Efficient Uncertainty Quantification using Gradient-Enhanced Kriging

Richard P. Dwight*† and Zhong-Hua Han*

German Aerospace Center (DLR), D-38102 Braunschweig, Germany.

A flexible non-intrusive approach to parametric uncertainty quantification problems is developed, aimed at problems with many uncertain parameters, and for applications with a high cost of functional evaluations. It employs a Kriging response surface in the parameter space, augmented with gradients obtained from the adjoint of the deterministic equations. The Kriging correlation parameter optimization problem is solved using the Subplex algorithm, which is robust for noisy functionals, and whose effort typically increases only linearly with problem dimension. Integration over the resulting response surface to obtain statistical moments is performed using sparse grid techniques, which are designed to scale well with dimensionality. The efficiency and accuracy of the proposed method is compared with probabilistic collocation, direct application of sparse grid methods, and Monte-Carlo initially for model problems, and finally for a 2d compressible Navier-Stokes problem with a random geometry parameterized by 4 variables.

I. Introduction

Computational methods in science and engineering have become key technologies in a wide range of application areas. A notable example is Computational Fluid Dynamics (CFD) in transport and aviation, which is one of the most numerically and computationally demanding fields of modeling, but which nevertheless plays important roles at all stages of the modern aircraft design process and is widely accepted as an essential tool. However the application of uncertainty quantification in this field has been limited due to the high computational cost of performing deterministic calculations.

The problem we are concerned with here is to quantify parametric uncertainty, i.e. to determine the probability density function (PDF) (or statistical moments such as mean and variance), of some output quantity \( J \) of a computational code, given a vector of stochastic input parameters \( \alpha \) of size \( M \), with known PDFs. So far, efforts in the literature have mainly concentrated on problems with only a few uncertain parameters. Due to the correspondence between the \( M \) and the dimension of the spaces which arise, the problem becomes exponentially more demanding for increasing \( M \) because of the curse of dimensionality - i.e. the exponential growth of (hyper-)volume as a function of dimensionality.\(^2\)

However complex multi-physics engineering problems routinely involve tens of physical and numerical parameters. Even for relatively simple geometries, taking into account shape variations for the purposes of shape optimization requires tens of parameters in 2d and potentially hundreds in 3d.\(^4\) Furthermore the solution of the deterministic problem is often already an extremely computationally demanding task, and performing a large number of such solves is unrealistic. There is therefore a pressing need for efficient uncertainty quantification methods in very high dimensions.

Uncertainty quantification approaches may be divided into two broad classes: intrusive and non-intrusive, referring to whether the approach requires modification of the modeling code or not. The most widely-used intrusive technique, polynomial chaos, offers efficiency gains over the current leading non-intrusive methods, but implementation requires a very large code-specific and uncertain parameter-specific effort. Such an effort

†Faculty of Aerospace - Aerodynamics Group, TU Delft.
is almost certainly not feasible for complete industrial multi-physics systems unless the process is automated in a similar way in which code differentiation may be automated.$^{14}$

In contrast non-intrusive methods require only calls of the computer model, allowing coupling to any code with respect to any set of parameters at a minimal implementation cost. While slightly more computationally expensive than intrusive polynomial chaos, the costs of e.g. the probabilistic collocation method, remain reasonable for a low number of uncertain variables. In the following we aim to develop a non-intrusive technique that scales well to at least 10-dimensional cases - although in the examples in the present paper a maximum dimension of 4 is considered.

To motivate the following, first note that the problem of determining statistical moments is reducible to the problem of evaluation of integrals over a unit hypercube - this will be demonstrated in the following section. There are well-developed mathematical methods for tackling such integrals in the general case. To obtain any improvement on these approaches it is therefore necessary to consider the special features of the uncertainty quantification problem:

(a) For many input parameters $\alpha$, $J$ varies only linearly over the range of integration.
(b) High accuracy is not critical as input parameter PDFs are unlikely to be known to high accuracy.
(c) High dimensional spaces are of interest.
(d) The uncertain integral is weighted with input parameter PDFs.
(e) Evaluations of $J$ are extremely expensive.
(f) The derivatives of $J$ with respect to all $\alpha$ may be available at a cost independent of $M$ (adjoint method).
(g) $J$ may contain discontinuities.

For example properties (a), (d) and (f) are exploited in the two-stage approach of Loeven et al.$^{21}$ First gradients of $J$ with respect to all uncertain parameters are computed about the mean state. Those (generally few) parameters for which $J$ is seen to vary rapidly are then examined more accurately with a probabilistic collocation method while other parameters are neglected.

There are additionally engineering requirements and desirable properties on an uncertainty quantification technique. The authors consider some of these to be:

(i) No more than 10-100 evaluations of $J$ are feasible.
(ii) Available error estimates on $J$ should be exploited.
(iii) It should be possible to incorporate any pre-existing evaluations of $J$ into the analysis.
(iv) Failure of evaluation of $J$ for a single point should not lead to failure of the entire method (robustness).
(v) Error estimates and error reduction methods for the uncertain output should be available.

Some explanation is required for points (iii) and (iv). In practice uncertainty calculations are likely to be performed as part of a wider study - it is to be expected, for example a deterministic calculation at the mean condition, $\bar{\alpha}$, is likely to be available, and it is desirable that this information be incorporated into the uncertainty quantification, hence (iii). Furthermore it should not be the case that failure of a single deterministic simulation at a single $\alpha$ condition causes failure of the entire uncertainty quantification process (which may involve 100 deterministic simulations). These requirements can not be satisfied with fixed integration rules, but call for the use of surrogate modeling.

Our proposed approach is as follows: using an adjoint method the derivatives of $J$ with respect to all $\alpha$ may be evaluated at a cost equivalent to about one evaluation of $J$ itself,$^5$ independent of the dimension of $\alpha$. As such the adjoint method provides a very efficient mean of gaining local information about the behaviour of $J$ in many dimensions.

In order to incorporate this information into a general model for $J$ a Gradient-Enhanced Kriging (GEK) response surface is used in the parameter space.$^{16}$ Kriging allows for samples at arbitrary locations, and its probabilistic framework admits incorporation of error information, allowing items (ii), (iii), (iv) and (v) of the engineering requirements to be satisfied. Building the Kriging models requires obtaining the correlation coefficients, by the solution of an optimization problem in a (often noisy) design space of the same dimension.
as the response surface. The Subplex method\textsuperscript{27} is seen to be particularly well suited to this task, providing convergence proportional to dimension, and robustness to noise.

Given a response surface it remains to evaluate integrals over the surface. We employ sparse grid integration techniques\textsuperscript{12,29} to provide reasonable scaling of effort with dimension - for even very cheap evaluation of response surface samples, increasing dimension makes integration rules based on products of one-dimensional rules rapidly impractical.

Finally, the choice of sample points to build the response surface must be made. This is based initially on a Latin-Hypercube sampling of the probability space (assuming that no prior samples are available). Thereafter sample points are added adaptively, currently using an ad-hoc adaptation indcator for the required integrals, based on the standard Kriging error estimator.

Note that the same approach may be used without alteration if gradient information is not available or is limited, at the cost of accuracy and high-dimensional efficiency.

In Section II the formulation of the uncertainty quantification problem is described. Section III gives a theoretical overview of the adjoint method, while Section IV discusses two issues associated with linearized CFD codes that may limit the application of gradient information in response surfaces. Section V briefly considers some standard quadrature techniques used for the uncertainty quantification problem, while Section VI introduces Kriging response surfaces, and their generalization to the use of gradient information. Finally Section VII combines all these methods into the above outlined uncertainty quantification approach. The resulting method is compared with existing non-intrusive methods, Monte-Carlo, the Gauss rule, probabilistic collocation, and sparse grid techniques (applied directly to $J$ rather than a response surface). These approaches are applied to uncertain simulation of turbulent compressible flow about a RAE 2822 aerofoil, with uncertain geometry parameterized by 4 variables.

II. Uncertainty quantification problem

Consider an arbitrary deterministic non-linear problem with unknowns $u$ and an $M$-dimensional deterministic parameter vector $\alpha$,

$$N(u, \alpha) = 0,$$

where the computed solution $u$ is not of principle interest, but rather only an $N$-dimensional vector of functionals of the solution $J := J(u, \alpha)$. Such a situation arises often, e.g. in aerodynamics, where the forces and moments on a body, and not the entire flow field, are of primary engineering relevance.

Now assume that $\alpha := \alpha(\omega)$ are random variables of a random event $\omega \in \Omega$, with known probability density functions (PDFs) $f_\alpha(\alpha)$. A probability space may then be formally defined by $(\Omega, \Sigma, P)$, with $\Omega$ the set of possible outcomes, $\Sigma \subset 2^\Omega$ the $\sigma$-algebra of events, and $P : \Sigma \rightarrow [0,1]$ a probability measure. The objective is to determine $f_J(J)$ the probability density function of the random variable $J$, and in particular the expectation, variance and higher moments of $J$, which may all be cast as integrals over the parameter space.

For example the expectation $\mathbb{E}[J]$ may be written

$$\mathbb{E}[J(u(\alpha(\omega)), \alpha(\omega))] = \int_0^\infty J(u(\alpha), \alpha)f_\alpha(\alpha)\,d\alpha = \int_0^1 J\left[\left[u(F_\alpha^{-1}(\xi)), F_\alpha^{-1}(\xi)\right]\right]d\xi \quad (2)$$

where $F_\alpha$ are the cumulative density functions (CDFs) of $\alpha$, and a change of variables has been used to obtain the second equality, employing the relation between the CDF and PDF: $dF_\alpha/d\alpha = f_\alpha$. This change of variables is sometimes referred to as the chaos transformation, and transforms the problem from parameter space into probability space. Thus the problem of parametric uncertainty quantification may be naturally reduced to the standard problem of integration over an $M$-dimensional hypercube. Unique features of the uncertainty quantification problem determine the character of the integrand, and hence which integration method will be most accurate and efficient.

III. Gradient evaluation via the adjoint method

One feature of the problem that may be exploited immediately if $N \ll M$ is that all gradients $dJ/d\alpha$ may be obtained at a cost proportional to $N$ using a dual problem approach.\textsuperscript{5} Briefly, let the Lagrangian $\mathcal{L}$, be defined as

$$\mathcal{L}(u, \alpha, \psi) = J(u, \alpha) + \psi^T N(u, \alpha),$$

where $\psi$ is a vector of Lagrange multipliers, and $N(u, \alpha)$ is the constraint matrix. The adjoint formulation is then

$$\frac{d}{d\alpha} \mathcal{L}(u, \alpha, \psi) = 0 \quad (3)$$

with solution $\psi = \psi(\alpha, \psi)$. The final equality is obtained by differentiating the constraints

$$\frac{d}{d\alpha} \left[ N(u, \alpha) \right] = 0 \quad (4)$$

and $\psi$ the Lagrange multipliers that satisfy

$$\psi = \left[ \frac{d}{d\alpha} \left[ N(u, \alpha) \right] \right]^{-1} \frac{d}{d\alpha} \left[ J(u, \alpha) \right] \quad (5)$$

The adjoint method is limited, at the cost of accuracy and high-dimensional efficiency.
where $\psi$ is the adjoint state, a vector of the same dimension as $N$. Now note that since $N \equiv 0$ for all $\alpha$ and $u$ satisfying (1), $L \equiv J$, $\forall \alpha$, and hence also

$$
\frac{dJ}{d\alpha} = \frac{dL}{d\alpha} = \left( \frac{\partial J}{\partial \alpha} + \frac{\partial J}{\partial u} \frac{du}{d\alpha} \right) + \psi^T \left( \frac{\partial N}{\partial \alpha} + \frac{\partial N}{\partial u} \frac{du}{d\alpha} \right)
$$

(4)

$$
= \left( \frac{\partial J}{\partial \alpha} + \psi^T \frac{\partial N}{\partial \alpha} \right) + \left( \frac{\partial J}{\partial u} + \psi^T \frac{\partial N}{\partial u} \right) \frac{du}{d\alpha}.
$$

(5)

Now if $\psi$ satisfies the linear adjoint equation:

$$
\frac{\partial J}{\partial u} + \psi^T \frac{\partial N}{\partial u} = 0,
$$

(6)

then the unknown $du/d\alpha$ term drops out of the expression for the gradient in (5), and we have

$$
\frac{dJ}{d\alpha} = \frac{\partial J}{\partial \alpha} + \psi^T \frac{\partial N}{\partial \alpha}.
$$

(7)

Since (6) is independent of $\alpha$, only $N$ such linear equations must be solved to obtain all gradients $dJ/d\alpha$.

The main prerequisite for application of this technique is the availability of the partial derivatives of the (typically discrete) operator $N$. For complex simulation codes, obtaining these is in principle a straightforward process, but requires access to the source code, and represents a considerable amount of effort.\(^5,^8\)

The compressible Navier-Stokes solver examined in this paper, the DLR TAU-Code, has an adjoint solver mode available - originally developed for the purposes of gradient-based optimization and error estimation\(^6,^7\) - and this method for determining gradients of $J$ is used in what follows.

### IV. Limitations of adjoint gradients in CFD

The adjoint methods described above were developed in CFD in the context of optimization problems, where the aim is typically to minimize the drag on an aerofoil by modifying its shape (parameterized by a large number of design variables), subject to constraints lift and aerofoil thickness. This type of problem is characterized by extremely regular flows. At least as the region of the optimum is approached, shocks and regions of separated flow have typically been eliminated because these flow features represent the greatest contribution to the removable drag. Furthermore - depending on the optimization algorithm - the demands on gradient accuracy are not particularly stringent. For example if a conjugate-gradient method combined with a line-search is used for descent, then provided the gradient points in roughly the right direction in the design space, the method is very likely to converge.\(^9\)

However when building a general-purpose response surface with gradients we can not afford to be so relaxed. Regions of interest in the parameter space are likely to include “difficult” flows with heavily separated flows, stall, and strong shocks. Furthermore the gradients must be accurate if spurious oscillations are not to be seen in the response surface.

To highlight these problems, we present two issues associated with obtaining accurate gradients in CFD, one associated with non-smooth resolution of shocks by the flow solver, and one associated with a standard approximation of freezing the turbulence equations in the linearization. The later may be resolvable with further development of numerical techniques, but appears at present intractable. The former is a more fundamental problem. In any case these results will show that strict enforcement of the gradient in a response-surface model is not always appropriate.

#### A. Oscillatory gradients due to shocks

The presence of discontinuities in solutions of the Euler equations causes a variety of discretization difficulties which modern numerical techniques are capable of handling well. However shock-capturing methods often suffer from an oscillatory dependence of the solution on flow parameters. Such a phenomena was first observed by Giles et al.\(^11\)

To demonstrate this effect we give the example of a NACA 0012 aerofoil at an angle-of-attack of $0.1^\circ$ on a sequence of three structured grids, and Mach number varying from a subsonic to a transonic regime. The lift coefficient is plotted in the top left, and (zoomed) top right of Figure 1. For small Mach numbers the
curve is smooth, but at a Mach number of about 0.78 a shock starts to form on the upper and lower surfaces and as the speed increases the shocks travel slowly along the aerofoil. The correct physical behaviour is a smooth Mach–lift curve, the oscillations seen in the figure are numerical artifacts caused by the varying position of the shocks in relation to the mesh points. If a shock occurs on the boundary of two mesh cells, it is likely to be better resolved than if it lies in interior of a cell, as the solution representation is continuous within cells and discontinuous on cell boundaries. This can be confirmed by noting that the frequency of the oscillations double when the spacing of the surface mesh points halves.

These oscillations are only visible because of the fine resolution of the curve, and are generally not considered a serious problem because they are aerodynamically not of significant amplitude, and as the mesh is refined their size decreases — if the discretization is consistent. However the linearization of the discretization is also aware of these oscillations, and the derivatives calculated from the linearization faithfully follow them. This can be seen again in the top two plots of Figure 1, the bar attached to each point is based on the adjoint derivative of lift with respect to Mach number at that point.

Now the problem arises that although the amplitude of these oscillations reduces as the mesh is refined, so their frequency increases, and their overall shape remains roughly constant. Hence the magnitude of the first derivatives of these oscillations does not decrease with mesh resolution. This is shown in the bottom plots of Figure 1, where the gradient obtained by adjoint is plotted directly. The oscillations are of large magnitude compared to the absolute value of the gradient, and their magnitude actually increases with mesh resolution. Clearly such gradients are of limited practical value despite being perfectly correct descriptions of the local behaviour of the discrete flow solver.

B. Gradient error due to a frozen-turbulence approximation

Although it is possible to completely linearize the $N$ of (1) for discretizations of the Reynolds Averaged Navier-Stokes (RANS) equation, including the turbulence model - if this is actually performed the resulting linear system is often so badly conditioned that it can not be solved using anything less than direct inversion. For this reason very often the turbulence model is not linearized, rather frozen by treating eddy-viscosity, turbulence energy and any other turbulence quantities as constant with respect to the linearization. This is termed the frozen-turbulence approximation, and modifies the solution of the linearized problem, and hence the calculated gradients. In $^9$ the present author examined the influence of this approximation of the Jacobian on gradient accuracy with the conclusion that the magnitude of the error is strongly dependent on the flow under consideration.

To demonstrate the type of gradient errors that can result we consider an RAE 2822 single-element 2d aerofoil, modeled with the Spallart-Almaras-Edwards one-equation turbulence model at a Reynolds number of $R = 6.5 \times 10^5$, Mach number of $M = 0.73$ and varying angle-of-attack $\phi$. Lift and drag are plotted in Figure 2. A variety of flow phenomena occur in this $\phi$ range. A shock develops at about $\phi = 2^\circ$, flow begins to separate at about $\phi = 3^\circ$, the flow is in the process of massive separation in the range $\phi = 4.5 - 8^\circ$ and the flow solver can not obtain a stationary solution here. From $\phi = 8^\circ$ the flow is fully separated from the upper surface, and the solver can again obtain a stationary solution.

The line segments plotted in Figure 2 represent gradients obtained with the adjoint method at each angle-of-attack, with and without linearization of the turbulence model. Apart from the region in which no stationary solutions are obtained, the fully-linearized adjoint gives accurate gradients (judging by agreement with the polar). The frozen-turbulence approximation performs well provided separation is either not present, or has no significant effect on the integrated forces. However, as maximum lift is approached the approximation breaks down completely, presumably as the importance of variation of turbulence to the flow grows.

To see the effect such gradient errors have on a surrogate model consider Figure 3, which shows Kriging and gradient-enhanced Kriging (GEK) response surfaces (to be described in the following sections) with exact and frozen-turbulence gradients. The surfaces are smooth, are required to pass through all the sample points, and in the case of GEK the surface is required to have the specified gradient at the sample point too. The surfaces are based on 4 sample locations deliberately chosen to lie outside the oscillatory region where the flow solver did not converge to a stationary state. Given accurate gradients GEK performs substantially better than basic Kriging. With poor quality gradients it is clear that the response surface is of no value, returning unphysical negative drag in some regions.

One chance of resolving these difficulties with linearization of turbulence models lies in the enforcement of realizability conditions to explicitly avoid the creation of un-physical states, and choosing variables in...
Figure 1. Lift coefficient and derivatives against Mach number for a transonic NACA0012 aerofoil on three grids. The bars in the top plots display the derivatives computed using the discrete adjoint.
Figure 2. Lift and drag against angle-of-attack for an RAE 2822 aerofoil. Line segments represent gradients computed with a discrete adjoint code, with a full linearization of the turbulence model (black), and a frozen-turbulence approximation (red).
Figure 3. One-dimensional Kriging and GEK response surfaces for lift and drag based on 4 support points for the RAE 2822 test-case with varying angle-of-attack.

which the solution is smoother e.g. using the variable log $\omega$ rather than $\omega$ directly in the implementation of the $k - \omega$ turbulence model. The former however involves some modifications that change the physical behaviour of the model, and it remains to be seen if such changes will be accepted by the aerodynamics community.

In the following where CFD gradients are used they are obtained with full linearization of the turbulence model in order to observe the uncertainty quantification algorithm in the absence of gradient error. At present this is only feasible in 2d, where direct inversion of the linear system is feasible, using e.g. SuperLU.3

V. Direct quadrature techniques

To evaluate the integral of (2) a variety of standard weighted techniques may be used. Monte-Carlo and pseudo Monte-Carlo methods are straight-forward to apply and are well known for converging at a rate of $O(n^{-1})$, independent of dimension. Spectral Gram-Schmidt Probabilistic collocation (PC) methods have found popularity in uncertainty quantification due to their optimality in the sense of exponential convergence of error for arbitrary input distributions. However integration rules based on simple tensor products of these methods scale poorly to high dimensions. This “curse of dimensionality” may be mitigated by the use of sparse grid methods. Rather than taking a products of high-order 1d rules in every Cartesian direction, these combine of high-order rules in each direction with low-order rules in all other directions. By choosing the baseline 1d rule to be hierarchical, further effort can be saved. Gram-Schmidt PC and sparse grid methods will not be described further here - for details see the above citations.

These three techniques, Monte-Carlo, probabilistic collocation and sparse grid integration are applied to a test integrand, and error against number of functional evaluations is plotted in Figure 4 for each method and a range of dimensionalities. The test integrand is Genz Integrand Number 1, namely:

$$f_1 = \cos \left(2\pi w + \sum_{1 \leq i \leq M} c_i x_i \right)$$

where $x$ is the independent variable, $w = 1/2$ and $c_i = 9/M$, over the unit hypercube. In the case of probabilistic collocation we take all $x$ to be distributed uniformly on $[0,1]$ to perform the integral, in which case PC corresponds to the Gauss rule. The exponential convergence of PC is visible, especially in the leftmost plot of Figure 4, as is its poor scaling to high dimensions. The substantial dimension independence but low accuracy of Monte-Carlo is seen, as is the remarkable dimension independence achieved by the sparse grid technique.
Note that the accuracy of the lowest order rule for PC improves with dimension, this is a feature of the integrand, which is an oscillation in only one direction in $M$-dimensional space, and therefore becomes effectively smoother in higher dimensions. This may or may not be representative of typical high-dimensional uncertainty quantification problems in CFD.

![Figure 4. Application of three integration methods to a test integral of varying dimensionality.](image1)

To better display the dependence on dimension, in Figure 5 effort (in terms of functional evaluations) required to achieve a given error tolerance is plotted for each method, and for three tolerances. Given that the absolute upper limit on acceptable number of function evaluation in a CFD context is about 100, we see that an error of 1% in a 20-dimensional problem is just about within reach using sparse grids, and completely impractical with either of the other techniques.

![Figure 5. Scaling of three integration methods in problem dimension for a test integral. Effort required to achieve three levels of error is plotted.](image2)

In the following sparse grid integration will be applied, not directly, but to obtain estimates of integrals over response surfaces. That this is necessary may be seen from the results above: even when applied to a cheap response surface, probabilistic collocation becomes computationally intractable for sufficiently high dimension.

**VI. Response surface techniques**

The methods of the previous section require samples at special locations, and are difficult to generalize to incorporate (possibly incomplete) gradients and sample error information. Furthermore although error estimators are available, error reduction methods generally require resampling unless the quadrature rule is hierarchical, which it is not in general for probabilistic collocation methods. A great deal more flexibility is needed in order to satisfy engineering requirements of adaptivity, re-use of samples and robustness.
set out in the introduction. For such requirements response surface techniques are appropriate, providing reconstruction of functions with no restrictions on sample locations.

Of the many methods available, such as radial basis function models, neural networks, smoothing spline models, and support vector machines, for this work we initially concentrate on Kriging. This technique is rooted in a statistical framework, thereby fitting well with the theme of uncertainty. It is highly generalizable: incorporation of sample error and gradient information (gradient-enhanced Kriging) is possible, as well as trend information from other models of lower fidelity (co-Kriging). Finally there are a variety of predictor error estimators and adaptation algorithms available.

Once a Kriging model of the parameter space is available, sparse grid techniques will be applied on the model to obtain approximations of expectation, variance, and other integrals of interest.

A. Kriging response surfaces

A particularly lucid exposition of Kriging may be found in, here we present a brief overview of the main idea. Consider standard curve-fitting by regression for a deterministic scalar function $y$ sampled at $n$ points $x_i$ in $M$ dimensions. In this technique the observations are treated as if they were generated from the following model:

$$ y(x) = \sum_j \beta_j f_j(x) + \epsilon, \quad \text{(8)} $$

where $f_j$ and $\beta_j$ are the regression functions and coefficients respectively and the errors $\epsilon$ are assumed to be independently normally distributed with mean zero. Usually $f_j$ are pre-defined and a least-squares measure is used to determine the best values for $\beta_j$. However unless $f_j$ capture the full non-linear behaviour of $y$, the assumption of independence of errors is blatantly false for a deterministic $y$. Rather we would expect that errors are a function of position $\epsilon = \epsilon(x)$, and errors at two close points are closely correlated.

Motivated by this argument the correlation between the errors at two points is modeled as

$$ \text{Corr} [\epsilon(x), \epsilon(x')] = R_{ij} = e^{-d(x, x')}, \quad \text{(9)} $$

where $d$ is some weighted distance function

$$ d(x, x') = \sum_{k=1}^{M} \theta_k |x_k - x'_k|^p, \quad \text{(10)} $$

where the weights $\theta_k > 0$ account for different levels of correlation in each dimension, and $p = 2$ (resulting in a Gaussian correlation) in what follows. This model turns out to be sufficiently general that it is possible to dispense with the original regression, and use the sample model

$$ y(x) = \mu + \epsilon(x) \quad \text{(11)} $$

where $\mu$ is the mean of the stochastic process, $\epsilon(x)$ are normally distributed with zero mean and variance $\sigma^2$ and are no longer independent, but correlated according to (9).

The fitting is then performed by determining values for $\mu$, $\sigma^2$ and $\theta$, such that the likelihood of achieving the observed sample with this model is maximized. The likelihood function is:

$$ L(\theta, \mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} |R|^{1/2} \exp \left[ -\frac{(y - \mu)^T R^{-1} (y - \mu)}{2\sigma^2} \right], \quad \text{(12)} $$

where $\mathbf{1}$ is a vector of ones of dimension $n$. Given $\theta$ there exist closed-form expressions for $\mu$ and $\sigma^2$ maximizing (12), namely

$$ \hat{\mu} = \frac{\mathbf{1}^T R^{-1} y}{\mathbf{1}^T R^{-1} \mathbf{1}}, \quad \hat{\sigma}^2 = \frac{(y - \hat{\mu})^T R^{-1} (y - \hat{\mu})}{n}, \quad \text{(13)} $$

so that an $M$ dimensional optimization problem must be solved for $\theta$. The inversion of the positive definite matrix $R$ in (12) is performed with a Cholesky factorization.

Given values for $\theta$, the model may be now used to predict the functional value at unsampled points. The best linear unbiased predictor of $y(x^*)$ may be shown to be

$$ y(x^*) = \hat{\mu} + r^T R^{-1} (y - \mathbf{1} \hat{\mu}), \quad \text{(14)} $$

where

$\text{American Institute of Aeronautics and Astronautics}$
where the first term is the regression prediction, and the second term is a correction for the error correlation. The correlation between the error at the untried point \( x^* \) and the error at the sample points \( r \) is of the form:

\[
r^i = \text{Corr} \left[ \epsilon(x^*), \epsilon(x^i) \right] = e^{-d(x^*,x^i)},
\]

(15)

**B. Gradient-enhanced Kriging (GEK)**

In order to incorporate gradients of \( y \) obtained by an adjoint method, we employ so-called gradient-enhanced Kriging, for which there are two formulations: direct and indirect.\(^{15}\) Both approaches have similar costs and give similar results, but the indirect approach is conceptually much simpler. There the sample set is augmented with samples obtained by linear reconstruction by a small distance \( \delta \), in each coordinate direction. The standard Kriging is then performed on this augmented sample.

In the direct approach, the Kriging MLE and predictor are re-derived, incorporating gradient information directly, after which the gradients of \( y \) may be used directly as sample data.\(^{15}\) Investigations of this approach have revealed it to deliver substantially the same results as the indirect approach. It has the advantage of not requiring the tuneable parameter \( \delta \), and is more accurate if a sample point happens to fall at a local minimum. Furthermore the likelihood \( L \) of (12), tends to be smoother in the indirect approach, where the sample points spaced with \( \delta \) causes poor conditioning of the correlation matrix \( R \), and thereby numerical noise in \( L \). This noise makes the optimization problem for \( L \) more demanding in the indirect case.

The left plot of Figure 6 shows a comparison of direct and indirect GEK models. The analytic test function used here is

\[
y(x) = \frac{3}{10} + \sin \left( \frac{16}{15}x - 1 \right) + \sin^2 \left[ 2 \left( \frac{16}{15}x - 1 \right) \right],
\]

with samples and gradient samples at \( \{-3/4, 0, 3/4\} \). The indirect GEK uses a constant step size of 0.01. In both cases the Gauss random function is employed as the correlation function, with the correlation parameter \( \theta \) optimized globally by exhaustive search (\( \theta_{\text{opt}} = 3.035 \) for direct GEK and \( \theta_{\text{opt}} = 3.384 \) for indirect GEK). Only small differences are visible between direct and indirect GEK, and both are much better than Kriging without gradient information. The right plot demonstrates the minor effect of step size on the result of indirect GEK. Figure 7 shows the design space of various GEK models, where single precision has been used to emphasize the influence of rounding error in the MLE. The cost function for indirect GEK becomes noisy with decreasing \( \delta \), as \( R \) becomes increasingly ill-conditioned when sample points are close to collocated.

![Figure 6. Comparison of Kriging, and direct and indirect gradient-enhanced Kriging for an analytic test function.](image)

Nevertheless in the following we use exclusively indirect gradient-enhanced Kriging. To deal with noise in the design space and high dimensionality the Subplex optimization algorithm is used, see Section VI.C.

Consider now the cost of building the Kriging and GEK models for \( n \) samples in \( M \)-dimensions. For basic Kriging each evaluation of (12) requires inversion of the \( n \times n \) dense matrix \( R(\theta) \). The Cholesky decomposition applied to \( R \) costs \( \mathcal{O}(n^3) \). Assume that the optimization problem for \( \theta \) may be solved at a cost linear in the dimension of the problem \( \mathcal{O}(M) \), which is expected to be the typical case for the Subplex
method described in the following section. Then the total cost of building the response surface is $O(n^3M)$. The predictor may then be evaluated at a cost of $O(nM)$ per point.

However in indirect GEK the augmented sample set has size $n(M+1)$ (assuming we know derivatives in all directions at all sample points), so that the total cost of building the response surface is $O(n^3M^4)$ and the predictor cost is $O(nM^2)$. The same scaling of costs applies to direct GEK. So for very high dimensions GEK could become a significant part of the cost of the entire calculation, even for very expensive function evaluations. Although in our intended applications $n$ will remain small, and hence the problem tractable, the poor scaling in $M$ is undesirable. This is a weak point in the generalizability of this method to very high dimensions, and will be tackled in future work with local-Kriging, where the influence of sample points far from each other (in terms of the weighted distance function $d$) is neglected, resulting in a sparse linear system.

C. Kriging correlation parameter optimization

Arguably the most critical aspect of a practical Kriging algorithm is the solution of the $M$-dimensional global optimization problem for the correlation parameters. Often this issue is glossed over in the literature, where declarations such as “the parameter optimization problem usually has only a single local minimum,” are not uncommon - and demonstrably untrue even for simple problems in 1d. In addition the $\theta$-design space is often noisy due to ill-conditioning of the correlation matrix $R$ as already discussed. In summary the optimization problem may be considered to be of substantial difficulty.

The consequences of using poor $\theta$ values for reconstruction are response surfaces that tend towards spurious oscillations, and are poorly representative of the data. In one- or two-dimensions such a situation may be identified by visual inspection of the response surface, at the cost of automation. In higher dimensions this task becomes increasingly impractical, and reliable global optimization methods are essential.

Our approach is as follows: firstly, in order to standardize the problem, the sample data and coordinates in each dimension are centered on zero and normalized using the sample means and standard deviations respectively. Secondly, by taking logs of $\theta$ the problem is transformed from a constrained optimization problem for $\theta_i > 0$, to an unconstrained problem for $\log(\theta_i)$. Since $\theta$ represents a scaling it has an exponential character, and taking logs serves to better condition the design space, as well as avoiding the necessity for setting arbitrary bounds on $\theta$.

To perform the optimization we employ the Subplex algorithm of Rowan. It is a generalization of the Nelder-Mead Simplex (NMS) method, and works by applying NMS on a sequence of low-dimensional sub-spaces extracted from the higher-dimensional design space. Subplex retains the advantages of NMS with respect to robustness to noisy functions, but the number of function evaluations required for convergence typically increases only linearly with the problem dimension.

As for NMS, Subplex find only local optima and therefore requires some globalization procedure. In this
instance we apply the Subplex method repeatedly with \( M + 2 \) randomly selected initial conditions. These are chosen using Latin-Hypercube sampling\(^{23} \) in order to encourage a wide spread of values.

### VII. Numerical Results

In order to compare the algorithms described so far, and in particular their performance with respect to problem dimensionality, we resort initially to a model problem. Once general bounds on efficiency (in terms of number of functional evaluations required) have been established, the most successful techniques are applied to problems in CFD.

The majority of the original implementation required to obtain the following results was performed in the Python programming language, with the packages Numpy and Scipy\(^{17} \) used to provide efficient array operations and a high-level syntax comparable to that of MATLAB. Furthermore Python admits calling code in other languages with relative ease, allowing the immediate use of the original FORTRAN implementation of Subplex by Rowan,\(^{27} \) and a C implementation of Smolyak sparse grid integration (SMOLPACK), by Petras.\(^{25,29} \) Thus the investigation and integration of a wide variety of quite complex numerical techniques could be performed rapidly at a relatively low effort.

#### A. Sandtimer model problem

For the purposes of testing we define an \( n \)-dimensional model problem as follows:

\[
J := \sum_{i=1}^{n-1} u_i^2 + S_q(x_n)
\]

where each \( u_i \) for \( 1 \leq i \leq n - 1 \) is defined implicitly by

\[
N(u_i, x) = (x_n^2 + 1)(u_i + u_i^3) - x_i = 0,
\]

and \( S_q \) is a quartic with coefficients chosen such that it is slightly skewed, but still has a unique minimum at \( x = 0 \) with value 0:

\[
S_q(x) = r (\tilde{x}^4 + (\tilde{x} - 1)^2 - q), \quad \tilde{x} = sx + p,
\]

\[
p = 0.589754512301, \quad q = 0.289273423937, \quad r = 0.05, \quad s = 0.25.
\]

The resulting surface in two dimensions can be seen in the upper left plot of Figure 8, and will be denoted it the sandtimer model problem. The goal it to evaluate the expected value of \( J \) for \( x \) with a specified PDF.

This choice of problem serves several purposes: firstly to model the situation in CFD where we have a cost-function \( J \) which is an explicit function of a solution \( u \), which is itself defined implicitly. Secondly \( J \) cannot be represented by a finite polynomial, otherwise the probabilistic collocation method would give an exact uncertainty result for sufficient order. Thirdly the resulting surface is sufficiently irregular that the uncertainty quantification problem is difficult if the variation of \( x \) is large enough. Finally we are looking towards robust optimization problems: the optimization problem for \( J \) has a unique minimum at \( x = 0 \), while the optimization problem for \( E_x J \) has at least two local minima away from zero.

1. **GEK reconstruction of \( J \)**

To begin we illustrate the effect of incorporating gradient information into the Kriging response surface for \( J \), see Figure 8. The response surface is built using Kriging, GEK and radial basis function (RBF) interpolation with 5 and 20 random Latin-hypercube samples (Figures 8 and 9 respectively). The RBF interpolation used Gaussian functions, the parameters of which were chosen by hand to produce reasonable results.

With 5 samples standard Kriging finds no correlation in the data in the vertical direction \( \theta \approx 0 \), and the response surface deteriorates to a one-dimensional model. This often occurs with low numbers of sample points in rapidly varying functions - a more satisfactory surface may be obtained by arbitrarily setting a minimum \( \theta \), however a suitable value is difficult to predict in advance. GEK already reproduces the main features of the exact surface with 5 samples. With 20 samples, the lack of a sample within the peak on the left causes this feature to be absent in the RBF spline and Kriging model, though it is reproduced well when gradient information is added.
Figure 8. Reconstruction of a test surface using 5 random sample points.
Figure 9. Reconstruction of a test surface using 20 random sample points.
2. Adaptive estimation of $E_xJ$

In order to estimate $E_xJ$ we employ an adaptive algorithm. Initially $J$ and $dJ/dx$ are sampled at $M + 1$ points - the mean of $x$ and an additional $M$ points taken from a Latin-hypercube sampling weighted with the distribution of $x$. GEK is performed on this sample set in parameter space, and a high-order sparse grid integration (here employing a 9th order 1d-rule) is performed on the resulting response surface mapped into probability space. This provides a first estimate of $EJ$.

The mean-squared estimate (MSE) for the error in the response surface provided by Kriging, is weighted with the PDF of $x$ in order to obtain a heuristic adaptation indicator. It is expected that where the indicator is large a large contribution to the error in the integral for $EJ$ is made, and new samples are added. This process is repeated until convergence is achieved.

An example of the action of the method is shown for the 2d sandtimer with $x_1 \sim \text{Norm}(3,3)$ and $x_2 \sim \text{Norm}(2,3)$ in Figure 10 after 9 adaptation iterations. The upper left plot is again $J$, and upper right the GEK approximation to $J$ based on the samples shown by dots. Lower left is the Kriging MSE estimator, which is small near sample locations, and large far away as expected. When the estimator is weighted with the PDF of $x$ a highly multi-modal adaptation indicator is obtained (lower right), which seems to compromise extending sampling further away from the mean, and refining locally. The locations of maximum indicator are determined with a differential evolution (DE) optimization, the red points in the lower right plot are the sample locations for the optimization.

In this work only one $J$ sample is added per adaptation step, but as the effect of adding a sample is mostly local, removing a single peak from the adaptation indicator, we do not expect to see significantly different behaviour when multiple $J$ samples are added simultaneously.

3. Convergence of $EJ$

The above algorithm is applied to estimation of $EJ$ for the 2d and 4d sandtimer problem, with $x$ normally distributed with standard deviation $\sigma = \frac{1}{2}, 1, 3, 5$, and mean $\bar{x} = (3, 2)$ in the 2d case, and $\bar{x} = (3, 2, 1, 2)$ in the 4d case. It is compared with Gauss rule integration on the probability space, probabilistic collocation and sparse grid integration using SMOLPACK. Also considered is the algorithm already presented, but without use of gradient information.

Results for the 2d case are plotted in Figure 11, and for the 4d case in Figure 12. The oscillatory behaviour of probabilistic collocation for larger $\sigma_x$ may be explained by the fact that a Taylor series is a poor approximation for $J$ over the larger range. In these cases the Gauss rule converges much more regularly, likely because the sample locations remain much more closely clustered around the mean, whereas with PC the sample points rapidly spread to the tails of the distribution and thereby the corners of the parameter space. This property of PC can be countered by using a truncated normal distribution as an input, in which case more regular convergence would expected to be observed. For small $\sigma_x$ PC beats all other methods convincingly.

Understandably sparse grid is not competitive in the two-dimensional case - in four dimensions it is already more efficient than PC in all but the $\sigma = 1/2$ case, and its advantage might be expected to grow with increasing dimension.

The oscillations present in the Kriging and GEK convergence histories are partly due to the sensitivity of Kriging to the addition of new samples. The introduction of a single sample can cause the Kriging MLE to change smoothly, but in such a way that the location of the global maximum jumps from one location to another, changing the values of the correlation parameter $\theta$ and thereby the character of the entire surface. This leads to a jump or a spike in $EJ$. In an attempt to suppress this behaviour somewhat the choice of correlation parameter is under-relaxed from one iteration to the next, and is not taken as the optimum for the given data set. Even so oscillations are still highly in evidence, especially in the 4-d case.

Despite the corresponding difficulty in judging the convergence of these methods, they perform effectively, with GEK being of comparable or better efficiency than the other approaches in all cases. With comparable efficiency and additional flexibility, the Kriging-based approaches show promise as a replacement for the other methods in some circumstances.

B. Shape uncertainty for the RAE 2822 aerofoil

Now we consider the above algorithms applied to estimation of expected drag, $E_{cD}$ for CFD simulation of the RAE 2822 aerofoil previously mentioned. The flow conditions are deterministic, and identical to those
Figure 10. Adaptive sampling of $J$ for estimation of $EJ$ for $x \sim \text{Norm}(\bar{x}, 3)$ where $\bar{x} = (3, 2)$. $J$ (top left), GEK reconstruction (top right), Kriging mean-squared error estimator (lower left), error estimator weighted with PDF (lower right). The latter also shows in red the samples made by a Differential Evolution optimizer, attempting to find a maximum.
Figure 11. Convergence of estimates for $\mathbb{E}J$ for the 2-dimensional sandtimer model problem for $x$ normally distributed with various standard deviations.
Figure 12. As for Figure 11 for the 4-dimensional sandtimer problem.
of Section IV.B. However the camber-line of the aerofoil is parameterized with \((a_1, \ldots, a_4)\) multiplying four Hicks-Henne bump functions, see Figure 13. The aerofoil thickness is held constant. This parameterization is of relevance in optimization, where \(c_D\) is to be minimized, and the lift \(c_L\) is held constant by varying the angle-of-attack, \(\phi\). Without this constraint, minimum drag would be achieved by eliminating lift and therefore also the induced drag. The constraint on the lift modifies the expression for the derivative of \(c_D\) as follows:

\[
\nabla_L c_D = \nabla_\phi c_D - \left( \frac{d c_D}{d \phi} / \frac{d c_L}{d \phi} \right) \cdot \nabla_\phi c_L,
\]

where \(\nabla_L\) specifies the derivative at constant lift, and \(\nabla_\phi\) at constant angle-of-attack. Therefore two adjoint solutions, one of lift and one for drag, are required for the gradient at each sample location.

Figure 13. Parameterization of the RAE 2822 with 4 Hicks-Henne bump-functions (upper) modifying the camber-line. The modified aerofoil shapes are compared to the original shape (lower). Deformations shown correspond to large parameters settings of \((100, 0, 0, 0)\) etc.

Here this same parameterization is used as a test case for uncertainty in aerofoil geometry. It is not intended to represent variation that might occur in practice, however the constraint on lift is still reasonable, as a pilot will always attempt to maintain constant lift, not constant angle-of-attack.

Two mean states are considered, the \(a = 0\) case corresponding to the original RAE 2822, and \(a = a_{\text{opt}}\), which minimizes drag at the given flow conditions. The design space near the former is likely to be locally linear, while near the latter it should be locally quadratic - providing two cases of varying character. For the former we consider \(a\) normally distributed with standard deviation 5, and for the latter standard deviations of 5 and 10.

Slices through the parameter space are plotted in Figure 14, where a Kriging response surface has been used to compute contours, the samples of \(c_D\) come from calculations used in the various uncertainty approximation techniques applied to this case. The parameter space appears to have a simple structure, but more slices would be needed to verify this.

Estimates of \(E c_D\) for various methods are plotted in Figure 15, as well as \(c_D\) for the zero and optimal states. In all three cases probabilistic collocation achieves an accurate solution with the least possible number of samples (16). Sparse grid integration obtains similar accuracy with 9 samples for the \(\sigma = 5\) cases, but fails for the larger standard deviation. The adaptive GEK approach described here performs comparably well.
Figure 14. Slices through the design space of RAE 2822 test case, for the $a_1 = a_2 = 0$ plane (left), and the $a_0 = a_3 = 0$ plane (right). Contours plotted from a GEK reconstructed surface, samples plotted as black dots.

with the more traditional methods in all three cases. Again the value of the Kriging correlation parameter was under-relaxed in order to smooth the convergence behaviour, and in this case this proved very effective. In particular the under-relaxation did not prevent the response surface in the case $a = a_{opt}$, $\sigma_a = 10$, jumping from an apparently incorrect state (before about 12 functional evaluations) to a correct state (thereafter).

In all cases the magnitudes of the errors made in $E_{CD}$ are small in comparison to the variation of $c_D$ between the different test cases, suggesting that the 4-parameter uncertainty quantification may be accurate enough for many purposes, e.g. robust design in this case.

VIII. Conclusions

It has been demonstrated that by combining a Kriging response surface method with cheap gradient information using an adjoint problem, an adaptive method may be constructed which is comparably effective at quantifying uncertainty in up to 4 dimensions as existing standard integration methods such as sparse grid and probabilistic collocation.

Based on these results none of the methods considered has a clear advantage, and the best choice depends strongly on test case and level of accuracy required. It is expected that as the dimension of the problem increases the advantage to be gained by employing gradient information will also increase, but this remains to be seen. The issue of gradient accuracy also requires strict attention.

The Kriging-based method is more flexible than the traditional methods with respect to sample locations. It may therefore have a significant advantage in situations in which some parameter space sampling has already been performed, for example in optimization, or multiple uncertainty quantification calculation on the same space (with varying distributions).

Several issues remain to be solved however. Determining convergence of the adaptation iteration is problematic because of the Kriging model’s discontinuous behaviour with respect to the addition of new samples. Efficient and robust means of solving (or avoiding) the various high-dimensional, multi-modal optimization problems that arise are necessary, if the method is to extend to very high dimensions.

IX. Acknowledgments

Thank you to Alex Loeven of TU Delft for the Gram-Schmidt probabilistic collocation code, a polynomial chaos code for a model problem and many helpful discussions. Joel Brezillon provided the parameterization of the RAE 2822, and useful discussions regarding the optimization problems. Thanks also to the authors of the various numerical codes used in this work, K. Petras for the SMOLPACK implementation of sparse grid integration, and T. Rowan for the Subplex implementation.
Figure 15. Convergence of various uncertainty quantification methods for the 4-parameter RAE 2822 case.

References


American Institute of Aeronautics and Astronautics
17E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001–.


