

Robust Optimizations of Structural and Aerodynamic Designs

Komahan Boopathy *

University of Dayton, Ohio, 45469, USA

Markus P. Rumpfkeil †

University of Dayton, Ohio, 45469, USA

This paper demonstrates the use of polynomial chaos and kriging surrogate models, which are enhanced with a dynamic training point selection framework, for the propagation of mixed epistemic and aleatory uncertainties in robust optimization problems. The selection of training points for the two surrogate models is guided by local surrogate models (multivariate interpolation and regression) which are built using a subset of the available training data. The aleatory uncertainties are propagated via extensive sampling of the surrogate models, whereas the epistemic uncertainties are propagated using a box-constrained optimization approach. Robust optimizations are demonstrated for two structural and one aerodynamic test problem. The structural test cases include designing a three-bar truss and a cantilever beam, whereas the aerodynamic test case involves the robust lift-constrained drag minimization of an airfoil under transonic flow conditions.

Nomenclature

α	angle of attack	$\frac{d\mathcal{J}}{d\boldsymbol{\eta}}$	epistemic variable gradients
β	epistemic realizations	f^*	extremum of the function
$\boldsymbol{\alpha}$	aleatory realizations	F_s	factor of safety
\boldsymbol{d}	vector of design variables	g	inequality constraint
μ	mean	g^*	extremum of the constraints
σ	standard deviation	M	number of variables or dimensions
σ^2	variance	M_∞	Mach number
\tilde{N}	number of Monte Carlo samples	N	number of surrogate training points
$\boldsymbol{\xi}$	aleatory random variables	n	number of exact function evaluations for box-constrained optimization
$\boldsymbol{\eta}$	epistemic random variables	$p(\boldsymbol{\xi})$	probability distribution
C_D	drag coefficient	P_k	probability of constraint satisfaction
C_L	lift coefficient	\mathcal{J}	objective function
f	exact function	\hat{f}	surrogate approximated function value
$\frac{d\mathcal{J}}{d\boldsymbol{\xi}}$	aleatory variable gradients		

I. Introduction and Motivation

When designing a system uncertainties associated with the inputs lead to uncertainties in the output quantities of interest. Some of the common sources of uncertainties are geometrical, modeling (physics, turbulence models), parametric (model parameters), operating environment, boundary conditions, and so on. A deterministic optimization approach assumes no variations in the design variables and other parameters *i.e.*, all the inputs are assumed to be precise which produces a deterministic output. In reality, this can easily lead to sub-optimal performance or failure of many deterministically optimized designs. For instance, when an aircraft designed to cruise at specific optimal settings (e.g. Mach number, angle of attack, shape) deviates from these settings (e.g. due to continuous wind gusts, ice accumulation, faulty calibration of instruments,

*Master's Candidate, Dept. of Mechanical and Aerospace Eng., KomahanBoopathy@gmail.com, Student Member AIAA

†Assistant Professor, Dept. of Mechanical and Aerospace Eng., Markus.Rumpfkeil@udayton.edu, Senior Member AIAA

wear and tear) the flight performance can be adversely affected leading to an increased fuel burn or other undesirable characteristics. Therefore, when a system is designed allowances must be made to accommodate likely variations that can disrupt the nominal performance. When such variations or uncertainties are not accounted for in the design process (as in the case of a deterministic optimization practice), a degraded performance of the optimized design is inevitable.

In a heavily optimized design, the optimum solution tends to lie either at the extremum of the objective function or at a constraint boundary.¹ Considering the random elements during the design process, a deterministic optimum is a vulnerable solution with a greater likelihood of violating the design requirements: even small perturbations in the input can lead to a poor performance or failure of the design. To alleviate some of these problems, a factor of safety is traditionally incorporated into the constraints. The factor of safety serves to move the optimum away from the constraint boundary by a considerable distance, thereby preventing the design from an imminent failure. However, the reality is that the designer assigning a factor of safety is seldom aware of the real effects of uncertainty and predominantly produces either over- or under-conservative designs leading to weight penalty or vulnerable products, respectively. With the continuous evolution of radically new types of design, it is increasingly difficult for a designer to assign an adequate factor of safety.¹

The ability to design a system that is less sensitive to variations of the input variables is thus far beyond the control of the designer, as stated by Genichi Taguchi,¹ and has therefore become the target of recent research efforts with the emergence and availability of uncertainty analysis procedures and computational resources. Uncertainty quantification (UQ) has grown to be a major field of interest, where the goal is to account for the effect of uncertainties on designs, through a modified optimization process known as *optimization under uncertainty* (OUU), where the inputs are treated as random variables. OUU can be subdivided into two fields as *robust design optimization* (RDO) and *reliability based design optimization* (RBDO).²⁻⁴ Though these two fields share many common attributes, they differ in their objectives: RDO techniques are used to produce a design that is more robust (less sensitive) to design parameter anomalies, whereas the goal of RBDO is to minimize the probability of failure of the system. Robust optimization minimizes the expected mean and variance of the output. This work focuses on methods to produce robust designs which involves finding an optimum that is less sensitive to input variability, as opposed to a deterministic optimum that can exhibit a sharp change in the objective function value for minor perturbations in the inputs.

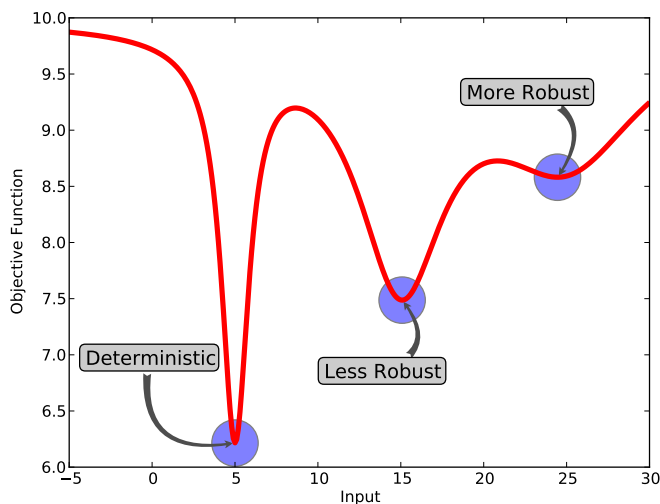


Figure 1: An illustration for the sensitivity of optimum designs to input variations.

Usually, a robust solution is obtained at the expense of an increased cost function value¹ as illustrated in Figure 1. Some of the main goals in OUU are listed below:

- To determine the effects of uncertainties on designs (knowing whether they are robust or vulnerable).
- To identify the limitations of designs and find potential improvements.

- To construct confidence intervals on output quantities that provide valuable information to the designer (e.g., output means and variances).
- To carry out reliability analysis for certification and quality assurance purposes.

“Quantifying uncertainties” is the key to all the potential outcomes including the ones listed above. Uncertainties fall into one of two categories; namely *aleatory* (Type A or reducible) and *epistemic* (Type B or irreducible) uncertainties.^{1,5-9} In recent years, design teams and regulatory agencies are increasingly being asked to specifically characterize and quantify different types of uncertainties and separate their individual effects.⁵⁻⁹ The most popular and easiest approach for the propagation of uncertainties is the Monte Carlo simulation (MCS), where the simulation output f is sampled many times to obtain output statistics or to determine worst case scenarios. However, multiple realizations of the output function f are not always computationally tractable; for example, high-fidelity physics-based simulations such as computational fluid dynamics (CFD) or finite element analyses (FEA) can be very time consuming.

To overcome the problem of exorbitant computational expenses, surrogate models can be constructed to model the uncertainties. Surrogate models are an approximate but inexpensive to evaluate representation of the output quantity of interest. They can be sampled exhaustively at a cheaper computational cost to propagate the input uncertainties and determine the output effects. This approach is referred to as the inexpensive Monte Carlo simulation (IMCS).¹⁰ The accuracy of surrogate models are influenced primarily by the choice of training point locations. Training point selection is typically done using design of experiments (DoE) techniques; for example, uniform design,¹¹ Monte Carlo (MC),¹² latin hypercube (LHS),¹³ quadrature nodes,^{14,15} and low-discrepancy sequences.¹⁶ To overcome the difficulties with these methods (e.g. missing important locations, correlated distributions, poorly conditioned linear systems) and to put the expensive surrogate training information to good use, the authors have recently developed a dynamic training point selection strategy that ably chooses training points in regions that are most viable to improve the accuracy of the surrogate models (see Boopathy and Rumpfkeil¹⁷⁻¹⁹). This work entails the application of the dynamically trained kriging and polynomial chaos (PC) surrogate models for propagating uncertainties in robust optimizations.

Outline of the paper

The remainder of this paper is organized as follows. Section II reviews practices for uncertainty analysis and provides a detailed account of the framework for robust optimization under mixed epistemic and aleatory uncertainties which is employed here. Sections III and IV discuss the results for a three-bar truss and cantilever beam design problem, respectively. Section V discusses the robust lift-constrained drag minimization of an airfoil and Section VI summarizes the results.

II. Design in the Presence of Uncertainty

This section provides a discussion of the different stages in optimization under uncertainty.^{1,5-9}

1. *Identification, modeling and representation of uncertainties* to translate the available data into mathematical models that are either probabilistic or non-probabilistic in nature,
2. *Propagation of uncertainties* through computer models to quantify their impact on system performance,
3. *Formulation and solution of an optimization problem* with appropriate objective and constraint functions ensuring that the optimum solution is robust against variations.

II.A. Uncertainty Modeling

The modeling of uncertainties begins with the treatment of inputs as random variables. Uncertainties can be classified as *aleatory* and *epistemic* uncertainties.^{1,5} Aleatory uncertainties are the inherent randomness or variation in a physical system, input parameters and variables, or operating environment.⁵ For example, operating conditions are predominantly dissimilar to the ones used in design calculations and typically fluctuate around some mean value. Epistemic uncertainties arise due to the lack of knowledge or information in any phase or activity of the modeling process. It is not an inherent property of the system and thus

can be eliminated (or converted to aleatory form) when sufficient data becomes available. As an example, situations can arise where only the bounds or intervals of the uncertain random variables are known (e.g. manufacturing tolerances) whereas the underlying probability distribution or other statistical parameters within the interval are unknown (unlike aleatory random variables).

II.A.1. Probabilistic Modeling

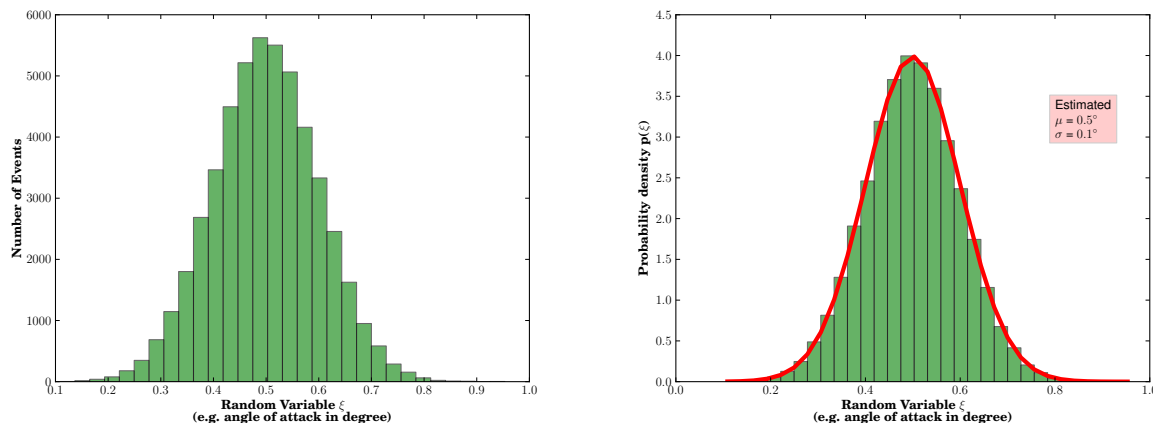


Figure 2: An illustration for the use of probabilistic methods (aleatory uncertainty representation). A histogram of the available angle of attack data (left) is shown with the fitted Gaussian probability density function (right) whose parameters (μ and σ) are estimated from the available data.

The use of probabilistic methods to model uncertainties is possible when sufficient data is available. When field data is available a probability density function can be fit to the available data. Alternatively, a distribution function (Gaussian, log-normal, exponential, etc.) can be assumed and its parameters can be estimated from the available data. Figure 2 shows an example of fitting the available angle of attack information to a normal distribution function. The mean and standard deviation are estimated from the data and can be used to define the probability density function (PDF) of the random variable as $p(\xi) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\xi-\mu)^2}{2\sigma^2}}$ (Gaussian PDF). The angle of attack (and similarly any other variables) can now be treated as random variable in optimization formulations.

Distributions should be assumed with caution when only limited data is available to the designer for assessment.¹ For example, a normal distribution can not be assumed for Young's modulus, as a Gaussian distribution supports $[-\infty, +\infty]$ and a zero probability would mean a negative Young's modulus which is unrealistic. In reliability calculations, the probability of failure is estimated near the tail end of the distribution, which can be very erroneous if a wrong distribution structure is assumed.

II.A.2. Non-Probabilistic Modeling

The use of probability theory to model the distribution of input uncertainties is questionable when not enough information is available. This naturally leads into non-probabilistic approaches such as possibility theory, interval analysis, convex modeling and evidence theory.¹ The simplest non-probabilistic approach is perhaps the interval representation of input uncertainties. The input random variable is represented by the interval $[\eta^-, \eta^+]$ where η^- and η^+ denote the lower and upper bounds on the input random variable, respectively. This scenario is illustrated with a two-variable example in Figure 3. The random process can take any value within the specified interval but the underlying probability distribution is unknown. The input bounds need to be processed into the analysis model to construct bounds on the output quantity of interest.¹

In summary, probabilistic approaches are apt for modeling aleatory uncertainties featuring an abundance of data and non-probabilistic approaches are suitable for epistemic uncertainties suffering from a data scarcity.

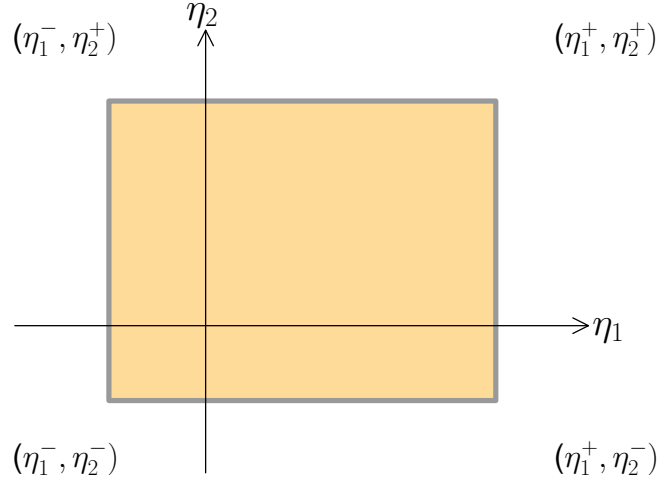


Figure 3: An illustration for the bounds on input variables (epistemic uncertainty representation).

II.B. Uncertainty Propagation

In this section, approaches for the propagation of input uncertainties are discussed. The goal is to quantify the uncertainties and model the input–output relationship through numerical methods. The aleatory variables are denoted as $\boldsymbol{\xi}$ and realizations of aleatory variables from their probability distribution are represented as $\boldsymbol{\alpha}$. The epistemic variables are denoted as $\boldsymbol{\eta}$ and their realizations within the specified bounds are denoted as $\boldsymbol{\beta}$.

II.B.1. Propagation of Aleatory Uncertainties

Sufficient input data is generally available for the analysis of aleatory uncertainties. Thus, probabilistic methods which mandate multitude of realizations are commonly used for computing the statistics based on the input probability distribution. In other words, the distribution type of the input random variables (e.g. $\boldsymbol{\alpha} \sim \mathcal{N}(\bar{\boldsymbol{\xi}}, \boldsymbol{\sigma}_{\boldsymbol{\xi}}^2)$) are known, whereas the functional dependence $f(\boldsymbol{\xi})$ on these random variables is not known, and are modeled using numerical simulations.

MONTE CARLO SIMULATION: The simplest approach to achieve uncertainty propagation is the Monte Carlo simulation (MCS). When information such as the input mean, standard deviation and PDF of design variables and other parameters (collectively known as inputs) are available, the statistics of the output function can be computed using MCS. In this method, numerous samples (realizations) $\boldsymbol{\alpha}^{(j)}$ are generated from the distribution $p(\boldsymbol{\xi})$ of the input random variables and the response function or simulation code is evaluated. This leads to the following estimates for the mean:

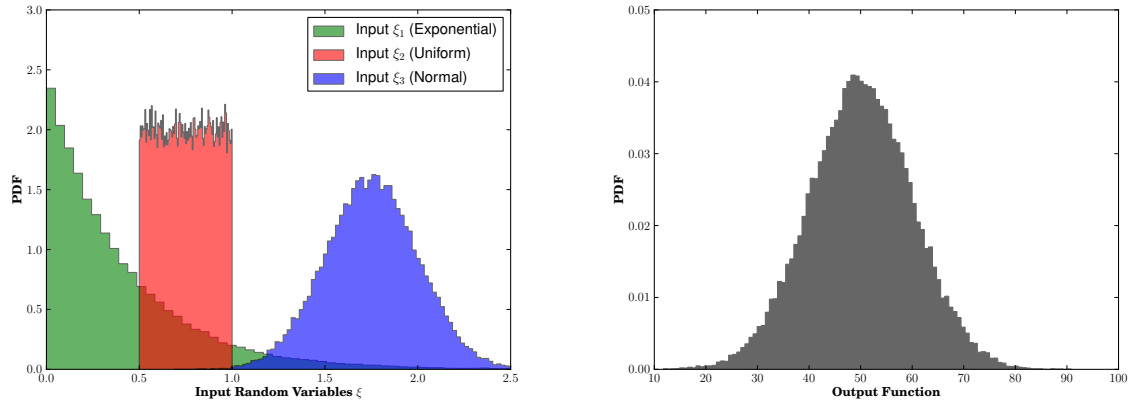
$$\bar{f} = \mu_f = \frac{1}{\tilde{N}} \sum_{j=1}^{\tilde{N}} f(\boldsymbol{\alpha}^{(j)}), \quad (1)$$

and variance of the output quantity:

$$\sigma_f^2 = \vartheta_f = \frac{1}{\tilde{N}} \sum_{j=1}^{\tilde{N}} (f(\boldsymbol{\alpha}^{(j)}) - \bar{f})^2, \quad (2)$$

where \tilde{N} is the number of Monte Carlo samples. MCS can be used on any output function $f(\boldsymbol{\xi})$ and is hence non-intrusive in nature.

INEXPENSIVE MONTE CARLO SIMULATION: It is well known that repeated evaluations of the exact function, $f(\boldsymbol{\xi})$, is prohibitively expensive or impractical most of the times. To overcome this problem, surrogate models can be built and inexpensively probed to yield approximated output function values $\hat{f}(\boldsymbol{\xi})$ for the calculation of approximate means and variances. The required number of function values (training points) for building



Set of input variables with distributions

Distribution of the output function

Figure 4: An illustration for modeling the input-output relationship of uncertainties (propagation of uncertainties).

an accurate surrogate is usually way less than the number required for statistically converged Monte Carlo simulation. In this work, the domain over which the surrogate model is built is taken to be *three standard deviations* in all aleatory input dimensions from the specified mean value *i.e.*, the surrogate domain is: $[\bar{\xi} \pm 3 \cdot \sigma_{\xi}]$. This implies that for normally distributed input variables more than 99 % of all samples (aleatory realizations) fall within the surrogate domain and the less accurate extrapolation capabilities of the surrogate model only need to be employed for a small fraction of the samples. A larger domain can be specified for the surrogate (e.g. 6σ from mean) to account for even remote possibilities, but is not used here (recommended for reliability calculations). In this work, polynomial chaos and kriging surrogate models trained using the dynamic framework developed by Boopathy and Rumpfkeil^{17,18} are employed for aleatory uncertainty propagation. Both surrogate models have the capability to incorporate higher-order derivative information (gradients and Hessian), however, the surrogates are built using function values only to reduce the complexity in presenting the results.

II.B.2. Propagation of Epistemic Uncertainties

Epistemic uncertainties represent the lack of knowledge about the appropriate value to use.⁷ For instance, manufacturers typically provide intervals in terms of tolerances (e.g. $\pm 0.5 \text{ mm}$ for length of a bolt), but the exact values are not known or cannot be guaranteed whatsoever. Here, the goal is to have bounds on the output quantity of interest or to determine worst case scenarios (e.g. maximum possible constraint violation, least possible lift), in order to minimize the sensitivity or variation of the design with respect to these uncertainties. In a situation where only the input interval $I(\boldsymbol{\eta}) = [\boldsymbol{\eta}^-, \boldsymbol{\eta}^+] = [\bar{\boldsymbol{\eta}} - \boldsymbol{\tau}, \bar{\boldsymbol{\eta}} + \boldsymbol{\tau}]$ is known, the above assessment can be accomplished in most straightforward manner by either one of the two methods discussed below.

SAMPLING: An extensively sampling of the interval $I(\boldsymbol{\eta})$ can be done and an ordering of the resulting outputs $f(\boldsymbol{\eta})$ is carried out to determine the extreme values (worst and best cases). However, the computational burden can be prohibitive in the case of high-fidelity physics-based simulations and for higher-dimensional spaces. As a remedy, a surrogate model over the domain $I(\boldsymbol{\eta})$ can be constructed (similar to aleatory uncertainties), which can then be sampled using inexpensive Monte Carlo simulations (IMCS). However, with increasing number of variables, building an accurate surrogate model requires thousands of simulation outputs (referred to as the curse of dimensionality) and quickly becomes prohibitively expensive as well.

BOUND-CONSTRAINED OPTIMIZATION: A bound- or box-constrained optimization (BCO)²⁰⁻²² can be employed to find the worst and best behavior of the constraint/objective function within the specified interval $I(\boldsymbol{\eta})$. A gradient-based BCO scales mildly with the number of input variables, making it computationally

more attractive than MCS for quantifying the effect of epistemic uncertainties, particularly for larger problems. In BCO, the problem of finding the extreme value of the function, f^* , (and the constraints, g_i^*) within the interval $I(\boldsymbol{\eta})$ can be cast as follows:

$$\begin{aligned} & \underset{\boldsymbol{\beta}}{\text{minimize/maximize}} && f = f(\boldsymbol{\eta}), \\ & \text{subject to} && \boldsymbol{\beta} \in I(\boldsymbol{\eta}) = [\bar{\boldsymbol{\eta}} - \boldsymbol{\tau}, \bar{\boldsymbol{\eta}} + \boldsymbol{\tau}]. \end{aligned} \quad (3)$$

In most cases the extremum occurs at either the upper or lower bound of the interval due to the quasi-linearity of the typically small space described by $I(\boldsymbol{\eta})$. Thus, BCO typically takes only about five to ten simulation output and gradient evaluations to reach f^* and is used throughout this work to propagate epistemic uncertainties. An L-BFGS^{23,24} algorithm which utilizes function and gradient information is used to solve the BCO problem given by Eq. (3).

II.B.3. Propagation of Mixed Uncertainties

Table 1 summarizes the four typical methods that can be employed for the propagation of mixed epistemic and aleatory uncertainties along with their corresponding approximate simulation requirements. The computational requirements can be interpreted assuming an approximate range of values for: (i) the number of Monte Carlo sample points ($\tilde{N} = 10^5 - 10^8$), (ii) the number of surrogate training points ($N = 50 - 5000$), and (iii) the number of simulation output evaluations (with gradients) for a BCO ($n = 10 - 100$). The most

Table 1: Methods for optimization under mixed uncertainties along with their simulation requirements per iteration.

Method	Propagation Method		No. of Evaluations		Total per iteration
	Aleatory	Epistemic	Aleatory	Epistemic	
1	MCS	MCS	\tilde{N}_1	\tilde{N}_2	$\tilde{N}_1 \tilde{N}_2$
2	MCS	BCO	\tilde{N}	n	$\tilde{N} \cdot n$
3	IMCS	IMCS	N_1	N_2	$N_1 \cdot N_2$
4	IMCS	BCO	N	n	$N \cdot n$

straightforward way to propagate mixed uncertainties is to carry out a nested-sampling approach (Method 1), where for each aleatory random variable realization, $\boldsymbol{\alpha}^{(i)}$, $i = 1, 2, \dots, \tilde{N}$, drawn from its probability distribution $p(\boldsymbol{\xi})$, a Monte Carlo sampling (or LHS for a better search performance) has to be performed over the epistemic variable realizations $\boldsymbol{\beta}^{(j)}$, $j = 1, 2, \dots, \tilde{N}$, to determine the extreme behavior. Method 2 uses BCO for epistemic uncertainties and is less expensive than Method 1, but can still represent an enormous computational endeavor for the aleatory uncertainties, and is hence impractical for high-fidelity simulations. It can be seen that the last two methods employing surrogate models for uncertainty propagation are several orders of magnitude cheaper. Method 3 turns out to be the cheapest for smaller problems (M less than six or so), but can easily suffer from the curse of dimensionality and thus lacks robustness, whereas it can be inferred that Method 4 is still computationally manageable for bigger problem sizes. Thus, this work preferably employs the IMCS-BCO approach (Method 4) for the propagation of mixed uncertainties in a robust optimization problem. The employed IMCS-BCO framework has been developed by Lockwood *et al.*²⁰⁻²² and Rumpfkeil.²⁵ A detailed discussion of all the steps involved is given in Section II.C.3.

The computational requirements in Table 1 are given for just one iteration of the numerical solution of the robust optimization problem. If the optimizer requires \mathcal{K} iterations to converge, the number in the last column has to be multiplied with \mathcal{K} to obtain an approximation for the number of simulation evaluations needed. As a last remark, a deterministic gradient-based optimization requires only on the order of $2\mathcal{K}$ (one function and one gradient evaluation per iteration) simulation output evaluations to reach the optimum.

In the mixed uncertainty problem, the trial design variable vector \boldsymbol{d} is comprised of both aleatory and epistemic components *i.e.*, $\boldsymbol{d} = [\bar{\boldsymbol{\xi}}, \bar{\boldsymbol{\eta}}]$, where $\bar{\boldsymbol{\xi}}$ represents the mean of aleatory uncertainties and $\bar{\boldsymbol{\eta}}$ refers to the midpoint of the epistemic uncertainty bounds. Here, the aleatory uncertainties are assumed to be *statistically independent* and *normally distributed* with $\boldsymbol{\alpha} \sim \mathcal{N}(\bar{\boldsymbol{\xi}}, \boldsymbol{\sigma}_{\boldsymbol{\xi}}^2)$. Equations for correlated and/or non-normally distributed aleatory variables can also be derived; however, the analysis and resulting equations become more complex²⁶ and are beyond the scope of this work. In addition, the assumed input standard

deviation σ for aleatory variables as well as the upper and lower bounds for epistemic variables defined by τ are treated as fixed throughout the optimization for simplicity (could be easily changed).

II.C. Optimization Problem Formulation

II.C.1. Deterministic Optimization

A conventional constrained optimization problem for an objective function, J , that is a function of input variables, \mathbf{d} , state variables, $\mathbf{q}(\mathbf{d})$, and simulation outputs, $f(\mathbf{d}) = F(\mathbf{q}(\mathbf{d}), \mathbf{d})$, can be written as:

$$\begin{aligned} \underset{\mathbf{d}}{\text{minimize}} \quad & J = J(f, \mathbf{q}, \mathbf{d}), \\ \text{subject to} \quad & R(\mathbf{q}, \mathbf{d}) = 0, \\ & g(f, \mathbf{q}, \mathbf{d}) \leq 0. \end{aligned} \tag{4}$$

Here, the state equation residuals, R , are expressed as an equality constraint, and other system constraints, g , are represented as general inequality constraints. In the case where the input variables are precisely known all functions dependent on \mathbf{d} are deterministic. However, in the presence of input uncertainties all functions in Eq. (4) can no longer be treated deterministically.

II.C.2. Robust Optimization

The setup of a robust optimization problem under mixed uncertainties is discussed below.

OBJECTIVE FUNCTION: A robust objective function, \mathcal{J} , can be written in terms of the mean values of the functional outputs μ_{f^*} and variance $\sigma_{f^*}^2$. The robust optimization problem is minimizing the weighted sum of mean extremum and variance of the function. Mathematically, the objective function assumes the form:

$$\mathcal{J} = w_1 \mu_{f^*} + w_2 \sigma_{f^*}^2, \tag{5}$$

where w_1 and w_2 are some user specified weights. In this work, the weights w_1 and w_2 are set to one. The asterisk (*) refers to the extremum of the BCO problem.

CONSTRAINT FUNCTIONS: The state equation residual equality constraint, R , is deemed satisfied for all values of α and β . The inequality constraints can be cast into a probabilistic statement such that the probability that the constraints are satisfied is greater than or equal to a desired or specified probability, P_k . The constraints are written as a function of mean values and their standard deviations.^{27,28}

$$g^r = g(\mu_{f^*}, \mathbf{q}, \boldsymbol{\xi}, \boldsymbol{\eta}) + k\sigma_{f^*} \leq 0, \tag{6}$$

where k is the number of standard deviations σ_{g^*} that the constraint g must be displaced in order to achieve the required P_k .

PROBLEM FORMULATION: Lastly, the deterministic optimization problem given by Eq. (4) can be recast into a robust design optimization problem^{26,29} as follows:

$$\begin{aligned} \underset{\boldsymbol{\xi}, \boldsymbol{\eta}}{\text{minimize}} \quad & \mathcal{J} = \mathcal{J}(\mu_{f^*}, \sigma_{f^*}^2, \mathbf{q}, \boldsymbol{\xi}, \boldsymbol{\eta}), \\ \text{subject to} \quad & R(\mathbf{q}, \boldsymbol{\xi}, \boldsymbol{\eta}) = 0, \\ & g^r = g(\mu_{f^*}, \mathbf{q}, \boldsymbol{\xi}, \boldsymbol{\eta}) + k\sigma_{f^*} \leq 0. \end{aligned} \tag{7}$$

OPTIMIZATION SOFTWARE: The software package IPOPT (Interior Point OPTimizer)³⁰ for large-scale nonlinear constrained optimization is used for the solution of the robust optimization problem given by Eq. (7). IPOPT allows users to impose bound or box constraints on the design variables, which can be very helpful in ensuring the stability of the simulation output analysis by preventing the exploration of too extreme regions of the design space.

II.C.3. Robust Optimization Framework

The steps involved in robust optimization under mixed uncertainties^{20–22,25} are detailed here (see Figure 5). Surrogate models are built to propagate aleatory uncertainties and bound-constrained optimizations are used to propagate epistemic uncertainties.

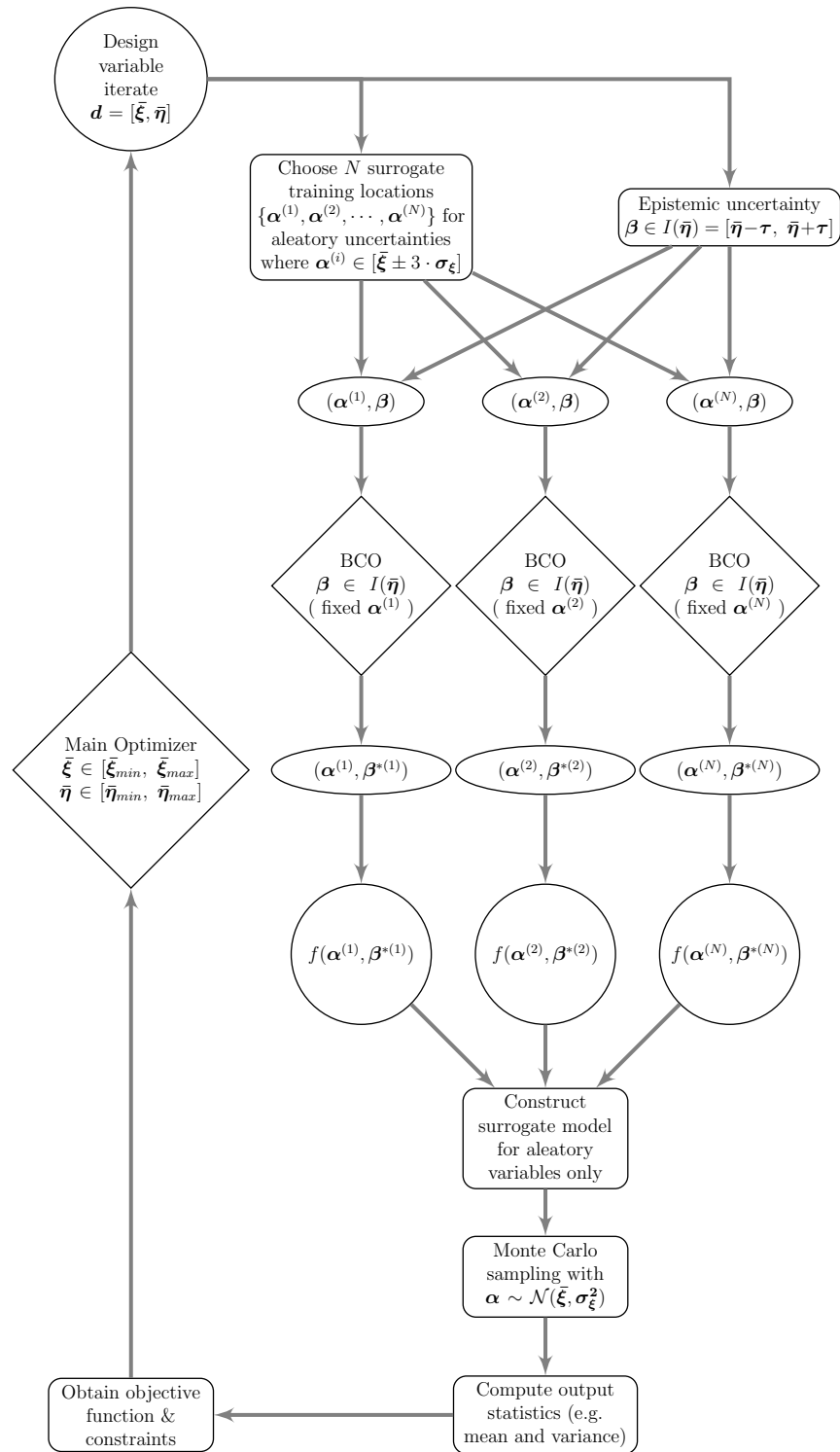


Figure 5: Framework for robust optimization under mixed epistemic and aleatory uncertainties.

1. INITIALIZE: The main optimizer IPOPT (see Subsection II.C.2) provides a trial design variable vector $\mathbf{d} = [\bar{\boldsymbol{\xi}}, \bar{\boldsymbol{\eta}}]$ at each iteration, based on which the surrogate domain is determined as: $[\bar{\boldsymbol{\xi}} \pm 3 \cdot \boldsymbol{\sigma}_{\boldsymbol{\xi}}]$, from which surrogate training point locations $\boldsymbol{\alpha}^{(i)}$, $i = 1, \dots, N$, are selected (using the dynamic training point selection framework^{17,18}). The interval for the bound-constrained optimization for epistemic random variables is

represented as $I(\boldsymbol{\eta}) = [\boldsymbol{\eta}^-, \boldsymbol{\eta}^+] = [\bar{\boldsymbol{\eta}} - \boldsymbol{\tau}, \bar{\boldsymbol{\eta}} + \boldsymbol{\tau}]$.

2. PROPAGATE EPISTEMIC EFFECTS: For each surrogate training point $\boldsymbol{\alpha}^{(i)}$, $i = 1, \dots, N$, a BCO problem is solved for determining the worst or best epistemic realization $\boldsymbol{\beta}^{*(i)}$ within the interval $I(\boldsymbol{\eta})$. Mathematically, this refers to the determination of the extremum f^* of the output function within the interval $I(\boldsymbol{\eta})$:

$$\begin{aligned} & \underset{\boldsymbol{\beta}}{\text{minimize/maximize}}, \quad f = f(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}) \\ & \text{subject to} \quad \boldsymbol{\beta} \in I(\boldsymbol{\eta}). \end{aligned} \quad (8)$$

The aleatory variables remain fixed during the BCO process, while the epistemic variables are allowed to change within the specified bounds. This completes the propagation of epistemic uncertainties. The BCO problem needs only a few exact function and gradient evaluations to reach the extremum.

3. OBTAIN SURROGATE TRAINING DATA: The exact function $f(\boldsymbol{\xi}, \boldsymbol{\eta})$ is evaluated at $(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{*(i)})$, $i = 1, \dots, N$, and the data is used to train the surrogate model. Note that, if the number of aleatory variables is large, the surrogate suffers from the curse of dimensionality, *i.e.*, tens of thousands of BCO results may be required as input training data to the surrogate.

4. PROPAGATE ALEATORY EFFECTS: Once the surrogate model is built using the training data, it can be probed inexpensively to yield the output statistics (e.g. mean μ_{f^*} and variance $\sigma_{f^*}^2$).

5. UPDATE OBJECTIVE/CONSTRAINTS: The aleatory statistics are used to update the objective function and constraints defined in section II.C.2.

Steps (1) to (5) are continued until meeting user-specified stopping criteria for the robust optimization loop.

II.C.4. Gradient Evaluation

ALEATORY GRADIENTS The gradient of the objective function with respect to design variables associated with aleatory uncertainties (random variables) is given by:²⁵

$$\frac{d\mathcal{J}}{d\boldsymbol{\xi}} = \frac{\partial \mathcal{J}}{\partial \mu_{f^*}} \frac{d\mu_{f^*}}{d\boldsymbol{\xi}} + \frac{\partial \mathcal{J}}{\partial \vartheta_{f^*}} \frac{d\vartheta_{f^*}}{d\boldsymbol{\xi}} = w_1 \frac{d\mu_{f^*}}{d\boldsymbol{\xi}} + w_2 \frac{d\vartheta_{f^*}}{d\boldsymbol{\xi}}, \quad (9)$$

where the mean μ_{f^*} and variance ϑ_{f^*} are computed using the surrogate model (kriging or polynomial chaos). The mean extremum of the simulation output is approximated as:

$$\mu_{f^*} \approx \frac{1}{\tilde{N}} \sum_{k=1}^{\tilde{N}} \widehat{f^*}(\boldsymbol{\alpha}^k). \quad (10)$$

The derivative of the mean extremum with respect to aleatory variables can be calculated as:

$$\frac{d\mu_{f^*}}{d\boldsymbol{\xi}} \approx \frac{1}{\tilde{N}} \sum_{k=1}^{\tilde{N}} \frac{d\widehat{f^*}(\boldsymbol{\alpha}^k)}{d\boldsymbol{\alpha}^k} \frac{d\boldsymbol{\alpha}^k}{d\boldsymbol{\xi}} = \frac{1}{\tilde{N}} \sum_{k=1}^{\tilde{N}} \frac{d\widehat{f^*}(\boldsymbol{\alpha}^k)}{d\boldsymbol{\alpha}^k}, \quad (11)$$

where $\frac{d\widehat{f^*}(\boldsymbol{\alpha}^k)}{d\boldsymbol{\alpha}^k}$ is obtained from the surrogate models. Likewise, the variance and its derivative can be approximated as follows:

$$\vartheta_{f^*} \approx \left(\frac{1}{\tilde{N}} \sum_{k=1}^{\tilde{N}} \widehat{f^*}^2(\boldsymbol{\alpha}^k) \right) - \mu_{f^*}^2 \quad (12)$$

$$\frac{d\vartheta_{f^*}}{d\boldsymbol{\xi}} \approx \left(\frac{2}{\tilde{N}} \sum_{k=1}^{\tilde{N}} \widehat{f^*}(\boldsymbol{\alpha}^k) \frac{d\widehat{f^*}(\boldsymbol{\alpha}^k)}{d\boldsymbol{\alpha}^k} \right) - 2\mu_{f^*} \frac{d\mu_{f^*}}{d\boldsymbol{\xi}}. \quad (13)$$

EPISTEMIC GRADIENTS The gradient of the objective function with respect to design variables associated with epistemic uncertainties (random variables) is given by:

$$\frac{d\mathcal{J}}{d\boldsymbol{\eta}} = \frac{\partial \mathcal{J}}{\partial \mu_{f^*}} \frac{d\mu_{f^*}}{d\boldsymbol{\eta}} + \frac{\partial \mathcal{J}}{\partial \vartheta_{f^*}} \frac{d\vartheta_{f^*}}{d\boldsymbol{\eta}} = w_1 \frac{d\mu_{f^*}}{d\boldsymbol{\eta}} + w_2 \frac{d\vartheta_{f^*}}{d\boldsymbol{\eta}}. \quad (14)$$

In this case, the calculation of $\frac{d\mu_{f^*}}{d\boldsymbol{\eta}}$ and $\frac{d\vartheta_{f^*}}{d\boldsymbol{\eta}}$ is not simple because moving the midpoint of the epistemic intervals will lead in general to different extrema for the training points and thus to a different surrogate model, which when sampled provides different values for μ_{f^*} and ϑ_{f^*} . In comparison, the aleatory gradient

was easy to obtain because the same surrogate model is used and only the change in sample points (random realizations) has to be accounted for. In this work the following approximations are used:²⁵

$$\frac{d\mu_{f^*}}{d\boldsymbol{\eta}} \approx \left. \frac{df}{d\boldsymbol{\eta}} \right|_{(\boldsymbol{\xi}=\bar{\boldsymbol{\xi}}, \boldsymbol{\eta}=\bar{\boldsymbol{\eta}})} \quad \text{and} \quad \frac{d\vartheta_{f^*}}{d\boldsymbol{\eta}} \approx 0, \quad (15)$$

i.e., the derivative of the mean extremum μ_{f^*} with respect to the epistemic variables $\boldsymbol{\eta}$, is approximated by the derivative of the function f with respect to $\boldsymbol{\eta}$, evaluated at the mean values of the aleatory variables and midpoints of the interval for the epistemic variables. Generally, this derivative is non-zero since for the epistemic optimizations via BCO, the extreme value is typically encountered at the interval bound. Since the variances are small in comparison with the mean values, their sensitivities are neglected: $\frac{d\vartheta_{f^*}}{d\boldsymbol{\eta}} \approx 0$.

III. Three-bar Truss Design

In this section the robust optimization of a three-bar truss is discussed.

III.A. Deterministic Problem

The truss shown in Figure 6 is subjected to a load inclined at an angle θ from the horizontal, which puts bars 1 and 2 under tension and bar 3 under compression. The nodes are represented with numbers 1 through 4. The goal is to minimize the total weight W of the structure. The mathematical formulation of the problem is given below in Eq (16).

$$\begin{aligned} \underset{\mathbf{d}}{\text{minimize}} \quad & W = \frac{A_1 \gamma H}{\sin(\phi_1)} + \frac{A_2 \gamma H}{\sin(\phi_2)} + \frac{A_3 \gamma H}{\sin(\phi_3)}, \\ \text{subject to} \quad & g_1 = \frac{\sigma_1}{\sigma_{1max}} - 1 \leq 0, \\ & g_2 = \frac{\sigma_2}{\sigma_{2max}} - 1 \leq 0, \\ & g_3 = \frac{\sigma_3}{\sigma_{3max}} - 1 \leq 0, \\ & g_4 = -\frac{\sigma_1}{\sigma_{1max}} - 1 \leq 0, \\ & g_5 = -\frac{\sigma_2}{\sigma_{2max}} - 1 \leq 0, \\ & g_6 = -\frac{\sigma_3}{\sigma_{3max}} - 1 \leq 0, \\ & g_7 = \frac{Q_{4x}}{Q_{4xmax}} - 1 \leq 0, \\ & g_8 = \frac{Q_{4y}}{Q_{4ymax}} - 1 \leq 0, \\ \text{bounds} \quad & 0.25 \text{ in}^2 \leq A_1, A_2, A_3 \leq 5.0 \text{ in}^2, \\ & 30^\circ \leq \phi_1 \leq 60^\circ, \\ & 60^\circ \leq \phi_2 \leq 120^\circ, \\ & 120^\circ \leq \phi_3 \leq 150^\circ. \end{aligned} \quad (16)$$

The problem has a total of six design variables $\mathbf{d} = [A_1, A_2, A_3, \phi_1, \phi_2, \phi_3]$, *i.e.*, the areas (A_1 , A_2 and A_3) and the orientations (ϕ_1 , ϕ_2 and ϕ_3) of the bars with respect to the horizontal. The structure has to be designed to withstand a total of 8 constraints $g_i(\mathbf{d})$. It is to be noted that the constraints are normalized with respect to their allowable values (denoted with subscript *max*). The first three, the next three, and the last two constraints impose tensile stress, compressive stress and displacement requirements, respectively. The axial stresses and nodal displacements used in Eq. (16) are calculated using a finite element procedure described in Appendix A. The other parameters used for this problem are listed in Table 2.

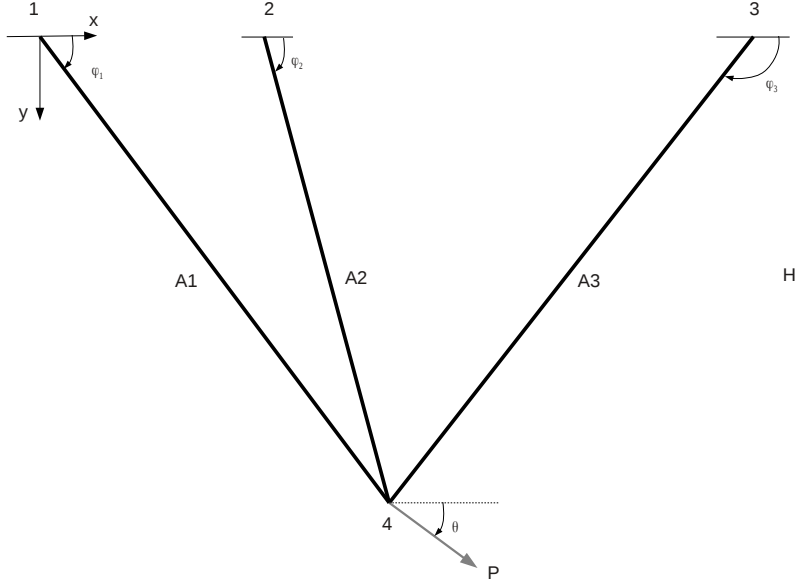


Figure 6: A schematic diagram of the three-bar truss structure.

Table 2: Design data for three-bar truss.

Quantity	Description	Value	Unit
P	Load	30000	lb
θ	Loading angle	50	deg
E	Young's modulus	10^7	psi
γ	Weight density	0.1	lb/in ³
H	Reference length (projection on y -axis)	10	in
$\sigma_{1_{max}}$	Allowable axial stress on bar 1	5000	psi
$\sigma_{2_{max}}$	Allowable axial stress on bar 2	10000	psi
$\sigma_{3_{max}}$	Allowable axial stress on bar 3	5000	psi
$u_{4x_{max}}$	Allowable x-displacement at 4	0.005	in
$u_{4y_{max}}$	Allowable y-displacement at 4	0.005	in
ϵ_1	Constraint violation tolerance	10^{-3}	-
ϵ_2	Norm of design change $\ \Delta \mathbf{d}\ $	10^{-3}	-

III.B. Robust Optimization Problem

The robust optimization problem involves minimizing the following objective function:

$$\underset{\xi, \eta}{\text{minimize}} \quad \mathcal{J} = \mu_W + \sigma_W^2, \quad (17)$$

$$\text{subject to} \quad g_i^r = \mu_{g_i} + k\sigma_{g_i} \leq 0, \quad \text{for } i = 1, \dots, 8,$$

i.e. the minimization of an equally weighted sum of the mean and variance of the weight subject to eight constraints. The area design variables A_i are assumed to have epistemic uncertainties with bounds $\tau_i = \pm 0.1 \text{ in}^2$ and the orientations ϕ_i are assumed to have aleatory uncertainties with standard deviation $\sigma_i = 1^\circ$. The input aleatory uncertainties are modeled as $\alpha^{(j)} \sim \mathcal{N}(\mu_{\phi_i}, \sigma_{\phi_i}^2)$ and the epistemic uncertainties are represented as an interval $\beta^{(j)} \in [A_i - \tau_i, A_i + \tau_i]$. All other input parameters are kept fixed throughout the optimization.

SURROGATE MODELS: The kriging surrogate model is built with seventy training points. The polynomial chaos surrogate is a fourth order polynomial with an oversampling factor of two which also requires seventy training points. The training points are chosen via the dynamic training point selection framework.^{17, 18} Note that each training data f^* comes from solving a BCO problem as discussed in Subsection II.B.2.

III.C. Optimization Results

III.C.1. Deterministic and Robust Designs

Table 3 compares the robust design optima with the deterministic optimum. From the optimum weights, it can be inferred that the deterministic design is the best in terms of lightness of the structure, but lacks robustness. A deterministic design with no assumed factor of safety is 15% lighter than a highly robust design specified by $k = 4$. However, a deterministic design with a small factor of safety of 1.3 is 29% heavier than a highly robust design specified by $k = 4$. The designers can capitalize on such a behavior for over-conservative designs that are in use today or the ones that need to be built in the future. A design corresponding to $k = 0$ with a weight of 14.65 ± 0.24 has a 50% chance of violating the constraints and is not as robust as a design corresponding to $k = 3$ with a weight of 16.54 ± 0.25 that has less than one percent probability of violating the constraints. As the desired robustness specified with k increases, an increase in the objective function value can be seen, meaning that robustness is obtained at the expense of additional weight to the structure. The designer can carry out a trade-off study between the weight of the structure and the required robustness specified with k or P_k . It can be seen that the kriging and polynomial chaos based results agree very closely with each other for all the tested cases.

Table 3: Optimization results for three-bar truss problem.

Type	k	P_k	A_1 <i>in</i> ²	A_2 <i>in</i> ²	A_3 <i>in</i> ²	ϕ_1 <i>deg</i>	ϕ_2 <i>deg</i>	ϕ_3 <i>deg</i>	μ_W <i>lb</i>	σ_W <i>lb</i>	C_v -	No. of F/FG Evals. & Iterations
Initial design	-	-	2.0	2.0	2.0	45.0	90.0	135.0	7.66	-	-	-
Det $F_s = 1.0$	-	-	5.00	1.42	2.30	37.6	60.0	150.0	14.45	-	-	108/108-12
Det $F_s = 1.3$	-	-	5.00	4.95	5.00	39.5	60.0	143.6	22.00	-	-	126/126-14
Robust-KR	0	0.5000	5.00	1.45	2.37	37.7	60.0	150.0	14.65	0.24	0.0162	17559/17559-12
Robust-PC	0	0.5000	5.00	1.45	2.37	37.7	60.0	150.0	14.65	0.24	0.0162	17615/17615-12
Robust-KR	1	0.8413	5.00	1.66	2.66	37.5	60.0	149.3	15.41	0.24	0.0159	21963/21963-14
Robust-PC	1	0.8413	5.00	1.66	2.66	37.5	60.0	149.3	15.41	0.24	0.0159	20555/20555-13
Robust-KR	2	0.9772	5.00	1.84	2.92	37.5	60.0	148.6	16.02	0.25	0.0155	23594/23594-13
Robust-PC	2	0.9772	5.00	1.84	2.92	37.5	60.0	148.6	16.02	0.25	0.0155	33555/33555-18
Robust-KR	3	0.9986	5.00	1.99	3.15	37.5	60.0	148.2	16.54	0.25	0.0153	20771/20771-12
Robust-PC	3	0.9986	5.00	1.99	3.15	37.5	60.0	148.2	16.54	0.25	0.0153	17938/17938-12
Robust-KR	4	0.9999	5.00	2.13	3.36	37.6	60.0	147.9	17.00	0.26	0.0151	31178/31178-17
Robust-PC	4	0.9999	5.00	2.13	3.36	37.6	60.0	147.9	17.00	0.26	0.0151	19500/19500-12

It can also be noted that area A_1 is pushed to its upper limit for all designs, while the other two areas (A_2 and A_3) and orientations generally govern the robustness of the structure.

Since the standard deviation is always associated with a mean it is advantageous to use a dimensionless number, the coefficient of variation C_v (also known as relative standard deviation or relative amount of uncertainty)² that measures the extent of variability in relation to the mean of the output. It can be used as a metric of comparison with different data sets or designs that involve different units, or different assumed input mean and standard deviations. A decrease in the coefficient of variation can be observed across the robust designs.

III.C.2. Simulation Requirements

Table 3 also presents the number of exact function and gradient evaluations needed to reach the final design. Here, each constraint evaluation is counted towards the total number of simulations. The box-constrained optimization takes 2 – 3 exact function and gradient evaluations for this test case. On average, the kriging

and polynomial chaos took roughly the same number of function and gradient evaluations to reach the optimum. Figure 7 plots the change in the objective function with the number of optimizer iterations.

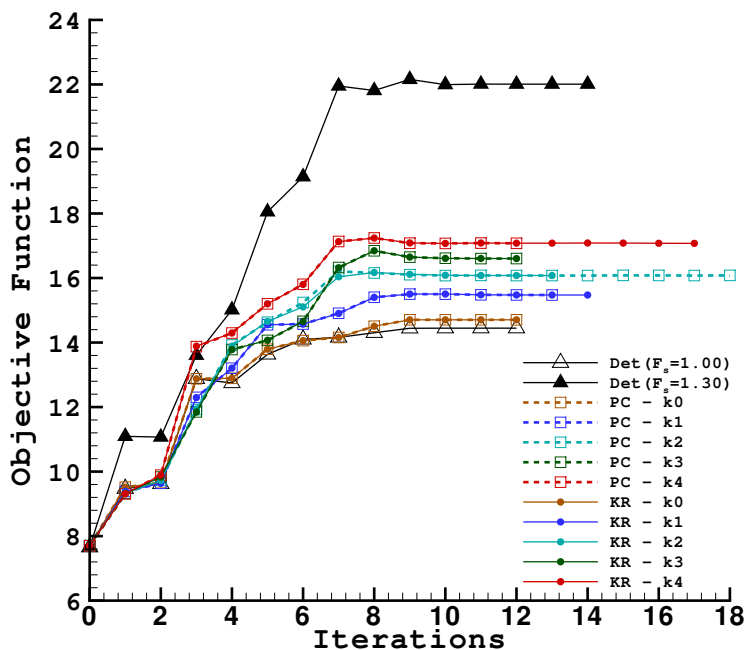


Figure 7: Change in the objective function with the number of optimizer iterations.

III.C.3. Constraint Status

Table 4: Constraint status for three-bar truss problem.

Type	k	g_1^r	g_2^r	g_3^r	g_4^r	g_5^r	g_6^r	g_7^r	g_8^r
Initial design	-	$0.12 \cdot 10^1$	$-0.43 \cdot 10^0$	$-0.21 \cdot 10^1$	$-0.32 \cdot 10^1$	$-0.16 \cdot 10^1$	$0.69 \cdot 10^{-1}$	$0.14 \cdot 10^0$	$0.23 \cdot 10^1$
Det $F_s = 1.0$	-	$-0.14 \cdot 10^0$	$-0.41 \cdot 10^0$	$-0.12 \cdot 10^1$	$-0.19 \cdot 10^1$	$-0.16 \cdot 10^1$	$0.82 \cdot 10^0$	$-0.48 \cdot 10^{-8}$	$-0.17 \cdot 10^{-8}$
Det $F_s = 1.3$	-	$-0.29 \cdot 10^0$	$-0.58 \cdot 10^0$	$-0.13 \cdot 10^1$	$-0.17 \cdot 10^1$	$-0.14 \cdot 10^1$	$0.71 \cdot 10^0$	$-0.46 \cdot 10^0$	$-0.34 \cdot 10^{-7}$
Robust-KR	0	$-0.15 \cdot 10^0$	$-0.41 \cdot 10^0$	$-0.12 \cdot 10^1$	$-0.18 \cdot 10^1$	$-0.16 \cdot 10^1$	$-0.82 \cdot 10^0$	$-0.56 \cdot 10^{-6}$	$-0.87 \cdot 10^{-5}$
Robust-PC	0	$-0.15 \cdot 10^0$	$-0.41 \cdot 10^0$	$-0.12 \cdot 10^1$	$-0.18 \cdot 10^1$	$-0.16 \cdot 10^1$	$-0.82 \cdot 10^0$	$-0.84 \cdot 10^{-5}$	$-0.78 \cdot 10^{-5}$
Robust-KR	1	$-0.17 \cdot 10^0$	$-0.42 \cdot 10^0$	$-0.11 \cdot 10^1$	$-0.18 \cdot 10^1$	$-0.15 \cdot 10^1$	$-0.79 \cdot 10^0$	$0.90 \cdot 10^{-4}$	$0.49 \cdot 10^{-4}$
Robust-PC	1	$-0.17 \cdot 10^0$	$-0.42 \cdot 10^0$	$-0.11 \cdot 10^1$	$-0.18 \cdot 10^1$	$-0.15 \cdot 10^1$	$-0.79 \cdot 10^0$	$-0.12 \cdot 10^{-4}$	$0.38 \cdot 10^{-4}$
Robust-KR	2	$-0.19 \cdot 10^0$	$-0.43 \cdot 10^0$	$-0.11 \cdot 10^1$	$-0.18 \cdot 10^1$	$-0.15 \cdot 10^1$	$0.77 \cdot 10^0$	$-0.73 \cdot 10^{-4}$	$0.13 \cdot 10^{-3}$
Robust-PC	2	$-0.19 \cdot 10^0$	$-0.43 \cdot 10^0$	$-0.11 \cdot 10^1$	$-0.18 \cdot 10^1$	$-0.15 \cdot 10^1$	$0.77 \cdot 10^0$	$-0.16 \cdot 10^{-4}$	$-0.12 \cdot 10^{-3}$
Robust-KR	3	$-0.20 \cdot 10^0$	$-0.43 \cdot 10^0$	$-0.11 \cdot 10^1$	$-0.17 \cdot 10^1$	$-0.15 \cdot 10^1$	$0.76 \cdot 10^0$	$0.28 \cdot 10^{-3}$	$-0.12 \cdot 10^{-3}$
Robust-PC	3	$-0.20 \cdot 10^0$	$-0.43 \cdot 10^0$	$-0.11 \cdot 10^1$	$-0.17 \cdot 10^1$	$-0.15 \cdot 10^1$	$0.76 \cdot 10^0$	$0.16 \cdot 10^{-3}$	$-0.76 \cdot 10^{-4}$
Robust-KR	4	$-0.21 \cdot 10^0$	$-0.43 \cdot 10^0$	$-0.11 \cdot 10^1$	$-0.17 \cdot 10^1$	$-0.14 \cdot 10^1$	$-0.74 \cdot 10^{-3}$	$0.90 \cdot 10^{-4}$	$0.20 \cdot 10^{-3}$
Robust-PC	4	$-0.21 \cdot 10^0$	$-0.43 \cdot 10^0$	$-0.11 \cdot 10^1$	$-0.17 \cdot 10^1$	$-0.14 \cdot 10^1$	$-0.74 \cdot 10^{-3}$	$0.75 \cdot 10^{-3}$	$-0.36 \cdot 10^{-3}$

Table 4 displays the status of all eight constraints at the initial design, deterministic optimum, and robust optimum. Here, a positive value for g represents a constraint-violation, whereas a negative value means that the constraint is satisfied. It can be inferred that constraints 1, 7 and 8 are the ones that significantly affect the design throughout the optimization (tight or ϵ -active constraints). The kriging and polynomial chaos based values are a little different for these tight constraints, yet within the specified tolerance to ensure that the constraints are not violated. All other constraints are inactive and both surrogate models provide the same values for these constraints.

III.C.4. Output PDF and CDF

Figures 8 and 9 show the probability density function (PDF) and cumulative distribution function (CDF) of the objective function (weight) as well as the constraints (normalized) at their optimum designs. As the desired robustness specified with k increases, an increase in the objective function value by means of a shift to the right can be seen. The robust optimization problem formulation serves to move the constraint values k standard deviations away from a potential violation which is evident from the PDF and CDF of the constraints. It can be seen that a design corresponding to $k = 4$ has less than a 1% chance of constraint violation, whereas $k = 0$ features a 50% chance of constraint violation due to the effect of input uncertainties. It can also be seen that the spread of values is reduced as the robustness increases (compare PDFs of $k = 0$ and $k = 4$ cases), which shows that the design is less sensitive to uncertainties/variations in the input. Overall, both surrogate models produce comparable distributions apart from occasional differences.

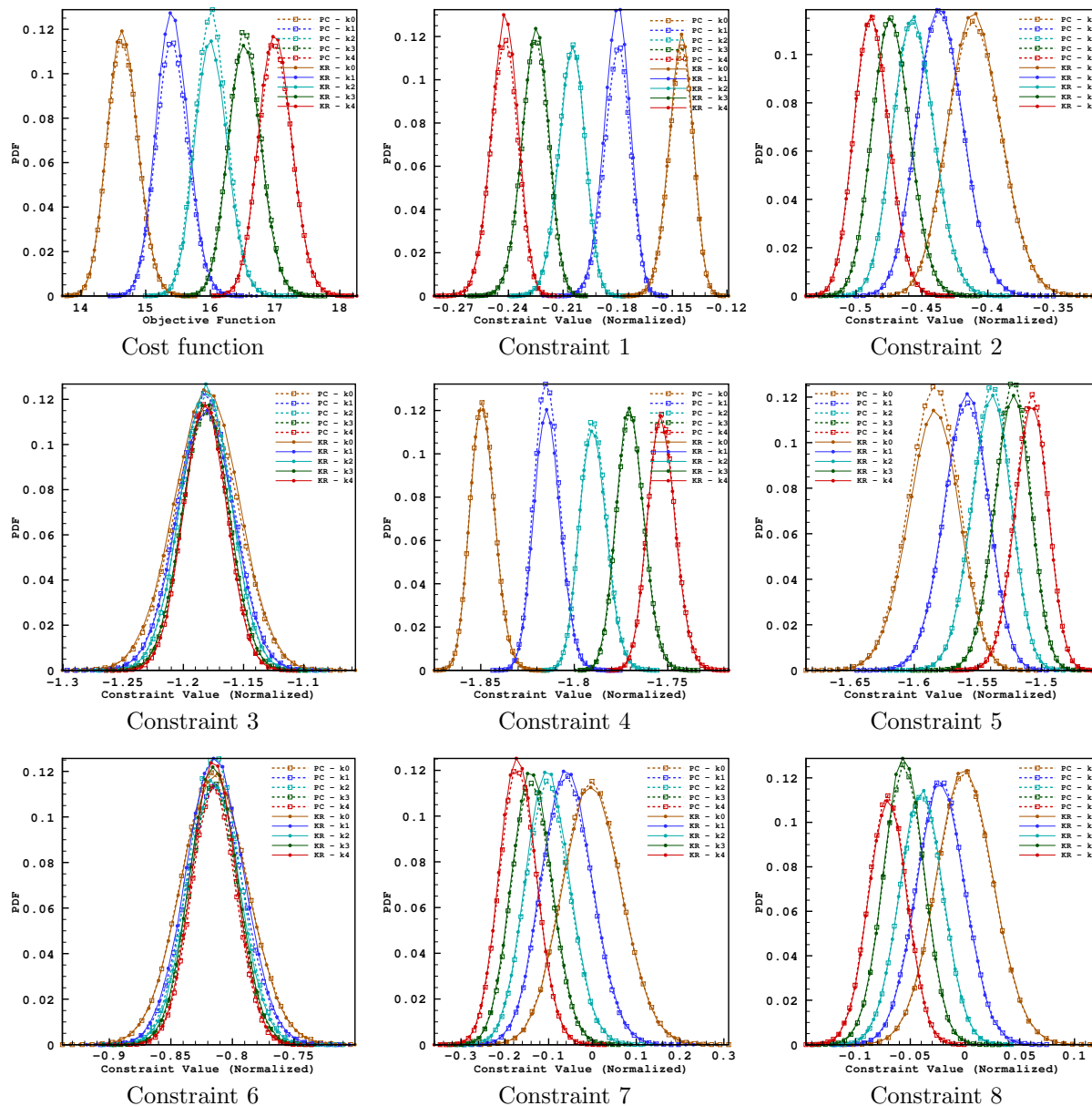


Figure 8: Probability density function of objective and constraint functions at robust optimum designs.

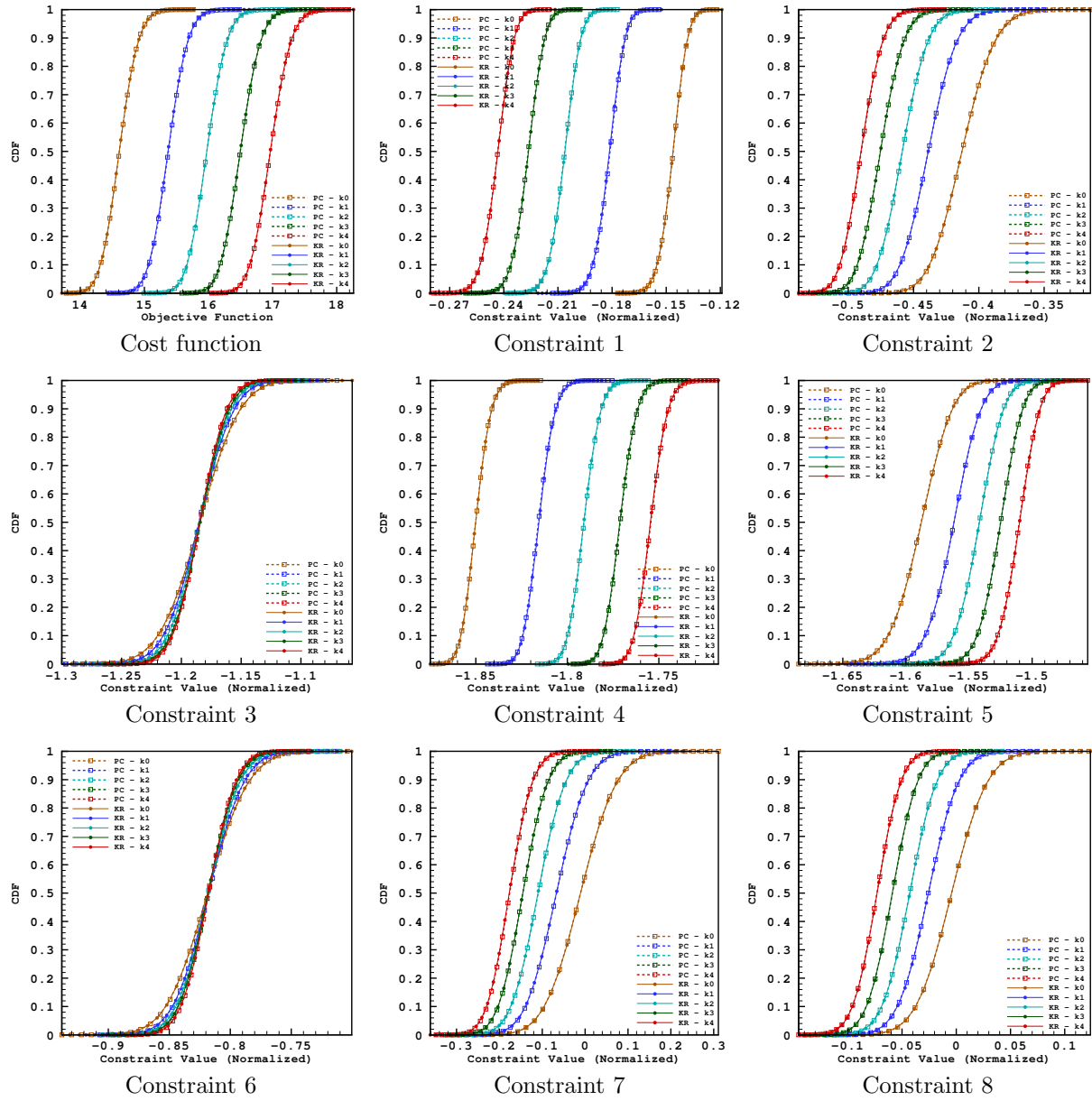


Figure 9: Cumulative distribution function of objective and constraint functions at robust optimum designs.

IV. Cantilever Beam Design

This section describes the robust optimization of a cantilever beam.

IV.A. Problem Description

A cantilever beam of rectangular cross-section is subjected to a bending moment \mathcal{M} ($N \cdot mm$) and shear force \mathcal{V} (N). The bending stress in the beam is calculated as $\sigma = 6\mathcal{M}/bd^2$ (N/mm^2) and the average shear stress is calculated as $\tau = 3\mathcal{V}/2bd$ (N/mm^2), where b is the width and d is the depth of the beam. The maximum allowable stress in the form of bending, σ_{allow} , is $10 N/mm^2$ and the maximum allowable shear, τ_{allow} , is $2 N/mm^2$. The goal is to minimize the cross-sectional area A (mm^2) of the beam. The mathematical

formulation of the problem is given below:

$$\begin{aligned}
& \underset{b,d}{\text{minimize}} && A(b,d) = bd, \\
& \text{subject to} && g_1(b,d,\mathcal{M}) = \frac{6\mathcal{M}}{bd^2\sigma_{allow}} - 1 \leq 0, \\
& && g_2(b,d,\mathcal{V}) = \frac{3\mathcal{V}}{2bd\tau_{allow}} - 1 \leq 0, \\
& && g_3(b,d) = \frac{d}{2b} - 1 \leq 0, \\
& \text{bounds} && 100 \text{ mm} \leq b,d \leq 600 \text{ mm},
\end{aligned} \tag{18}$$

where the constraints g_1 and g_2 enforce bending and shear stress requirements, respectively, while g_3 imposes an aspect-ratio requirement for the rectangular cross-section. All the constraints are represented in standard normalized form. The design variables are the width and depth of the beam.

IV.B. Robust Optimization Problem

The two allowable stresses (σ_{allow} and τ_{allow}) in Eq. (18) are assumed to be precise (hence kept fixed) whereas the remaining parameters are assumed to have uncertainties and are therefore treated as random variables as listed in Table 5. The bending moment and shear force are assumed to have normally distributed aleatory

Table 5: Data and assumed uncertain parameters for cantilever beam design problem.

Random Variable	Description	Uncertainty Type	τ_{min}	τ_{max}	Mean	Standard Deviation	Unit
b	Width	Epistemic	-10	10	-	-	mm
d	Depth	Epistemic	-10	10	-	-	mm
\mathcal{M}	Bending Moment	Aleatory	-	-	$40 \cdot 10^6$	40000	$N \cdot mm$
\mathcal{V}	Shear Force	Aleatory	-	-	$150 \cdot 10^3$	1500	N

uncertainties with specified mean and standard deviations as shown in Table 5. Only the constraints g_1 and g_2 , which are functions of the bending moment and shear force, are influenced by these aleatory variables. Unlike the three-bar truss problem where all random variables were also considered as design variables, the cantilever beam problem involves random variables which are not design variables, but their effects will be considered in the optimization procedure. The robust optimization problem can be written as:

$$\begin{aligned}
& \underset{b,d}{\text{minimize}} && A(b,d) = \mu_A + \sigma_A^2, \\
& \text{subject to} && g_1^r(b,d,\mathcal{M}) = \mu_{g_1} + k\sigma_{g_1} \leq 0, \\
& && g_2^r(b,d,\mathcal{V}) = \mu_{g_2} + k\sigma_{g_2} \leq 0, \\
& && g_3^r(b,d) = \mu_{g_3} + k\sigma_{g_3} \leq 0.
\end{aligned} \tag{19}$$

In this problem only the epistemic random variables (width and depth) govern the cost function and the aspect-ratio constraint and therefore the output standard deviations are unavailable for these functions: $\sigma_A = \sigma_{g_3} = 0$.

SURROGATE MODELS: The robust optimization results will be compared using both kriging and polynomial chaos. The kriging surrogate model is built with 20 training points and the polynomial chaos metamodel is a third order polynomial which is also built with 20 training points.

IV.C. Optimization Results

Table 6 presents the optimization results. It can be seen that the objective function value increases with the desired robustness, for example, the cross-sectional area increases by roughly 17% for a design corresponding to $k = 4$ compared to a deterministic design with no factor of safety. However, the robust design ($k = 4$) has 29% less cross-sectional area (hence lighter) than a deterministic design with a factor of safety of 1.5.

Figure 10 shows all three constraints plotted along with the objective function contours. The objective function is parallel to the constraint g_2 , therefore, any point on the curve A–B is a feasible deterministic

Table 6: Optimization results for cantilever beam design problem.

Type	k	P_k	Width b mm	Depth d mm	Area A $\cdot 10^3 \text{ mm}^2$	No. of F/FG Evals. & Iterations
Initial Design	-	-	300	300	90.0	-
Det ($F_s = 1.0$)	-	-	335.5	335.4	112.5	33/33-7
Det ($F_s = 1.5$)	-	-	595.5	283.4	168.7	45/45-8
Robust-KR	0	0.5000	347.4	343.4	126.3	7046/3523-7
Robust-PC	0	0.5000	347.4	343.4	126.3	7917/7917-8
Robust-KR	1	0.8413	349.7	344.5	127.5	7146/3573-7
Robust-PC	1	0.8413	349.7	344.5	127.5	8037/8037-8
Robust-KR	2	0.9772	398.5	305.4	128.8	7686/3843-7
Robust-PC	2	0.9772	398.5	305.4	128.8	9661/9661-9
Robust-KR	3	0.9986	386.5	317.8	130.0	8694/4347-8
Robust-PC	3	0.9986	386.5	317.8	130.0	11669/11669-10
Robust-KR	4	0.9999	356.6	347.5	131.1	7286/3643-7
Robust-PC	4	0.9999	356.6	347.5	131.1	8196/8196-8

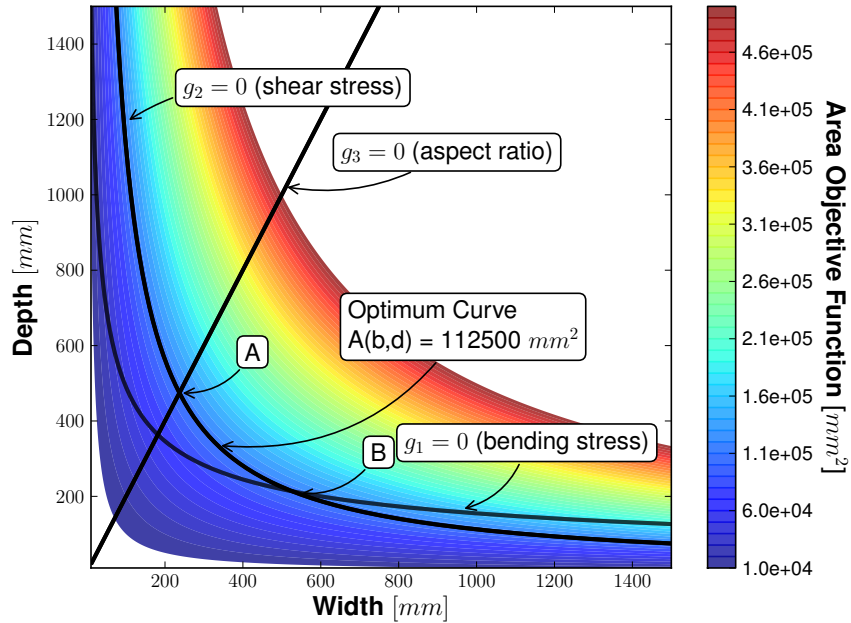


Figure 10: Graphical solution to the minimum area beam design problem.

optimum. At point A, the constraints g_2 and g_3 are active; at point B, the constraints g_1 and g_2 are active; while any point on the curve A–B has the constraint g_2 active. Through robust optimization, the optimum solution is moved by a distance of k standard deviations away from the deterministic solution, which is shown by an increment in the objective function values in Table 6. However, the robust optimization accounts for the exact amount of uncertainty in the problem and achieves a reduced cost function compared to deterministic designs employing an arbitrary factor of safety which could easily be over- or under-conservative.

IV.C.1. Simulation Requirements

The box-constrained optimizations took only 2 to 3 function and gradient evaluations to reach the extremum. On average, the robust optimization takes about 7500 function and gradient evaluations (including the constraint evaluations) and the simulation requirements for the polynomial chaos method are roughly 20% higher than that of kriging.

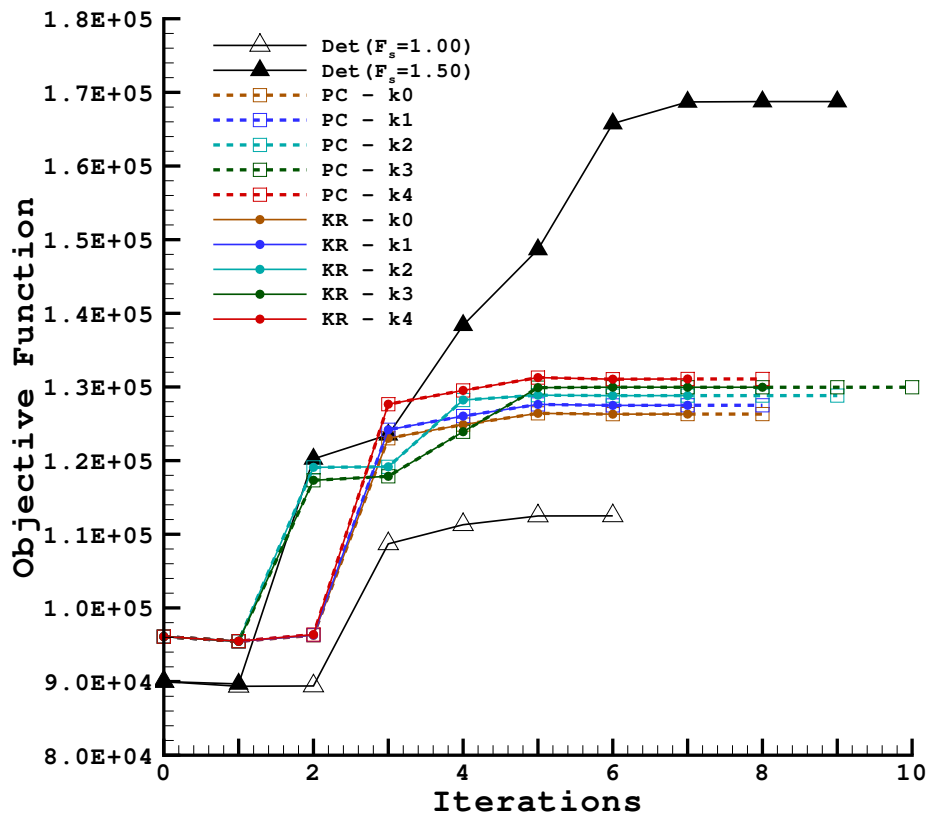


Figure 11: Optimization history for the beam design problem.

Figure 11 plots the change in the objective function with the number of optimization iterations for the different tested cases (deterministic and robust).

IV.C.2. Output PDF and CDF

Figure 12 shows the probability density and cumulative distribution functions of constraints g_1 and g_2 . The PDFs show the spread of possible values taken by the constraints corresponding to different robust designs, whereas the CDFs show the probability of obtaining a specified value or less. Knowing the spread of values helps a designer to make informed decisions about the performance of the design. For example, a robust design corresponding to $k = 4$ (red lines) features negligible occurrence of $g_i > 0$ (signifies a constraint violation). It can also be observed that the constraint values are normally distributed.

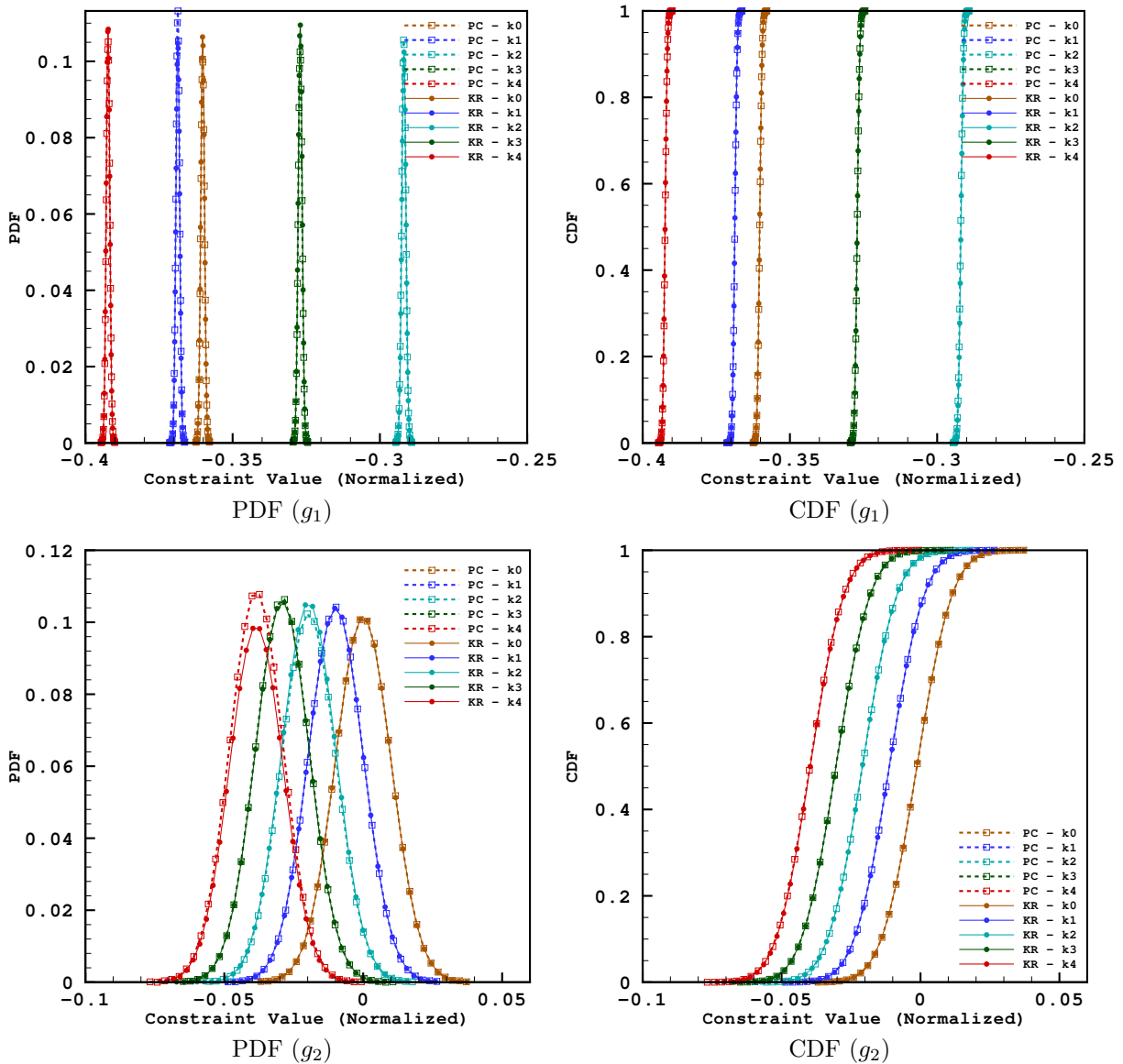


Figure 12: Output PDF (left) and CDF (right) of constraint g_1 and g_2 .

V. Airfoil Design

In this section the robust optimization of an airfoil is discussed.

V.A. Aerodynamic Analysis

The steady inviscid flow around an airfoil governed by the Euler equations is solved by using a second-order accurate finite-volume approach.^{31,32} The computational mesh is shown in Figure 13. Hicks-Henne sine bump functions³³ are used to control the shape of the airfoil resulting from perturbations of shape design variables. The resulting deformation of the mesh is calculated via a linear tension spring analogy.^{34,35}

V.B. Robust Optimization Problem

Seven shape design variables are placed on the upper surface and seven on the lower surface of the airfoil (at 20%, 30%, 40%, 50%, 60%, 80%, and 90% chord). The bounds on the flow variables, angle of attack and Mach number, are taken as $0^\circ \leq \alpha \leq 4^\circ$ and $0.6 \leq M_\infty \leq 0.78$. All fourteen shape design variables are assumed

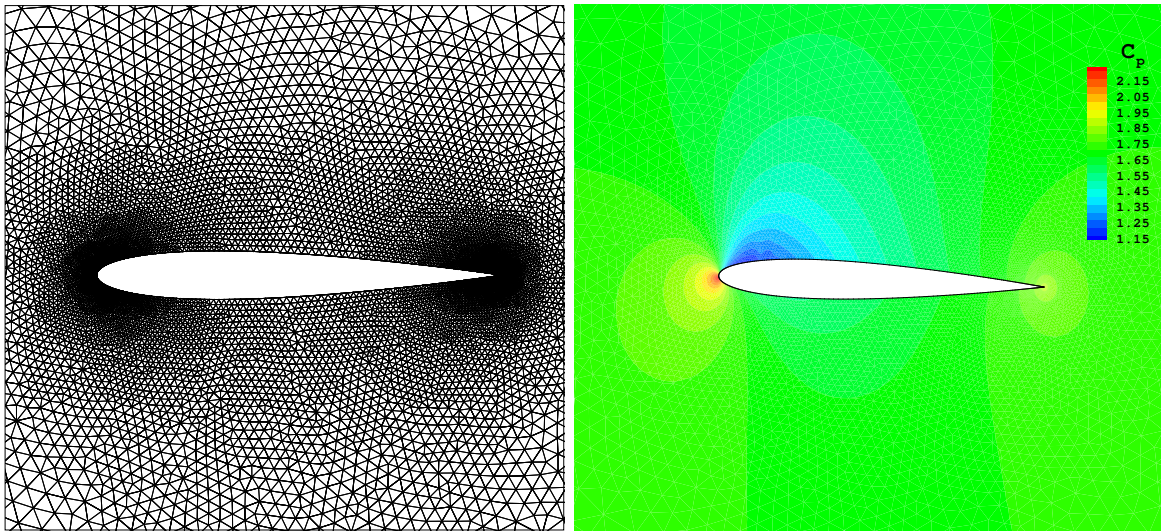


Figure 13: Computational mesh for NACA 0012 airfoil with 19,548 elements (left), pressure distribution at $\alpha = 2.0^\circ$ and $M_\infty = 0.65$ (right).

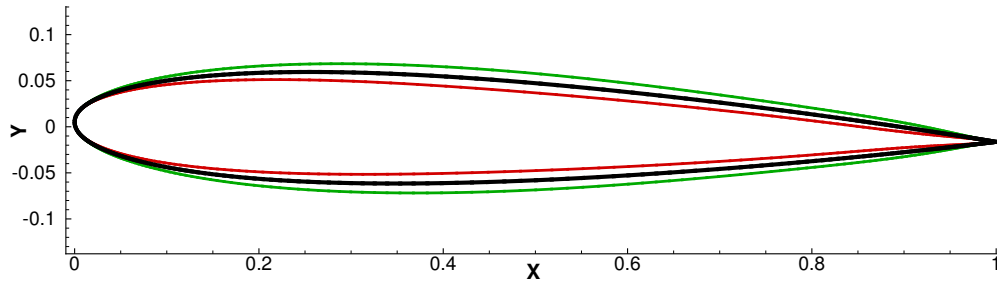


Figure 14: The NACA 0012 airfoil (in black) and airfoils resulting from perturbations of ± 0.0025 (in red and green).

to have epistemic uncertainties and the two flow variables are assumed to have aleatory uncertainties.

Table 7: Data for robust optimization of airfoil.

Random Variable	Description	Uncertainty Type	τ_{min}	τ_{max}	Standard Deviation
$\eta_{1,2,13,14}$	Shape design variables	Epistemic	-0.00125	0.00125	-
η_{3-12}	Shape design variables	Epistemic	-0.01	0.01	-
ξ_α	Angle of attack	Aleatory	-	-	0.1°
ξ_{M_∞}	Mach number	Aleatory	-	-	0.01

Figure 14 shows the baseline NACA 0012 airfoil used as the initial (starting) design and the airfoils resulting from perturbations of the fourteen shape design variables within the bounds specified in Table 7. The initial value of the angle of attack is 2° and the Mach number is 0.65. The mathematical formulation of the problem is given as follows,

$$\begin{aligned} & \underset{\xi, \eta}{\text{minimize}} && \mathcal{J} = \mu_{C_{d_{max}}} + \sigma_{C_{d_{max}}}^2, \\ & \text{subject to} && g = (\mu_{C_{l_{min}}} + k\sigma_{C_{l_{min}}}) - C_l^+ \geq 0, \end{aligned} \quad (20)$$

where C_l^+ refers to a target lift coefficient of 0.6, and $C_{l_{min}}$ and $C_{d_{max}}$ are the least possible lift and highest possible drag within the specified set of epistemic variables, respectively.

SURROGATE MODELS: The kriging surrogate model is built with 11 training points. The polynomial chaos surrogate is a second-order polynomial with an over sampling factor of two forming a regression surface which requires 12 training points. The surrogate models are built only over the assumed aleatory variables: the angle of attack and Mach number. The domain of the surrogate model is three standard deviations wide from the mean values ξ provided by the main optimizer (IPOPT) at every iteration.

V.C. Optimization Results

Table 8: Optimization results for airfoil design problem.

Type	k	P_k	$\mu_{c_{d_{max}}}$	$\sigma_{c_{d_{max}}}^2$	$\mu_{c_{l_{min}}}$	$\sigma_{c_{l_{min}}}$	α	M_∞	No. of F/FG Evals. & Iterations
Initial	-	-	$4.72 \cdot 10^{-4}$	-	0.335	-	2.000°	0.650	
Deterministic	-	-	$1.17 \cdot 10^{-3}$	-	0.600	-	2.510°	0.600	49/49-24
Robust-KR	0	0.5000	$2.72 \cdot 10^{-3}$	$2.03 \cdot 10^{-7}$	0.600	$1.84 \cdot 10^{-2}$	2.013°	0.600	844/844-23
Robust-PC	0	0.5000	$2.62 \cdot 10^{-3}$	$5.80 \cdot 10^{-8}$	0.600	$1.82 \cdot 10^{-2}$	2.389°	0.600	675/6751-16
Robust-KR	1	0.8413	$2.93 \cdot 10^{-3}$	$3.07 \cdot 10^{-7}$	0.619	$1.86 \cdot 10^{-2}$	2.065°	0.600	434/434-13
Robust-PC	1	0.8413	$2.73 \cdot 10^{-3}$	$2.50 \cdot 10^{-7}$	0.618	$1.84 \cdot 10^{-2}$	3.058°	0.600	434/434-15
Robust-KR	2	0.9772	$3.10 \cdot 10^{-3}$	$4.46 \cdot 10^{-7}$	0.637	$1.88 \cdot 10^{-2}$	2.179°	0.600	831/831-19
Robust-PC	2	0.9772	$3.20 \cdot 10^{-3}$	$8.58 \cdot 10^{-7}$	0.637	$1.89 \cdot 10^{-2}$	2.193°	0.600	710/710-22
Robust-KR	3	0.9986	$3.28 \cdot 10^{-3}$	$6.23 \cdot 10^{-7}$	0.657	$1.90 \cdot 10^{-2}$	2.301°	0.600	650/650-21
Robust-PC	3	0.9986	$3.25 \cdot 10^{-3}$	$9.83 \cdot 10^{-7}$	0.658	$1.92 \cdot 10^{-2}$	2.352°	0.600	1145/1145-21
Robust-KR	4	0.9999	$3.56 \cdot 10^{-3}$	$9.50 \cdot 10^{-7}$	0.677	$1.93 \cdot 10^{-2}$	2.421°	0.600	620/620-15
Robust-PC	4	0.9999	$3.65 \cdot 10^{-3}$	$1.25 \cdot 10^{-6}$	0.677	$1.93 \cdot 10^{-2}$	2.427°	0.600	2104/2104-36

Table 8 compares the robust design optima with the deterministic optimum. The average drag and mean angle of attack increase as the desired probability of achieving the target lift C_l^+ is increased. The optimum solution is sought from the optimizer at a distance of k -standard deviations away from the lift-constraint hyperplane. As a result, the amount of lift produced is higher as more robustness is expected from the design *i.e.*, the additional lift produced defines the robustness of the design and the design will be less prone to failure (violation of the lift-constraint). On the contrary, during a deterministic optimization an optimum design is sought at the constraint boundary, that can very well violate design requirements when the underlying variables are not representative of the ones considered during optimization. Another inference is that robustness is achieved at the expense of the objective function (drag penalty). Also by observing the optimum aleatory variables on the right, it can be seen that the Mach number remains the same for all designs (at its lower bound), whereas the angle of attack varies.

V.C.1. Airfoil shape

Figure 15 shows the original, deterministic and robustly optimized ($k = 4$, with polynomial chaos) airfoils. It can be inferred that the deterministically optimized airfoil (shown in blue) is thinner than the robustly optimized airfoil (shown in red).

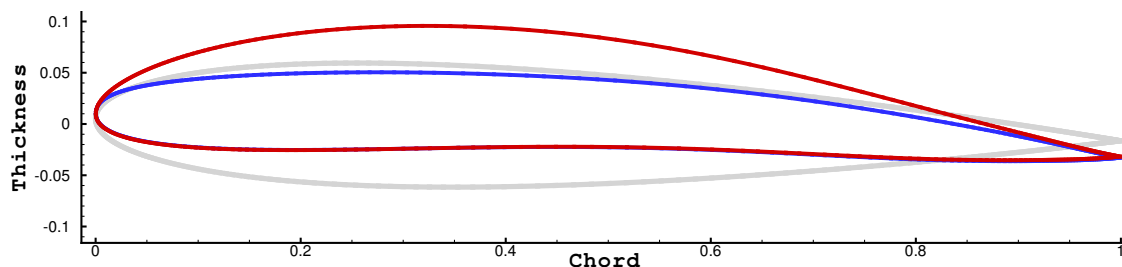


Figure 15: Original NACA 0012 (gray), deterministic (blue) and robust with $k = 4$ (red) airfoils produced using polynomial chaos. The kriging produced very similar airfoils (hence not shown).

The robust airfoils corresponding to increasing k are shown in Figure 16. Except for the first two cases ($k = 0$ and $k = 1$), the epistemic shape design variables attain similar values for both models. For the $k = 1$

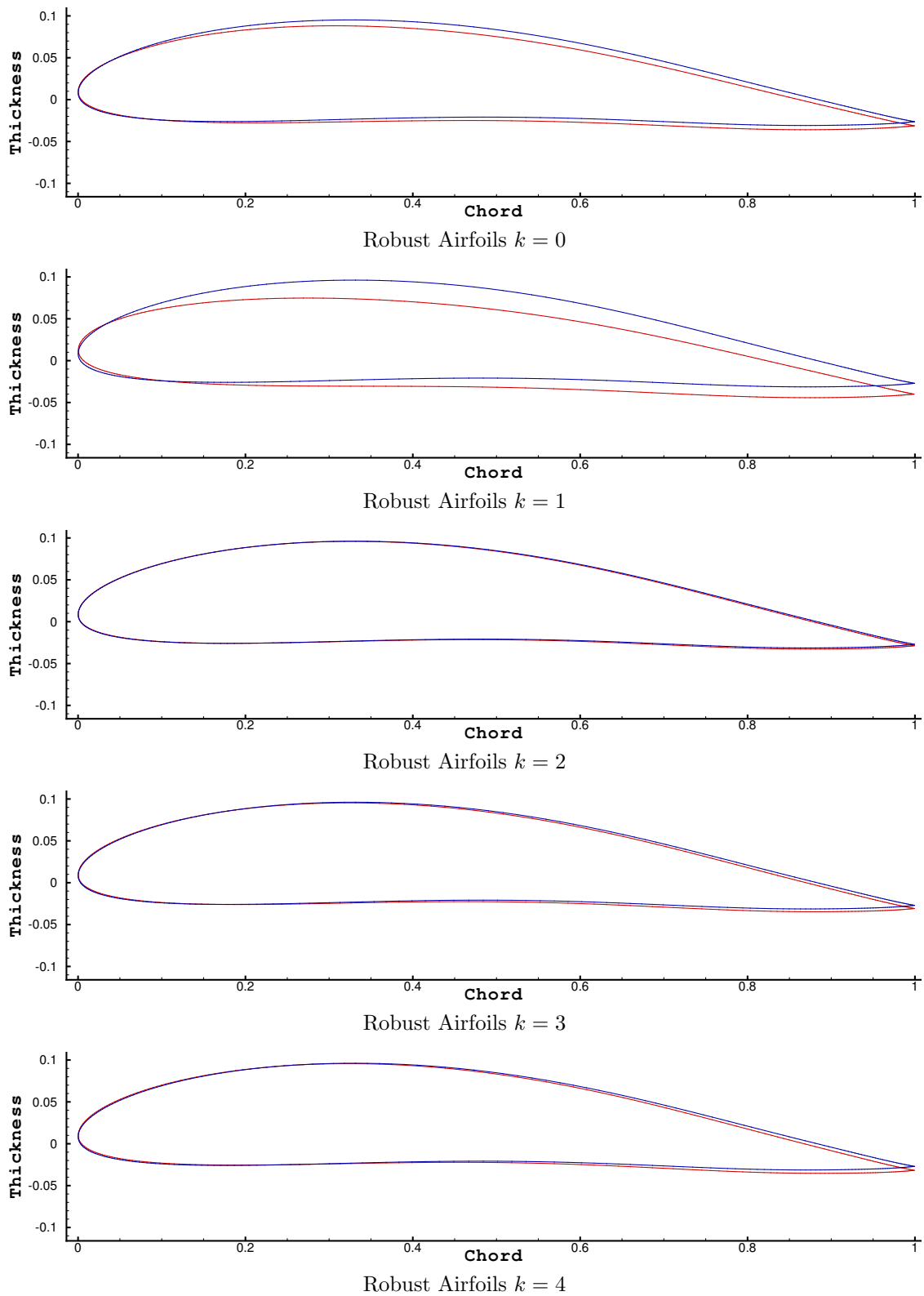


Figure 16: Plots showing the shape (also angle of attack) of different robust airfoils. Red and blue lines correspond to polynomial chaos and kriging, respectively.

case, the kriging based robust airfoil is thicker and has a lower angle of attack, whereas the polynomial chaos based airfoil is thinner and attains the target lift with an increased angle of attack.

Overall, the kriging surrogate based robust designs (shown as blue lines) tend to have a lower angle of attack than its polynomial chaos counterpart (shown as red line). This behavior can be observed across all five robust designs. In general, it may be advantageous for an airplane to fly faster rather than having an increased angle of attack for generating more lift. The solution to the Euler equations ignores important viscous effects, such as boundary layers, wakes and flow separation. If a Navier-Stokes solver is used, it can be expected that the optimizer places a greater emphasis on the shape optimization rather than the flow parameter optimization.

V.C.2. Simulation Requirements

The last column of Table 8 presents the number of function and gradient evaluations needed as well as the number of iterations taken by the optimizer to converge. Here a single flow solve provides the lift (constraint) and drag (objective) values. For this airfoil optimization test case, the robust optimization needs on average roughly 1000 flow and adjoint solutions, compared to close to 50 evaluations needed for the deterministic optimization, placing a roughly 20 times higher simulation requirement on the designer. The box-constrained optimizations took 2 to 3 flow and adjoint solutions to determine the worst possible lift and the highest possible drag within the specified epistemic uncertainty bounds. For the aleatory uncertainty propagation, though kriging and polynomial chaos involve similar amounts of training information (eleven and twelve points, respectively), at the end of the main optimization, the latter's simulation requirements are 50% higher than that of the kriging (on average). This shows that kriging is more effective for non-smooth functions such as this aerodynamic test case.

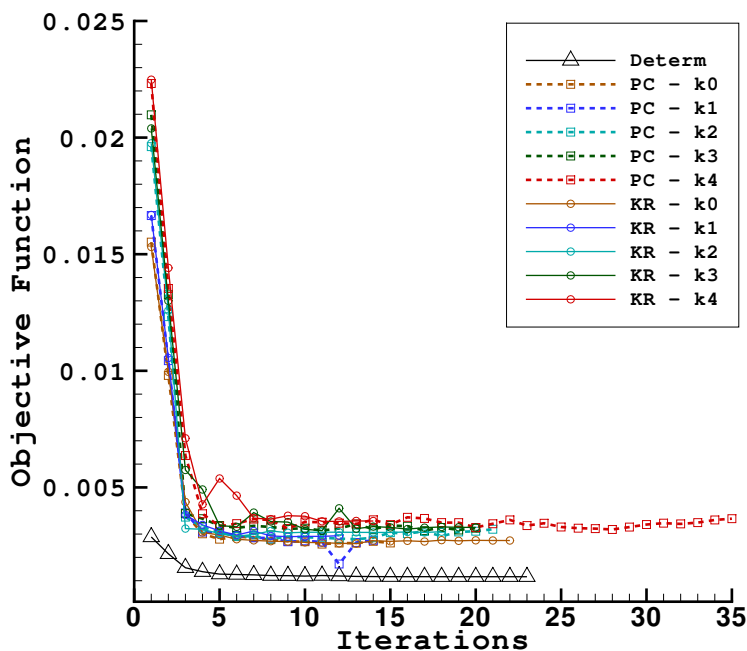


Figure 17: Optimizer iteration history for airfoil design problem.

Figure 17 plots the change in the objective function with the number of optimizer iterations.

V.C.3. Output PDF and CDF at the Optimum

Figures 18 and 19 show the PDF as well as CDF of the drag and lift coefficients, respectively, at several robust designs using kriging and polynomial chaos. The PDFs shown in the left show the distribution of possible drag and lift coefficient values due to the effect of uncertainties, whereas the CDFs show the probability of obtaining a specified value or less. For example, the distribution of drag (see the left of Figure 18) helps the designer to construct confidence bounds on possible drag values. Similarly, the probability that the target

lift coefficient ($C_L^+ = 0.6$) is not attained is 50% for the $k = 0$ case and is less than 1% for the $k = 4$ case. As the required robustness increases, the distributions shift to the right, which signifies a higher lift generation as well as drag penalty. A Gaussian distribution is seen for the lift coefficient, whereas the distribution of drag coefficient resembles a log-normal one.

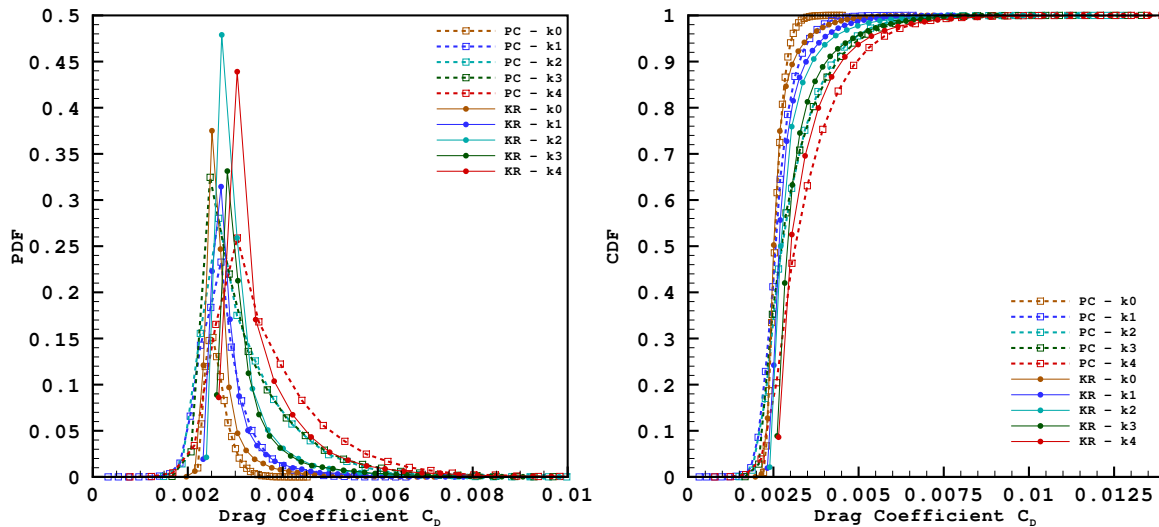


Figure 18: PDFs and CDFs of the drag coefficient at optimum designs.

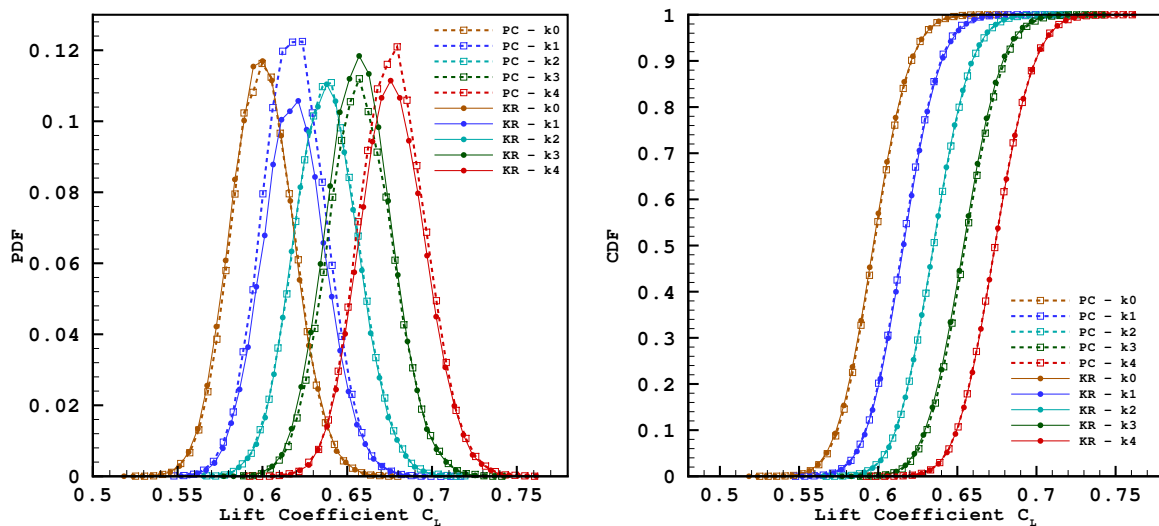


Figure 19: PDFs and CDFs of the lift coefficient at optimum designs.

V.C.4. Pressure Contours at the Optimum

Figures 20 and 21 show the pressure distribution around deterministically and robustly optimized airfoils using kriging and polynomial chaos, respectively. It can be seen that the pressure distributions are very similar among the robust airfoils, whereas a distinct difference can be noticed between the robust and deterministic ones.

V.D. Validation with Exact Monte Carlo Simulation

Here, validations for the IMCS-BCO approach are provided by a selective comparison of the $k = 1$ case with exact Monte Carlo simulation (MCS) and BCO *i.e.*, the surrogate models are replaced with exact function evaluations (Euler flow solutions). Due to the expense of the Euler flow solutions, only 3000 Monte Carlo

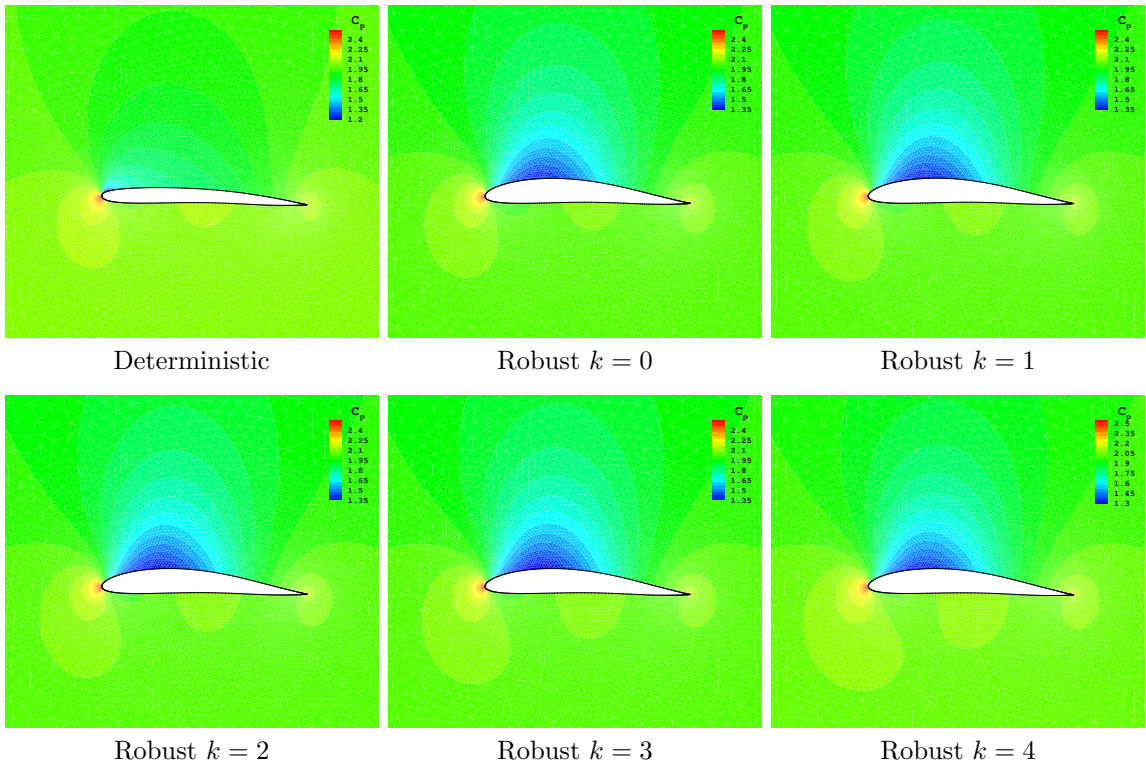


Figure 20: Contour plots of pressure coefficients C_p at different optimum designs using kriging.

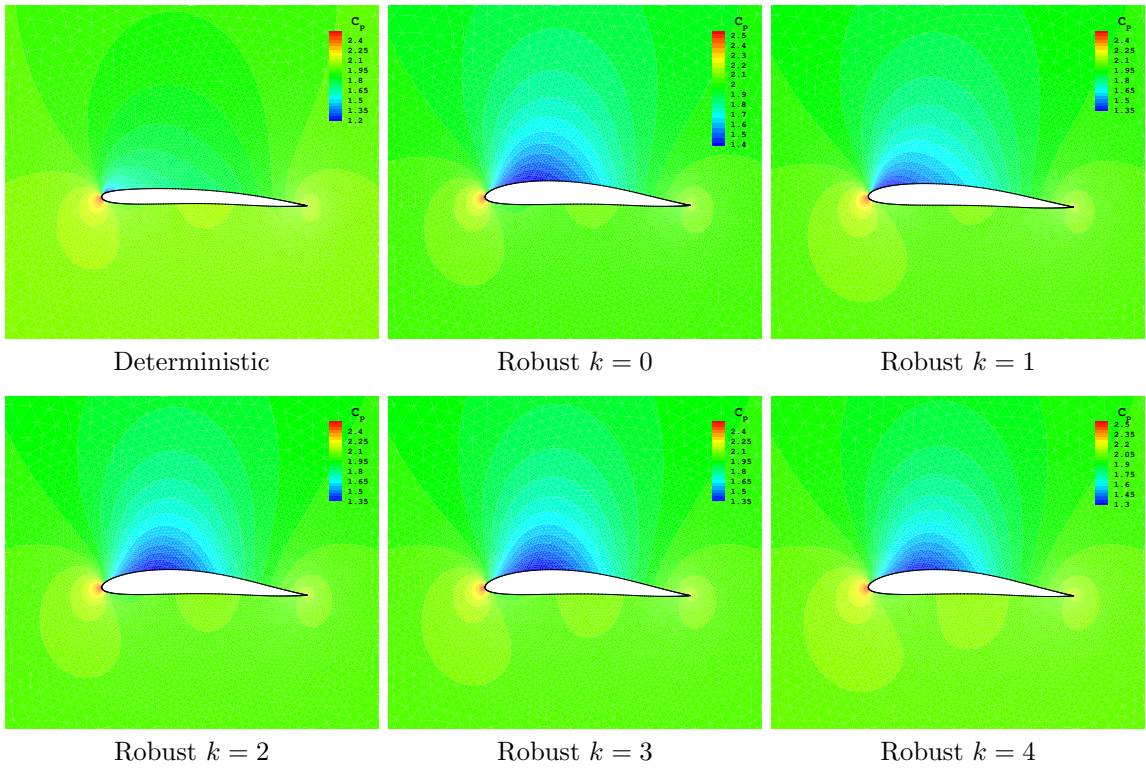


Figure 21: Contour plots of pressure coefficients C_p at different optimum designs using polynomial chaos.

samples are used for this test. For each Monte Carlo sample, a BCO problem is solved and statistics obtained are presented in Tables 9 and 10.

Table 9: Validations for $k = 1$ robust case at the initial design (NACA 0012, $\alpha = 2^\circ$ and $M_\infty = 0.65$) for kriging and polynomial chaos.

Type	$\mu_{c_{d_{max}}}$	$\sigma_{c_{d_{max}}}^2$	$\mu_{c_{l_{min}}}$	$\sigma_{c_{l_{min}}}$	No. of Function/ Gradient Evals.
IMCS-BCO (Kriging)	$8.85 \cdot 10^{-4}$	$4.32 \cdot 10^{-9}$	0.186	$1.72 \cdot 10^{-2}$	38/38
IMCS-BCO (PC)	$9.27 \cdot 10^{-4}$	$3.97 \cdot 10^{-8}$	0.186	$1.72 \cdot 10^{-2}$	36/36
MCS-BCO	$8.98 \cdot 10^{-4}$	$2.98 \cdot 10^{-8}$	0.186	$1.72 \cdot 10^{-2}$	6153/6153

Table 10: Validations for $k = 1$ robust case at the final design for kriging (robust shape, $\alpha = 2.065^\circ$ and $M_\infty = 0.6$) and polynomial chaos (robust shape, $\alpha = 3.058^\circ$ and $M_\infty = 0.6$).

Type	$\mu_{c_{d_{max}}}$	$\sigma_{c_{d_{max}}}^2$	$\mu_{c_{l_{min}}}$	$\sigma_{c_{l_{min}}}$	No. of Function / Gradient Evals.
IMCS-BCO (Kriging)	$2.93 \cdot 10^{-3}$	$3.07 \cdot 10^{-7}$	0.619	$1.86 \cdot 10^{-2}$	23/23
MCS-BCO	$2.73 \cdot 10^{-3}$	$2.50 \cdot 10^{-7}$	0.618	$1.84 \cdot 10^{-2}$	6153/6153
IMCS-BCO (PC)	$2.96 \cdot 10^{-3}$	$2.86 \cdot 10^{-7}$	0.619	$1.86 \cdot 10^{-2}$	23/23
MCS-BCO	$2.65 \cdot 10^{-3}$	$2.90 \cdot 10^{-8}$	0.620	$1.81 \cdot 10^{-2}$	6152/6152

Overall, it can be noticed that the surrogate models produce reasonably accurate statistics for a fraction of the computational cost compared to MCS. Also, kriging is more accurate than polynomial chaos in predicting the statistics.

VI. Conclusion

In this paper, surrogate models (kriging and polynomial chaos) enhanced with a dynamic training point selection methodology have been employed for robust optimizations under mixed epistemic and aleatory uncertainties. As a result of robust optimization, designs that are less sensitive to the randomness in design variables and input parameters are produced. The distribution of the possible outcomes are shown in the form of probability density functions and the probability of getting a specified outcome or less are shown in the form of cumulative distribution functions, whose availability helps the designer to make informed decisions and tune the performance accordingly. It is observed that robustness comes at the expense of an increased cost function and that designs employing a factor of safety featured higher cost penalties compared to robust designs. The performances of both surrogate models are also assessed when applied to uncertainty quantification and robust optimization under uncertainty (mixed epistemic/aleatory) on structural and aerodynamic test problems. It is observed that the kriging is better than polynomial chaos by virtue of its superior capability to approximate non-smooth functions. The accuracy of the surrogate models is reflected in the total number of exact function evaluations needed by the surrogate models during the robust optimizations as well as in the quality of output statistics.

A. Finite Element Procedure for Three-bar Truss Analysis

The finite element procedure adopted for obtaining the nodal displacements (u_x and u_y) as well as the elemental stresses (σ_1 , σ_2 , and σ_3) for the three-bar truss problem from Section III is discussed here. The structure is assumed to have three elements with two degrees of freedom at each node (*i.e.*, a two-dimensional truss analysis). The element connectivity information is given in Table 11.

Table 11: Connectivity of elements.

Element	Node		Element length l_e	$\lambda_e = \cos(\phi_e)$	$\mu_e = \sin(\phi_e)$
	i	j			
1	1	4	$L_1 = H/\sin(\phi_1)$	$\cos(\phi_1)$	$\sin(\phi_1)$
2	2	4	$L_2 = H/\sin(\phi_2)$	$\cos(\phi_2)$	$\sin(\phi_2)$
3	3	4	$L_3 = H/\sin(\phi_3)$	$\cos(\phi_3)$	$\sin(\phi_3)$

The stiffness matrix of the e -th element in global coordinate system is given by,

$$[\mathbf{k}]_e = \frac{E_e A_e}{L_e} \begin{bmatrix} \lambda_e^2 & \lambda_e \mu_e & -\lambda_e^2 & -\lambda_e \mu_e \\ \lambda_e \mu_e & \mu_e^2 & -\lambda_e \mu_e & -\mu_e^2 \\ -\lambda_e^2 & -\lambda_e \mu_e & \lambda_e^2 & \lambda_e \mu_e \\ -\lambda_e \mu_e & -\mu_e^2 & \lambda_e \mu_e & \mu_e^2 \end{bmatrix}. \quad (21)$$

The individual stiffness matrices of each element: $[\mathbf{k}]_1$, $[\mathbf{k}]_2$, and $[\mathbf{k}]_3$ are obtained using Eq. (21) and Table 11. The elemental stiffness matrices are assembled to form the global stiffness matrix \mathbf{K} of size 8×8 (not shown here). After applying the boundary conditions (i.e. x - and y -displacements at nodes 1, 2 and 3 are

zero) and performing elimination, a simplified linear system is obtained: $\mathbf{K}\mathbf{Q} = \mathbf{F}$, where $\mathbf{Q} = \begin{Bmatrix} Q_{4x} \\ Q_{4y} \end{Bmatrix}$, $\mathbf{F} = \begin{Bmatrix} F_{4x} \\ F_{4y} \end{Bmatrix}$ and $\mathbf{K} = \begin{bmatrix} \frac{E_1 A_1}{L_1} \lambda_1^2 + \frac{E_2 A_2}{L_2} \lambda_2^2 + \frac{E_3 A_3}{L_3} \lambda_3^2 & \frac{E_1 A_1}{L_1} \lambda_1 \mu_1 + \frac{E_2 A_2}{L_2} \lambda_2 \mu_2 + \frac{E_3 A_3}{L_3} \lambda_3 \mu_3 \\ \frac{E_1 A_1}{L_1} \lambda_1 \mu_1 + \frac{E_2 A_2}{L_2} \lambda_2 \mu_2 + \frac{E_3 A_3}{L_3} \lambda_3 \mu_3 & \frac{E_1 A_1}{L_1} \mu_1^2 + \frac{E_2 A_2}{L_2} \mu_2^2 + \frac{E_3 A_3}{L_3} \mu_3^2 \end{bmatrix}$.

The x - and y -displacements at node 4 are given by, $\mathbf{Q} = \mathbf{K}^{-1}\mathbf{F}$. Once the nodal displacements are found, the stresses in element $i - j$ can be calculated using:

$$\sigma_e = \frac{E_e}{L_e} \begin{Bmatrix} -\lambda_e & -\mu_e & \lambda_e & \mu_e \end{Bmatrix} \begin{Bmatrix} Q_{ix} \\ Q_{iy} \\ Q_{jx} \\ Q_{jy} \end{Bmatrix}. \quad (22)$$

Using Eq. (22) and Table 11, expressions for the stresses acting on each element can be obtained:

$$\begin{aligned} \sigma_1 &= \frac{E_1}{L_1} \begin{Bmatrix} -\lambda_1 & -\mu_1 & \lambda_1 & \mu_1 \end{Bmatrix} \begin{Bmatrix} 0 \\ 0 \\ Q_{4x} \\ Q_{4y} \end{Bmatrix} = \frac{E_1}{L_1} (Q_{4x} \lambda_1 + Q_{4y} \mu_1), \\ \sigma_2 &= \frac{E_2}{L_2} \begin{Bmatrix} -\lambda_2 & -\mu_2 & \lambda_2 & \mu_2 \end{Bmatrix} \begin{Bmatrix} 0 \\ 0 \\ Q_{4x} \\ Q_{4y} \end{Bmatrix} = \frac{E_2}{L_2} (Q_{4x} \lambda_2 + Q_{4y} \mu_2), \\ \sigma_3 &= \frac{E_3}{L_3} \begin{Bmatrix} -\lambda_3 & -\mu_3 & \lambda_3 & \mu_3 \end{Bmatrix} \begin{Bmatrix} 0 \\ 0 \\ Q_{4x} \\ Q_{4y} \end{Bmatrix} = \frac{E_3}{L_3} (Q_{4x} \lambda_3 + Q_{4y} \mu_3). \end{aligned} \quad (23)$$

The constraints (for the optimization problem) can be evaluated by substituting Eq (23) into Eq. (16). The gradients of the constraints and objective function with respect to the design variables are obtained via differentiation with Maple.

Acknowledgments

The authors thank Wataru Yamazaki for his kriging model and acknowledge the financial support from the University of Dayton Office for Graduate Academic Affairs through the Graduate Student Summer Fellowship Program.

References

- ¹Keane, A. and Nair, P., *Computational Approaches for Aerospace Design*, John Wiley & Sons, 2005.
- ²Arora, J. S., *Optimization of Structural and Mechanical Systems*, World Scientific Publishing Co. Pte. Ltd., 2007.
- ³Agarwal, H., *Reliability Based Design Optimization: Formulations and Methodologies*, Ph.D. thesis, University of Notre Dame, 2004.
- ⁴Padmanaban, D., *Reliability-Based Optimization for Multidisciplinary System Design*, Ph.D. thesis, University of Notre Dame, 2003.
- ⁵Alvin, K. F., Oberkampf, W. L., Rutherford, B. M., and Diegert, K. V., "Methodology for Characterizing Modeling and Discretization Uncertainties in Computational Simulations," Tech. Rep. SAND2000-5015, Sandia National Laboratories, 2000.
- ⁶Pilch, M., Trucano, T. G., and Helton, J. C., "Ideas Underlying Quantification of Margins and Uncertainties (QMU): A white paper," Tech. Rep. SAND2006-5001, Sandia National Laboratories, 2006.
- ⁷Helton, J. C., Johnson, J. D., Oberkampf, W. L., and Storlie, C. B., "A sampling-based computational strategy for the representation of epistemic uncertainty in model predictions with evidence theory," Tech. Rep. SAND2006-5557, Sandia National Laboratories, 2006.
- ⁸Diegert, K., Klenke, S., Novotny, G., Paulsen, R., Pilch, M., and Trucano, T., "Toward a More Rigorous Application of Margins and Uncertainties within the Nuclear Weapons Life Cycle - A Sandia Perspective," Tech. Rep. SAND2007-6219, Sandia National Laboratories, 2007.
- ⁹Helton, J. C., Oberkampf, J. D. J. W. L., and Sallaberry, C. J., "Representation of Analysis Results Involving Aleatory and Epistemic Uncertainty," Tech. Rep. SAND2008-4379, Sandia National Laboratories, 2008.
- ¹⁰Ghate, D. and Giles, M. B., "Inexpensive Monte Carlo uncertainty analysis," *Recent Trends in Aerospace Design and Optimization*, Tata McGraw-Hill, New Delhi, 2006, pp. 203–210.
- ¹¹Fang, K. T., Lin, D., and Winker, P., "Uniform design: Theory and application," *Technometrics*, Vol. 42, No. 3, 2000, pp. 237–248.
- ¹²Metropolis, N. and Ulam, S., "The Monte Carlo method," *Journal of the American Statistical Association*, Vol. 44, 1949, pp. 335–341.
- ¹³McKay, M. D., Conover, W. J., and Beckman, R. J., "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, Vol. 21, No. 2, 1979, pp. 239–245.
- ¹⁴Xiu, D., *Numerical Methods for Stochastic Computations: A Spectral Method Approach*, Princeton University Press.
- ¹⁵Maitre, O. P. L. and Knio, O. M., *Spectral Methods for Uncertainty Quantification*, Scientific Computation, Springer.
- ¹⁶Wong, T. T., Luk, W. S., and Heng, P. A., "Sampling with Hammersley and Halton points," *J. Graph. Tools*, Vol. 2, No. 2, Nov. 1997, pp. 9–24.
- ¹⁷Boopathy, K. and Rumpfkeil, M. P., "A Unified Framework for Training Point Selection and Error Estimation for Surrogate Models," *AIAA Journal*, Accepted.
- ¹⁸Boopathy, K. and Rumpfkeil, M. P., "A Multivariate Interpolation and Regression Enhanced Kriging Surrogate Model," AIAA Paper, 2013-2964.
- ¹⁹Boopathy, K. and Rumpfkeil, M. P., "Building Aerodynamic Databases Using Enhanced Kriging Surrogate Models," AIAA Region III Student Conference, Illinois Institute of Technology, Chicago, 2013.
- ²⁰Lockwood, B. A. and Anitescu, M., "Gradient-Enhanced Universal Kriging for Uncertainty Propagation in Nuclear Engineering," Preprint ANL/MCS-P1833-0111, 2011.
- ²¹Lockwood, B., Anitescu, M., and Mavriplis, D. J., "Mixed Aleatory/Epistemic Uncertainty Quantification for Hypersonic Flows via Gradient-Based Optimization and Surrogate Models," AIAA Paper, 2012-1254, 2012.
- ²²Lockwood, B., Rumpfkeil, M. P., Yamazaki, W., and Mavriplis, D. J., "Uncertainty Quantification in Viscous Hypersonic Flows using Gradient Information," AIAA Paper, 2011-885, 2011.
- ²³Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C., "A Limited Memory Algorithm for Bound Constrained Optimization," *SIAM Journal on Scientific Computing*, Vol. 16(5), 1995, pp. 1190–1208.
- ²⁴Zhu, C., Byrd, R. H., Lu, P., and Nocedal, J., "L-BFGS-B: A Limited Memory FORTRAN Code for Solving Bound Constrained Optimization Problems," Tech. Rep. NAM-11, Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, Illinois, USA, 1994.
- ²⁵Rumpfkeil, M. P., "Optimizations Under Uncertainty Using Gradients, Hessians, and Surrogate Models," *AIAA Journal*, Vol. 51, No. 2, 2013, pp. 444–451.
- ²⁶Putko, M. M., Newmann, P. A., Taylor III, A. C., and Green, L. L., "Approach for uncertainty propagation and robust design in CFD using sensitivity derivatives," AIAA Paper, 2001-2528, June, 2001.
- ²⁷Du, X. and Chen, W., "Methodology for Managing the Effect of Uncertainty in Simulation-Based Design," *AIAA Journal*, Vol. 38(8), 2000, pp. 1471–1478.
- ²⁸Parkinson, A., Sorensen, C., and Pourhassan, N., "A general approach for robust optimal design," *Trans. ASME*, Vol. 115, 1993, pp. 74–80.
- ²⁹Putko, M. M., Taylor III, A. C., Newmann, P. A., and Green, L. L., "Approach for Input Uncertainty Propagation and Robust Design in CFD Using Sensitivity Derivatives," *Journal of Fluids Engineering*, Vol. 124(1), 2002, pp. 60–69.
- ³⁰Waechter, A. and Biegler, L. T., "On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming," *Mathematical Programming*, Vol. 106(1), 2006, pp. 25–57.
- ³¹Mani, K. and Mavriplis, D. J., "An Unsteady Discrete Adjoint Formulation for Two-Dimensional Flow Problems with Deforming Meshes," AIAA Paper, 2007-60, 2007.
- ³²Mani, K. and Mavriplis, D. J., "Discrete Adjoint Based Time-Step Adaptation and Error Reduction in Unsteady Flow Problems," AIAA Paper, 2007-3944, 2007.

³³Hicks, R. and Henne, P., “Wing Design by Numerical Optimization,” *Journal of Aircraft*, Vol. 15 No. 7, 1978, pp. 407 – 412.

³⁴Batina, J. T., “Unsteady Euler Airfoil Solutions Using Unstructured Dynamic Meshes,” *AIAA Journal*, Vol. 28, No. 8, 1990, pp. 1381 – 1388.

³⁵Mani, K. and Mavriplis, D. J., “Unsteady Discrete Adjoint Formulation for Two-Dimensional Flow Problems with Deforming Meshes,” *AIAA Journal*, Vol. 46 No. 6, 2008, pp. 1351–1364.