization process.

As no parameterization is used, every mesh point on the surfaces to optimize is a design variable. This gives a lot of freedom in the shape modifications which can lead to unrealistic oscillatory shapes. An implicit smoothing algorithm is employed to make sure the optimized surface remains smooth. In order to maintain the quality of the mesh in the laminar test cases the mesh is smoothed by solving a Laplacian equation, see [14]. For the turbulent test cases, in order to maintain a proper boundary layer resolution, the problem is remeshed.

A optimization problem with deterministic design variables and using a steepest descent optimization is described in algorithm 1.

Algorithm 1 Non-Design Stochastic Variables

Set tol, $U_{initial}$
while $\langle G_0 \rangle > tol$ **do**
    {
    Sample $\xi^1, \xi^2 ... \xi^{n_s}$
    for $i = 1 \rightarrow n_s$ **do**
        {
        Solve primal problem, $U(\alpha, \xi^i)$
        Compute Objective $J(\alpha, \xi^i)$
        Solve adjoint problem, $\lambda(\alpha, \xi^i)$
        Compute gradient $G(\alpha, \xi^i))$
        }
    end for
    Use regression to find $\hat{G}_0, ..., \hat{G}_P$ and $\hat{J}_0, ..., \hat{J}_P$
    Compute gradient of the variance $\nabla \sigma_J^2$
    Compute G_t using Eq. 18
    $\alpha = \alpha - \beta G_t$
    }
end while
End

3 NUMERICAL IMPLEMENTATION

Both the primal and adjoint problems are solved using the the **SIMPLE** algorithm in OpenFOAM. The steady state solution is achieved using an Euler implicit time scheme. Bounded central schemes are used for the numerical discretization of both the primal and the adjoint problems, see [15]. In the turbulent test cases, the turbulence in the primal problem is modeled using a RANS approach. The Spalart-allmaras model is chosen and the boundary layer is modeled by means of wall functions. For the adjoint solver, a frozen turbulence approach is used.

The sampling needed for the Latin Hypercube scheme, ensuring a good spread of the samples in the stochastic space. The random number generator is connected to the time and date of the machine in order to guarantee randomness. The samples are all obtained in $[0, 1]^n$ and then mapped into the stochastic variable distribution. For a uniform distribution $U[x_a, x_b]$, the mapping is simply a projection, $[0, 1] \rightarrow [x_a, x_b]$. The PCE terms are found by solving the overdetermined problem by using a regression approach.

4 TEST CASES

4.1 2D U-bend

A common fluid mechanics problem found in engineering applications is the minimization of total pressure loss in flows through piping systems.

For an arbitrary pipe shape, the total pressure loss (the objective to minimize) can be written as:

$$J = \int_{\partial\Omega} P_t d\dot{m} \tag{20}$$

where, P_t is the total pressure, $\partial\Omega$ the domain's boundary, and d$\dot{m}$ the elementary mass flux

through the boundary.

In this test case we consider the robust optimization of a U-bend under operating uncertainties. This type of bend is commonly found as part of a larger serpentine in applications such as heat exchangers. For serpentines with a high number of U-bends, the accumulated pressure drop through all the U-bends may severely impair the total efficiency of the system. As a result it is crucial to the reduce the pressure drop at each bend.

In a robust optimization setting the problem can be written as:

$$minimize : \mu(J), \sigma(J) \tag{21}$$

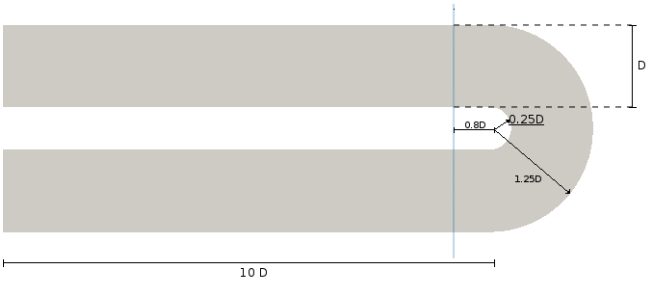


Figure 1: U-bend pipe.

The initial shape and dimensions of the U-bend are shown in Figure 1. For this test case $D = 0.075m$ and the deterministic Reynolds number is $Re = 400$. The laminar Navier-Stokes equations are therefore considered for the primal and the adjoint. In this test case the optimization is restricted to the outer wall of the U-Bend. The inlet velocity is considered as uncertain with a uniform distribution $U_{inlet} = [0.066, 0.084]m/s$. A 2^{nd} order polynomial chaos method is used in J, requiring six samples at each optimization step to get the PCEs of the objective and its gradient.

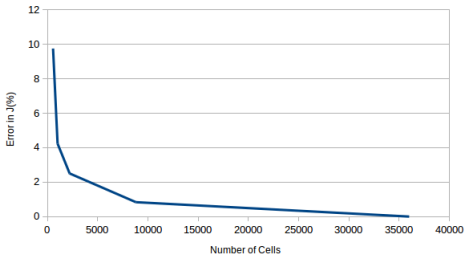


Figure 2: Mesh independence test.

Figure 2 shows the error in the calculated total pressure drop on meshes of different size. The solution on a very fine mesh of 36000 cells is considered as a reference solution. It is observed that for a mesh with ≈ 9000 cells, the error on the pressure drop is less than 1%. This mesh is therefore considered as a realistic mesh for the problem at hand, suitable for use in the optimization process.

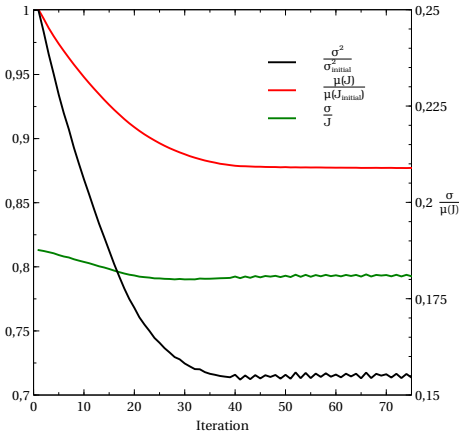


Figure 3: $\mu(J)$ and σ^2 evolution during optimization using $\omega = 1$.

In Figure 3 the evolution of the mean objective and the variance during the optimization are shown for $\omega = 1$. Both the variance and the mean objective are nondimensionalized by their values at the beginning of the optimization process ($\mu(J_{initial})$, $\sigma^2_{initial}$). It can be observed that the optimization of the mean objective also leads to a lower variance, hence the optimal solution (lowest pressure drop) is additionally a robust one. This conclusion is further supported by observing that $\frac{\sigma}{J}$ is almost constant through the optimization process.

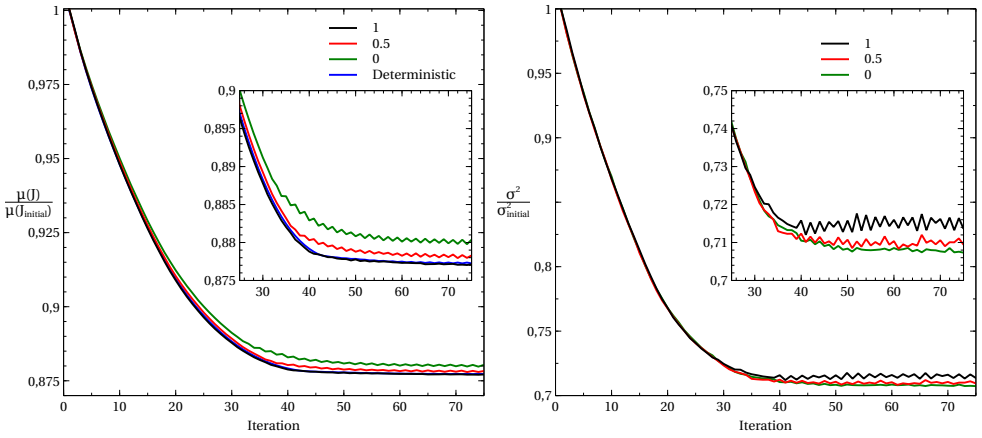


Figure 4: Deterministic and Robust optimization using $\omega = 1,\ 0.5$ and 0.

Figure 4 shows the evolution of the variance and the mean objective for different values of ω. Taking into account that the optimal is a robust one, optimizing the mean objective, $\omega = 1$, or the variance, $\omega = 0$, will lead to very similar results. However, it can be observed that there are small differences between the different cases. For $\omega = 1$, the optimal mean objective reaches its lowest value and the variance is the highest. On the other hand, for $\omega = 0$, the variance reaches its lowest value whereas the objective is at its highest. Furthermore, for $\omega = 1$ the mean objective presents a smooth optimization but the variance is oscillating, which indicates that a minimum has not been reached for the latest. The opposite happens for $\omega = 0$.

It can be observed that for an intermediate value of $\omega = 0.5$, both the mean objective and the variance take intermediate values. This shows that combining both gradients can be used to obtain a mix between optimal and robust results.

Table 1: Deterministic operational conditions.

U	$102m/s$
Re	6×10^6
AoA	1°

Table 2: Lift and Drag coefficients for NACA 0012 at AoA = 1° and Re 6×10^6.

	Cl	Cd	L/D
Numerical	0.109	0.0089	12.25
Reference NASA-SA	0.110	0.0085	12.94
error [%]	0.9	4.7	5.3

Finally it is important to notice that the optimization of the mean objective, $\omega = 1$, shows an almost perfect correspondence with the deterministic optimization.

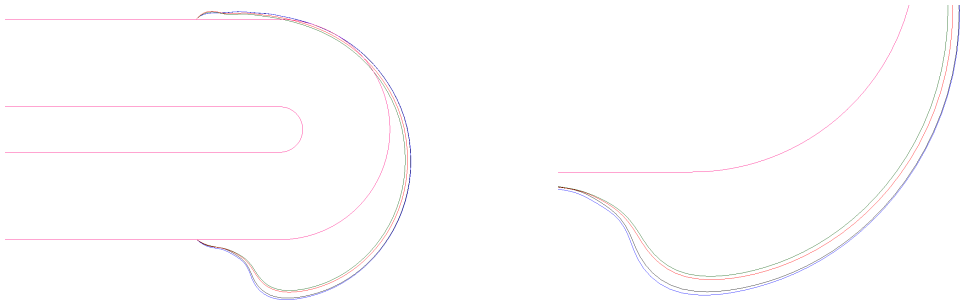


Figure 5: Initial(Pink) and optimal shapes. Deterministic(Blue), $\omega = 1$(Black), 0.5(Red) and 0(Green) .

Figure 5 depicts the optimal shapes obtained with the different approaches. As expected from the previous results analysis, all the obtained optimal shapes are similar. In all cases, there is an asymmetrical swelling of the U-bend more pronounced on the side where the flow exits the U-bend. Furthermore, the optimal shape obtained using $\omega = 1$ is the closest to the one obtained in the deterministic problem whereas the shape obtained using $\omega = 0$ shows the largest differences.

4.2 2D Airfoil

In this section the robust optimization of a subsonic airfoil with uncertainties in the operating conditions is considered. The deterministic objective is to maximize the Lift-to-Drag ratio (L/D). The robust optimization problem is the defined as:

$$minimize : \mu(-L/D), \sigma(-L/D) \tag{22}$$

The initial airfoil is a NACA 0012 and the deterministic operational conditions are given in Table 1.

The chosen turbulence model is the Spalart-Almaras with wall functions. A C-type mesh of 80000 cells is used. The numerical results are compared to the reference results of [16] in Table 2. A good comparison is observed confirming that the size and the quality of the mesh is appropriate.

Table 3: Random variable distributions.

Random Variable	Probability distribution
AoA °	Uniform [0.5,1.5]
U $[m/s]$	Uniform [98.84,105.2]

The random variables for this test case are the freestream velocity and the angle of attack with uniform distributions as listed in Table 3.

The number of samples required to find the PCE coefficients dramatically increases with the order of the PCE expansion. As each sample involves a primal and an adjoint CFD solutions, the methodology becomes prohibitively costly for high order PCEs.

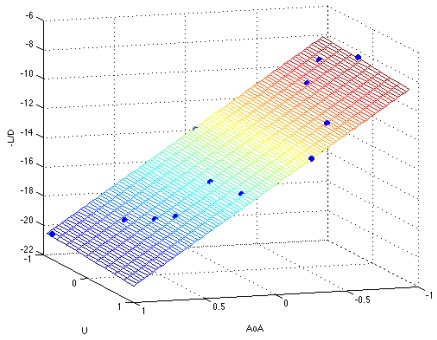


Figure 6: Response surface for $-L/D$.

Figure 6 shows the response surface for $-L/D$ in relation to the stochastic variables (mapped to [-1 1]). It can be observed that all the samples lie in a plane. This indicates that a first order PCE is sufficient to accurately capture the response.

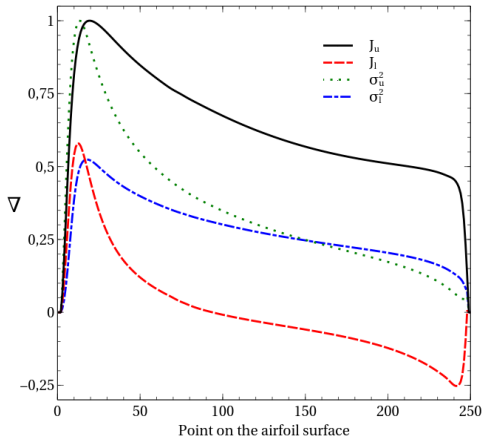


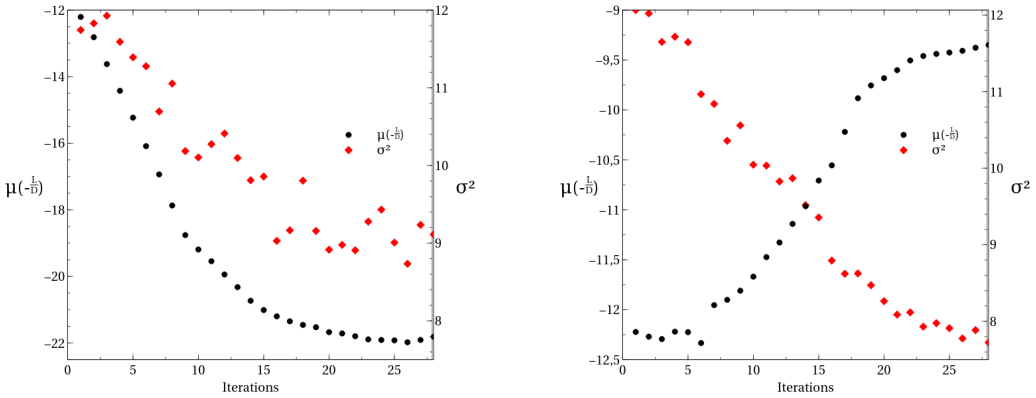
Figure 7: Sensitivities of the mean objective ($\mu(J)$) and variance (σ^2). u and l refer respectively to the upper and lower surfaces of the airfoil.

In Figure 7 the initial surface sensitivities for the mean objective and variance are shown. It can be noted that for the upper surface both sensitivities are positive corresponding to an outward movement of the nodes. On the lower surface, the variance sensitivities are also positive but the mean objective sensitivities change sign, meaning that towards the trailing edge the

Table 4: Optimal Cl and Cd at AoA = 1°.

	Cl	Cd	$\mu(L/D)$	σ^2	$\mu(L/D)-\sigma$	$\mu(L/D)+\sigma$
Initial	0.109	0.0089	12.25	12	8.79	15.71
$\omega = 1$	0.204	0.0094	21.17	9.19	18.14	24.20
$\omega = 0$	0.099	0.0109	9.34	7.72	6.59	12.12

nodes will move inwards leading to slightly cambered airfoil. It is also to be mentioned that the leading and trailing edges are fixed.



(a) $\omega = 1$ (b) $\omega = 0$

Figure 8: Evolution of $\mu(-L/D)$ and σ^2 during the optimization.

The evolution of the variance and the mean objective during the optimization using $\omega = 1$ and $\omega = 0$ are shown on Figure 8 and the final results are summarized in Table 4. For $\omega = 1$ the evolution of the $\mu(J)$ and σ^2 are similar, indicating that the airfoil corresponding to the optimal mean value is also more robust than the initial airfoil. On the other hand, when $\omega = 1$, $\mu(J)$ and σ^2 have opposite behaviors indicating that a very robust airfoil will have a poor mean performance. It can also be observed that although the variance is reduced in both cases, the lowest value is obtained when the gradient of the variance is used. Moreover, the evolution of the variance during optimization with $\omega = 0$ is much smoother than when optimizing with $\omega = 1$. Finally, it is interesting to notice that even in a pessimistic scenario, $L/D = \mu(L/D) - \sigma$ on the airfoil obtained when optimizing the mean of the objective, is still better than the L/D of original airfoil or $\mu(L/D)$ of the airfoil obtained using $\omega = 0$.

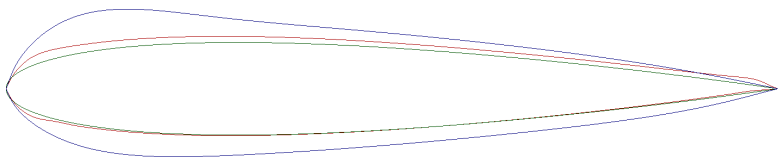
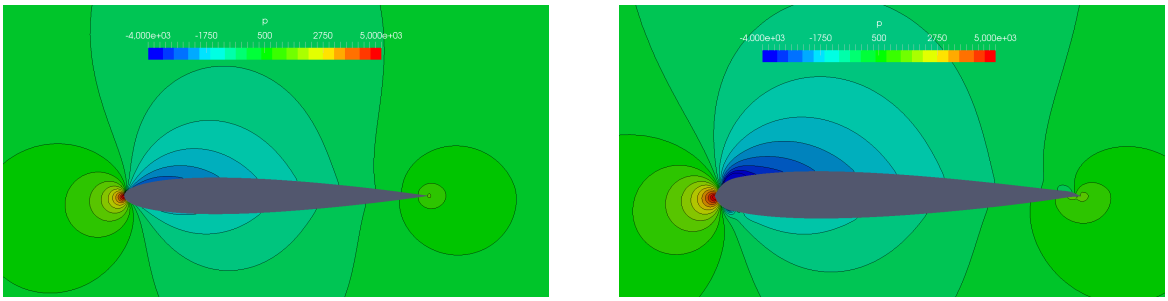


Figure 9: Initial(Green) and optimal shapes: $\omega = 1$(Red) and 0(Blue) .

The optimal airfoil shapes are shown in Figure 9. For the airfoil obtained using $\omega = 1$, the most noticeable differences can be observed on the upper surface and the leading edge.

The increase in thickness on the leading edge results in a higher acceleration of the flow over the airfoil creating a region of low pressure and increasing lift. Moreover, the upper surface experiences a much larger deformation than the lower surface, thus, the resulting airfoil will have a small curvature which additionally contributes to the increase of lift.

On the other hand, for the case with $\omega = 0$, the optimal airfoil is overall much thicker than the initial configuration. This explains the increase in drag and suggests a lower sensitivity to the angle of attack explaining the smaller variance. It is important notice that no constraints where imposed in the optimization.



(a) (b)

Figure 10: Initial (a) and final (b) pressure distribution for $\omega = 1$.

Figure 10 depicts the pressure contours for the initial and optimal airfoil using $\omega = 1$. It can be observed that the low pressure region in upper surface on the optimized airfoil is much more intense than on the initial one. This explains the large increase in the lift on the optimized airfoil when compared to the initial configuration.

As a last comment, given the relativity small change of the airfoil shape, especially if $\omega = 1$, it seems that the current solution is only a local optimum. Other shapes might be found starting from another initial airfoil. In order to retrieve the Pareto front it seems advisable to combine the gradient based approach with an evolutionary algorithm.

5 CONCLUSIONS

A robust gradient based shape optimization using PCE has been developed. The gradient used in the steepest descent is a weighted average of the gradient of the mean objective and the gradient of its variance. by changing the weights, solutions favoring either mean or variance can be found.

Two test cases are presented.

In the first test case, a U-bend, the optimal solution obtained using the gradient of mean objective was found to be also a very robust one. Therefore, and as expected, the optimization of the variance led to a similar optimal solution. The small differences found on the optimal solutions agreed with the chosen value for ω.

In the second test case, a 2D airfoil under turbulent subsonic flow, the optimization of the mean objective and the variance led to two very distinct configurations with somewhat conflicting objectives. The observed changes in the airfoil shape are rather small, especially if a high weight was given to to optimizing the mean objective. This suggests a local optimum, and a dependency on the initial airfoil configuration. In order to retrieve a global optimum and the Pareto front, a combination with evolutionary algorithms seems indicated. This will be the subject of future research.

6 ACKNOWLEDGEMENTS

This research was partly funded by the Flemish Science Foundation FWO under a grant G0B7113N. This support is gratefully acknowledged.

REFERENCES

[1] Deb, Kalyanmoy *Optimization for engineering design: Algorithms and examples*. PHI Learning Pvt. Ltd., 2012.

[2] Snyman, Jan *Practical mathematical optimization: an introduction to basic optimization theory and classical and new gradient-based algorithms, Vol. 97*. Springer, 2005.

[3] Papadimitriou, Dimitrios and Giannakoglou, Kyriakos, Aerodynamic Shape Optimization Using First and Second Order Adjoint and Direct Approaches. *Archives of Computational Methods in Engineering, Vol. 15* 447–488, 2008.

[4] Deb, Kalyanmoy *Multi-objective optimization using evolutionary algorithms, Vol. 16*. John Wiley & Sons, 2001.

[5] Samareh, Jamshid A., Survey of shape parameterization techniques for high-fidelity multidisciplinary shape optimization *AIAA journal, Vol. 39* 877–884, 2001.

[6] Castonguay, Patrice and Nadarajah, Siva K, Effect of shape parameterization on aerodynamic shape optimization *45th AIAA Aerospace Sciences Meeting and Exhibit* 1–20, 2007.

[7] Jameson, A. Aerodynamic design via control theory. *Journal Scientific Computating Vol. 3* 233–260, 1988.

[8] G. van Bloemen Waanders, et Al. Sensitivity technologies for large scale simulation *Sandia Nationa Laboratories*, 2005.

[9] Nadarajah, Siva Kumaran The discrete adjoint approach to aerodynamic shape optimization *Stanford University*, 2003.

[10] Lacor, C. and Smirnov, S., Non-Deterministic Compressible Navier-Stokes Simulations using Polynomial Chaos *Proc. ECCOMAS Conf*, 2008.

[11] Dinescu, Cristian, Smirnov, Sergey, Hirsch, Charles and Lacor, Chris. Assessment of intrusive and non-intrusive non-deterministic CFD methodologies based on polynomial chaos expansions *International Journal of Engineering Systems Modelling and Simulation Vol. 2,* Inderscience Publishers, 87–98, 2010.

[12] Xiu, Dongbin and Karniadakis, George Em, The Wiener–Askey polynomial chaos for stochastic differential equations *SIAM journal on scientific computing Vol. 24,* 619–644, 2002.

[13] Hosder, Serhat and Walters, Robert W and Balch, Michael, Efficient sampling for non-intrusive polynomial chaos applications with multiple uncertain input variables. *Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, AIAA paper*, 2007.

[14] Jasak, HRVOJE and Tukovic, Zeljko, Dynamic mesh handling in openfoam applied to fluid-structure interaction simulations. *Proceedings of the V European Conference Computational Fluid Dynamics, Lisbon, Portugal, June*, 14–17 2010.

[15] Weller, H. G. and Tabor, G. and Jasak, H. and Fureby, C., A tensorial approach to computational continuum mechanics using object-oriented techniques *Computers in Physics, Vol. 12*, 620–631, 1998.

[16] Rumsey, Chris Turbulence modeling resource. *NASA, 2D NACA 0012 Airfoil Validation Case* 2013.