AIAA-2002-5585

FORMULATIONS FOR SURROGATE-BASED OPTIMIZATION UNDER UNCERTAINTY

M. S. Eldred*, A. A. Giunta†, S. F. Wojtkiewicz, Jr.‡, and T. G. Trucano§
Sandia National Laboratories¶
Albuquerque, NM 87185

Abstract

In this paper, several formulations for optimization under uncertainty are presented. In addition to the direct nesting of uncertainty quantification within optimization, formulations are presented for surrogate-based optimization under uncertainty in which the surrogate model appears at the optimization level, at the uncertainty quantification level, or at both levels. These surrogate models encompass both data fit and hierarchical surrogates. The DAKOTA software framework is used to provide the foundation for prototyping and initial benchmarking of these formulations. A critical component is the extension of algorithmic techniques for deterministic surrogate-based optimization to these surrogate-based optimization under uncertainty formulations. This involves the use of sequential trust region-based approaches to manage the extent of the approximations and verify the approximate optima. Two analytic test problems and one engineering problem are solved using the different methodologies in order to compare their relative merits. Results show that surrogate-based optimization under uncertainty formulations show promise both in reducing the number of function evaluations required and in mitigating the effects of non-smooth response variations.

Introduction

Many optimization problems must be performed in the presence of inherent variability (aleatory/irreducible uncertainty) or uncertainty resulting from a lack of knowledge (epistemic/reducible uncertainty). The challenge is to efficiently and reliably include these uncertainties in the optimization procedures, so that prescribed robustness or reliability can be achieved in the optimal designs.

To perform optimization under uncertainty (OUU), optimization techniques must be combined with statistical uncertainty quantification (UQ) techniques. Depending upon the specifics of this combination, OUU formulations support both "Design for Robustness" and "Design for Reliability." The former is generally regarded as a simpler problem, since the mean performance is of interest (for which statistics are less costly to compute and more reliable for small sample sizes). In this case, some optimization formulations presented in the literature neglect UQ entirely, instead relying on local derivatives to assess robustness. The authors do not advocate this approach, as a local derivative is generally an insufficient metric of robustness. In the latter case of design for reliability, performance statistics in the tails of the distributions are of interest. These statistics are more difficult to obtain accurately, placing greater demands on the UQ portion of the OUU study.

When combining optimization and UQ techniques to perform OUU, the most direct approach is to nest the iterative loops by performing a complete uncertainty estimation (inner loop) for each optimization data request (outer loop). This can be prohibitively expensive, especially if sampling techniques are used for the uncertainty estimation. For this reason, techniques which can break this nested relationship and reduce the overall expense are highly desirable. A variety of such approaches exist and will be categorized into three areas: sampling-based, analytic reliability-based, and stochastic finite element-based.

Sampling-based OUU

When using sampling approaches to perform uncertainty estimation within OUU, surrogate modeling techniques naturally arise out of the desires to reduce the computational expense associated with sampling and to smooth out any noise in the variations of the statistics generated from sampling. Possible formulations for surrogate-based optimization under uncertainty (SBOUU) can include approaches where

---

*Principal Member of Technical Staff, Optimization and Uncertainty Estimation Dept., Mail Stop 0847, AIAA senior member.

†Senior Member of Technical Staff, Optimization and Uncertainty Estimation Dept., Mail Stop 0847, AIAA senior member.

‡Senior Member of Technical Staff, Structural Dynamics Research Dept., Mail Stop 0847, AIAA member.

§Distinguished Member of Technical Staff, Optimization and Uncertainty Estimation Dept., Mail Stop 0819.

¶Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

the surrogate model appears at the optimization level, at the uncertainty quantification level, or at both levels. These surrogate models encompass both data fit surrogates (e.g., a polynomial response surface) and hierarchical surrogates (e.g., an Euler CFD model used in place of a Navier-Stokes CFD model). An important direction of research is the extension of algorithmic techniques for deterministic surrogate-based optimization to these SBOUU formulations. This involves the use of sequential trust region-based approaches to manage the extent of the approximations and verify the approximate optima. Whereas a predicted optimum from an approximate optimization cycle can be easily verified in deterministic surrogate-based optimization, the introduction of statistical quantities in SBOUU makes a rigorous verification in the absolute sense unobtainable in general. What is needed to be able to achieve provable-convergence is the ability to verify improvements from a trust region cycle in a relative sense, using adaptive UQ methods which allow a statistical verification based on confidence bounds.

A related approach to these trust-region based SBOUU methods is the stochastic approximation approach [1]. Rather than performing a rigorous verification on each optimization cycle, these methods allow some erroneous trust region steps. By analyzing the statistical form of the approximation errors and bounding the number of erroneous steps, convergence of these methods can still be verified using probabilistic arguments.

Another approach for OUU with sampling-based UQ involves the use of nongradient-based optimization techniques, e.g., pattern search [2]. The motivation for nongradient-based techniques is similar to that of surrogate-based approaches, i.e., they tend to be more tolerant of noise in the variations of the statistics generated from sampling.

A final sampling-based concept targeted at addressing OUU expense involves the blurring of boundaries between optimization and uncertainty quantification (UQ) through the leveraging of the same simulation data by both components of the OUU process [3]. For example, certain types of optimization algorithms (e.g., genetic algorithms) may be able to utilize sampling data from the uncertainty analysis to accelerate the optimization search (assuming some crossover between design and uncertain variables).

Exploiting analytic reliability structure

Another approach is to exploit the structure of analytic reliability techniques such as AMV/AMV+/FORM/SORM [4]. This can involve accelerating a gradient-based optimization through reuse of the gradient of the reliability index, use of a “cross-iterated” approach [5], or use of a combined optimization/most probable point (MPP) search [6]. The former approach simply makes use of gradient data available from the MPP search at the UQ level to avoid having to finite difference at the optimization level. In the cross-iterated approach, the nested relationship is broken by iterating back and forth between deterministic design and analytic reliability-based UQ. Based on statistics from the UQ analysis, the safety factors on the deterministic design constraints can be adjusted in order to exchange performance for higher reliability, and the iterative process is continued until the desired reliability is obtained. In the combined optimization/MPP search approach, analytic relationships between the MPP and the design variables are exploited to form a single-level combined optimization problem.

Each of these approaches can provide an efficient means for solving OUU problems. The weakness is that each of the analytic reliability UQ methods employs a set of approximating assumptions. Thus, it is essential that the validity of these assumptions for a particular problem domain be understood before employing these techniques.

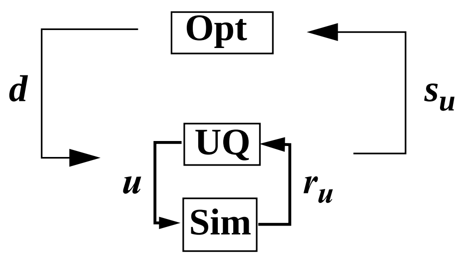
SFE/SAND

Finally, it is anticipated that the combination of simultaneous analysis and design (SAND) methods [7] and stochastic finite element (SFE) techniques [8] will be fruitful. These two techniques use intrusive approaches to couple design optimization and stochastic analysis, respectively, with simulation. In the SFE approach for nonlinear systems, a set of coupled, block-structured equations is created for each design point, where each block is the size of the original deterministic system. Applying these equations as equality constraints in SAND optimization formulations results in very large-scale optimization problems. Stochastic objectives and constraints can either be formulated from statistics generated from the polynomial chaos representations or from the polynomial chaos coefficients themselves [9]. Developing the capability to converge the SFE and design optimality conditions simultaneously on large parallel computers is a future direction that could result in considerable efficiency gains for OUU problems.

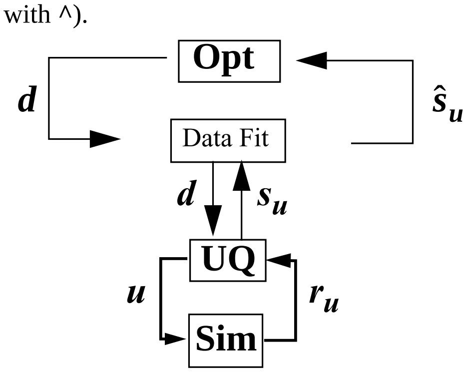
OUU Formulations

Four optimization under uncertainty (OUU) approaches are of primary interest in this paper. In addition to the direct nesting formulation, surrogate-based approaches are presented where the surrogate model appears at the optimization level, at the uncertainty quantification level, or at both levels. These surrogate models encompass both data fit surrogates and hierarchical surrogates. Each of these formulations is equally applicable to uncertainty of optima (UOO) problems, which involve an inversion of the nested loops is order to compute statistical data on optimal solutions. For clarity, only the OUU approaches are presented and the “optional interface” portions (see Implementation section) are omitted.

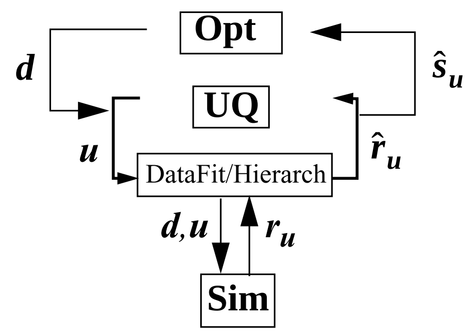
Formulation 1: Nested. Optimization is performed directly on the UQ results, and the UQ analyses are performed directly on the simulation code. This involves use of a nested model (see Implementation section) without any surrogates where $\boldsymbol{d}$ are the design variables, $\boldsymbol{u}$ are the uncertain variables characterized by probability distributions, $\boldsymbol{r_u}(\boldsymbol{d},\boldsymbol{u})$ are the response functions from the simulation, and $\boldsymbol{s_u}(\boldsymbol{d})$ are the statistics generated from the uncertainty quantification on these response functions.



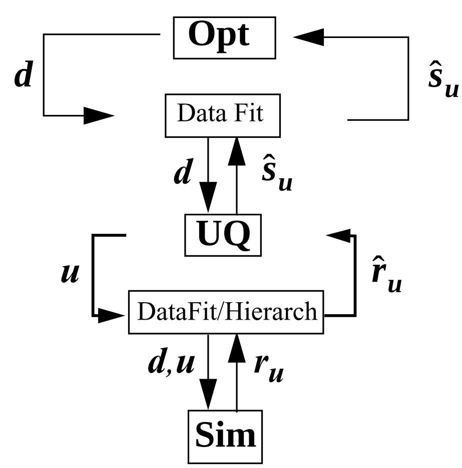
Formulation 2: Layered/Nested. Optimization is performed on a data fit surrogate (e.g., polynomial response surface) which fits statistical data $\boldsymbol{s_u}$ generated from a set of UQ analyses performed over the range of $\boldsymbol{d}$ (approximate quantities marked with ^).



Formulation 3: Nested/Layered. Optimization is performed directly on the UQ results, and UQ analyses are performed on either a data fit surrogate built over $\boldsymbol{d}$ and $\boldsymbol{u}$ or a hierarchical surrogate [10] for which a correction is applied at $\boldsymbol{d}$ and mean $\boldsymbol{u}$. Note that there are several possibilities for the data used to build data fit surrogates at the UQ level. The data fit can be performed:
(1) once covering the full range of $\boldsymbol{d}$ and $\boldsymbol{u}$, or
(2) over the range of $\boldsymbol{u}$ only for each $\boldsymbol{d}$ instance.
In the latter case, there may be a reduction in the total number of samples from not populating the full interactions. In addition, the form of the fit could be influenced; that is, breaking $\boldsymbol{u}$ and $\boldsymbol{d}$ apart could simplify the form of the surrogate model over $\boldsymbol{u}$, e.g., a quadratic model over $\boldsymbol{u}$ might be sufficient if decoupled from $\boldsymbol{d}$, whereas a higher order model would be best if fit over both $\boldsymbol{u}$ and $\boldsymbol{d}$ [2].



Formulation 4: Layered/Nested/Layered. Optimization is performed on a data fit surrogate which fits statistical data $\boldsymbol{s_u}$ generated from a set of UQ analyses performed over the range of $\boldsymbol{d}$. UQ analyses are performed on either a data fit surrogate built over $\boldsymbol{d}$ and $\boldsymbol{u}$ or a hierarchical surrogate for which a correction is applied at $\boldsymbol{d}$ and mean $\boldsymbol{u}$ (formulations 2 and 3 combined). In this formulation, there is an additional possibility for the data used to build data fit surrogates at the UQ level beyond the two listed in formulation 3:
(3) periodically over $\boldsymbol{u}$ and a restricted range of $\boldsymbol{d}$ for each new design trust region.



For this paper, formulation 3 will employ the first UQ surrogate approach of generating a single data fit covering the full range of $\boldsymbol{d}$ and $\boldsymbol{u}$, and formulation 4 will employ the third UQ surrogate approach of regenerating data fits over $\boldsymbol{u}$ and a restricted range of $\boldsymbol{d}$ for each new design trust region.

Implementation

The DAKOTA toolkit [11], [13], [14] is an open source software framework for systems analysis, encompassing optimization, parameter estimation, uncertainty quantification, design of computer experiments, and sensitivity analysis. It interfaces with a variety of simulation codes from a range of engineering disciplines, and it manages the complexities of a broad suite of capabilities through the use of object-oriented abstraction, class hierarchies, and polymorphism [15]. Through implementation of OUU approaches in DAKOTA, the latest capabilities for

- optimization (e.g., gradient and nongradient, mixed integer, simultaneous analysis and design) [16],
- uncertainty quantification (e.g., sampling, analytic reliability, and stochastic finite element) [17],
- surrogate modeling (e.g., polynomial regression, kriging, neural networks, splines) [18], and
- parallel processing (e.g., multilevel parallelism for massively parallel architectures) [19]

can be brought together to perform SBOUU for engineering applications using complex, high-fidelity simulations on high performance computers.

Supporting Model Class Hierarchy

The **DakotaModel** class hierarchy shown in Figure 1 provides a new set of software components that allow both the nesting of iterative studies and the layering of approximations. The **Single** model class preserves previous functionality for a standard parameter to response mapping through a single interface. The **Nested** model provides a capability for executing a complete iterative study as part of every evaluation

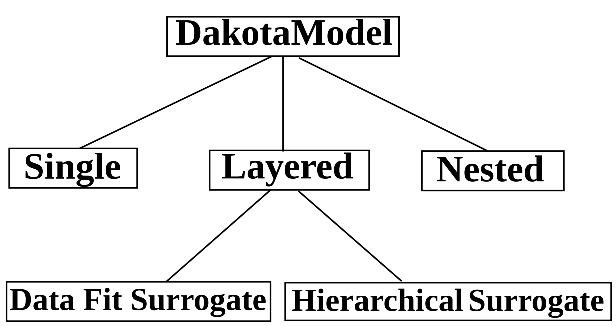


Figure 1. DakotaModel class hierarchy.

performed on this model. Examples of studies which can employ nested models include:

- optimization under uncertainty (a full UQ on every optimization function evaluation)
- uncertainty of optima (a full optimization on every UQ function evaluation)
- Multi-start optimization for global optimization (an optimization nested within stratified sampling)
- Pareto-frontier mapping for multiobjective optimization (an optimization nested within a parameter study)

Components of the **Nested** model include:

- a subordinate iterator/model pair which is used to perform the inner iterative study. An essential feature of the subordinate model is that it is of arbitrary type, i.e., it can be of any of the derived classes shown in Figure 1. Thus, a nested model may contain a layered model, and so on.
- an optional interface for evaluating non-nested portions of the parameter to response mapping. In the case of OUU, this interface provides the deterministic data that is combined with statistical data from the subordinate iterator/model pair (see OUU Mappings).

The next model type is the **Layered** model, which like the **Nested** model, contains internal iterators and models. However, these internal iterators and models are not used on every function evaluation in a nested process; rather, they are used periodically for update and verification. **Layered** model has two derived class implementations. The first of these is the **Data Fit Surrogate** model, which uses an inexpensive, data fit approximation as a surrogate for an actual, expensive model. Examples of data fit approximations include:

- global approximation surrogates (kriging, splines, neural networks, polynomial regression)
- local approximation surrogates (Taylor series: direct, reciprocal, intermediate)
- multipoint approximation surrogates (e.g., two-point adaptive nonlinear approximation (TANA))

Components of the **Data Fit Surrogate** model include:

- an approximation interface which contains the data fit that approximates the actual parameter to response mapping.
- an optional iterator/model pair which is used to perform design and analysis of computer experiments (DACE) in order to generate the data to build the approximation. Again, an essential feature of the DACE model is that it is of arbitrary type, i.e., it can be of any of the derived classes shown in Figure 1. Thus, a layered model may contain a nested model, and so on. These components are optional since a surrogate can be

built with restart data, new data (from DACE), or both. Also, these components are only relevant for global approximations (not local or multipoint).

The second layered model is the **Hierarchical Surrogate** model, which uses a low fidelity model (with correction) as an approximation to a high fidelity model. Examples of hierarchical modeling fidelity [20] include:

- variable fidelity physics (e.g., aerodynamics using panel methods vs. Euler equations vs. Navier-Stokes equations; could be different codes or the same code with different physics options active)
- variable resolution (same code/physics, different mesh)
- variable accuracy (same code/physics, same mesh, different simulation convergence controls)
- reduced basis/order (modal coordinates, proper orthogonal decomposition)

The **Hierarchical Surrogate** model contains:

- a low fidelity interface which provides parameter to response mappings that are much less expensive than would be available directly from the high fidelity model.
- a high fidelity model which is used to provide corrections to the low fidelity approximation as well as verifications of actual improvement. Again, an essential feature of this model is that it is of arbitrary type, so that model classes can be used in recursive fashion.

A future extension to these layered model capabilities would involve the ability to use a data fit surrogate to fit the response ratios between low and high fidelity models [2]. This would combine the **Data Fit Surrogate** and **Hierarchical Surrogate** capabilities.

OUU Mappings

As mentioned previously, **Nested** models support an optional interface in addition to the subordinate iterator/model pair. This permits evaluation of deterministic components (e.g., weight) in addition to stochastic components (e.g., probability of failure) within the top level response computation.

First consider the usual nonlinear programming formulation of

$$\begin{aligned} \text{Minimize} \quad & \boldsymbol{f}(\boldsymbol{d}) \\ \text{Subject to} \quad & \boldsymbol{g_l} \le \boldsymbol{g}(\boldsymbol{d}) \le \boldsymbol{g_u} \\ & \boldsymbol{h}(\boldsymbol{d}) = \boldsymbol{h_t} \\ & \boldsymbol{d_l} \le \boldsymbol{d} \le \boldsymbol{d_u} \end{aligned}$$

where $\boldsymbol{f}$, $\boldsymbol{g}$, and $\boldsymbol{h}$ are deterministic objective functions, inequality constraints, and equality constraints, respectively, defined over the design variables $\boldsymbol{d}$. Extending this formulation for OUU problems, a linear mapping for combining deterministic and stochastic components is:

$$\begin{aligned} \text{Minimize} \quad & \boldsymbol{f}(\boldsymbol{d}) + \boldsymbol{W}\boldsymbol{s_u}(\boldsymbol{d}) \\ \text{Subject to} \quad & \boldsymbol{g_l} \le \boldsymbol{g}(\boldsymbol{d}) \le \boldsymbol{g_u} \\ & \boldsymbol{h}(\boldsymbol{d}) = \boldsymbol{h_t} \\ & \boldsymbol{a_l} \le \boldsymbol{A_i}\boldsymbol{s_u}(\boldsymbol{d}) \le \boldsymbol{a_u} \\ & \boldsymbol{A_e}\boldsymbol{s_u}(\boldsymbol{d}) = \boldsymbol{a_t} \end{aligned}$$

where the optional interface provides $\boldsymbol{f}$, $\boldsymbol{g}$, and $\boldsymbol{h}$, the subordinate iterator/model provides $\boldsymbol{s_u}$ (statistics including mean μ, standard deviation σ, and probability of failure p_{fail}), and the user specifies the constraint bounds and targets $\boldsymbol{g_l}$, $\boldsymbol{g_u}$, $\boldsymbol{h_t}$, $\boldsymbol{a_l}$, $\boldsymbol{a_u}$, and $\boldsymbol{a_t}$ and the coefficient matrices $\boldsymbol{W}$ and $\boldsymbol{A}$ ($\boldsymbol{A_i}$ and $\boldsymbol{A_e}$ are sub-matrices of $\boldsymbol{A}$ for inequality and equality constraints, respectively). Note that multiobjective optimization is supported, and $\boldsymbol{f}$ and $\boldsymbol{W}\boldsymbol{s_u}$ may be of different lengths to accommodate purely deterministic objective functions, purely stochastic objective functions, or any combination. It is currently assumed that there is no need to combine deterministic and stochastic constraints, although this could be easily accommodated if the need arises.

Computational Experiments

Computational experiments have been performed with several test problems in order to compare the proposed OUU formulations. In each of these examples, the UQ analyses are performed using sampling techniques.

Test Problem 1: DAKOTA test functions

This analytic test problem is formulated as follows:

$$\begin{aligned} \text{Minimize} \quad & f + p_{fail\_r1} + p_{fail\_r3} \\ \text{Subject to} \quad & g_i \le 0, \text{ for } i = 1,2,3 \\ & \mu_{r2} + 3\sigma_{r2} \le 1.6e5 \\ & 1.5 \le d_1 \le 2.164 \\ & 0.0 \le d_2 \le 4.0 \\ & \text{normal: } u_1, u_2\text{; uniform: } u_3, u_4\text{; weibull: } u_5, u_6 \end{aligned}$$

where u_1 and u_2 have means of 248.89 and 593.33 and standard deviations of 12.4 and 29.7, respectively; u_3 and u_4 have lower bounds of 199.3 and 474.63 and upper bounds of 298.5 and 712.0, respectively; and u_5 and u_6 have alpha parameters of 12. and 30. and beta parameters of 250. and 590., respectively. All six uncertain variables are independent. The f and g_i are deterministic functions defined by the DAKOTA "cylinder head" analytic regression test problem [12]. After insertion of constants, these functions simplify to:

$$f = -1.8 - \frac{4d_1}{9.165} + 0.15d_2$$

$$g_1 = \frac{1}{1500}\left(1.132 - \frac{d_1}{2}\right)^{-2.5} - 0.5$$

$$g_2 = 0.15d_2 - 0.6$$

$$g_3 = 0.075(4 - d_2)^{1.5} - 0.25$$

Each of the statistics (p_{fail_r1}, p_{fail_r3}, μ_{r2}, and σ_{r2}) are computed from an uncertainty estimation performed on the DAKOTA "textbook OUU" regression test problem [12]. These three response functions are defined as:

$$r_1 = \sum_{i=1}^{3}(u_i - 10d_1)^4 + \sum_{i=4}^{6}(u_i - 10d_2)^4$$

$$r_2 = d_1(u_1^2 - 0.5u_2)$$

$$r_3 = d_2(u_2^2 - 0.5u_1)$$

with the associated failure probabilities defined as $P(r_1 \ge 4.2e11)$ and $P(r_3 \ge 7.5e5)$. These two regression test problems are not naturally related and the combined problem does not have a physical interpretation; rather these functions were chosen to test the overlaying of deterministic and stochastic data in the OUU mappings.

For formulations involving surrogates, quadratic polynomials are used and are built using the minimum amount of data required. For a data fit over 2 design variables, this equates to 6 samples; and for a data fit over 2 design variables and 6 uncertain variables, this equates to 45 samples. Uncertainty estimations are performed using either 50 Latin hypercube samples, in the cases where sampling is interfaced directly with the simulation (formulations 1 and 2), or 5000 Latin hypercube samples, in the cases where sampling is interfaced with the data fit surrogate (formulations 3 and 4). Random number seeds are reused on each sample generation. This is critical, since it guarantees similar/identical stencils of samples between design changes and allows for much smoother variation of statistical quantities with respect to the design parameters.

OUU without trust regions

Table 1 shows results for the four OUU formulations and some basic trends are evident. In general, use of surrogate models was successful for this problem, reducing the expense by up to a factor of 20 from the best nested run using DOT's sequential quadratic programming (SQP) method [21] (NPSOL SQP [22] was much more expensive at 5500 UQ plus 110 optional interface evaluations due to its inability to verify optimality and terminate in the presence of under-resolved p_{fail} statistics). The four solutions are all comparable, with formulations 1 and 3 finding a slightly better solution [$(d_1, d_2) \approx (1.98, 1.77)$] than formulations 2 and 4 [$(d_1, d_2) \approx (1.75, 1.77)$]. This is expected, since a single global data fit using only six samples over the two design variables will have some inaccuracy. Each of the approximate solutions (marked with "*") has been veri-

Table 1: Test problem 1 results for the four OUU formulations.

OUU Formulation	**Function Evaluations**	**Objective Function**	**Constraint Violation**	**Qualitative Comments**
1 (nested)	900 UQ + 18 deterministic	-2.237	0.0	optimization gradients are somewhat inaccurate/ noisy due to under-resolved sampling (DOT does well, NPSOL takes much longer to terminate).
2 (layered/ nested)	300 UQ + 6 deterministic	-2.004* (verified: -2.139)	9.6e-15* (verified: 0.0)	*Advantages:* smooth NPSOL navigation, amenable to trust-region techniques, deterministic data easily incorporated into surrogates. *Disadvantages:* sampling must be under-resolved (accurate p_{fail} difficult), verification is critical.
3 (nested/ layered)	45 UQ + 18 deterministic	-2.250* (verified: -2.243)	0.0* (verified: 0.007)	*Advantages:* UQ sampling can be extensive *Disadvantages:* opt. gradients potentially noisy, although extensive sampling may partially mitigate. Deterministic data outside surrogate loop.
4 (layered/ nested/ layered)	45 UQ + 6 deterministic	-2.002* (verified: -2.139)	0.0* (verified: 0.0)	*Advantages:* least expensive. Combines advantages of approaches 2 and 3. *Disadvantages:* verification is critical.

fied and the actual objective function and constraint violation are shown (marked with "verified:").

In other runs of this test problem with alternate OUU mapping data combinations (not shown), inaccuracy of the surrogates at the design level had a much more severe effect. In this case, formulations 1 and 3 generated completely different solutions for (d_1, d_2) from formulations 2 and 4. Upon verification, the constraints were badly violated by formulations 2 and 4. Again, this is not surprising for a quadratic polynomial fit over the entire parameter range using the minimum number of data points, and it highlights the need for restricting the steps in the approximate optimizations and the need for regular verifications, both of which are central features of trust-region approaches.

Trust-region SBOUU

Trust-region approaches to surrogate-based optimization manage the extent of the approximations and perform periodic verifications to maintain the quality of results [18]. Table 2 shows results for formulations 2 and 4 since they are amenable to the application of trust regions for surrogates at the optimization level. In formulation 4, surrogate fits at the UQ level are regenerated for each new trust region at the design level.

Table 2: Trust-region SBOUU results, test problem 1.

OUU Formulation	**Fn. Evals.**	**Obj. Fn.**	**Constraint Violation**
2 + trust regions	2150 UQ + 43 deterministic	-2.237	6.2e-11
4 + trust regions	405 UQ + 64 deterministic	-2.200* (verified: -2.224)	0.0* (verified: 0.0)

With the addition of trust region verifications, formulations 2 and 4 now find the better solution at [(d_1, d_2) ≈ (1.98, 1.77)]. However, the cost of rebuilding the design surrogate for each trust region has increased the expense by approximately an order of magnitude over the corresponding approaches from Table 1. Formulation 4 with trust regions is half the expense of the nested approach (formulation 1 from Table 1). In general, optimizer navigation over $\boldsymbol{d}$ appears to be sufficiently reliable without surrogates for this analytic problem; trust-region surrogate approaches are more strongly motivated in engineering applications where there may be inherent nonsmoothness in the variation of response functions over $\boldsymbol{d}$.

Test Problem 2: Cantilever beam

The next OUU test problem involves the simple uniform cantilever beam [5],[23] shown in Figure 2.

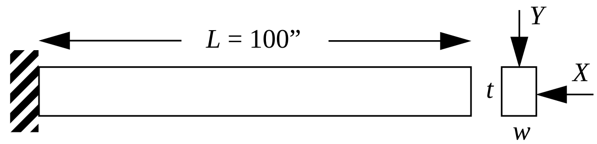


Figure 2. Cantilever beam test problem.

The design problem is to minimize the weight (or, equivalently, the cross-sectional area) of the beam subject to a displacement constraint and a stress constraint. Random variables in the problem include the yield stress R of the beam material, the Young's modulus E of the material, and the horizontal and vertical loads, X and Y, which are modeled with normal distributions using N(40000, 2000), N(2.9E7, 1.45E6), N(500, 100), and N(1000, 100), respectively. The constraints have the following analytic form:

$$stress = \frac{600}{wt^2}Y + \frac{600}{w^2 t}X \le R$$

$$displacement = \frac{4L^3}{Ewt}\sqrt{\left(\frac{Y}{t^2}\right)^2 + \left(\frac{X}{w^2}\right)^2} \le D_0$$

or when scaled:

$$g_S = \frac{stress}{R} - 1 \le 0$$

$$g_D = \frac{displacement}{D_0} - 1 \le 0$$

When seeking a 3-sigma reliability level (probability of failure = 0.00135 if normally-distributed) on these scaled constraints, the design problem can be summarized as follows:

Minimize	wt
Subject to	$\mu_D + 3\sigma_D \le 0$
	$\mu_S + 3\sigma_S \le 0$
	$1.0 \le w \le 4.0$
	$1.0 \le t \le 4.0$
	normal: E, R, X, Y

If the random variables are fixed at their means, the resulting deterministic design problem has the solution (w, t) = (2.35, 3.33) with an objective function of 7.82. For the OUU solutions, quadratic polynomials are again used as the surrogate models and are built using the minimum amount of data required. For 2 design variables, this equates to 6 samples; and for 2 design variables plus 4 uncertain variables, this equates to 28 samples. Uncertainty estimations are performed using either 50 Latin hypercube samples (when sampling on the simulation in formulations 1 and 2), or 5000 Latin hypercube samples (when sampling on the data fit surrogate in formulations 3 and 4), and random number seeds are again reused on each sample generation. A separate interface for deterministic data is not required for this problem, so function evaluation counts only involve simulations performed at the UQ level.

OUU without trust regions

Solutions to the cantilever beam problem for the four OUU formulations are shown in Table 3. The solution for formulation 1 is $(w, t) = (2.53, 3.69)$, which has a slightly lower objective function than solutions from the literature [5],[23]. The solution differences are likely attributable to the differences between sampling-based and analytic reliability-based approaches.

Formulations 2, 3, and 4 are showing significant inaccuracy in the constraint surrogates, again not surprising due to a quadratic polynomial fit over the entire parameter range using the minimum number of data points. Formulation 2 underpredicts the constraints, and formulations 3 and 4 overpredict the constraints. Trust region restriction of steps and regular verifications of approximate optima are needed.

Table 3: OUU results, test problem 2

OUU Formul-ation	Fn. Evals.	Obj. Fn.	Constraint Violation
1	3250	9.32	1.79e-10
2	300	7.49* (verified: 7.49)	3.4e-10* (verified: 3.72)
3	28	30.1* (verified: 30.1)	8.22* (verified: 0.0)
4	28	34.7* (verified: 34.7)	8.09* verified: 0.0)

Trust-region SBOUU

Adding trust regions to formulations 2 and 4 generates the results shown in Table 4. In formulation 4, surrogate fits at the UQ level are regenerated for each new trust region at the design level. It is evident that, relative to the Table 3 results, the trust regions maintain the quality of results but the cost of rebuilding surrogates for each trust region has increased the expense by approximately an order of magnitude. It is also important to note that the results for formulation 4 were sensitive to the initial seed and a representative result is shown. Increasing the number of samples at the UQ level from 28 samples (the minimum) to 42 samples (50% overfit) improved the reliability of the formulation 4 runs with a modest increase in expense.

For this test problem, formulation 2 with trust regions is competitive with the direct nested approach (formulation 1), and formulation 4 with trust regions is much less expensive. This shows improvement in the important metric of computational expense, but an equally important metric is robustness. As for test problem 1, optimizer navigation over ***d*** appears to be sufficiently reliable without surrogates for this analytic problem; the strongest motivation for surrogate approaches is in addressing noisy response variations in real engineering problems. For this reason, the final test problem performs OUU using an engineering simulation code.

Table 4: Trust-region SBOUU results, test problem 2

OUU Formul-ation	Fn. Evals.	Obj. Fn.	Constraint Violation
2 + trust regions	3550	9.32	0.0
4 + trust regions	392	9.36* (verified: 9.38)	0.0* (verified: 0.0)

Test Problem 3: ICF capsule robust design

The final OUU test problem is an engineering design problem which seeks an inertial confinement fusion (ICF) capsule design that is robust with respect to manufacturing variability [10]. It uses a large finite element code to simulate the shock physics involved in imploding the capsule. The simple capsule design shown in Figure 3 has an outer layer of plastic ablator material and an inner core of hydrogen fuel. The ablator material absorbs the radiation pulse, vaporizes and blows off at very high velocities (ablates), and consequently compresses the fuel through momentum reaction forces.

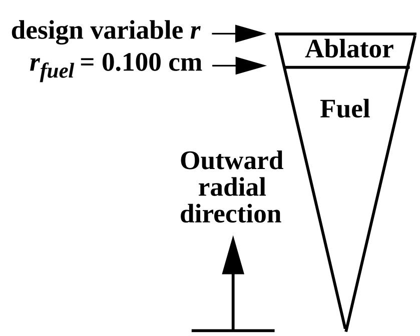


Figure 3. Cross-section of a spherical ICF capsule showing the fuel and ablator layers.

This problem was previously formulated in [10] using probability of failure metrics since this was the information provided by the UQ methods at that time. The problem can now be formulated using a more natural standard deviation metric as follows:

Minimize $V(r)$
Subject to $\sigma_V(r) \le 1.e6$ cm/s
$0.105 \text{ cm} \le r \le 0.135 \text{ cm}$
uniform: $u = [-0.005 \text{ cm}, 0.005 \text{ cm}]$

where r is the outer radius of the ablator layer, u is a uniformly distributed uncertain perturbation on r, $V(r)$ is the implosion velocity of the capsule at the ablator/ fuel interface, and $\sigma_V(r)$ is the standard deviation of the implosion velocity computed using 10 Latin hypercube samples over u for each fixed value of r. Prior to performing the OUU problem, a parameter study was performed to assess the variability in V and σ_V. The results of this study are shown in Figures 4 and 5. Note that the sign on the velocity value is negative due to the orientation of the coordinate axes in the physics simulation code. Thus, minimizing the objective function is equivalent to maximizing the absolute value of the implosion velocity. In addition, local nonsmoothness is evident. This behavior is a common occurrence in engineering problems and provides a primary motivation for surrogate-based techniques. Figure 4 shows that the objective function has a global minimum at r=0.106 cm and a local minimum at r=0.129 cm. However, the local minimum is more robust to perturbations in r than is the global minimum point. In some engineering design problems, a more robust local optimum is preferred to a less robust global optimum. This OUU problem has been formulated to

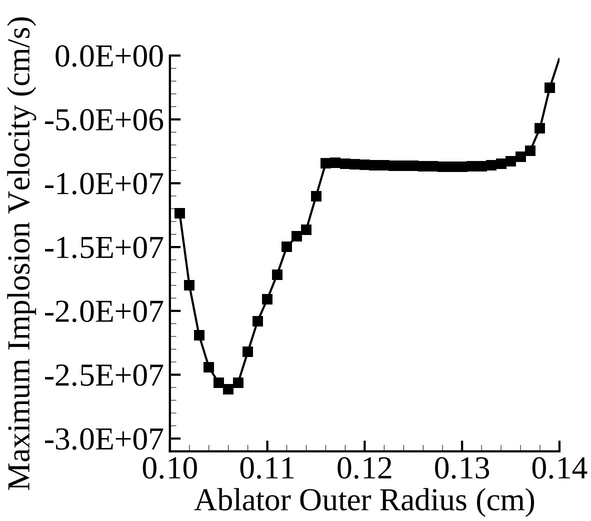


Figure 4. Capsule implosion velocity, $V(r)$, versus ablator radius, r (radius perturbation, u, held at zero).

duplicate such a scenario by constraining the standard deviation of the velocity to be less than 1.0e6 cm/s. Hence, the global optimum point at r = 0.106 cm is infeasible in this OUU problem formulation. Due to computational expense, all problem formulations were not able to be investigated. Rather, formulation 1 and formulation 2 augmented with trust regions were applied to the problem.

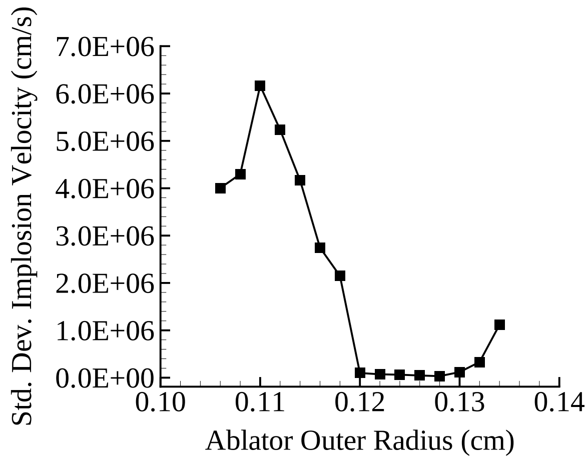


Figure 5. Standard deviation of the implosion velocity, $\sigma_V(r)$, (computed using 10 samples over radius perturbation, u) versus ablator radius, r.

OUU without trust regions

Formulation 1 used DOT's Modified Method of Feasible Directions [21] with central finite difference gradients estimated using a relative step size of 0.01. The starting point for the optimizer was r=0.112 cm. The results of this optimization case are shown in Table 5. The optimizer was successful in moving from the initially infeasible point at r=0.112 cm to the local optimum point at approximately r=0.129 cm using 242 function evaluations.

Table 5: OUU results, test problem 3

OUU Formul-ation	Fn. Evals.	Obj. Fn.	Constraint Violation
1	242	-8.713e6 cm/s	0.0 (σ_V = 5.39e4 cm/s)
2 + trust regions	363	-8.712e6 cm/s	0.0 (σ_V = 3.94e4 cm/s)

Trust-region SBOUU

Next, OUU problem formulation 2 with trust regions was applied to the problem with the expectation that the use of surrogate functions for V and σ_V would improve performance in the presence of nonsmooth response variations. This formulation creates a sequence of quadratic polynomial approximations for V and σ_V, each of which is built using three data points in the current trust region. The initial trust region size was set to coincide with the bounds on the ablator radius, r, i.e., from 0.105 cm to 0.135 cm. Subsequent iterations of the SBOUU algorithm resized the trust region to maintain accurate surrogate models. The results of this study are also shown in Table 5.

The optimization history of the objective and constraint functions is shown for the OUU and SBOUU formulations in Figures 6 and 7, respectively. Note that the horizontal axis on both figures has been truncated to 300 function evaluations to focus on the initial convergence behavior of both formulations. The optimization history results show that the SBOUU algorithm found a feasible design point in a single cycle compared to several cycles for the OUU algorithm. That is, it was successful in smoothing the noisy response variations and jumping to the vicinity of the optimal solution in a single step. However, the SBOUU algorithm required more iterations to terminate than did the OUU algorithm. This is typical of SBOUU algorithms using “soft” convergence criteria, and more work is needed on refining these criteria.

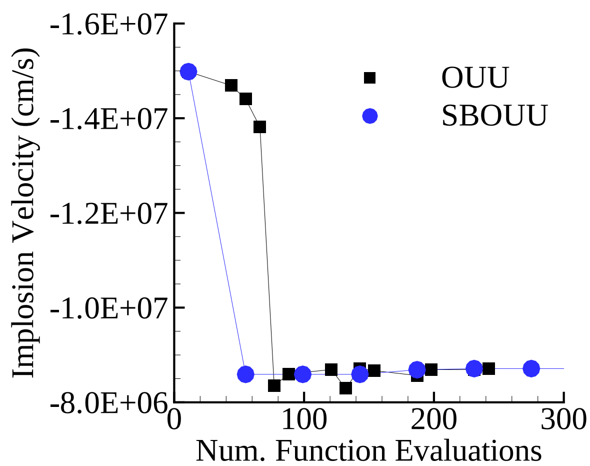


Figure 6. Objective function iteration history for formulation 1 and for formulation 2 with trust regions.

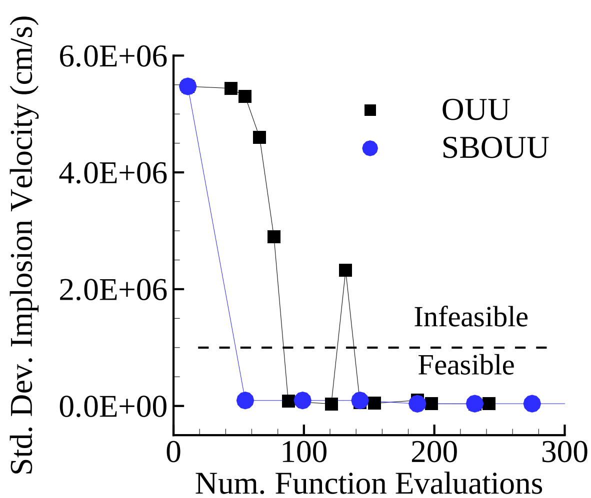


Figure 7. Constraint iteration history for formulation 1 and for formulation 2 with trust regions.

Conclusions

This paper explores a variety of formulations for optimization under uncertainty, with an emphasis on surrogate-based approaches. Preliminary work points to the utility of surrogate fits to statistical data generated from sampling-based uncertainty quantification, both in terms of reducing computational expense and mitigating the effects of nonsmooth response variations. However, without restricting and rigorously verifying the steps in the approximate optimization cycles, weaknesses in the data fits can be exploited and poor solutions may be obtained. It is clear that trust-region approaches to surrogate-based optimization under uncertainty maintain the quality of results in these classes of problems and provide an effective alternative to direct nested approaches.

In future work, the verification steps in trust-region SBOUU can be extended to be more rigorous. Techniques like ordinal optimization [24] can be used for rigorous selection among stochastic alternatives by non-overlapping the confidence bounds (for a user-selected confidence level) on the statistics of the distributions. This is an important component of achieving provable convergence in trust-region surrogate-based optimization under uncertainty algorithms.

Acknowledgements

The authors would like to express their thanks for informative discussions with Prof. Raphael Haftka of the University of Florida, Dr. Natalia Alexandrov of NASA Langley, Prof. C.H. Chen of George Mason University, Prof. Larry Biegler of Carnegie-Mellon University, Prof. Roger Ghanem of Johns Hopkins University, and Prof. Takeru Igusa of Johns Hopkins University.

References

[1]Igusa, T., and Wan, Z., “Response surface methods for optimization under uncertainty,” abstract

submitted for *International Conference on Applications of Statistics and Probability in Civil Engineering (ICASP9)*, Berkeley, CA, July 6-9, 2003.

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

[3]Haftka, R.T., private communication.

[4]Haldar, A. and Mahadevan, S., *Probability, Reliability and Statistical Methods in Engineering Design*, Wiley, New York, NY, 2000.

[5]Wu, Y.-T., Shin, Y., Sues, R., and Cesare, M., "Safety-Factor Based Approach for Probability-Based Design Optimization," paper AIAA-2001-1522 in *Proceedings of the 42rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Seattle, WA, April 16-19, 2001.

[6]Wang, L., and Kodiyalam, S., “An Efficient Method for Probabilistic and Robust Design with Non-Normal Distributions,” paper AIAA-2002-1754 in *Proceedings of the 43rd AIAA/ASME/ASCE/ASC Structures, Structural Dynamics, and Materials Conference* (Nondeterministic Approaches Forum), Denver, CO, April 22-25, 2002.

[7]Orozco, C.E., and Ghattas, O.N., “A Reduced SAND Method for Optimal Design of Non-linear Structures,” *International Journal for Numerical Methods in Engineering*, Volume 40, 1997, pp. 2759-2774.

[8]Ghanem, R.G., and Red-Horse, J., “Propagation of probabilistic uncertainty in complex physical systems using a stochastic finite element approach,” *Physica D*, 133 (1999), pp. 137-144.

[9]Ghanem, R.G., private communication.

[10]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.

[11]Eldred, M.S., Giunta, A.A., van Bloemen Waanders, B.G., Wojkiewicz, S.F., 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 SAND2001-3796P, Dec. 2001.

[12]DAKOTA regression tests available for download from http://endo.sandia.gov/DAKOTA/ licensing/license.html.

[13]Eldred, M.S., Giunta, A.A., van Bloemen Waanders, B.G., Wojkiewicz, S.F., 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 Reference Manual." Sandia Technical Report SAND2001-3515W, Nov. 2001.

[14]Eldred, M.S., Giunta, A.A., van Bloemen Waanders, B.G., Wojkiewicz, S.F., 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 Developers Manual." Sandia Technical Report SAND2001-3514W, Nov. 2001.

[15]Stroustrup, B., *The C++ Programming Language*, 2nd ed., Addison-Wesley, New York, 1991.

[16]Bartlett, R.A. and Biegler, L.T., "rSQP++: An Object-Oriented Framework for Successive Quadratic Programming," abstract for *First Sandia Workshop on Large-scale PDE-constrained Optimization*, Santa Fe, NM, April 4, 2001, to appear in *Springer-Verlag Lecture Notes in Computational Science and Engineering*.

[17]Wojtkiewicz, S.F., Jr., Eldred, M.S., Field, R.V., Jr., Urbina, A., and Red-Horse, J.R., "A Toolkit For Uncertainty Quantification In Large Computational Engineering Models," paper AIAA-2001-1455 in *Proceedings of the 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Seattle, WA, April 16-19, 2001.

[18]Giunta, A.A., and Eldred, M.S., "Implementation of a Trust Region Model Management Strategy in the DAKOTA Optimization Toolkit," paper AIAA-2000-4935 in *Proceedings of the 8th AIAA/ USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, September 6-8, 2000.

[19]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.

[20]Alexandrov, N.M., Nielsen, E.J., Lewis, R.M., and Anderson, W.K., “First-Order Model Management with Variable-Fidelity Physics Applied to Multi-Element Airfoil Optimization,” paper AIAA-2000-4886 in *Proceedings of the 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, September 6-8, 2000.

[21]*DOT Users Manual*, Version 4.10, VMA Engineering, Colorado Springs, CO, 1994.

[22]Gill, P.E., Murray, W., Saunders, M.A., and Wright, M.H., *User's Guide for NPSOL (Version 4.0): A Fortran Package for Nonlinear Programming*, System Optimization Laboratory, TR SOL-86-2, Stanford University, Stanford, CA, Jan. 1986.

[23]Sues, R., Aminpour, M., and Shin, Y., "Reliability-Based Multidisciplinary Optimization for Aerospace Systems," paper AIAA-2001-1521 in *Proceedings of the 42rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Seattle, WA, April 16-19, 2001.

[24]Chen, C.H., Lin, J., Yucesan, E., and Chick, S.E., "Simulation Budget Allocation for Further Enhancing the Efficiency of Ordinal Optimization," *Discrete Event Dynamic Systems: Theory and Applications*, 10(3): 251-270; Jul 2000.

NOTE:

many of the Sandia references are available online from http://endo.sandia.gov/DAKOTA/references.html