10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference
30 August -1 September 2004, Albany, New York

AIAA 2004-4451

# PERSPECTIVES ON OPTIMIZATION UNDER UNCERTAINTY: ALGORITHMS AND APPLICATIONS

Anthony A. Giunta*
*Validation and Uncertainty Quantification Dept.*

Michael S. Eldred*, Laura P. Swiler, Timothy G. Trucano
*Optimization and Uncertainty Estimation Dept.*

Steven F. Wojtkiewicz, Jr.†
*Structural Dynamics and Smart Materials Dept.*

*Sandia National Laboratories‡, Albuquerque, NM 87185 USA*

## Abstract

This paper provides an overview of several approaches to formulating and solving optimization under uncertainty (OUU) engineering design problems. In addition, the topic of high-performance computing and OUU is addressed, with a discussion of the coarse- and fine-grained parallel computing opportunities in the various OUU problem formulations. The OUU approaches covered here are: sampling-based OUU, surrogate model-based OUU, analytic reliability-based OUU (also known as reliability-based design optimization), polynomial chaos-based OUU, and stochastic perturbation-based OUU.

## 1. Introduction

### 1.1 Overview

One of the criticisms of the traditional engineering design optimization process is that it often produces optimal designs that are not robust to perturbations in the design parameters or other uncertain parameters (e.g., manufacturing tolerances, atmospheric conditions, etc). An example of such an occurrence is shown in Figure 1 (Giunta, 2002a) where the objective function is highly sensitive to perturbations in design parameter *x2*. In this application, the optimization process found the design parameter values that maximized the objective function, but the resulting optimal design was not acceptable from an engineering standpoint, i.e., the optimal design was not robust.

While there are some post-optimality criteria that provide insight into the sensitivity of an optimal design to parameter perturbations (e.g., gradients and Hessians), these criteria only provide a local measure of sensitivity at the optimal design point. In many engineering system design applications, broader measures of objective and constraint function sensitivity are often needed. Statistical measures, such as mean value, standard deviation, and probability of failure can provide such information. These statistical measures are available to the engineer through the various types of uncertainty quantification and reliability estimation software tools that are now available. Thus, it is a natural extension for engineers to incorporate statistical measures directly into the design optimization process. This has led to areas of study known as robust design, reliability-based design optimization, and optimization under uncertainty, among others. We have elected to use the optimization under uncertainty designation, as it reflects our interests in both design for robustness, which focuses on optimizing the mean response of a system, and design for reliability, which focuses on optimizing the extreme response of a system.

The optimization under uncertainty approaches described in this paper combine optimization methods with uncertainty quantification and reliability estimation methods. As one might expect given the computa-

* Senior Member, AIAA
† Member, AIAA
‡ Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

tional burden that is created both in optimization alone and in uncertainty quantification alone when applied to realistic engineering applications, OUU becomes computationally expensive for all but the most trivial of problems (Trosset, 2000). For this reason, there is much research underway that is aimed at decreasing the computational expense of solving OUU problems. This paper provides an overview of some of these research activities, along with a discussion of the opportunities for both coarse- and fine-grained parallel computing within OUU methods. Clearly, parallel computing must be exploited if OUU methods are to have an impact on simulation-based engineering design activities.

## 2. Background

Optimization under uncertainty is distinct from the field of stochastic programming (cf., Birge, 1997), since OUU problem formulations typically lack the specific mathematical structure often found in stochastic programming (e.g., linear objective and constraint functions, future time recourse). In some sense, OUU can be thought of as nonlinear programming under uncertainty. That is, OUU problems often involve nonlinear implicit objective and constraint functions, where in a function evaluation may require the execution of a computationally expensive “black box” engineering simulation code. However, OUU problem formulations differ from traditional nonlinear programming in that OUU problems contain non-deterministic parameters as well as statistical measures in the objective function and/or constraint specifications. Since these statistical measures often cannot be integrated analytically due to the nonlinear, implicit nature of the response functions, they must be approximated numerically.

Note that this paper focuses on optimization problem formulations that contain uncertain parameters which are characterized with statistical probability distributions, i.e., aleatoric uncertainties. This paper does not cover optimization problem formulations that contain non-probabilistic (e.g., epistemic, fuzzy or possibilistic) uncertain parameters. Ferson, et al., (2004) provides an overview of the state of the art in epistemic uncertainty propagation methods. The topic of optimization under non-probabilistic uncertainty is discussed in other sources (cf., Maglaras, et al., 1997).

A notional deterministic optimization process is shown in Figure 2, where the objective function and constraint function evaluations obtain data through an execution of one or more simulation codes. A notional OUU process is shown in Figure 3, in which the objective/constraint function evaluation requires the execution of an uncertainty quantification (UQ) procedure. In this case, the UQ procedure could be one of the many variants of Monte-Carlo sampling, or it could be one of the several types of analytic reliability estimation methods, or other uncertainty propagation approaches. The use of a UQ procedure to estimate statistical metrics within the optimization process often makes the OUU process computationally expensive. Hence, solving OUU problems often requires the use of parallel computing, both within the simulation code where possible and also within the UQ and optimization methods. This report describes some of the current approaches to OUU that are under investigation, along with the opportunities for parallel computing in each approach.

### 2.1 OUU and High Performance Computing

The U.S. Department of Energy (DOE) currently operates numerous multi-teraflop (1 teraflop = 1 trillion floating point operations per second) computing assets at various laboratory sites. As of mid-2004, several of these assets have peak performance levels in the tens of teraflops. Near- and long-term DOE plans call for continued computing performance improvements in order to reach the 100 teraflop level and eventually the 1,000 teraflop (1 petaflop) level. As an example, currently under construction at Sandia National Laboratories is a computer with 11,648 processors that should, by January 2005, perform calculations at a peak rate of 41.5 teraflops. This computer will be upgraded to achieve 100 teraflops in the 2005-2006 timeframe (Sandia, 2004).

Historically, DOE computing needs have been based on a “capability model,” that is, the ability to execute a single simulation code job that performs fine-grained massively parallel calculations on hundreds or thousands of processors for days or weeks of run time. More recently, however, is a move within DOE to procure supercomputers that also support a “capacity model” of computing, that is, the ability to execute concurrently a large number of serial and/or small-scale parallel simulation code jobs. Typically, these jobs run on tens or hundreds of processors for minutes or hours of run time. Current computational resources at the DOE laboratory sites are a mixture of capability computing machines and capacity computing machines, with some machines serving both roles.

When optimization, UQ, and OUU methods are applied to real-world engineering applications, one often encounters computational bottlenecks unless both capacity and capability computing resources can be employed in the solution process. That is, coarse-grained parallel computing is exploited for aspects of the optimization/UQ/OUU methods that have inherent independence, and fine-grained parallel computing is exploited in solving the large system of state equations that occurs in many physics-based simulations. For example, coarse-grained computing could be used in an uncertainty quantification study that employs Monte Carlo sampling, since all of the simulation jobs are independent and could be executed concurrently. Similarly, in a gradient-based optimization study there is either $n+1$ (forward difference) or $2n+1$ (central difference) independent simulation code jobs when estimating the terms in an $n$-dimensional gradient vector via finite difference methods. Fine-grained parallel computing is needed when the storage requirements for the state equations exceeds the memory available on a single computer processor. Hence, the system of state equations must be decomposed and solved in a distributed manner across multiple processors.

The DAKOTA toolkit developed at Sandia National Laboratories (Eldred, et al., 2002a) has been extensively used for both routine and large-scale massively parallel optimization studies and UQ studies (Eldred, et al., 2002a). In the past several years, DAKOTA's methods for optimization and UQ have been coupled to enable optimization under uncertainty studies. One of the key attributes of the DAKOTA toolkit that enables its use on high-performance computers is its capability for "multilevel parallelism" (Eldred et al., 2000). In multilevel parallelism, the user exploits opportunities for parallel execution both within a physics simulation code and within an optimization algorithm or an uncertainty quantification method. For example, two-level parallelism is accomplished via concurrent execution of simulation code jobs that are needed for a finite difference gradient estimate (coarse-grained parallelism), with each simulation code job running on multiple processors (fine-grained parallelism). Three-level parallelism is accomplished if, for example, multiple independent simulation codes are needed to compute the objective function. Four-level parallelism is accomplished if, for example, multiple optimization sub problems must be solved within an overall optimization strategy, as occurs in solving mixed integer nonlinear programming problems using a branch-and-bound algorithm (Eldred, et al., 2002a).

While there has been significant recent interest in OUU algorithms and methods, there has been relatively little discussion of the opportunities for parallel computing in the different OUU approaches. Since parallel computing can have a significant impact on the utility of an optimization or an uncertainty quantification approach, some consideration of the impact of parallel computing on OUU is warranted. The text below provides a description of various approaches to OUU, as well as a discussion of how parallel computing can be applied to each approach in order to exploit opportunities for both coarse- and fine-grained parallel computing.

## 3. Description of OUU Methods

In this paper, OUU methods are separated into different categories based on the uncertainty quantification approach that is employed. These categories are: sampling-based methods, surrogate modeling methods, analytic reliability methods, and polynomial chaos methods. Provided below is a limited technical description of each category of OUU method, along with a discussion of the opportunities for parallel computing in that category of OUU method.

For discussion purposes, we formulate a notional OUU problem as:

$$\text{minimize: } f(d,u), \qquad (1)$$

where $f(\ )$ is the objective function, $d$ is the vector of design parameters, and $u$ is the vector of probabilistic uncertain parameters. In this formulation, the design parameters are those that can be controlled by the user, whereas the uncertain parameters are outside the control of the user. If the mean response of the objective function is of interest, then a robust design OUU formulation is:

$$\text{minimize: } E[f(d,u)], \qquad (2)$$

where E[ ] denotes the expected value operator. If the extreme response (i.e., low probability response) of the objective function is of interest, then a reliability-based design OUU formulation is:

$$\text{minimize: } \mathrm{Prob}[f(d,u) > f^*], \qquad (3)$$

where Prob[ ] denotes an estimate of the probability that the objective function exceeds a critical threshold, $f^*$. The methods discussed below can be applied to both the robust design and reliability design variants of OUU.

## 3.1 OUU with Sampling Methods

Figure 5 illustrates the sampling-based OUU approach in which the uncertainty quantification process is performed via Monte Carlo sampling or one of its variants (Giunta, et al., 2002). This is the simplest approach to OUU, in that the UQ sampling process is repeated over the uncertain variables for each vector of design parameters generated by the optimization algorithm. While this approach is attractive due to its simplicity, it has numerous pitfalls. For example, a large number of samples must be taken in each UQ evaluation in order to resolve low probability events. Given that the optimization algorithm may generate hundreds or thousands of trial design variable vectors, the computational expense of this approach can quickly become intractable.

Coarse-grained parallel computing can address some of the computational expense of this approach, since all of the objective function evaluations in each UQ process are independent and can be executed concurrently. In addition, there is opportunity for coarse-grained parallel computing in the optimization algorithm such as in finite difference gradient evaluations, line search points, genetic algorithm population members, or pattern search points.

## 3.2 OUU with Surrogate Models and Sampling Methods

Figure 4 shows the surrogate-based approach to OUU. In this approach, sampling is performed on some type of surrogate model, where the surrogate model is a low-cost alternative to the original simulation code. There is considerable flexibility in the form of the surrogate model. For instance, it can be a mathematical model that is created by fitting a multidimensional surface function (e.g., low-order polynomials, neural networks, radial basis functions, etc.) to a set of simulation code data points (cf., Giunta, et al., 2004). Alternatively, the surrogate model can be physics-based if it in some way neglects some of the detail or complexity that is modeled in the original simulation code (e.g., the surrogate can be a coarse finite element model of the original simulation finite element model).

The key aspect of the surrogate-based OUU approach is that use of a surrogate model trades accuracy for efficiency in the UQ portion of the OUU algorithm. That is, the surrogate model is cheap to evaluate but it is less accurate than the original model. In previous studies by Eldred, et al. (2002b), it was found that a trust-region optimization algorithm was needed to prevent the optimizer from exploiting inaccuracies of the surrogate model. On a series of test problems investigated in this study, surrogate-based OUU approach provided a 2x-10x savings in the number of function evaluations versus a sampling-based OUU approach.

Coarse-grained parallel computing opportunities in the surrogate-based OUU approach depend on the form of the surrogate model. For surface fit surrogate models, relatively few samples are needed from the original simulation code (e.g., $O(n$ to $n^2)$ samples for an $n$-dimensional vector $d$). Once the surface fit surrogate model is constructed, the UQ samples usually can be evaluated at virtually no cost, so parallel execution of the UQ samples is usually not an issue. For physics-based surrogate models, the cost of the UQ samples may or may not be negligible. Thus, the user must decide if coarse-grained parallel execution of the samples is warranted.

## 3.3 OUU with Analytic Reliability Methods

Figure 6 depicts the use of analytic reliability methods (Haldar and Mahadevan, 2000) within OUU. The acronyms in the figure are: MV - mean value method, AVM - advanced mean value method, AMV+ - iterated advanced mean value methods, FORM - first-order reliability method, and SORM - second-order reliability method. The primary distinguishing feature of these UQ methods versus the sampling-based UQ methods is that the analytic reliability methods employ either first- or second-order approximations of the limit state function. Thus, the statistical metrics needed for the objective/constraint function values are computed analytically from the approximate state function, i.e., sampling of the state function is not used. With the exception of the MV method, all of the analytic reliability methods employ an iterative process to locate the most-probable-point (MPP) on the limit state function. This requires an optimization sub problem to be solved each time the UQ portion of Figure 6 is evaluated. For additional information on the various forms of the analytic reliability methods and reliability-based design optimization (RBDO) examples, see the work of Eldred, et al. (2004b).

The analytic reliability OUU method (a.k.a. RBDO) offers opportunities for coarse-grained parallel computing in both the construction of the analytic approximation of the limit state function and in the solution of the MPP optimization sub problem. For example, if the optimizer generates *k* trial design parameter vectors, $d_1,\ldots,d_k$, then the gradient terms $\partial f / \partial d$ can be evaluated via finite differences in parallel, and the subsequent *k* MPP searches can be performed in parallel.

## 3.4 OUU with Polynomial Chaos Methods

Figure 7 illustrates the use of polynomial chaos expansion (PCE) methods (Ghanem and Spanos, 1991; Ghanem and Red-Horse, 1999; Walters, 2003) in the OUU process. This is a fundamentally different approach than either the sampling-based or analytic reliability OUU methods. In the PCE-based approach to OUU, the statistical properties of the uncertain parameters, *u*, are decomposed using spectral expansion methods, in a manner analogous to the use of eigenvalues and eigenvectors to decompose the displacements and vibration modes of a structural system. The PCE decomposition process generates a large system of coupled state equations. When this system is solved, the resulting data can be aggregated, using the PCE basis functions, to produce statistical metrics on the response function.

For example, consider a system of state equations of the form $Ax=b$, where the *x*-vector is unknown, and where the uncertain parameters, *u*, only affect the coefficients in *A*. If the PCE approach employs *m* basis functions, then a single large-scale linear system can be formed, $A'x'=b'$, where $A'$ is an *m*-by-*m* block matrix that contains the terms generated by the PCE method. Once this system is solved, the terms in the solution vector, $x'$, and the PCE basis functions are used to approximate the statistical metrics that are needed as objective/constraint data in the OUU problem.

The opportunity for parallel computing with the PCE-based OUU occurs in solving the large-scale system of coupled state equations. In this case, fine-grained parallel computing is necessary to solve $A'x'=b'$, since $A'$ is likely to be extremely large ($O(10^5 - 10^7)$ terms) and sparse. Note that this PCE-based approach to OUU is the only approach, beyond that of a parallel simulation, for which fine-grained parallel computing can be exploited within the UQ process.

## 3.5 Other OUU Methods of Interest

An OUU method that merits discussion here, but one with which the authors have little experience, is the stochastic search and optimization methods of Spall (2003). These methods are specifically intended for use in design-for-robustness applications (see Equation 2). Interestingly, this OUU approach has the structure of a deterministic optimization approach (Figure 2) since it does not include a specific procedure for uncertainty quantification. The key aspects of Spall's OUU approach is the use of *stochastic* finite-difference approximations in which the *n* gradient terms for the objective function, $\partial f / \partial d$, are computed by employing a random vector of design parameter perturbations, $\Delta d$. In Spall's simultaneous perturbation stochastic approximation (SPSA) optimization method, $\partial f / \partial d$ is estimated via finite differences using <u>two</u> objective function evaluations. Following the notation used in this paper (and omitting the "gain" terms shown in Spall's text), $\partial f / \partial d$ is estimated as:

$$\frac{\partial f}{\partial d} \approx \begin{bmatrix} \dfrac{f(d+\Delta d,u)-f(d-\Delta d,u)}{2\Delta d_1} \\ \vdots \\ \dfrac{f(d+\Delta d,u)-f(d-\Delta d,u)}{2\Delta d_n} \end{bmatrix} \tag{4}$$

where *u* is a randomly selected vector of uncertain parameters that is different in each of the $f(d+\Delta d, u)$ and $f(d-\Delta d, u)$ evaluations. In performing this approximation, the SPSA approach makes a tradeoff between gradient accuracy and optimization method convergence. That is, the SPSA optimization algorithm exhibits convergence only in the limit as the number of optimization algorithm iterations goes to infinity. A comparison of the SPSA approach versus some of the other OUU approaches described in this paper is planned in our future work.

The SPSA optimization algorithm is primarily a serial process, as there is not a line search or trust region search phase in the algorithm that would provide an opportunity to exploit coarse-grained parallel computing. However, coarse-grained parallel computing can be employed in evaluating the two function needed to estimate the SPSA gradient. The finite-difference stochastic approximation (FDSA) algorithm, which is a variant of SPSA, is more amenable to coarse-grained parallel computing since FDSA gradient estimates require *2n* function evaluations.

## 4. Summary

This paper has provided an overview of several approaches to optimization under uncertainty, along with a description of the parallel computing opportunities that are afforded in each OUU approach. Due to the computational expense of coupling non-deterministic uncertainty quantification methods with nonlinear optimization algorithm and high-fidelity physics-based simulations, high-performance parallel computing assets must be employed when OUU methods applied to engineering design activities.

## Acknowledgements

The authors wish to thank the participants of the Optimization Under Uncertainty Workshop held at Sandia National Labor oratories in January 2004. The presentations and discussions held at the workshop contributed to the content of this paper. In particular, the authors wish to acknowledge contributions of Prof. Roger Ghanem (Johns Hopkins University), Prof. Takeru Igusa (Johns Hopkins University), Prof. Kurt Maute (University of Colorado), Dr. Victor Perez (General Electric Aircraft Engines), Dr. Johannes Royset (Naval Postgraduate School), Prof. Christine Shoemaker (Cornell University), and Prof. Michael Trosset (College of William and Mary).

## References

Birge, J.R, and Louveaux, F., *Introduction to Stochastic Programming*, Springer-Verlag, 1997.

Eldred, M.S., Hart, W.E., Schimel, B.D., and van Bloemen Waanders, B.G., "Multilevel Parallelism for Optimization on MP Computers: Theory and Experiment," paper AIAA-2000-4818 in Proceedings of the 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, CA, September 6-8, 2000.

Eldred, M.S., Giunta, A.A., van Bloemen Waanders, B.G., Wojtkiewicz, S.F., Jr., Hart, W.E., and Alleva, M.P., "DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis. Version 3.0 Users Manual." Sandia Technical Report SAND2001-3796, April 2002(a). (Updated April 2003, Version 3.1)

Eldred, M.S., Giunta, A.A., Wojtkiewicz, S.F., Jr., and Trucano, T.G., "Formulations for Surrogate-Based Optimization Under Uncertainty," paper AIAA-2002-5585 in Proceedings of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, GA, Sept. 4-6, 2002(b).

Eldred, M.S., Giunta, A.A., and van Bloemen Waanders, B.G., "Multilevel Parallel Optimization Using Massively Parallel Structural Dynamics," *Structural and Multidisciplinary Optimization*, Springer-Verlag, Vol. 27, Nos. 1-2, pp. 97-109, 2004(a).

Eldred, M.S., Agarwal, H., Perez, V.M., Wojtkiewicz, S.F., Jr., and Renaud, J.E., "Investigation of Reliability Method Formulations in DAKOTA/UQ," in Proceedings of the 9th ASCE Joint Specialty Conference on Probabilistic Mechanics and Structural Reliability, Albuquerque, NM, July 26-28, 2004(b).

Ferson, S., Joslyn, C.A., Helton, J.C., Oberkampf, W.L, and Sentz, K., “Summary from the Epistemic Uncertainty Workshop: Consensus amid Diversity,” Sandia Technical Report SAND2004-1061J, Feb. 2004. (to appear: *Reliability Engineering and System Safety*)

Giunta, A. A., "Use of Data Sampling, Surrogate Models, and Numerical Optimization in Engineering Design," paper AIAA-2002-0538 in Proceedings of the 40th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, Jan. 2002(a).

Giunta, A.A., Eldred, M.S., Trucano, T.G., and Wojtkiewicz, S.F., Jr., "Optimization Under Uncertainty Methods for Computational Shock Physics Applications," paper AIAA-2002-1642 in Proceedings of the 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference (Nondeterministic Approaches Forum), Denver, CO, April 22-25, 2002(b).

Giunta, A.A., Eldred, M.S., and Castro, J.P., "Uncertainty Quantification Using Response Surface Approximations," in Proceedings of the 9th ASCE Joint Specialty Conference on Probabilistic Mechanics and Structural Reliability, Albuquerque, NM, July 26-28, 2004.

Ghanem, R., and Spanos, P., *Stochastic Finite Elements: A Spectral Approach*, Springer Verlag, 1991. (reissued by Dover Publications, 2004.)

Ghanem, R., and Red-Horse, J., “Propagation of Probabilistic Uncertainty in Complex Physical Systems Using a Stochastic Finite Element Approach,” *Physica D*, 133, pp. 137-144, 1999.

Haldar, A., and Mahadevan, S, *Probability, Reliability and Statistical Methods in Engineering Design*, Wiley, 2000.

Maglaras, G., Nikolaidis, E., Haftka, R.T., and Cudney, H.H., ``Analytical-Experimental Comparison of Probabilistic Methods and Fuzzy Set based Methods for Designing Under Uncertainty,” *Structural Optimization*, 13(2/3), pp. 69-80, 1997.

Sandia National Laboratories, News Release, July 2004. (http://www.sandia.gov/news-center/news-releases/2004/comp-soft-math/redstormrising.html)

Spall, J.C. *Introduction to Stochastic Search and Optimization*, John Wiley & Sons, Hoboken, NJ, 2003.

Trosset, M.W., “On the Use of Direct Search Methods for Stochastic Optimization,” Technical Report 00-20, Department of Computational & Applied Mathematics, Rice University, 2000.

Walters, R.W., “Towards Stochastic Fluid Mechanics via Polynomial Chaos,” paper AIAA-2003-0413 in Proceedings of the 41st AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, Jan. 6-9, 2003.

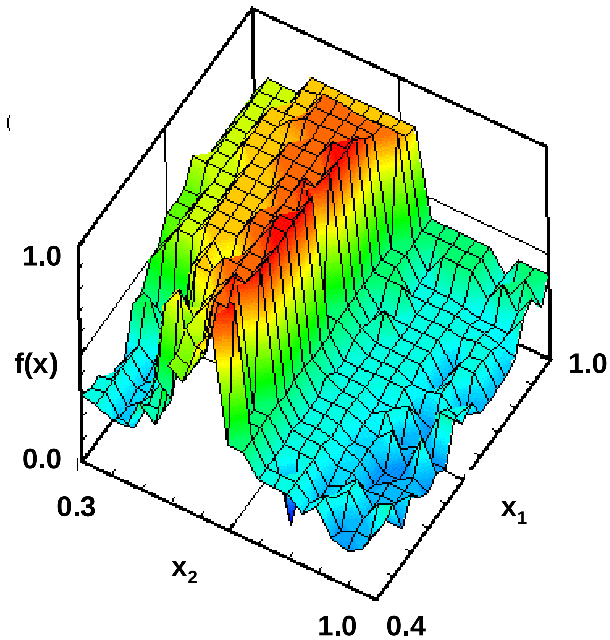


Figure 1. Parameter study data from Giunta (2002a), showing a steep-walled cliff adjacent to the maximum values (dark/red region) of the objective function, *f(x1,x2)*. The global maxima are not robust to perturbations in *x2*.

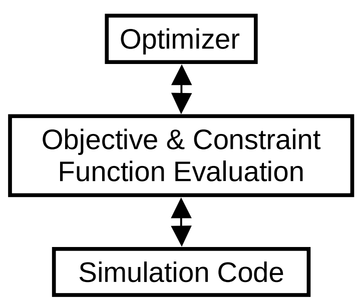


Figure 2. A notional optimization loop between an optimization algorithm and a simulation code.

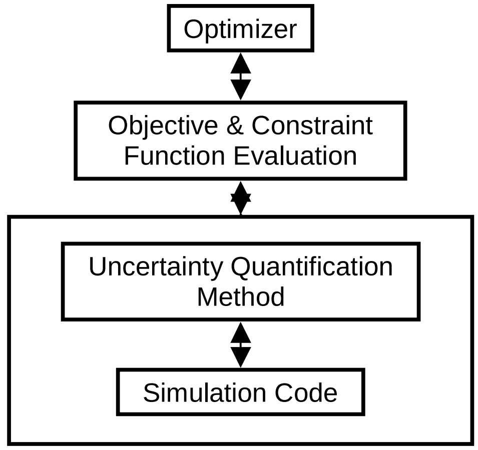


Figure 3. A notional optimization under uncertainty problem formulation, with an uncertainty quantification (UQ) method embedded inside an optimization loop.

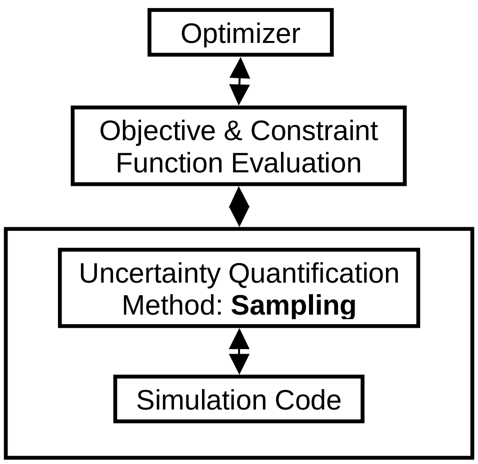


Figure 4. Optimization under uncertainty with a sampling-based method for uncertainty quantification.

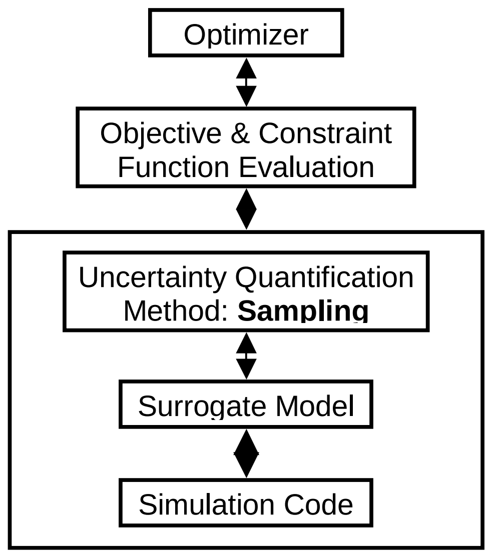


Figure 5. Optimization under uncertainty with a surrogate-based sampling-based method for uncertainty quantification.

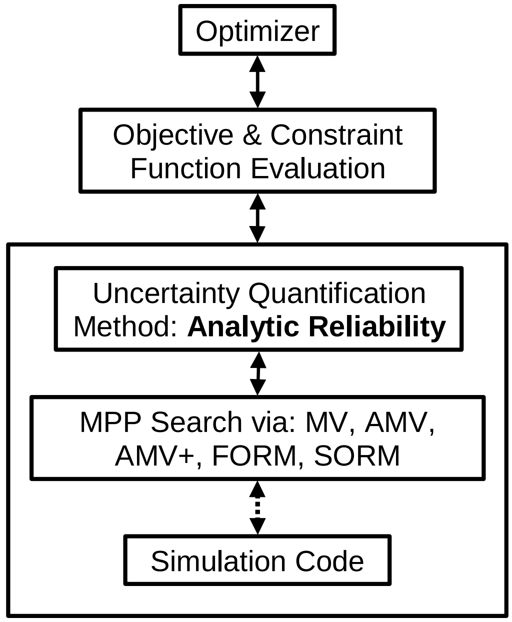

Figure 6. Optimization under uncertainty with an analytic reliability method for uncertainty quantification.

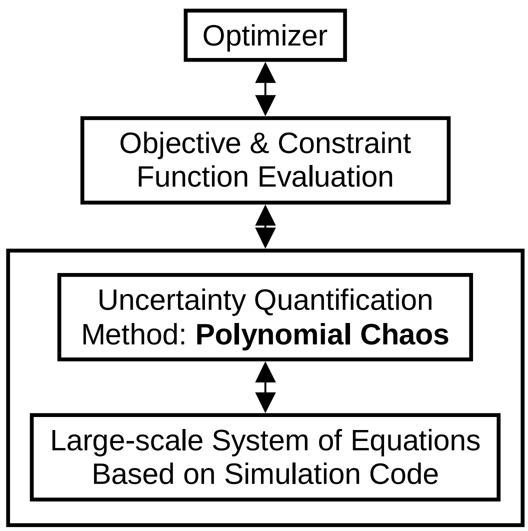

Figure 7. Optimization under uncertainty with a polynomial chaos representation of uncertainty.