Uncertainty Analysis Utilizing Gradient and Hessian Information

Markus P. Rumpfkeil, Wataru Yamazaki and Dimitri J. Mavriplis

Abstract In this paper gradient and Hessian information computed using automatic differentiation and the adjoint method is applied to approximate Monte Carlo simulations for a geometric uncertainty analysis involving a sinusoidally pitching airfoil.

1 Introduction and Motivation

Many real-world problems involve input data that is noisy or uncertain, due to measurement or modeling errors, approximate modeling parameters, manufacturing tolerances, in-service wear-and-tear, or simply the unavailability of the information at the time of the decision [1]. These imprecise or unknown inputs are important in the design process and need to be quantified in some fashion. To this end, uncertainty quantification (UQ) has emerged as an important area in modern computational engineering. Today, it is no longer sufficient to predict specific objectives using a particular physical model with deterministic inputs. Rather, a probability distribution function (PDF) or interval bound of the simulation objectives is required as a function of the uncertainties inherent in the simulation input parameters, in order to establish confidence levels over a range of performance predictions.

The use of surrogate models for both UQ and global optimizations has become popular. The idea of a surrogate model is to replace expensive function evaluations with an approximate but inexpensive functional representation which can be probed exhaustively if required. A gradient enhanced direct as well as indirect Kriging (called direct or indirect co-Kriging) has been developed in the surrogate model community and has shown very beneficial results [5]. While adjoint methods provide an effective approach for computing first-order derivatives, the ability to compute second-order derivatives is also highly desirable [2, 3, 8]. Second-order sen-

Markus P. Rumpfkeil
Dept. of Mech. Eng., U of Wyoming, Laramie, 82071, USA, e-mail: mrumpfke@uwyo.edu

Wataru Yamazaki
Dept. of Mech. Eng., U of Wyoming, Laramie, 82071, USA, e-mail: wyamazak@uwyo.edu

Dimitri J. Mavriplis
Dept. of Mech. Eng., U of Wyoming, Laramie, 82071, USA, e-mail: mavripl@uwyo.edu

sitivity information can be used effectively to devise efficient uncertainty propagation methods and inexpensive Monte-Carlo (IMC) techniques [3] for characterizing PDFs of computed simulation results. Since an efficient Hessian evaluation method has been developed in our group [7], it is very promising to utilize the Hessian information within surrogate models in addition to the gradient information [9].

2 Dynamically Sampled Kriging Model

Recently we have developed a gradient enhanced direct as well as indirect Kriging model [9]. In the direct co-Kriging approach, the covariances between function values, function values and gradients, as well as gradients have to be considered within the correlation matrix as opposed to the original Kriging formulation, where only the covariances between function values are important. In the indirect co-Kriging model additional sample points are constructed around a real sample point by using a Taylor series extrapolation. The principal advantage of the indirect co-Kriging approach is ease of implementation since the original Kriging model can be used. The major disadvantage is ill-conditioned correlation matrices which are produced because the additional sample points will be close to a real sample point. Since we also have the capability of computing the Hessian matrix [7], we can also utilize this information within the surrogate model. This is not a trivial task, since for the indirect approach the correlation matrix becomes even more ill-conditioned and for the direct approach up to fourth-order derivatives of the covariance are required, which we determine using automatic differentiation.

In order to obtain a globally accurate surrogate model, we refine the building of the model by a dynamic sample point selection with a stopping criteria rather than just to specify the sample size in the beginning and to pick the sample points through latin hypercube sampling. We construct a local response surface using a hybrid of extrapolation and interpolation involving a few, already existing, sample points D_i, $i = 1\ldots,I$ in order to guide the sampling process. The function values $J(D_i)$ and available derivatives at each sample point are used to construct an extrapolating function. At the location of a test candidate, D, the extrapolations from a sufficient amount of sample points are then weighted with a low-order interpolant. This approach has been coined Dutch Intrapolation [4] (DI) and it has been shown that the order of accuracy of the intrapolant is equal to the combined extrapolation and interpolation polynomial order, that is, using function, gradient, and Hessian information for the extrapolations and second-order interpolation leads to a fourth-order accurate intrapolant. The Dutch extrapolation functions are normal multivariate Taylor expansions of order n with a correction term given in multi-index notation by [4]

$$\mathscr{T}^n(D,D_i) = \sum_{|k|\geq 0}^{|k|\leq n} \frac{a_k^n}{k!}(D-D_i)^k \partial^k J(D_i) \quad \text{for} \quad i = 1,\ldots,I \qquad a_k^n = 1 - \frac{k}{n+1}.$$

The dynamic sampling method works as follows: Start only with the function (gradient and Hessian) values of the mean values of the samples and the corners of the boundary of the domain in question. Then repeat the following steps until convergence or a maximum amount of function (gradient and Hessian) evaluations has been reached

1. Generate a mesh using Delaunay triangulation in the sample space
2. Specify a set of test candidates randomly or geometrically (we pick the centers of the hypertriangles and the midpoints of the edges)
3. Construct a local function value for each test candidate using Dutch Intrapolation
4. Compare the global Kriging surrogate model function value predictions for the test candidates with the local Dutch Intrapolations
5. Add a user-specified number of test candidates with the worst discrepancy between the two to the set of sample points, only then evaluating the real function (gradient and Hessian)

We define convergence as having the worst discrepancy below a certain threshold. We also augment the selection process by geometric criteria, for example, we make sure that the distance of a test candidate to the nearest existing sample point is above the average distance of all test candidates to their respective closest sample point.

3 Uncertainty Analysis

Probabilistic assessment of uncertainty in computational models consists of three major phases: (i) characterization of input parameter variability from observations and physical evidence (ii) propagation of input variabilities through the model; and (iii) calculation of statistical properties of the output [6]. Arguably, the computationally most expensive part of UQ is the second phase. The simplest approach to obtain the output statistics in response to both aleatory and epistemic inputs is the Monte Carlo (MC) method, in which typically a large number of independent calculations are performed; however, in many practical cases the number of realizations required is too large which results in prohibitively high computational cost. Thus, the development of efficient surrogate models which capitalize on the availability of gradient and Hessian information constitutes an important avenue for reducing the cost of UQ.

To address the "curse of dimensionality" whereby the cost of quantifying uncertainties increases rapidly with the number of inputs, we combine two different strategies: firstly, select only the input parameters that are truly relevant to the simulation outcome through a sensitivity analysis and thus reduce the dimension of the problem at the outset; secondly, exploit the information gain at reduced additional cost with sensitivities. For example, the function and gradient provide $M+1$ pieces of information for M inputs for the constant cost of roughly two function evaluations if targeting a single output objective, using adjoint techniques. Similarly, the Hessian provides $M \cdot (M+1)/2$ pieces of information for the cost of roughly M function evaluations [2, 7]. Thus, one can reasonably expect to have to compute the output functional far fewer times to obtain good results when using gradient and Hessian information, which should also scale better to higher dimensions.

If one is only interested in the mean and standard deviation of an objective function, moment methods can be a good choice [8]. Moment methods are based on Taylor series expansions of the original non-linear objective function $J(D)$ about the mean of the input, μ_D, given standard deviations, σ_{D_j}. The resulting mean, μ_J, and standard deviation, σ_J, of the objective function are given to first order (MM1) by

$$\mu_J^{(1)} = J(\mu_D) \quad \text{and} \quad \sigma_J^{(1)} = \sqrt{\sum_{j=1}^{M} \left(\left. \frac{dJ}{dD_j} \right|_{\mu_D} \sigma_{D_j} \right)^2},$$

and to second order (MM2) by

$$\mu_J^{(2)} = \mu_J^{(1)} + \frac{1}{2} \sum_{j=1}^{M} \left(\left. \frac{d^2J}{dD_j^2} \right|_{\mu_D} \sigma_{D_j}^2 \right)$$

$$\sigma_J^{(2)} = \sqrt{\sum_{j=1}^{M} \left(\left. \frac{dJ}{dD_j} \right|_{\mu_D} \sigma_{D_j} \right)^2 + \frac{1}{2} \sum_{j=1}^{M} \sum_{k=1}^{M} \left(\left. \frac{d^2J}{dD_j dD_k} \right|_{\mu_D} \sigma_{D_j} \sigma_{D_k} \right)^2}.$$

Note that in the latter case, the non-linear shift between the mean of the output and the output of the mean is accounted for by the Hessian diagonal elements. If a complete PDF of the objective function is desired, a full non-linear MC simulation represents the most straight-forward approach for propagating uncertainties through the simulation process. Because this approach relies on a large number of repeated simulations, it is most often not practical for use with high-fidelity simulations. However, we can probe the Kriging surrogate exhaustively instead for relatively cheap computational cost. An even cheaper method for an IMC simulation is to simply use extrapolation [3]. A linear extrapolation (Lin) around the mean of the input, μ_D, for the sample point, D, is given by

$$J_{\text{Lin}} = J\Big(\mu_D\Big) + \left. \frac{dJ}{dD} \right|_{\mu_D} \cdot (D - \mu_D),$$

and a quadratic extrapolation (Quad) by

$$J_{\text{Quad}} = J_{\text{Lin}} + \frac{1}{2} \left. \frac{d^2J}{dD^2} \right|_{\mu_D} \cdot (D - \mu_D)^2.$$

We apply all of these proposed methods to to a sinusoidally pitching airfoil problem in the next Section.

4 Results for Sinusoidally Pitching Airfoil

As a test case we consider an inviscid NACA 0012 airfoil sinusoidally pitching about its quarter-chord location [7]. The free-stream Mach number is 0.755 with a mean angle of attack of 0.016 degrees, the time-dependent pitch has an amplitude of 2.51 degrees and a reduced frequency of 0.0814. One pitching period is divided into 32 discrete time steps and the entire simulation consists of 40 time steps after a steady-state solution with the mean angle of attack. The computational mesh has about $20,000$ triangular elements and the code is second-order accurate in both space and time. We allow one shape design variable on the upper and one on the lower surface to vary, which are defined as the magnitude of Hicks-Henne sine bump functions. The two design variables are treated as random variables with normal distribution. The mean is set to zero (corresponding to the NACA 0012 airfoil) and the standard deviations are taken to be $\sigma_{D_1} = \sigma_{D_2} = 0.01$.

The required MC samples are generated using latin hypercube sampling with a sample size of $10,000$. One flow solve takes about 15 minutes on four AMD processors with 2 GHz each and the adjoint solve for the gradient as well as the forward solves for each design variable for the Hessian calculation take about the same time. Comparisons of the mean and variance predictions of the objective function (time-averaged lift) using the various methods as well as approximate running times are displayed in Table 1.

Table 1 Mean and Variance predictions. Ten dynamically sampled points with function (F), function and gradient (FG), or function, gradient and Hessian (FGH) information are used for Kriging.

	Mean	% Error Mean	Variance	% Error Variance	Run time (minutes)
Non-linear	5.55×10^{-2}	-	1.07×10^{-2}	-	$150,000$
MM1	5.81×10^{-2}	4.74	1.05×10^{-2}	1.47	30
Lin	5.82×10^{-2}	4.69	1.05×10^{-2}	1.14	30
MM2	5.39×10^{-2}	2.93	1.05×10^{-2}	1.21	60
Quad	5.39×10^{-2}	2.93	1.06×10^{-2}	0.77	60
Kriging 10 F	5.45×10^{-2}	1.82	1.10×10^{-2}	3.07	150
Kriging 10 FG	5.51×10^{-2}	0.81	1.08×10^{-2}	1.26	300
Kriging 10 FGH	5.56×10^{-2}	0.04	1.07×10^{-2}	0.02	600

The 99 per cent confidence interval for the mean calculated with the full non-linear MC simulation is $[5.52 \times 10^{-2}, 5.58 \times 10^{-2}]$. As can be seen MM1 and Lin yield very similar results as expected from the leading error. Also, MM2 and Quad give similar results for the same reason. In Figure 1 we show the error in the mean

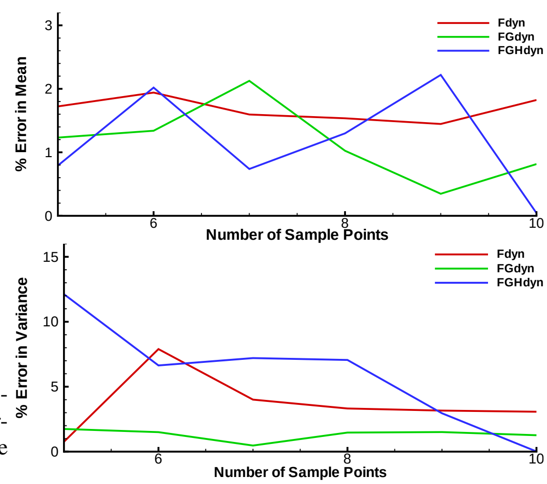

Fig. 1 Error in the mean and variance predictions using the Kriging surrogate model vs number of sample points.

and variance of the Kriging surrogate models compared to the values obtained from the non-linear MC simulations versus the number of sample points used to construct the surrogate. One can see that the gradient ($FGdyn$) as well as the gradient and Hessian enhanced Kriging models ($FGHdyn$) perform better than the model that is only based on function evaluations ($Fdyn$). Note that a certain number of sample points may be required to get a better estimate with the gradient/Hessian enhanced

Kriging model. Finally, as can be seen in Figure 2 the IMC methods capture the actual histograms and consequently PDFs of the time-averaged lift distribution quite well.

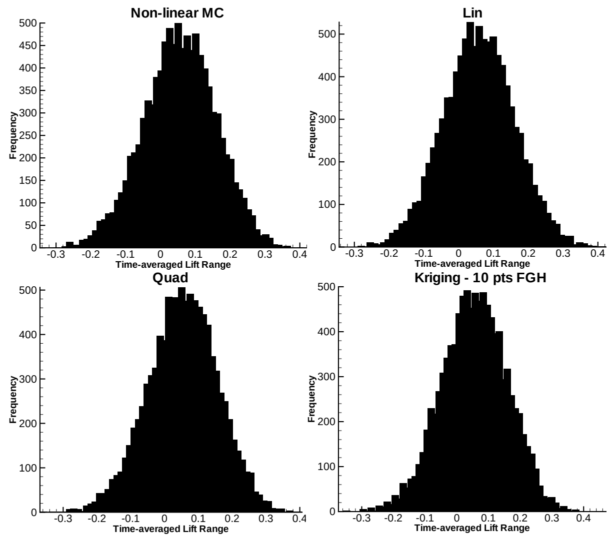

Fig. 2 Histograms for time-averaged lift perturbations using various methods.

5 Conclusion

We described a gradient and Hessian enhanced Kriging surrogate model with dynamic sample point selection. We applied the Kriging model as well as simple extrapolations to uncertainty quantification using inexpensive Monte Carlo simulations. All results benefited from the additional gradient and Hessian information.

References

1. A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. Foreword: special issue on robust optimization. *Mathematical Programming*, 107(1-2):1–3, 2006.
2. F. Chalot, Q. Dinh, E. Herbin, L. Martin, M. Ravachol, and G. Roge. Estimation of the impact of geometrical uncertainties on aerodynamic coefficients using CFD. AIAA Paper, 2008-2068.
3. D. Ghate and M. B. Giles. Inexpensive Monte Carlo uncertainty analysis. In *Recent Trends in Aerospace Design and Optimization, Tata McGraw-Hill, New Delhi*, pages 203–210, 2006.
4. D. A. Kraaijpoel. *Seismic ray fields and ray field maps: theory and algorithms*. PhD thesis, Universiteit Utrecht, The Netherlands, 2003.
5. J. Laurenceau and P. Sagaut. Building Efficient Response Surfaces of Aerodynamic Functions with Kriging and Cokriging. *AIAA Journal*, Vol. 46, No. 2:498–507, 2008.
6. W.L. Oberkampf and M.F. Barone. Measures of agreement between computation and experiment: validation metrics. *Journal of Computational Physics*, 217:5–36, 2006.
7. M. P. Rumpfkeil and D. J. Mavriplis. Efficient Hessian Calculations using Automatic Differentiation and the Adjoint Method. AIAA Paper, 2010-1268, 2010.
8. L. L. Sherman, A. C. Taylor III, L. L. Green, and P. A. Newman. First- and second-order aerodynamic sensitivity derivatives via automatic differentiation with incremental iterative methods. *Journal of Computational Physics*, 129:307 – 331, 1996.
9. W. Yamazaki, M. P. Rumpfkeil, and D. J. Mavriplis. Design Optimization Utilizing Gradient/Hessian Enhanced Surrogate Model. AIAA Paper, 2010-4363, 2010.