2015

NASA Uncertainty Quantification Challenge: An Optimization-Based Methodology and Validation

Anirban Chaudhuri
University of Florida

Garrett Waycaster
University of Florida

Nathaniel Price
University of Florida

Taiki Matsumura
University of Florida

Raphael Haftka
University of Florida

NASA Uncertainty Quantification Challenge: An Optimization-Based Methodology and Validation

Article *in* Journal of Aerospace Information Systems · January 2015

DOI: 10.2514/1.I010269

CITATIONS
8

READS
192

5 authors, including:

Anirban Chaudhuri
Massachusetts Institute of Technology
30 PUBLICATIONS **215** CITATIONS
SEE PROFILE

Garrett Waycaster
Siemens
20 PUBLICATIONS **69** CITATIONS
SEE PROFILE

Nathaniel B Price
University of Nebraska at Lincoln
13 PUBLICATIONS **27** CITATIONS
SEE PROFILE

Taiki Matsumura
Siemens
13 PUBLICATIONS **54** CITATIONS
SEE PROFILE

JOURNAL OF AEROSPACE INFORMATION SYSTEMS
Vol. 12, No. 1, January 2015

NASA Uncertainty Quantification Challenge: An Optimization-Based Methodology and Validation

Anirban Chaudhuri,* Garrett Waycaster,† Nathaniel Price,‡
Taiki Matsumura,§ and Raphael T. Haftka¶
University of Florida, Gainesville, Florida 32601

DOI: 10.2514/1.I010269

The focus of this work is on uncertainty characterization, sensitivity analysis, uncertainty propagation, and extreme-case analysis. To deal with the computationally expensive and complex NASA problem, a simpler toy problem is devised to mimic the NASA problem for which the true results were known. The toy problem helped in thoroughly testing the current methods and their repeatability. For uncertainty characterization, a novel cumulative density function matching method is proposed, which gave similar results as a standard Markov chain–Monte Carlo-based Bayesian approach. An efficient reliability reanalysis-based probability-box sensitivity analysis method is employed to identify the most sensitive parameters to the risk analysis metrics. Uncertainty propagation to find extreme values for the risk analysis metrics is done using a single-loop efficient reliability reanalysis-based method. A modified version of the efficient reliability reanalysis is proposed that uses self-normalizing weights and caps on the weights; this is referred to as a capped self-normalizing efficient reliability reanalysis. This method showed considerably better performance at estimating risk analysis metrics for this application as compared to the generic efficient reliability reanalysis. The use of efficient reliability reanalysis was dictated by the cost of the black-box functions provided by NASA.

Nomenclature

D	=	modified Kolmogorov–Smirnov statistic
F	=	cumulative density function
f	=	probability density function
$\boldsymbol{g}$	=	requirement metrics
I	=	indicator function
J_1	=	expected value of worst-case requirement metric
J_2	=	probability of failure
n	=	given number of observations
n'	=	number of aleatory samples for generating an empirical cumulative density function
p	=	uncertain random variables
q	=	sampling probability density function for efficient reliability reanalysis
w	=	worst-case requirement metric
x	=	intermediate variables
θ	=	subparameters (epistemic uncertainty)
Θ	=	vector of subparameters

I. Introduction

THIS work aims to address the challenge problem posed by NASA [1] that comprises the entire uncertainty quantification process, including the characterization, propagation, and robust design. The different subproblems in the challenge tackled in this work are as follows: Subproblem A is the uncertainty characterization or improvement in the initial uncertainty model based on experimental data, subproblem B is the sensitivity analysis of the parameters on the system performance to decide which uncertainty models should be improved, subproblem C is the uncertainty propagation to determine the range of system performance metrics, and subproblem D is the identification of realizations that lead to extreme cases of system performance.

Uncertainty quantification is becoming increasingly important in an effort to make design optimization more robust and prepare to face the worst. It is helpful in trying to reduce surprises in designs. There is epistemic uncertainty due to lack of knowledge or imprecise data that can be reduced or possibly eliminated with more information. The aleatory uncertainty, on the other hand, is due to the inherent variability of the physical system and is irreducible. Conducting more experiments or gathering more information cannot reduce this uncertainty. Oberkampf et al. [2,3] explained these two types of uncertainties in detail. Although there is a consensus that aleatory uncertainty is modeled by probability distributions, various modeling techniques for epistemic uncertainty have been studied, such as interval theory, Dempster–Shafer evidence theory [4,5], possibility theory [6,7], and probability theory [8,9].

Presented as Paper 2014-1498 at the 16th AIAA Non-Deterministic Approaches Conference, National Harbor, MD, 13–17 January 2014; received 7 April 2014; revision received 24 October 2014; accepted for publication 4 December 2014; published online 30 January 2015.

*Research Assistant, Mechanical and Aerospace Engineering; anirban.chaudhuri01@gmail.com. Student Member AIAA (Corresponding Author).
†Graduate Research Assistant, Mechanical and Aerospace Engineering; gcwaycaster@ufl.edu. Student Member AIAA.
‡Graduate Research Assistant, Mechanical and Aerospace Engineering; natprice@ufl.edu. Student Member AIAA.
§Mechanical and Aerospace Engineering; taikimatsumura@gmail.com. Member AIAA.
¶Distinguished Professor, Mechanical and Aerospace Engineering; haftka@ufl.edu. Fellow AIAA.

In this work, we will consider three different categories of uncertain random variables [1], including layered uncertainties or mixed aleatory and epistemic uncertainties:

1) Only aleatory (irreducible) uncertainty is present. It is modeled as a given distribution with known parameters. There is no epistemic space associated with this category.

2) Only epistemic (reducible) uncertainty is present. It is modeled as having fixed but unknown constants that lie within given intervals.

3) Mixed aleatory and epistemic uncertainty is present. This is modeled as a random variable with a fixed functional form but unknown parameters. The unknown parameters have epistemic uncertainty prescribed in a given interval that can be reduced.

A cumulative distribution function (CDF) matching method for uncertainty quantification using optimization to identify reduced epistemic uncertainty based on some additional information is proposed in this work. In this method, we compare different confidence bands (or envelopes) for the empirical CDF of the given samples using a modified Kolmogorov–Smirnov statistic to get a refined range for the epistemic variables that well describe the 95% confidence bands associated with the empirical CDF of the given observations. We also compare this to a more brute force method using Bayesian approach.

A sensitivity analysis of the uncertain (epistemic) subparameters defining the random parameters is conducted to determine if we can fix some parameters without incurring error, and for which parameters more information would be most helpful. The sensitivity of the parameters is identified by the amount of change brought about in the p-boxes of each intermediate variable or ranges of values of interest due to a reduction in uncertain subparameter bounds. We use the efficient reliability reanalysis (ERR) [10] method, which is an importance sampling-based approach, as an approximation for the expensive NASA black-box functions. We propose some modifications to the generic ERR method to handle constant parameters with epistemic uncertainty and a version of self-normalized weights for ERR with a cap on the weights that is proportional to the sample size. We dubbed this modified version as a capped self-normalizing ERR (CSN-ERR). The cap on the weights and self-normalizing weights deters a single sample from essentially dominating all the other samples, which we found to be an obvious issue with the calculation of probability of failures. The uncertainty propagation process is done using a single-loop CSN-ERR-based approach to find the ranges of system performance metrics identifying the epistemic realization that prescribe extreme values of system performance. These results were compared to a brute force double-loop Monte Carlo simulation.

The NASA challenge is a complex problem having unknown structure and solution. This makes it difficult to judge the performance of the methods devised in this work. To validate and thoroughly test our methods, we devised a simpler toy problem to mimic the NASA problem. This helped us in identifying the level of confidence for each method accompanied by rigorous debugging of our algorithms.

II. Problem Formulation

A. NASA Challenge Problem

The multidisciplinary system under investigation is defined by a mathematical model S. The system performance is evaluated based on a set of requirement metrics $\boldsymbol{g}$ (here, eight) that define the safe domain when $\boldsymbol{g} < 0$. The requirement metrics depend on a vector of uncertain random variables (here, we call it a parameter) $\boldsymbol{p}$ (here, 21) and a vector of design variables $\boldsymbol{d}$ (here, 14) that can be set by the designer. The relationship between the outputs $\boldsymbol{g}$ and inputs, $\boldsymbol{p}$ and $\boldsymbol{d}$, are given by some black-box functions:

$$\boldsymbol{g} = f(\boldsymbol{x}, \boldsymbol{d}) \tag{1}$$

where $\boldsymbol{x}$ is a vector of intermediate variables (here, five) based on $\boldsymbol{p}$ as given by

$$x_1 = h_1(p_1, p_2, p_3, p_4, p_5) \tag{2}$$

$$x_2 = h_2(p_6, p_7, p_8, p_9, p_{10}) \tag{3}$$

$$x_3 = h_3(p_{11}, p_{12}, p_{13}, p_{14}, p_{15}) \tag{4}$$

$$x_4 = h_4(p_{16}, p_{17}, p_{18}, p_{19}, p_{20}) \tag{5}$$

$$x_5 = p_{21} \tag{6}$$

The components of $\boldsymbol{x}$, which can be interpreted as outputs of single discipline analyses, are the inputs to the cross-discipline analyses in Eq. (1). The components of $\boldsymbol{g}$ and $\boldsymbol{x}$ are continuous functions of the inputs that prescribe them. The software to evaluate these black-box functions is provided by NASA [1]. The system performance is defined by the expected value of a worst-case requirement metric J_1 or a probability of failure J_2. These two metrics are defined in terms of the largest violation:

$$w(\boldsymbol{p}, \boldsymbol{d}) = \max_{1 \le i \le 8} g_i = \max_{1 \le i \le 8} f_i(\boldsymbol{x}, \boldsymbol{d}) = \max_{1 \le i \le 8} f_i(h(\boldsymbol{p}), \boldsymbol{d}) \tag{7}$$

J_1 and J_2 are given by Eqs. (8) and (9), respectively:

$$J_1 = E[w(\boldsymbol{p}, \boldsymbol{d})] \tag{8}$$

$$J_2 = 1 - P[w(\boldsymbol{p}, \boldsymbol{d}) < 0] \tag{9}$$

The design variables are fixed by the analyst as $\boldsymbol{d}_{\text{baseline}}$ (specified here by NASA). A flowchart showing the progression of the problem is shown in Fig. 1.

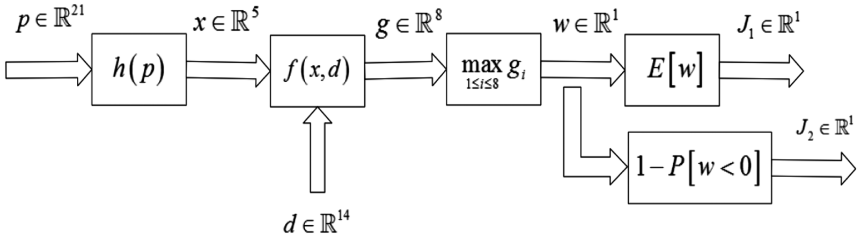


Fig. 1 Flowchart for the problem formulation.

B. Toy Problem

To debug our methods for the NASA problem, we first test the approach with a toy problem. The toy problem, for which we know the answer (or can easily calculate), is intended not only to hold basic procedures of the original NASA problem but also to be simple enough to analyze the results easily. The toy problem has two requirement metrics: $\boldsymbol{g} = [g_1, g_2]^T$, where g_1 and g_2 are the functions of three uncertain variables, $\boldsymbol{p} = [p_1, p_2, p_3]^T$, and two design variables, $\boldsymbol{d} = [d_1, d_2]^T$. The functions of g_1 and g_2 are given by the following equations:

$$g_1 = -2p_1 + 10p_2 - 5(p_3 - d_1) \tag{10}$$

$$g_2 = d_2 - p_3 \tag{11}$$

Note that g_1 and g_2 are treated as black-box functions when we solve the problem. If $\boldsymbol{g} < 0$, the system requirements are satisfied. Otherwise, the system is considered to be in a failure state. This is a trivial problem, and it is posed in this manner to help us quickly debug our methods and rigorously test diverse theories. System performance is measured by two metrics: J_1 and J_2. J_1 is the worst-case requirement metric defined
by

$$J_1 = E[w(\boldsymbol{p}, \boldsymbol{d})] \tag{12}$$

where $w(\boldsymbol{p}, \boldsymbol{d}) = \max(g_1, g_2)$. J_2 is the failure probability defined by

$$J_2 = 1 - P[w(\boldsymbol{p}, \boldsymbol{d}) < 0] \tag{13}$$

Table 1 Uncertainty models and true values of the parameters for toy problem

Symbol	Category	Uncertainty model	True value
p_1	2	$\Delta = [0, 1]$	$p_1 = 0.5$
p_2	3	Normal, $-2 \le E[p_2] \le 1$, $0.5 \le V[p_2] \le 1.1$	$E[p_2] = 0$, $V[p_2] = 1$
p_3	3	Unimodal Beta, $0.6 \le E[p_3] \le 0.8$, $0.02 \le V[p_3] \le 0.04$	$E[p_3] = 0.7$, $V[p_3] = 0.03$

Table 2 Uncertain parameters for random variables $\boldsymbol{p}$[a]

Symbol	Category	Uncertainty model
p_1	3	Unimodal Beta, $3/5 \le E[p_1] \le 4/5$, $1/50 \le \mathrm{Var}[p_1] \le 1/25$
p_2	2	Constant, $\Delta = [0, 1]$
p_3	1	Uniform, $\Delta = [0, 1]$
p_4, p_5	3	Normal, $-5 \le E[p_i] \le 5$, $1/400 \le \mathrm{Var}[p_i] \le 4$, $\lvert\rho\rvert \le 1$ for $i = 4, 5$
p_6	2	Constant, $\Delta = [0, 1]$
p_7	3	Beta, $0.982 \le a \le 3.537$, $0.619 \le b \le 1.080$
p_8	3	Beta, $7.450 \le a \le 14.093$, $4.285 \le b \le 7.864$
p_9	1	Uniform, $\Delta = [0, 1]$
p_{10}	3	Beta, $1.520 \le a \le 4.513$, $1.536 \le b \le 4.750$
p_{11}	1	Uniform, $\Delta = [0, 1]$
p_{12}	2	Constant, $\Delta = [0, 1]$
p_{13}	3	Beta, $0.412 \le a \le 0.737$, $1.000 \le b \le 2.068$
p_{14}	3	Beta, $0.931 \le a \le 2.169$, $1.000 \le b \le 2.407$
p_{15}	3	Beta, $5.435 \le a \le 7.095$, $5.287 \le b \le 6.945$
p_{16}	2	Constant, $\Delta = [0, 1]$
p_{17}	3	Beta, $1.060 \le a \le 1.662$, $1.000 \le b \le 1.488$
p_{18}	3	Beta, $1.000 \le a \le 4.266$, $0.553 \le b \le 1.000$
p_{19}	1	Uniform, $\Delta = [0, 1]$
p_{20}	3	Beta, $7.530 \le a \le 13.492$, $4.711 \le b \le 8.148$
p_{21}	3	Beta, $0.421 \le a \le 1.000$, $7.772 \le b \le 29.621$

[a]Category 1 variables have only aleatory uncertainty, category 2 variables have an unknown single value (only epistemic uncertainty), and category 3 variables have aleatory uncertainty in terms of a known distribution type with uncertain parameters (epistemic uncertainty).

The uncertainty models for $\boldsymbol{p}$ and their true values are given in Table 1. The nominal design variables are set as $\boldsymbol{d} = [1.5, 0.5]^T$. All the results for the toy problem are provided in Appendix C.

III. Uncertainty Characterization (Task A)

In this section, we describe a few ways of practically refining the epistemic uncertainties present in the random variables using information in the form of observed values. We demonstrate the process using observed values of x_1 ($x_{1,\text{obs}}$) that have been provided by NASA in order to update the epistemic uncertainty of parameters of p_{1-5}, given in Table 2. Each epistemic parameter (here, we refer to it as a subparameter) for category 2 and category 3 random variables is denoted by θ. The category 1 random variables have only aleatory uncertainty. At first, 25 observations of x_1 are used to update the uncertainty (task A1) and an additional 25 observations are used to validate the updated model (task A2). Then, all 50 observations of x_1 are used to update the uncertainty model (task A3) and the effect of the number of observations on the fidelity of the resulting uncertainty models is checked (task A4). The refinement of epistemic parameters is done using a novel CDF matching method, which is described in the next section. A Bayesian approach [11–13] was also implemented to compare results to the CDF matching method, and the details are provided in Appendix A.

A. CDF Matching Uncertainty Quantification Approach

The CDF matching approach uses the concept of the two-sample Kolmogorov–Smirnov test to compare the empirical distribution function or empirical CDF (ECDF) for the $x_{1,\text{obs}}$, with the ECDF for the generated x_1 values using aleatory uncertainties for some realization of Θ. The ECDF F_n for n independent and identically distributed samples X_i is given by Eq. (14):

$$F_n(X) = \frac{1}{n}\sum_{i=1}^{n} I_{X_i \le x} \tag{14}$$

where $I_{Xi<x}$ is an indicator function that is equal to one when $X_i \le x$ and zero otherwise. A modified Kolmogorov–Smirnov (K-S) statistic, used here for comparing two ECDFs with n (given observations) and n' (randomly generated using aleatory uncertainty; 100 in this work) samples ($D_{n,n'}$), is given by Eq. (15). It is the sum of distances at the observed values between the two ECDFs. For each Θ realization, we sample 100 random $\boldsymbol{p}$ and generate an ECDF using that to compare to the ECDF from $x_{1,\text{obs}}$ and calculate $D_{n,n'}$. The random stream for generating 100 samples of $\boldsymbol{p}$ for a particular Θ realization is fixed to reduce the noise in objective function calculation. The optimization is repeated several times with different random streams to placate this assumption:

$$D_{n,n'} = \sum_{x_{1,\text{obs}}} |F_n(x) - F_{n'}(x)| \tag{15}$$

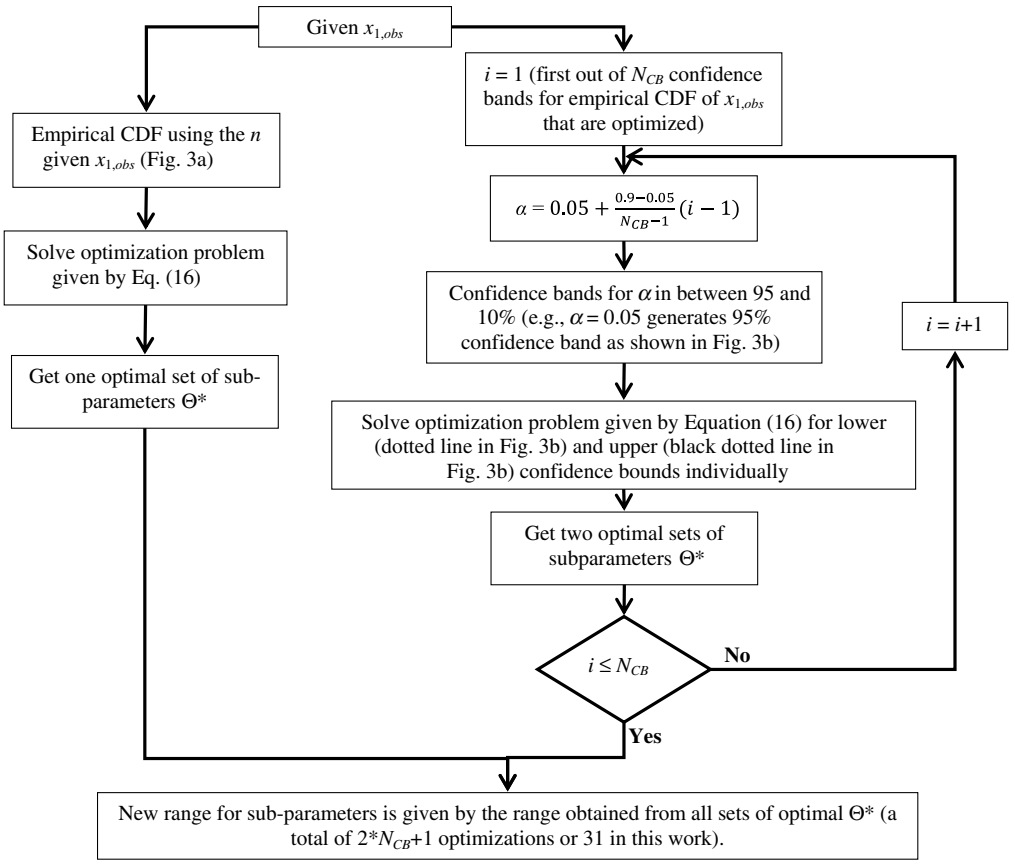
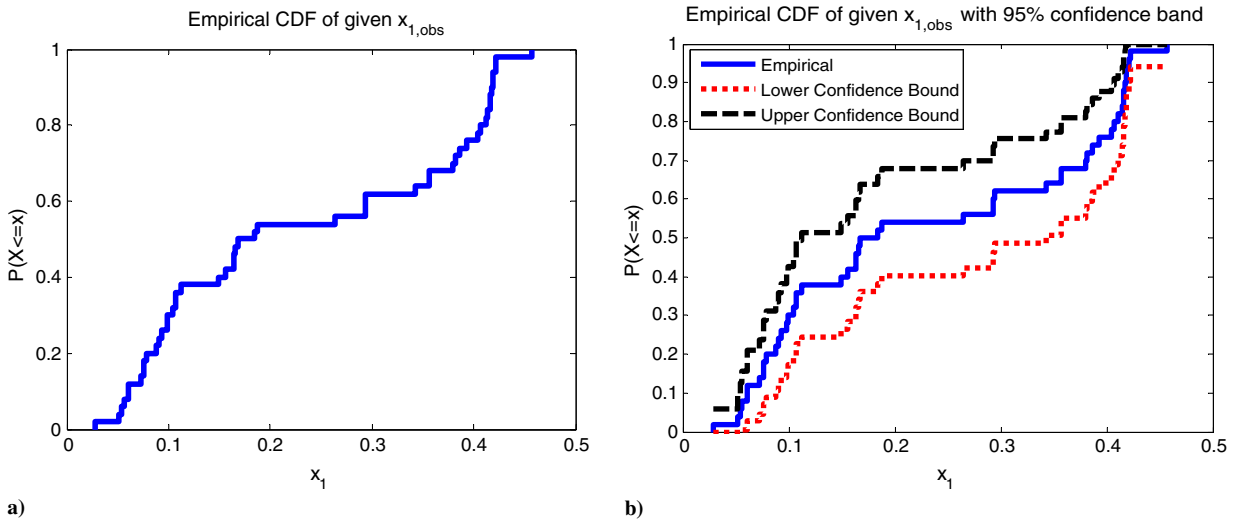


Fig. 2 Flowchart describing CDF matching method.



a) b)

Fig. 3 Illustration showing a) empirical CDF, and b) confidence bands for empirical CDF of $x_{1,obs}$.

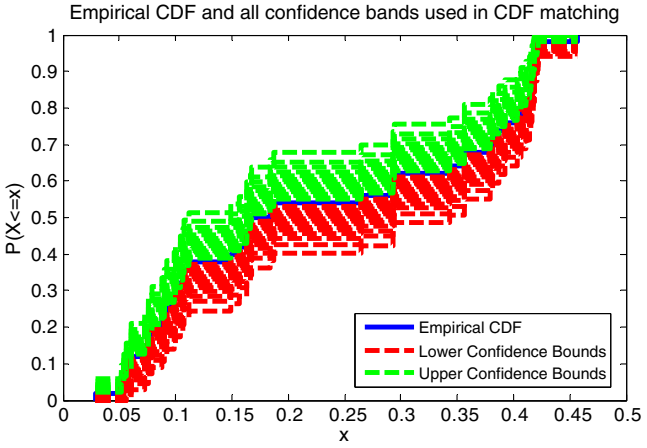


Fig. 4 Empirical CDF for $x_{1,obs}$ and all its confidence bands for which the optimization problem in Eq. (16) is solved during the CDF matching method.

$D_{n,n'}$ is minimized to obtain optimum Θ^*, as shown by the optimization problem in Eq. (16). A flowchart describing the CDF matching method is shown in Fig. 2 (see also Fig. 3). We find the optimal Θ^* for matching the given ECDF of $x_{1,obs}$ and for matching different lower and upper confidence bounds (10–95% confidence bands) of the given ECDF for $x_{1,obs}$, as shown in Fig. 4. This optimization is done for matching the given ECDF and repeated for matching 15 lower-bound and 15 upper-bound confidence bands (Fig. 4) of the given ECDF between 10 and 95%** confidence bands. This gives us 31 optimal values for each Θ (the optimization problem is solved 31 times, requiring 3.1e6 $p\_to\_x_1$ function evaluations). Then, the refined bounds of Θ are found by taking the range of those values. Note that we do not assume any monotonicity between epistemic parameters and the response quantity as we solve for a set of confidence bands and take the range from the final optimal sets of Θ^*:

$$\underset{\Theta}{\text{Minimize}}\ D_{n,n'} \quad \text{Such that } \theta_{\text{lower}_i} \le \theta_i \le \theta_{\text{upper}_i} \quad \text{for } i = 1, 2 \ldots 8 \tag{16}$$

In this case. there are eight θ for p_{1-5} with upper and lower bounds as provided in Table 2. The beta subparameters of a and b are found from the expected value and variance of the distribution, as shown in Appendix B. The unimodality condition for p_1 is enforced during the optimization by penalizing the objective function when $a \le 1$ or $b \le 1$. The confidence bands are evaluated using Greenwood's formula (*ecdf* function in MATLAB). The optimization problem is solved using the DIRECT optimizer developed by Finkel et al. [14]. This method actually gives us 31 separate sets of optimal Θ^* and can give us an idea of the correlation between the different θ. In this case, we only consider the refined range of θ to be used in the sensitivity analysis and uncertainty propagation, as it is simpler to implement and we choose not to trust the correlations from a limited dataset. This means that we continue to treat each a subparameter as an interval, as the information is originally posed in the challenge problem. Note that none of our methods preclude us from using correlation information in the methods for sensitivity analysis and extreme-case analysis, but we choose to ignore it to simplify the methods. An improvement on the current CDF matching method might be to consider optimizing for any ECDF that can be bounded by the p-box defined by the 95% confidence bands of the given ECDF. This would also help in choosing more ECDF combinations to optimize for and get a better understanding of the correlation between each θ at the expense of additional computational effort.

**This is a user-defined number related to the computational time available.

Table 3 Refined epistemic bounds using CDF matching UQ method for first set of 25 observations and full set of 50 observations of x_1 for NASA problem with five repetitions

		Using 25 observations		Using 50 observations	
Symbol	Given prior	Reduced bounds (median)	Reduction in range (% of prior range), %	Reduced bounds (median)	Reduction in range (% of prior range), %
$E[p_1]$	[0.6, 0.8]	[0.6037, 0.7444]	29.6	[0.6333, 0.7667]	33.3
$V[p_1]$	[0.02, 0.04]	[0.0210, 0.0349]	30.5	[0.0230, 0.0393]	18.5
p_2	[0,1]	[0.1173, 0.9938]	12.4	[0.0926, 0.9938]	9.9
$E[p_4]$	[−5, 5]	[−4.4444, 4.4444]	11.1	[−4.8148, 4.4444]	7.4
$V[p_4]$	[0.0025, 4]	[0.0765, 3.9260]	3.7	[0.2246, 3.8273]	9.9
$E[p_5]$	[−5, 5]	[−3.4568, 1.1111]	54.3	[−4.4444, 0]	55.6
$V[p_5]$	[0.0025, 4]	[0.5207, 3.7779]	18.5	[1.5077, 3.9260]	39.5
ρ	[−1, 1]	[−0.8148, 0.8230]	18.1	[−0.8889, 0.8916]	11

B. Results for Uncertainty Characterization of NASA Problem

The epistemic uncertainties are updated using a first set of 25 given observations and then a second set of 25 given observations of x_1 for the NASA problem using the CDF matching uncertainty quantification (UQ) approach. Some results for comparing the CDF matching method to the Bayesian approach are given in Appendix A. The CDF matching method is initially verified and debugged using the simple toy problem, and the results can be seen in Appendix C. It faired reasonably well for the toy problem, showing substantial uncertainty reduction while keeping the true values in the range. The repeatability of the CDF matching method was also found to be satisfactory for the toy problem.

1. *Tasks A1 and A3: Update Uncertainty Model Using Various Observation Sets*

The refined median bounds of Θ for five repetitions (with different random seeds) of the CDF matching UQ method using the first set of 25 and full set of 50 observations ($x_{1,\text{obs}}$) are given in Table 3. Using the first set of 25 observations, the CDF matching approach shows a trend of reducing the upper bound of $E[p_1]$. There were also significant changes observed in the upper bounds of $V[p_1]$ and $E[p_5]$ and the lower bound of $V[p_5]$. The maximum reduction in range is for $E[p_5]$.

Using the full set of 50 observations, the ranges of $E[p_1]$ and $E[p_5]$ are substantially reduced. But, these reductions in ranges were also identified while using the first set of 25 observations. The most significant difference in the reduction of range while using all 50 observations as compared to the first 25 observations can be seen for $V[p_5]$. The other difference is that, while using the first 25 observations, there was almost a 30% decrease in the range for $V[p_1]$ but, after using all 50 observations, the decrease was only around 16%. This shows that using the smaller set of observations was more consequential and led to an excessive reduction of the range. This could be due to a biased set of small samples (as we see in the results for the toy problem with a small sample set, shown in Fig. C1 in Appendix C). The results from the Bayesian approach are mostly similar to the CDF matching method, as can be seen from Appendix A.

2. *Tasks A2 and A4: Effect of Number of Observations on Uncertainty Models*

For the validation of the updated uncertainty models, we use the Kolmogorov–Smirnov test. The K-S test measures the maximum distance between two ECDFs, called the K-S statistic, as an indicator of the agreement of the functions. We first evaluate the K-S statistics of ECDFs of x_1 based on random sets of Θ generated from the prior range (provided by NASA) to be compared with the ECDF of the given observations, $x_{1,\text{obs}}$. The K-S test is conducted by generating 7256 Θ realizations in the refined bounds (7000 using Latin hypercube sampling and 256 corner points in eight dimensions). For each Θ realization, 1000 x_1 samples are generated using the aleatory uncertainty and the ECDF from these generated samples is compared against the full set of 50 given observations of x_1. This process is repeated for all the Θ realizations to obtain the variation in the K-S statistic. This is compared to results from generating 7256 Θ realizations in the prior range (as provided by NASA) with the same random seed to make it a fair comparison.

For the K-S test, all the ECDFs based on prior and refined ranges were compared to the ECDF of all 50 observations. As shown in Fig. 5, the third quartile (the top of the box and the 75th percentile value) of the K-S statistic was substantially reduced as the number of observations used for the uncertainty quantification is increased for both the approaches. The median of the K-S statistic also decreased with the number of observations. This reflects the fact that the uncertainty quantification successfully discarded the combinations of subparameters that poorly approximated the ECDF of x_1 with 50 observations.

The results for rejection percentages of the null hypothesis of the K-S test with $\alpha = 5\%$ for different sample observation sizes are given in Table 4. It can be seen that the rejection percentage decreases after updating the epistemic uncertainties. More than a 50% decrease in the rejection

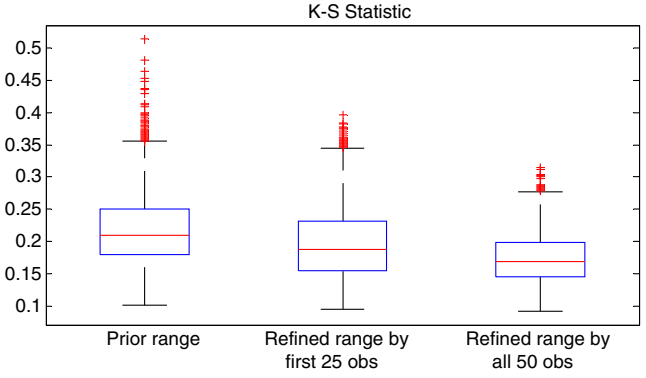


Fig. 5 Comparison of K-S statistic between prior range and refined range tested by all 50 observations (obs) for NASA problem using CDF matching UQ approach.

Table 4 K-S test rejection percentage for the prior range compared to refined epistemic uncertainty range from CDF matching method using first set of 25 observations and all 50 observations for NASA problem

Epistemic uncertainty range	Value, %
Prior range	62.4
Refined range by 25 observations	46.5
Refined range by 50 observations	28.3

percentage is observed when a CDF matching method is used with all 50 observations as compared to the prior range. The CDF matching method performs efficiently, using a total of 1.5e − 7 aleatory samples (five repetitions) and compares well to the Bayesian approach that uses 1e − 8 samples (Appendix A).

IV. Sensitivity of Subparameters to Intermediate Variables (Task B1)

The first step of our sensitivity analysis as specified by the NASA challenge problem is to determine the degree of refinement in the p-box of the intermediate variables x that could be obtained from a refined uncertainty model for our uncertain subparameters θ. One important consideration in performing the sensitivity analysis is that we are concerned with the sensitivity of the p-box of each intermediate variable x to the bounds of their corresponding subparameters and not to the value of the subparameter. This is because we are seeking to acquire refined bounds for some parameter models; and the fact that the value of an intermediate variable is highly sensitive to the value of a subparameter does not necessarily imply that we will benefit from reducing the bounds of that subparameter. Consider a simple case of a single normal variable with the mean and variance prescribed as intervals. Though the value of this variable is highly sensitive to the value of the variance, a change in the lower bound of this variance will have no effect on the p-box of the variable. The relationship between the value of a subparameter and the area of the p-box of an intermediate variable is in fact only a measure of the interaction effects of the subparameter in question and the remaining interval-valued subparameters, and it may therefore be misleading.

To determine the sensitivity of each subparameter on the uncertainty of the intermediate variable x, we use a probability-box (p-box) area-based sensitivity analysis using a double-loop Monte Carlo approach. Starting with the updated bounds obtained for each subparameter in Sec. III, we randomly generate realizations of each subparameter within these bounds. For each of these subparameter realizations, we then generate random samples of each parameter and calculate the corresponding values of each intermediate variable. This means that, for each subparameter realization, we can calculate an empirical CDF for the intermediate variable (Fig. 6b). By calculating the maximum and minimum values of each of these many empirical CDFs at each value x (Fig. 6c), we may obtain an estimate of the p-box of the intermediate variable corresponding to the given subparameter bounds. Then we calculate the p-box area (Fig. 6d) that is used as the metric in the sensitivity analysis. This technique is described graphically in Fig. 6.

We may then simulate a reduction in the bounds of each subparameter that we might expect from requesting a refined uncertainty model. These reduced bounds will affect our simulation of ECDFs of x as shown in Fig. 6b. Since every updated set of subparameter interval bounds is a subset of

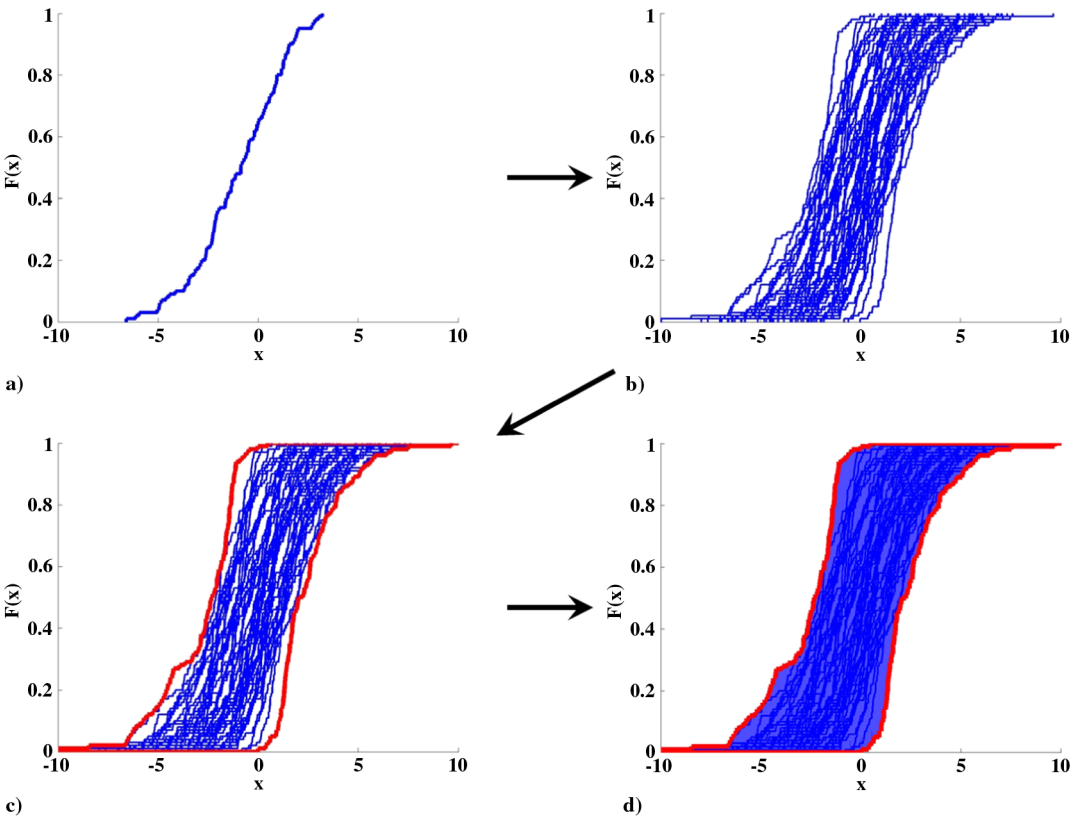


Fig. 6 Method of estimating the p-box of intermediate variables.

Table 5 Average percent change in the p-box of intermediate variables with reduced subparameter bounds for each parameter

x_1		x_2		x_3		x_4	
Parameter	Percent change, %	Parameter	Percent change, %	Parameter	Percent change, %	Parameter	Percent change, %
1	19.9	6	31.1	12	41.5	16	40.2
2	2.9	7	25.5	13	4.2	17	14.2
4	17.3	8	7.5	14	3.7	18	10.8
5	16.7	10	6.3	15	3.0	20	6.0

Table 6 Percentage change in p-box area of intermediate variables x due to fixing of parameters

x_1		x_2		x_3		x_4	
Parameter	Percent change, %	Parameter	Percent change, %	Parameter	Percent change, %	Parameter	Percent change, %
1	56	6	36	12	74	16	57
2	4	7	0	13	0	17	0
4	4	8	24	14	0	18	46
5	0	10	33	15	0	20	43

Table 7 Optimal values of fixed parameters corresponding to percent changes in Table 6

x_1		x_2		x_3		x_4	
Parameter	Fixed value	Parameter	Fixed value	Parameter	Fixed value	Parameter	Fixed value
1	0.647	6	1	12	0.352	16	1
2	0.6	7	0	13	0.068	17	0
4	−7.25	8	0	14	0.475	18	1
5	−2.876	10	1	15	0.43	20	0.001

the initial bounds, every p-box generated with the reduced bounds will be strictly contained within the initial p-box. This means that if the change in area is very small, the new p-box will be very similar to the initial p-box. If the change is significant, it is then possible that the effect on J_1 or J_2 may be different for the same change in area, depending on how the p-box has changed. At this first stage of sensitivity, we are only focused on the degree of refinement of the p-box, as requested in challenge problem subpart B1.

Although there are infinitely many possible reduced bounds that may be obtained from this refinement, we approximate the updated bounds using three simplified cases for a percent reduction in the range of possible subparameter values: a reduction entirely from the upper bound, entirely from the lower bound, or a centered reduction from both the upper and lower bounds. We elect to use a 25% reduction in the subparameter bounds to calculate the change in the p-box. We simulate each of these three cases for each of our 31 uncertain subparameters and again estimate the p-box of each intermediate variable. To compare the effect of these reduced bounds, we calculate the average change in the area of the p-box across these three cases for each subparameter, where a large change indicates a significant refinement in the p-box of the intermediate variable. Since a revised model of a given parameter will affect all related subparameters, we sum the expected change in the p-box of the intermediate variables for subparameters corresponding to the same parameter.

We use 5000 subparameter realizations with 250 parameter samples each to calculate the area of the p-box of each of the five intermediate variables. These subparameter realizations are generated using Latin hypercube sampling, with realizations not satisfying the unimodality condition for the parameter p_1 being randomly resampled. The average percent change in the p-box brought about by a change in subparameter bounds for each of our 16 parameters is shown in Table 5.

Our next task is to determine if it is possible to fix any parameters without incurring a significant error in the intermediate variables. It should be noted that a subparameter that is found to be insensitive or able to be fixed when considering that subparameter's effect on the p-box of the intermediate variable does not imply that the subparameter is not important for calculating our values of interest: J_1 and J_2. However, we will choose to make this assumption in order to reduce the dimensionality of the problem in later steps. In practice, this assumption could be based on information from experts in each subdiscipline; we use this assumption in order to satisfy our limited computational budget.

To calculate the optimal fixed parameters, we again use a double-loop Monte Carlo simulation to calculate the area of the p-box of each intermediate variable x; but in this case, we will define the value of one of the parameters $\boldsymbol{p}$ as a decision variable to be fixed across all samples. We use the DIRECT global optimization algorithm to find the optimal value of this fixed parameter that maximizes the area of the p-box of the intermediate variable. If the change in the area of the p-box is small, the representation of the intermediate variable remains relatively unchanged, even without the parameter in question varying freely. Again, we use 5000 subparameter realizations from Latin hypercube sampling and 250 parameter samples each. These 1.25×10^6 parameter realizations are fixed across DIRECT optimization cycles to avoid random noise in the optimization, meaning the only change will be in the value of the parameter that we are attempting to fix. Table 6 shows the optimized percent change in the intermediate variables for the case of attempting to fix each parameter, with the corresponding optimal parameter values in Table 7.

The results of this analysis reveal eight candidate parameters for which the change in the p-box of their corresponding intermediate variable is small (less than 10%): 2, 4, 5, 7, 13, 14, 15, and 17. Of these eight candidates, four parameters (4, 5, 7, and 17) were found to be sensitive to the area of the p-box of x (greater than 10%), as shown in Table 5. We choose not to fix these parameters, as their higher sensitivity means they may be more affected by the limited sample size available for our fixed-value optimization.

V. Testing Surrogates Models for Approximating J_1 and J_2

One common practice for dealing with computationally expensive problems such as this one is to use surrogates. We observe from the first round of results for the NASA challenge problem that many participants, including ourselves, elected to use surrogates to replace some of the

Table 8 Surrogate comparison case study

Number of training points	Surrogate type	Quantity estimated
1) 100	1) Second-order polynomial response surface	1) $g(x) = \mathbb{R}^8$
2) 500	2) Kriging	2) $g_{\max}(x) = W$
3) 1000		

provided challenge problem functions. However, the types of surrogates, number of training points, and the quantities used in the approximation varied greatly [15–21]. It is therefore of interest to better understand how well some common surrogates perform for this problem, particularly since the challenge problem functions have several difficult attributes for fitting surrogates, such as high nonlinearity and high dimensionality.

To perform this comparison, we perform a case study of multiple surrogate types. We consider three different characteristics: the number of training points, the type of surrogate, and the quantity being estimated. For each case, we fit 100 random initial designs of experiments (DOEs) in the intermediate variable space x in order to estimate the potential variation. Table 8 shows the characteristics used to develop each case. and all possible combinations are tried.

We therefore have a total of 12 cases: each with 100 surrogates based on our 100 random DOEs. We then generate a random sample of epistemic realizations θ and construct a sample of our random variables $\boldsymbol{p}$ that can be used to calculate the intermediate variables x using the challenge problem function. We finally evaluate each of our fitted surrogates at these new intermediate variable values and calculate J_1 and J_2. The distribution of J_1 and J_2 predicted by these surrogates is compared to the distribution calculated with double-loop sampling (DLS) [22] using the original NASA black-box functions with 1000 subparameter realizations and 1000 parameter realizations. These results are shown in Figs. 7–14, where the thick black line represents the value from DLS and each of the small lines represent one surrogate prediction. The combination of these surrogates can be represented as a p-box, given by the shaded region.

It can be seen that none of these cases considered offer any reliability in predicting the true values of J_1 and J_2. For J_1, each of the cases predicts a much wider range than the true values, which lie between zero and one for these test points. This overestimation of J_1 tends to get worse with the addition of more test points. For J_2, most cases show predictions that range from the extreme possible values of J_2: zero and one. In cases where this range does shrink, it actually moves away from the true distribution of J_2. The cross-validation metrics for all of these surrogates would suggest that any of these surrogates would be appropriate to be used; the issue is therefore the high nonlinearity and high dimensionality of this problem. It should be noted that, for the kriging surrogates, the hyperparameters are not optimized for each surrogate due to computational costs; this could potentially show some improvement in the kriging predictions. Overall, based on this study of surrogate performance, we determined that the surrogates that we considered were not fit for the NASA problem.

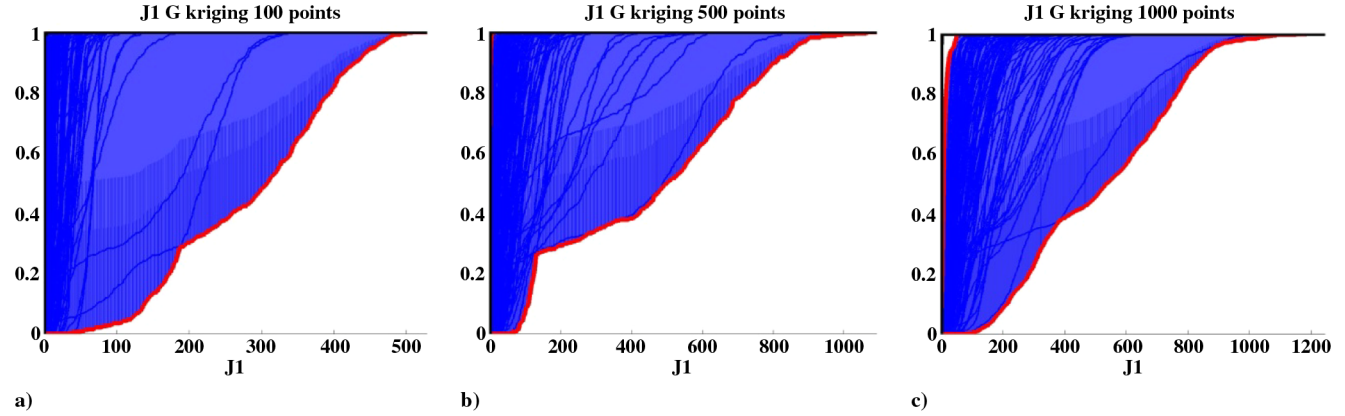


Fig. 7 J_1 estimate using kriging to approximate all constraints with a) 100, b) 500, and c) 1000 samples.

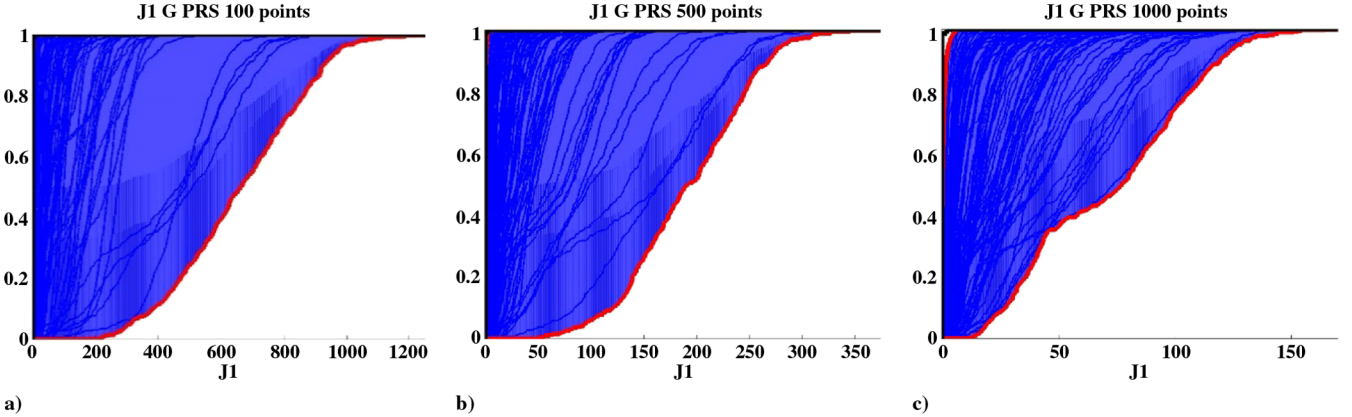


Fig. 8 J_1 estimate using polynomial response surface (PRS) to approximate all constraints with a) 100, b) 500, and c) 1000 samples.

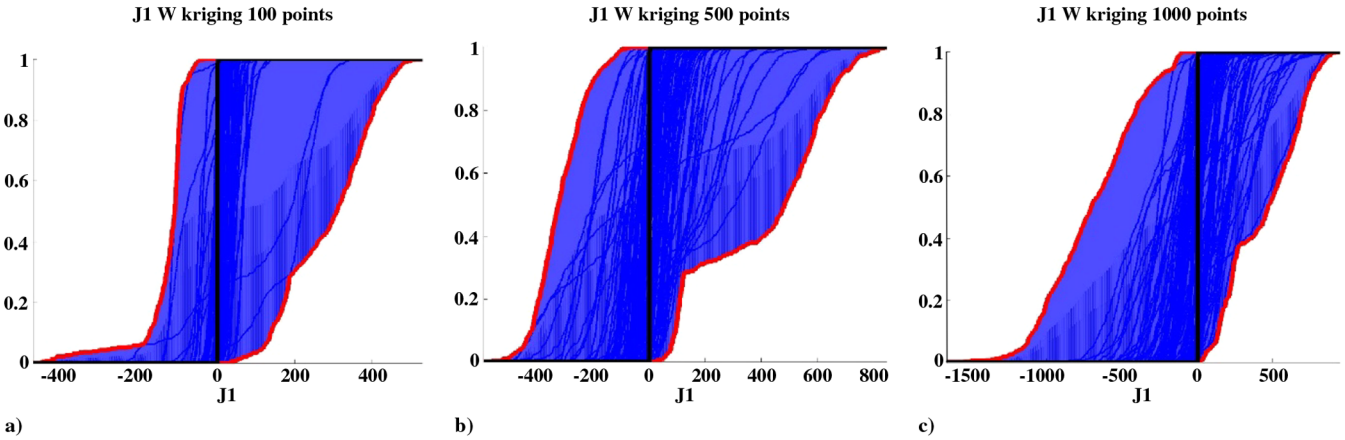


Fig. 9 J_1 estimate using kriging to approximate the worst-case constraint with a) 100, b) 500, and c) 1000 samples.

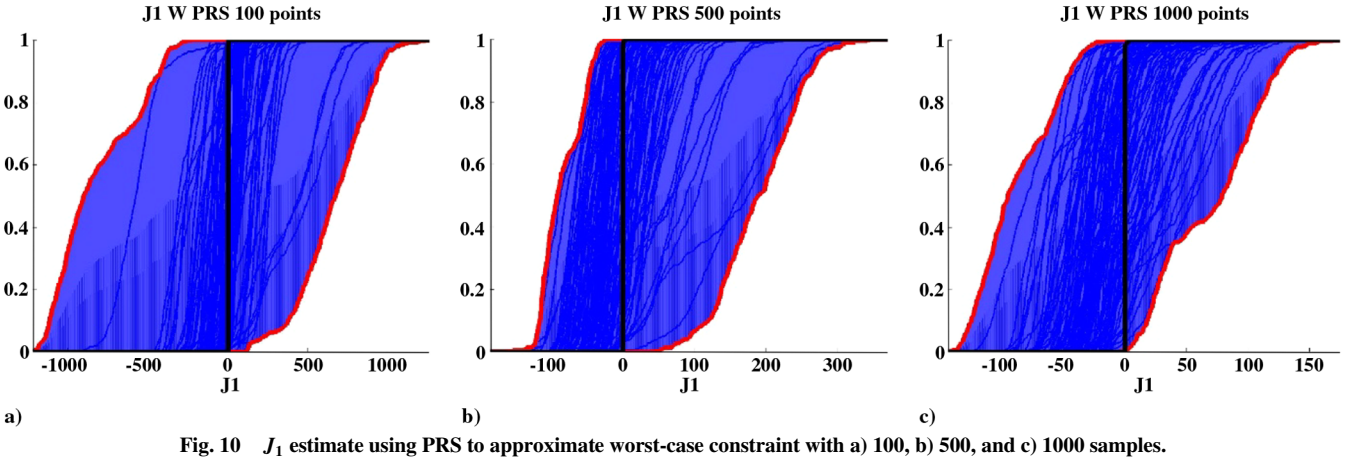


Fig. 10 J_1 estimate using PRS to approximate worst-case constraint with a) 100, b) 500, and c) 1000 samples.

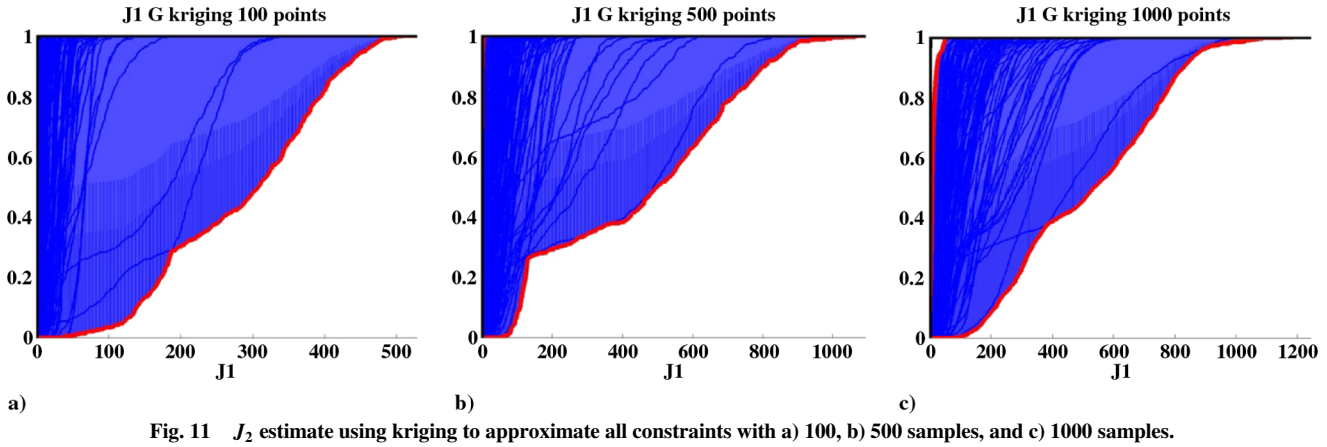


Fig. 11 J_2 estimate using kriging to approximate all constraints with a) 100, b) 500 samples, and c) 1000 samples.

VI. Efficient Reliability Reanalysis

Since we were unable to determine a surrogate model that would be acceptable for the challenge problem, we elect to explore another approach: efficient reliability reanalysis. ERR is a method of reducing the computational cost of a Monte Carlo simulation (MCS) when it is necessary to repeat the MCS for many different subparameters. In efficient reliability reanalysis, a set of initial samples is generated from a sampling probability density function (PDF) $q(\boldsymbol{p})$, the failure indicator function (or other function of interest) is evaluated at the sample points, and then the probability of failure (or mean of function of interest) can be estimated for a new PDF $f(\boldsymbol{p}|\boldsymbol{\theta})$ by weighting the initial samples based on the likelihood ratio $f(\boldsymbol{p}|\boldsymbol{\theta})/q(\boldsymbol{p})$ [10]. Here, we use lowercase theta ($\boldsymbol{\theta}$) to denote a particular realization of the random subparameters Θ. Using ERR, the function of interest is only evaluated at the initial sample points from $q(\boldsymbol{p})$, and then the expected value of the function can be estimated for any new realization of subparameters $\boldsymbol{\theta}$ without any additional function evaluations. ERR is very computationally efficient when the subparameters are random variables because, instead of generating new random samples from $f(\boldsymbol{p}|\boldsymbol{\theta})$ and evaluating the function at the new samples for every

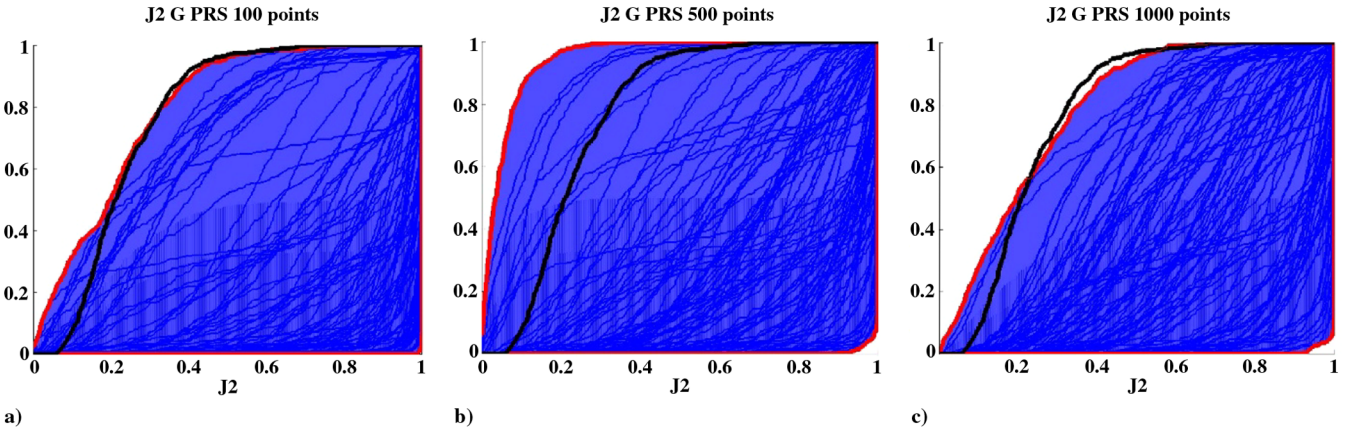


Fig. 12 J_2 estimate using PRS to approximate all constraints with a) 100, b) 500, and c) 1000 samples.

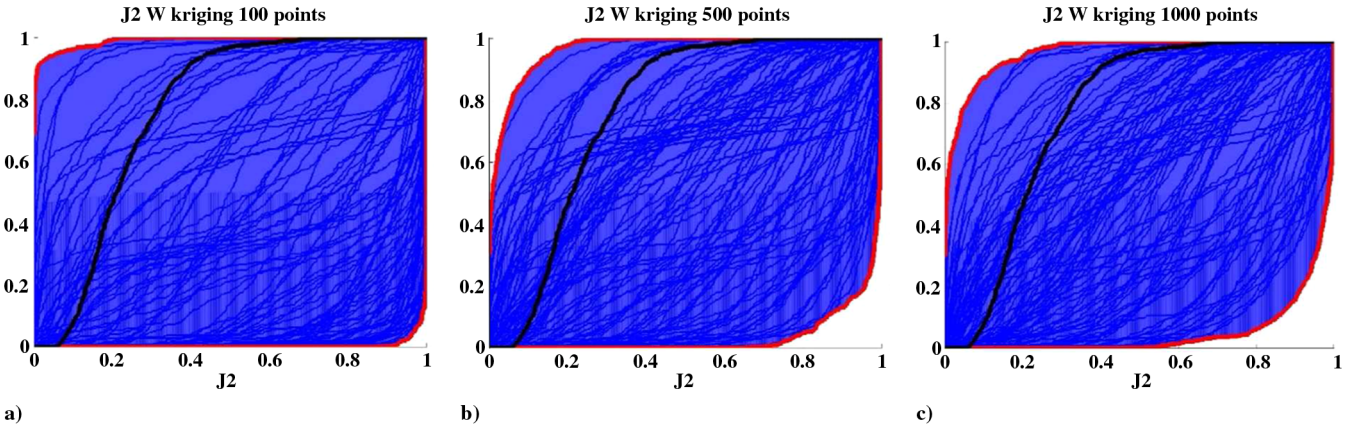


Fig. 13 J_2 estimate using kriging to approximate the worst-case constraint with a) 100, b) 500, and c) 1000 samples.

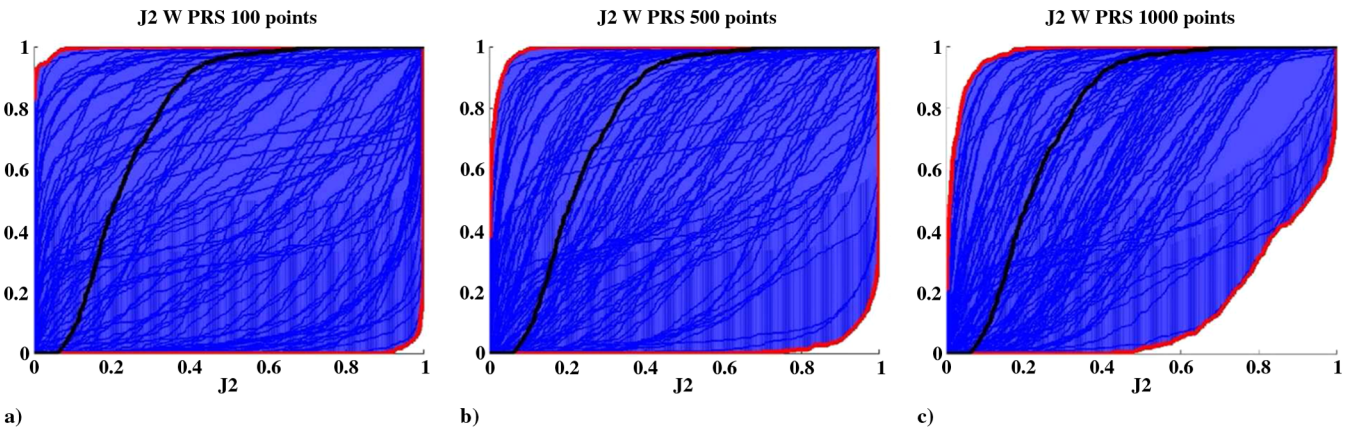


Fig. 14 J_2 estimate using PRS to approximate the worst-case constraint with a) 100, b) 500, and c) 1000 samples.

realization $\boldsymbol{\theta}$, it is only necessary to update the sample weights $f(\boldsymbol{p}|\boldsymbol{\theta})/q(\boldsymbol{p})$. Typically, the function of interest is the failure indicator function and the estimated mean is therefore the probability of failure. However, the method is applicable to other functions, such as the maximum constraint violation.

A. Efficient Reliability Reanalysis Approximations for J_1 and J_2

In the NASA UQ Challenge Problem, we are interested in the expected value of the maximum constraint violation J_1 and the probability of failure J_2 for many different realizations of the subparameters. The value of J_1 conditional on a particular realization of subparameters $\boldsymbol{\theta}$ is defined as shown in Eq. (17). This value of J_1 can be calculated using ERR as shown in Eq. (18). It should be noted that the formulation of ERR requires the support of $f(\boldsymbol{p}|\boldsymbol{\theta})$ to not be outside the support of $q(\boldsymbol{p})$. Similarly, the value of J_2 conditional on a particular realization of subparameters $\boldsymbol{\theta}$ is defined as shown in Eq. (19) and the ERR formulation is shown in Eq. (20):

$$J_1 = E[\max_{1\le i\le 8} g_i] = E[w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}})] = \int_{-\infty}^{+\infty} w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}}) f(\boldsymbol{p}|\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{p} \tag{17}$$

$$\tilde{J}_1 = \int_{-\infty}^{+\infty} w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}}) \frac{f(\boldsymbol{p}|\boldsymbol{\theta})}{q(\boldsymbol{p})} q(\boldsymbol{p})\, \mathrm{d}\boldsymbol{p} = \frac{1}{N}\sum_{i=1}^{N} w(\boldsymbol{p}_i, \boldsymbol{d}_{\text{baseline}}) \frac{f(\boldsymbol{p}_i|\boldsymbol{\theta})}{q(\boldsymbol{p}_i)} \tag{18}$$

$$J_2 = 1 - P[w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}}) < 0] = E[I[w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}}) > 0]] = \int_{-\infty}^{+\infty} I[w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}}) > 0] f(\boldsymbol{p}|\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{p} \tag{19}$$

$$\tilde{J}_2 = \int_{-\infty}^{+\infty} I[w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}}) > 0] \frac{f(\boldsymbol{p}|\boldsymbol{\theta})}{q(\boldsymbol{p})} q(\boldsymbol{p})\, \mathrm{d}\boldsymbol{p} \quad = \frac{1}{N}\sum_{i=1}^{N} I[w(\boldsymbol{p}_i, \boldsymbol{d}_{\text{baseline}}) > 0] \frac{f(\boldsymbol{p}_i|\boldsymbol{\theta})}{q(\boldsymbol{p}_i)} \tag{20}$$

B. Approximating Fixed but Unknown Constants (Type 2 Parameters) as Normal Variables

Before the ERR method could be applied to the NASA UQ Challenge Problem, it was necessary to develop a modification to handle the fixed but unknown constants. Fixed constants present a challenge for ERR because the PDF for the fixed constants is a Dirac delta function centered at the realization of the constant. If the Dirac delta function is used to calculate the sample weights $f(\boldsymbol{p}|\boldsymbol{\theta})/q(\boldsymbol{p})$, all the initial samples (or nearly all) would be weighted zero, since the finite sample size is unlikely to contain the precise realization of interest. To apply the ERR method to the NASA UQ Challenge Problem, we approximated the fixed but unknown constants as normal random variables with an unknown mean and fixed standard deviation. The range of the mean for the normal approximation was selected to be equal to the range of the fixed but unknown constant. A realization of the constant was interpreted as a realization of the mean for the narrow, normal PDF. Selecting an appropriate value for the fixed variance of the normal PDF is challenging because, if the variance is too large, the approximation will be significantly different than the true value. However, if the variance is too small, the majority of the samples will be weighted very low and the effective sample size will be too small to obtain a reasonable estimate. The standard deviation for the normal approximations was selected by comparing the relative error between ERR predictions and Monte Carlo simulation results at 100 test points. Plots of the mean, median, and maximum absolute relative errors in ERR prediction versus MCS at 100 test points are shown in Fig. 15. Based on these plots, we used a standard deviation of 0.1 when using ERR to calculate J_1 and 0.15 when calculating J_2.

C. Selection of Sampling Distributions for ERR

One challenge in the implementation of ERR is the selection of the sampling PDF. Note that the sampling PDF $q(\boldsymbol{p})$ could correspond to a particular epistemic realization Θ^* but does not necessarily have to. That is, we could use $q(\boldsymbol{p}) = f(\boldsymbol{p}|\Theta^*)$ as a sampling PDF where Θ^* is a particular epistemic realization. It is useful to examine the p-box of the $\boldsymbol{p}$ when selecting a sampling PDF. Since the p-boxes for all $\boldsymbol{p}$ except p_4, p_5, and p_{21} cover the entire [0,1] interval with high probability, we use $q(\boldsymbol{p}) \sim \text{Uniform}(0, 1)$ for these $\boldsymbol{p}$. For p_4 and p_5, we use $q(p_4) \sim \text{Normal}(0, 4^2)$ and $q(p_5) \sim \text{Normal}(-2, 4.5^2)$ to cover the range of $\pm 3\sigma$ from the refined uncertainty model with the $\pm 2\sigma$ range of the sampling PDFs. Since the p-box for p_{21} shows that high values of p_{21} have a very low probability, we use $q(p_{21}) \sim \text{Beta}(1, 7.772)$.

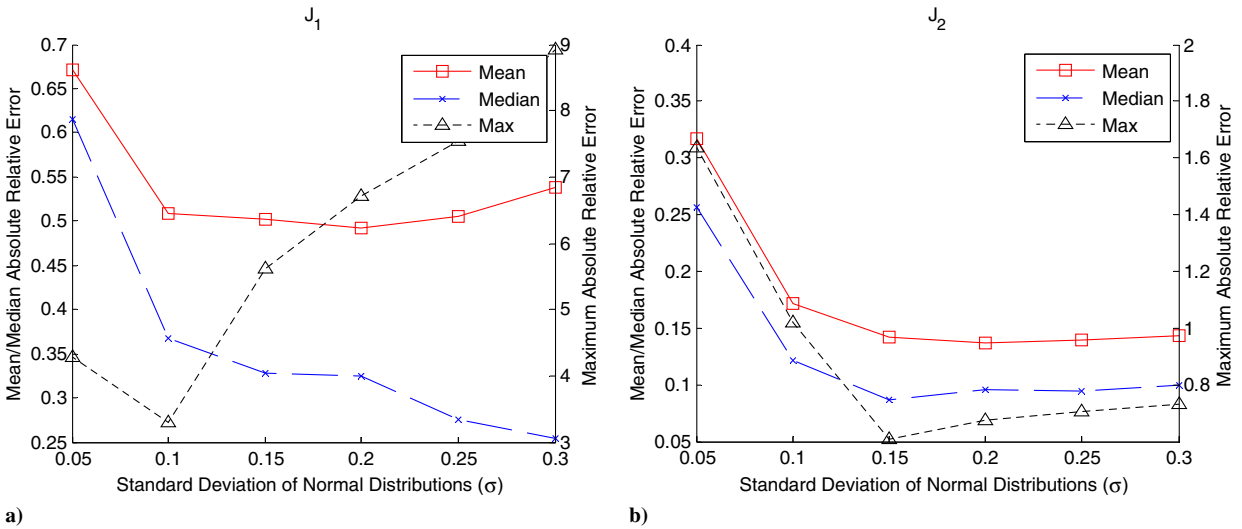


Fig. 15 Mean, median, and maximum (max) absolute relative error in ERR prediction versus MCS at 100 test points for a) J_1, and b) J_2.

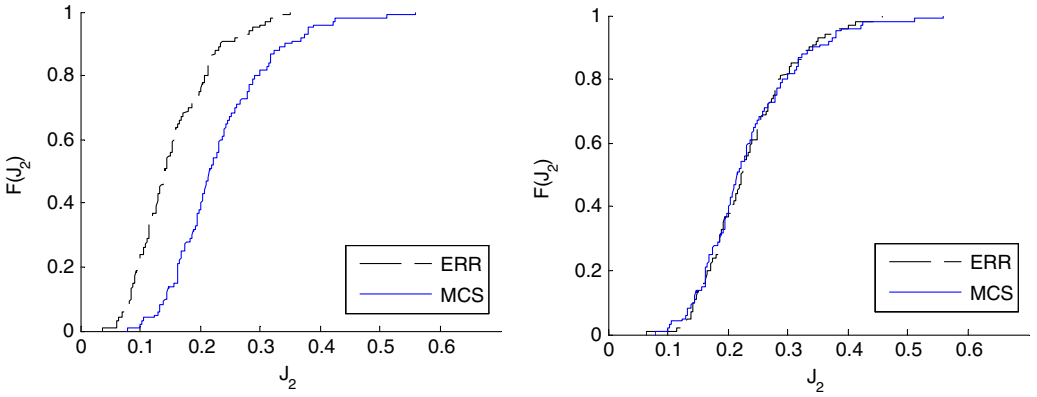


Fig. 16 Comparison of CDF's based on ERR prediction versus MCS at 100 test points for a) standard ERR, and b) self-normalized ERR.

D. Self-Normalizing ERR

Another modification to the ERR formulation was to use a "self-normalized" formulation. In the self-normalized formulation, the sum of the weighted samples is divided by the sum of the weights instead of by the total number of samples. The self-normalized version of ERR is shown in Eqs. (21) and (22). The self-normalized version of ERR is based on the importance sampling formulation for nonnormalized PDFs [23,24]. It should be noted that, for sufficiently large sample sizes, there is unlikely to be a significant difference between the self-normalized and the standard formulations. A comparison study of the relative error between ERR predictions and Monte Carlo simulation results at 100 test points showed a significant improvement in the J_2 estimate when using this self-normalized formulation, as shown in Fig. 16:

$$\hat{J}_1 = \int_{-\infty}^{+\infty} w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}}) \frac{f(\boldsymbol{p}|\boldsymbol{\theta})}{q(\boldsymbol{p})} q(\boldsymbol{p})\, \mathrm{d}\boldsymbol{p} = \frac{\sum_{i=1}^{N} w(\boldsymbol{p}_i, \boldsymbol{d}_{\text{baseline}})[f(\boldsymbol{p}_i|\boldsymbol{\theta})/q(\boldsymbol{p}_i)]}{\sum_{i=1}^{N}[f(\boldsymbol{p}_i|\boldsymbol{\theta})/q(\boldsymbol{p}_i)]} \tag{21}$$

$$\hat{J}_2 = \int_{-\infty}^{+\infty} I[w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}}) > 0] \frac{f(\boldsymbol{p}|\boldsymbol{\theta})}{q(\boldsymbol{p})} q(\boldsymbol{p})\, \mathrm{d}\boldsymbol{p} = \frac{\sum_{i=1}^{N} I[w(\boldsymbol{p}, \boldsymbol{d}_{\text{baseline}}) > 0][f(\boldsymbol{p}_i|\boldsymbol{\theta})/q(\boldsymbol{p}_i)]}{\sum_{i=1}^{N}[f(\boldsymbol{p}_i|\boldsymbol{\theta})/q(\boldsymbol{p}_i)]} \tag{22}$$

E. Capped Self-Normalizing ERR

Another issue that we faced in the implementation of ERR is that, for some samples, the combination of the sampling PDF and true PDF can lead to unreasonably large sample weights. For example, if there is a singularity in the Beta distribution, then the weight for a particular sample can go to infinity (e.g., $p_{21} \sim \text{Beta}(0.421, 7.772)$ has singularity at zero). However, there does not have to be a singularity present for the sample weight to become too large. In our implementation, we cap the maximum sample weight in order to limit the influence a single sample can have on the estimated mean. Although this cap could affect the extreme values that we are interested in, we believe the extreme values we do find with the cap will be more likely to correspond to true extreme values in J_1 and J_2, since they are not as strongly influenced by a single sample. The selection of the appropriate cap for the maximum sample weight is another challenge in the ERR implementation. It is more difficult to select a cap based on only a small number of test points, since the motivation is to have a cap that is only active in the cases where gaps in the samples, or unusual combinations of subparameters, lead to large errors. To select an appropriate value for the cap on maximum sample weight, we looked at the distribution of the ratio of the maximum weight to the sum of all weights at 100 test points and selected a cap to eliminate the outliers in this distribution. This quantity is used as we use the self-normalizing ERR, which normalizes the weights by the sum of all weights. We selected a cap of 10% of the sum of the weights based on the box plot of maximum normalized weight, shown in Fig. 17 (more than 10% maximum normalized weight corresponds to outliers). We dub this method the capped self-normalizing ERR. Henceforth, when we say ERR, we are referring to the capped self-normalizing ERR method.

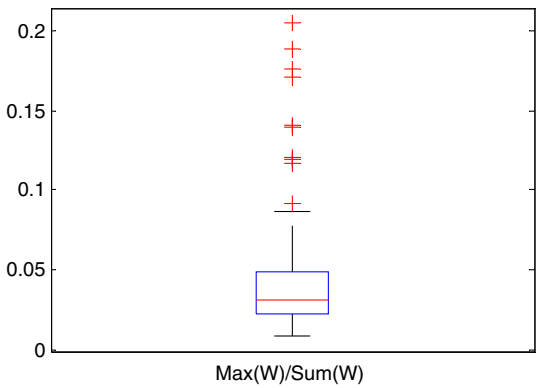


Fig. 17 Maximum sample weight divided by the sum of the weights was calculated at each of the 100 test points.

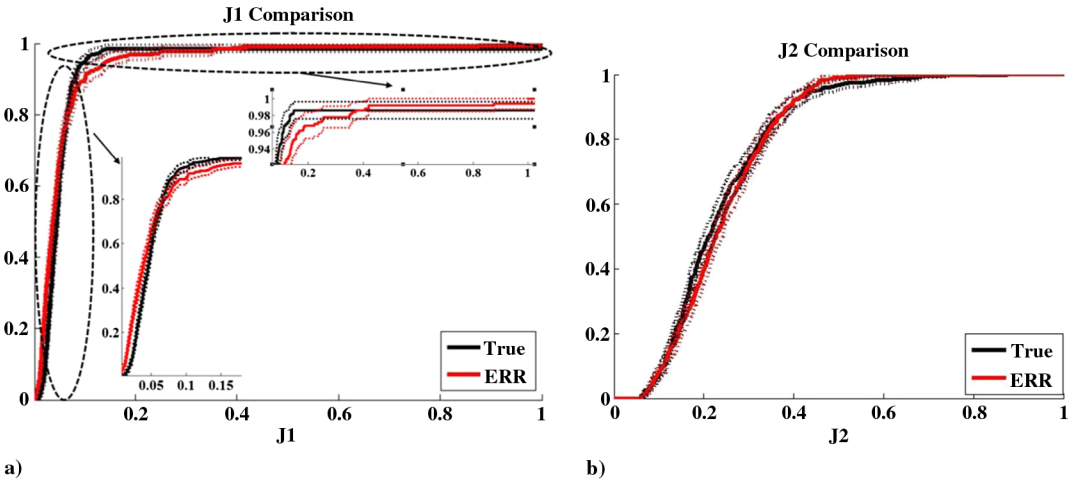


Fig. 18 MCS of the true function and CSN-ERR predictions for a) J1, and b) J2.

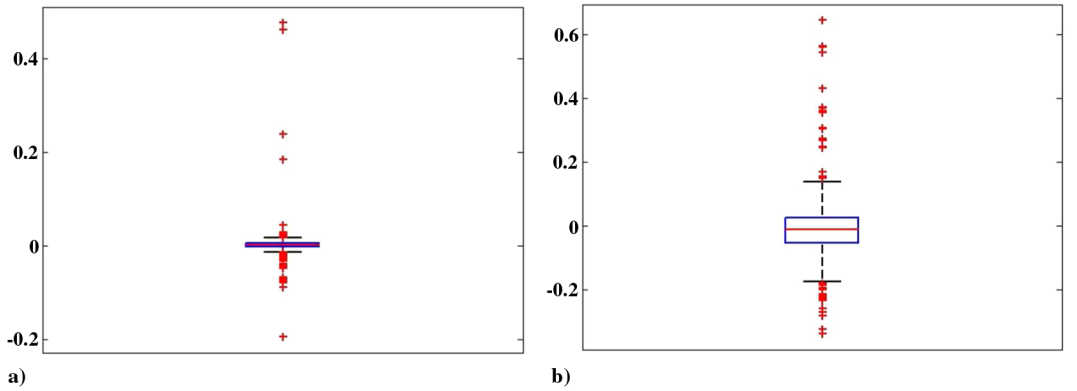


Fig. 19 Percent error in 500 predictions of a) J_1, and b) J_2 for CSN-ERR compared to MCS.

F. CSN-ERR Error Analysis

Before applying the CSN-ERR method in subsequent analysis steps, it was necessary to assess the accuracy of the CSN-ERR predictions. In assessing the CSN-ERR accuracy, we took the perspective that CSN-ERR acts as an approximation (similar to surrogates) for J_1 and J_2 as a function of subparameters $\boldsymbol{\theta}$. To check the accuracy of the CSN-ERR method, we used the same 500 Θ realizations as used in the surrogate error analysis in Sec. V. We then performed a Monte Carlo simulation at each of the 500 test points using 1000 random realizations of $\boldsymbol{p}$ to estimate J_1 and J_2, using the true function for comparison.

Figure 18 shows the comparison of the CDFs for J_1 and J_2 from MCS to the CDFs generated using CSN-ERR. The 95% confidence intervals for the CSN-ERR and MCS ECDFs (shown by dashed lines) overlap nearly everywhere for both quantities of interest. Additionally, box plots for the percent error in the predicted values of J_1 and J_2 for CSN-ERR are provided in Fig. 19. We see that the majority of errors for J_1 and J_2 are within about ± 5 and $\pm 20\%$, respectively, and that the most extreme percent errors are on the order of 40 and 60%, respectively. The CSN-ERR method provides acceptable accuracy for the CDF of both J_1 and J_2, as compared to the surrogates we tested, and can be used as a computationally cheap method of approximating J_1 and J_2. Some strengths and limitations of the different ERR approaches are presented in Appendix E.

VII. Sensitivity of Subparameters to Quantities of Interest (Tasks B2 and B3)

Next, we must calculate the sensitivity of J_1 and J_2 separately to the ranges of each subparameter. Using the information from our intermediate variable sensitivity, we first fix parameters that were found to have little influence on the p-box of the intermediate variables in order to reduce the number of analyses to run on our limited computational budget. As before, we consider three cases of reductions in subparameter bounds: upper, lower, and centered reductions. Since the area of the p-box is not applicable for J_1 and J_2, we consider the range of the value of interest as a scoring method to quantify the effect of this reduction in bounds. Since we are primarily concerned with extreme cases of J_1 and J_2, the sensitivity of the range of these values will allow us to balance the effect on both the maximum and minimum extrema. These results for both ranges of J_1 and J_2 are provided in Table 9.

To select four candidate parameters to get a reduced uncertainty model, we use a rank sum scoring, adding the ranking of each parameter based on the expected effect of both J_1 and J_2 separately in order to estimate the effect of a revised parameter model on both values of interest; these scores are shown in Table 9, where a high rank indicates high combined sensitivity. Based on these findings, we would choose to acquire reduced uncertainty models for parameters 5, 4, 18, and 20. However, we have already requested updated models during the first round of the NASA challenge problem [15] using surrogate methods we later found to be inaccurate. Because of this, we actually have reduced uncertainty models for parameters 21, 1, 18, and 16.

We may now quantify the effect of fixing a value of a subparameter on the range of J_1 or J_2, again using DIRECT optimization to find the optimal value of the parameter to maximize the remaining range. We perform this optimization for J_1 and J_2 separately using 100 random realizations of the nonfixed subparameters over 20 DIRECT evaluations. Since our CSN-ERR model provides us with values for J_1 and J_2 directly

Table 9 Total effect of revised parameter uncertainty on range of J_1 and J_2

	J_1	J_2	
Parameter	Percent change, %	Percent change, %	Rank sum score
1	184	46	16
4	269	83	24
5	271	89	26
6	79	27	3
7	157	52	15
8	178	50	16
10	180	50	16
12	91	28	6
16	85	14	3
17	132	41	8
18	172	55	17
20	182	47	17
21	181	47	15

Table 10 Effect of fixing subparameter values on the range of J_1 and J_2

	J_1		J_2	
Subparameter	Fixed parameter value	Percent change, %	Fixed parameter value	Percent change, %
$E[p_1]$	0.73	91.5	0.63	33.2
$V[p_1]$	0.02	90.8	0.02	30.0
$E[p_4]$	1.61	89.4	−4.80	21.4
$V[p_4]$	0.14	85.8	0.14	16.6
$E[p_5]$	−3.05	91.5	−3.45	28.7
$V[p_5]$	2.04	89.6	3.35	32.0
$\rho_{4,5}$	0.89	85.9	0.89	16.6
p_6	1.00	52.8	0.25	11.7
a_7	3.52	90.3	3.53	29.7
b_7	1.03	91.6	0.62	25.0
a_8	12.08	91.5	7.45	31.2
b_8	5.10	91.6	7.86	30.4
a_{10}	4.51	85.1	1.52	27.9
b_{10}	1.54	74.9	1.56	32.1
p_{12}	0.69	87.9	0.00	27.5
p_{16}	0.01	86.8	1.00	19.3
a_{17}	1.66	91.4	1.66	31.2
b_{17}	1.49	91.0	1.09	32.1
a_{18}	1.06	91.4	4.26	30.1
b_{18}	1.00	90.8	0.55	29.0
a_{20}	13.48	88.5	7.53	26.9
b_{20}	4.72	91.3	8.14	25.6
a_{21}	1.00	56.9	0.66	31.7
b_{21}	7.79	73.6	29.61	29.5

from the subparameters and does not consider parameter values, we choose to fix subparameter values rather than parameters as requested by the NASA challenge; this is a strictly weaker assumption. The remaining ranges and corresponding percent changes for J_1 and J_2 are shown in Table 10. In this case, we use a more aggressive threshold on the percent change in the range of the values of interest in order to meet the requirements of our computational budget in finding extreme-case values. We elect not to fix subparameters that were found to be sensitive due to the fact that we have only a limited sample that we use to justify our fixed subparameter values. If J_1 or J_2 are insensitive to a certain subparameter, we have higher confidence that that subparameter can be fixed without incurring a loss in the range of J_1 or J_2. However, the ability of a subparameter to be fixed and its sensitivity are not equivalent; e.g., if the range of J_1 or J_2 is monotonically increasing with respect to a given subparameter, we can fix this subparameter at its maximum value and still preserve the range of J_1 or J_2, even though the range of J_1 or J_2 maybe highly sensitive to the bounds of this same subparameter (specifically, a decrease in the upper bound, in this example). Based on these results, we determine that we may be able to fix subparameters p_6 and a_{21} when calculating J_1, and subparameters $V[p_4]$ and p_{16} when calculating J_2, without losing a significant portion of the possible range of values. However, by losing any range at all, we recognize we are losing at least one of the true extreme values; by maximizing the remaining range, we should still be close to the extreme value and are able to perform optimization in one fewer dimension for each subparameter that is fixed.

VIII. Uncertainty Propagation and Extreme-Case Analysis (Tasks C and D)

The approach used in this study was to solve "task C: uncertainty propagation" and "task D: extreme case analysis" together. Together, these tasks consist of finding the epistemic realizations of the subparameters that result in extreme values of J_1 and J_2. These tasks therefore require the solution of eight stochastic global optimization problems each in 31 dimensions. The eight problems are to minimize and maximize, J_1 and J_2, for the refined uncertainty model and for the reduced uncertainty model, as shown in Table 11. The domain of the optimization problems is a hyperrectangle, since our refined and reduced uncertainty models are treated as intervals and we do not attempt to infer any correlation between subparameters (see Sec. III.A). For p_1, we optimize the expected value and variance over the rectangular region and map the pairs of mean and

Table 11 Finding the range of J_1 and J_2 and the epistemic realizations that correspond to the extreme values requiring the solution of eight stochastic global optimization problems, each in 31 dimensions

Uncertainty model	J_1	J_2
Refined	$\max_{\boldsymbol{\theta}} J_1$ subject to $\boldsymbol{\theta}^{\text{lower}}_{\text{refined}} \le \boldsymbol{\theta} \le \boldsymbol{\theta}^{\text{upper}}_{\text{refined}}$	$\max_{\boldsymbol{\theta}} J_2$ subject to $\boldsymbol{\theta}^{\text{lower}}_{\text{refined}} \le \boldsymbol{\theta} \le \boldsymbol{\theta}^{\text{upper}}_{\text{refined}}$
	$\min_{\boldsymbol{\theta}} J_1$ subject to $\boldsymbol{\theta}^{\text{lower}}_{\text{refined}} \le \boldsymbol{\theta} \le \boldsymbol{\theta}^{\text{upper}}_{\text{refined}}$	$\min_{\boldsymbol{\theta}} J_2$ subject to $\boldsymbol{\theta}^{\text{lower}}_{\text{refined}} \le \boldsymbol{\theta} \le \boldsymbol{\theta}^{\text{upper}}_{\text{refined}}$
Reduced	$\max_{\boldsymbol{\theta}} J_1$ subject to $\boldsymbol{\theta}^{\text{lower}}_{\text{reduced}} \le \boldsymbol{\theta} \le \boldsymbol{\theta}^{\text{upper}}_{\text{reduced}}$	$\max_{\boldsymbol{\theta}} J_2$ subject to $\boldsymbol{\theta}^{\text{lower}}_{\text{reduced}} \le \boldsymbol{\theta} \le \boldsymbol{\theta}^{\text{upper}}_{\text{reduced}}$
	$\min_{\boldsymbol{\theta}} J_1$ subject to $\boldsymbol{\theta}^{\text{lower}}_{\text{reduced}} \le \boldsymbol{\theta} \le \boldsymbol{\theta}^{\text{upper}}_{\text{reduced}}$	$\min_{\boldsymbol{\theta}} J_2$ subject to $\boldsymbol{\theta}^{\text{lower}}_{\text{reduced}} \le \boldsymbol{\theta} \le \boldsymbol{\theta}^{\text{upper}}_{\text{reduced}}$

variance to beta subparameters alpha and beta using a simple analytical relationship. To enforce the unimodal condition of p_1, we apply a penalty to the objective function when $a \le 1$ or $b \le 1$. Solving the optimization problems is a very challenging task due to the stochastic nature of the objective functions, the high dimensionality of the optimization, and the computational cost of evaluating the black-box function x_to_g.

To overcome these challenges, we used a multifaceted approach. We used the results from $\boldsymbol{p}$ to x sensitivity analysis to fix p_2, p_{13}, p_{14}, and p_{15}. By fixing these p, the subparameters related to these p (seven θ in total) could be removed and the dimensionality of the optimization reduced. To reduce the computational cost of the objective functions, we performed the optimization using the CSN-ERR. It allows values of J_1 and J_2 to be estimated for different sets of subparameters using only a single set of initial samples (see Sec. VI). By reusing the same initial samples throughout the optimization, it was not necessary to evaluate the expensive black-box functions during the optimization. The use of CSN-ERR also has the benefit that the stochastic objective functions were replaced with the deterministic CSN-ERR function. The CSN-ERR function is deterministic, since the same initial samples are used for approximating every evaluation. We used the DIRECT optimization algorithm, which is a nongradient-based optimization method [14], to solve the optimization problems posed in Table 11. We limited the number of function evaluations for each optimization problem to 20,000. The proposed method was tested on the toy problem and showed very good performance (Appendix C). Before performing the optimization on the NASA problem, we performed an analysis using CSN-ERR to try to fix subparameters (see Sec. VII) and found that we were able to fix subparameters p_6 and a_{21} when optimizing J_1 and subparameters $V[p_4]$ and p_{16} when optimizing J_2.

As a point of reference, we compared the CSN-ERR results to the extreme values found using a double-loop Monte Carlo simulation (DLS) using 1000 θ samples each with 1000 p samples. We ran the DLS first, using the refined uncertainty model and then using the reduced uncertainty model. The combined computational cost of the two DLSs was 2e6 constraint function (x_to_g) evaluations. The computational cost of CSN-ERR was equivalent to the two DLSs, since it also required 2e6 constraint function evaluations for the initial samples and the same samples could be used with both uncertainty models. We neglect the additional computational cost of performing the optimization with CSN-ERR, since the computational cost of evaluating the true constraint functions using Monte Carlo is much higher than the computational cost of calculating the weights in CSN-ERR. To check the CSN-ERR and DLS results at the extreme locations, we ran a larger Monte Carlo simulation at these points with 10,000 $\boldsymbol{p}$ samples.

The extreme values of J_1 and J_2 for the refined uncertainty model are presented in Table 12. For J_1, the maximum value was estimated by CSN-ERR to be 275.5. However, when a MCS with 10,000 $\boldsymbol{p}$ samples was performed at this location, the calculated value of J_1 was only 5.7. Despite the large error in CSN-ERR approximation at this location, the optimization with CSN-ERR found a more extreme value of J_1 than was found using DLS (5.7 vs 1.6). The minimum value of J_1 found from optimization with CSN-ERR is also more extreme than the value found using DLS (0.02 vs 0.14). For J_2, CSN-ERR has large errors at the maximum and minimum locations (0.84 vs 0.40 at maximum/0.11 vs 0.07 at minimum). In addition, the minimum and maximum J_2 values found with CSN-ERR are not as good as those obtained from DLS (0.11 vs 0.07 at minimum/0.40 vs 0.73 at maximum). The results are similar for the reduced uncertainty model, as shown in Table 13. For both uncertainty models, optimization with CSN-ERR outperforms DLS when finding the extreme values of J_1 but does not work as well as DLS for finding the extreme values of J_2.

The values presented in Tables 12 and 13 may be misleading if not taken within the context of the overall J_1 and J_2 distributions. In Fig. 20, the CDFs of J_1 and J_2 from MCS for the reduced uncertainty model are compared to the extreme values found using optimization with CSN-ERR. For J_1, the CSN-ERR results bound the entire range of the CDF, since the values are more extreme than any found during the DLS. For J_2, we see that, although there is a large difference between the maximum CSN-ERR value of 0.40 and the MCS value of 0.73, the value of 0.40 is a high percentile of the overall distribution. Similarly, the minimum CSN-ERR value of 0.11 corresponds to a low percentile of the J_2 distribution. The optimization with CSN-ERR has bounded a large majority of the J_2 distribution, as indicated by the dashed vertical lines in the figure. In Fig. 21, the same comparison is made for the reduced uncertainty model. For J_1, we see that several DLS samples were more extreme than the CSN-ERR maximum. However, since these values are based on only 1000 $\boldsymbol{p}$ samples, we are not sure if the true value at these locations is actually larger than the CSN-ERR maximum or if this is due to random error (as can be seen from the comparison of values from 1000 to 10,000 Monte Carlo samples in Table 12). For example, when the maximum value from DLS was checked with more samples, the value dropped from 1.12 to 0.11, as shown in Table 13. Some strengths and limitations of the different approaches are presented in Appendix E.

Table 12 Refined uncertainty model: estimated range of J_1 and J_2 from DLS and optimization with CSN-ERR[a]

	Worst-case requirement metric (J_1)		Probability of failure (J_2)	
Method	Min	Max	Min	Max
DLS (1000 Θ samples by 1000 $\boldsymbol{p}$ samples)	0.02	3.15	0.07	0.73
Optimization with CSN-ERR	0.00	275.51	0.01	0.84
MCS at DLS extrema (Θ at extrema, 10,000 $\boldsymbol{p}$ samples)	0.14	1.58	0.07[b]	0.73[b]
MCS at CSN-ERR optima (Θ at optima, 10,000 $\boldsymbol{p}$ samples)	0.02[b]	5.73[b]	0.11	0.40

[a]A MCS was performed at extreme values to obtain a more accurate estimate.

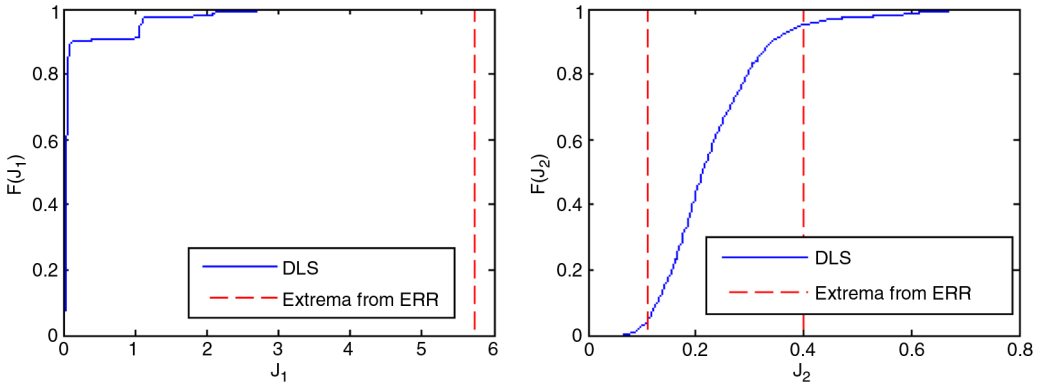
[b]Most extreme values.

Table 13 Reduced uncertainty model: estimated range of J_1 and J_2 from DLS and optimization with CSN-ERR[a]

	Worst-case requirement metric (J_1)		Probability of failure (J_2)	
Method	Min	Max	Min	Max
DLS (1000 θ samples by 1000 $\boldsymbol{p}$ samples)	0.03	1.12	0.15	0.79
Optimization with CSN-ERR	0.00	3.32	0.14	0.69
MCS at DLS extrema (θ at extrema, 10,000 $\boldsymbol{p}$ samples)	0.04[b]	0.11	0.16[b]	0.77[b]
MCS at CSN-ERR optima (θ at optima, 10,000 $\boldsymbol{p}$ samples)	0.05	0.16[b]	0.31	0.46

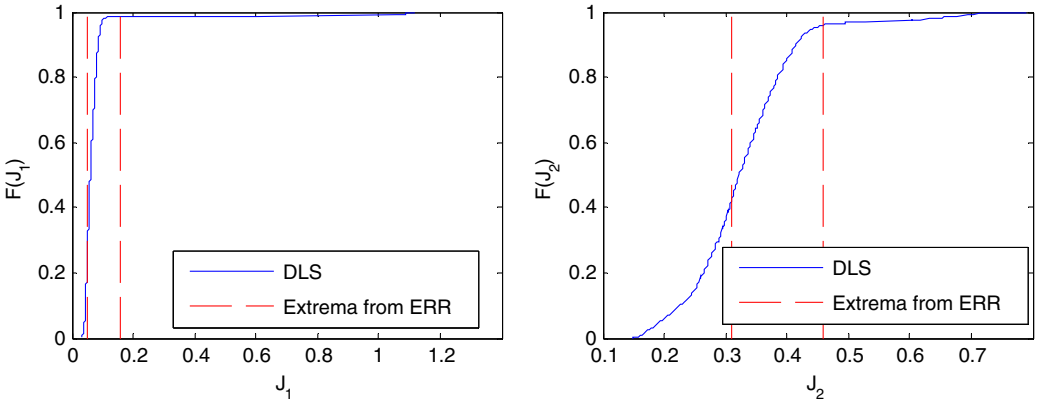
[a]A MCS was performed at extreme values to obtain a more accurate estimate.
[b]Most extreme values.

There are several issues that affect the confidence in the results. We take steps to reduce uncertainty in the results whenever possible but do not have an overall estimate of the confidence due to the computational cost of the NASA problem. The first issue is whether we have in fact located the global optimum for ERR. We cannot know for sure that we have located the true global optimum since the DIRECT optimization algorithm is a sampling-based search algorithm and our computational time is finite. However, by using an optimization algorithm, we expect to have a better chance of locating the global optimum than random sampling (i.e. DLS). The second issue is, even if we locate the global optimum for the CSN-ERR objective function, it may not coincide with the true optimum due to error in CSN-ERR or bias due to the fixed initial sample. For example, in both cases for the optimization for minima of J_2, we found that the CSN-ERR approximation does not well identify the minima found by DLS. This is due to the noise in CSN-ERR, which becomes comparable to range of the J_2 function. However, since we cannot afford to optimize the true function using DLS as an objective function, we must accept this risk that comes with any sort of approximation-based technique (either surrogates or ERR). This risk is also unavoidable, as the computational cost of the NASA problem dictates that we consider some approximation techniques. We also take steps to improve the accuracy of CSN-ERR in order to limit this risk. In fact, the results showed that we can successfully identify an extreme percentile value as compared to a Monte Carlo test sample. The third issue is how accurate our estimate of J_1 and J_2 are at that point, assuming it is the global optimum. To check the CSN-ERR and DLS results at the predicted extreme locations, we ran a larger Monte Carlo simulation at these points. A potential improvement could be to use an adaptive sampling strategy to refine CSN-ERR by adding the information from the predicted optimum. This would refine our model in these regions, and we can continue iteratively. This would require the use of multiple sampling PDFs, which requires further research.



a) b)

Fig. 20 Refined uncertainty model: Comparison of CDF from DLS to CSN-ERR extrema for a) J_1, and b) J_2.



a) b)

Fig. 21 Reduced uncertainty model: Comparison of CDF from DLS to CSN-ERR extrema for a) J_1, and b) J_2.

Table 14 Epistemic realizations leading to extreme values of J_1 and J_2 based on most extreme values from optimization with CSN-ERR

		Refined uncertainty model				Reduced uncertainty model			
		Min J_1	Max J_1	Min J_2	Max J_2	Min J_1	Max J_1	Min J_2	Max J_2
p_1	$E[p_1]$	0.7642	0.7667	0.7659	0.6336	0.6400	0.6400	0.6400	0.6333
	$V[p_1]$	0.0233	0.0230	0.0231	0.0340	0.0360	0.0360	0.0320	0.0320
p_2	p_2	— —	— —	— —	— —	— —	— —	— —	— —
p_4, p_5	$E[p_4]$	−0.1852	−3.2815	0.0434	−0.0328	−1.6600	−1.6600	0.1577	−4.7576
	$E[p_5]$	−4.3621	−3.2735	0.1449	0.1449	−2.4688	−2.4688	0.1449	0.1449
	$V[p_4]$	0.5582	0.2249	0.2468	0.4841	2.6267	2.6267	0.2320	3.4492
	$V[p_5]$	3.1647	1.5079	2.4482	2.3188	1.5079	1.5079	3.9111	1.5226
	ρ	0.8586	0.8915	0.8806	0.8879	0.8915	0.8915	−0.3064	0.8586
p_6	p_6	0.9815	0.4445	0.2531	0.2531	0.4262	0.4262	0.2531	0.2531
p_7	a	3.4897	3.5368	3.5212	2.6801	2.6851	2.6851	3.5212	3.5212
	b	0.6275	0.6190	0.6218	0.6199	1.0800	1.0800	0.6218	0.6218
p_8	a	13.9700	9.6638	11.8377	12.8491	14.0925	14.0925	14.0520	9.6233
	b	4.3513	7.7317	7.5326	6.2660	7.8637	7.8637	6.7373	7.8419
p_{10}	a	4.3467	4.5128	1.6124	1.5262	2.5006	2.5006	4.4945	4.4945
	b	3.8572	4.5122	4.6905	4.7434	4.7498	4.7498	1.5558	1.5558
p_{12}	p_{12}	0.4630	0.7105	0.8827	0.0021	0.5428	0.5428	0.7716	0.0062
p_{13}	a	— —	— —	— —	— —	— —	— —	— —	— —
	b	— —	— —	— —	— —	— —	— —	— —	— —
p_{14}	a	— —	— —	— —	— —	— —	— —	— —	— —
	b	— —	— —	— —	— —	— —	— —	— —	— —
p_{15}	a	— —	— —	— —	— —	— —	— —	— —	— —
	b	— —	— —	— —	— —	— —	— —	— —	— —
p_{16}	p_{16}	0.5000	0.4412	0.9979	0.9979	0.8100	0.8100	0.9979	0.9979
p_{17}	a	1.0934	1.0600	1.4502	1.6608	1.6620	1.6620	1.2570	1.6583
	b	1.4609	1.4880	1.2259	1.4850	1.0000	1.0000	1.4850	1.1657
p_{18}	a	4.2055	3.1771	4.2458	4.2593	1.0001	1.0001	2.5901	1.0099
	b	0.5778	1.0000	0.5558	0.5539	1.0000	1.0000	0.8708	0.8836
p_{20}	a	11.6151	11.5042	10.9526	12.7928	13.4915	13.4915	13.4552	13.4552
	b	8.0844	7.0021	6.1749	8.0136	4.7113	4.7113	5.4960	4.7322
p_{21}	a	0.4317	1.0000	0.4246	0.5104	1.0000	1.0000	0.8708	0.9575
	b	29.2164	7.7737	29.4861	29.5760	19.0003	19.0003	22.9753	22.9753

Epistemic realizations leading to extreme values of J_1 and J_2 based on the most extreme values from optimization with CSN-ERR are presented in Table 14. When based on optimization with ERR, epistemic realizations for p_2, p_{13}, p_{14}, and p_{15} are not available, since they are fixed by a sensitivity analysis.

IX. Conclusions

This work is aimed at tackling the uncertainty quantification challenge posed by NASA, consisting of mixed aleatory and epistemic uncertainties. One of the major challenges was dealing with the dimensionality and the computational cost of the challenge problem. Another challenge was testing the validity of all the different methods that were tested and debugging them. When faced with such a complex problem, it is difficult to test the validity of the results. A simpler toy problem was created, mimicking the performance of the NASA problem, which was faster and easier to analyze. The true solution for the toy was known, which enabled testing of the confidence in the results of different methods. It helped in rigorous debugging, as well as in learning useful lessons to enable modifications to make the methods more efficient.

A CDF matching UQ method is proposed for uncertainty reduction in epistemic parameters based on some available data. A standard Markov chain Monte Carlo-based Bayesian method is also tested for comparison. The CDF matching approach gave similar results as the Bayesian approach using considerably fewer samples. The effect of the number of given observations was also checked, and it showed that using a very low number of observations could be misleading. The methods were applied to the NASA problem, and results were validated to show the effectiveness of the uncertainty reduction. The two different methods for UQ showed similar reduction in bounds for epistemic uncertainty, which further increased confidence in the results. The complementary nature of the CDF matching method also opened ways for future research on using the CDF matching method to guide the Bayesian approach instead of Markov chain Monte Carlo.

A key challenge was the computational cost of the quantities of interest in the rest of the work. Efficient reliability analysis was leveraged to reduce the computational burden of the double-loop Monte Carlo simulation for the NASA problem computation. A way to handle fixed but unknown constants was proposed in the ERR methodology using narrow normal distributions with fixed standard deviations. In addition, two modifications to the ERR method were proposed by introducing a cap on the maximum sample weight and by using a self-normalized formulation. This method was dubbed capped self-normalized ERR or CSN-ERR. It was also shown that CSN-ERR outperformed the surrogates tested in approximating J_1 and J_2.

A sensitivity analysis was conducted to understand the effect of revised uncertainty models. The analysis was based on a double-loop Monte Carlo simulation using the efficient reliability analysis and p-box area as a metric describing the remaining uncertainty. Estimates for the p-boxes of each parameter were obtained by comparing empirical CDFs from multiple uncertain subparameter realizations. The CSN-ERR method was used to reduce the computational burden. Finally, the DIRECT global optimization algorithm was used to attempt to fix values of certain parameters while maximizing the remaining uncertainty captured in the model.

The key contribution in uncertainty propagation and extreme-case analysis was the use of CSN-ERR to perform the optimization. The effectiveness of these methods was evaluated by comparing the results of CSN-ERR with optimization to the results of a double-loop Monte Carlo simulation (DLS) that had the same computational cost in terms of the number of constraint function evaluations. The CSN-ERR method outperformed the DLS method for finding extreme cases of J_1, but DLS worked better on J_2. But, for all the cases, it was seen that an extreme percentile value could be successfully identified as compared to a Monte Carlo test sample. It is hypothesized that the better performance of ERR

on J_1 may be because the larger range of J_1 was clearly distinguishable from the noise in the ERR approximation. Since J_2 is confined to [0,1], it may be more difficult to find the extreme values, depending upon the noise in the ERR approximation.

Appendix A: Bayesian-Based Approach for Uncertainty Characterization

A Bayesian-based approach was also used for the uncertainty characterization. Let $\Theta = (\theta_1, \ldots, \theta_8)$ be a set of eight parameters for the uncertainty variables p_{1-5}, such as the means and variances. By applying the Bayes's theorem [24], the probability density function of the set of the parameters for the given observation $x_{1,\text{obs}}$, called posterior, is defined by

$$f(\Theta|\boldsymbol{x}_{1,\text{obs}}) = \frac{L(\Theta|\boldsymbol{x}_{1,\text{obs}})P(\Theta)}{\int L(\Theta|\boldsymbol{x}_{1,\text{obs}})P(\Theta)\,\mathrm{d}\Theta} \tag{A1}$$

where $L(\Theta|x_{1,\text{obs}})$ is the likelihood function of a particular set of the parameters for given observation $x_{1,\text{obs}}$.

We assume that the ranges of the parameters are known to the designer, as NASA provides, but the designer cannot tell which value is more likely to other values of each parameter. Thereby, the distribution of each parameter is modeled as a uniform distribution; that is, all possible values within the range are equally likely.

Since calculating a marginal distribution for eight parameters involves multiple numerical integrations, its computational cost is very expensive. To overcome this issue, we used an efficient sampling technique, the so-called Markov chain Monte Carlo (MCMC) simulation [11]. MCMC is a technique of drawing the values of parameters θ and keeping the ones that approximate the target distribution, i.e., posterior distributions. To obtain good convergence of the MCMC sampling, we applied the Gelman and Rubin method [12] to evaluate the convergence, and repeated the MCMC 10 times, for each of which $10,000 (= 2n)$ posterior samples of Θ were calculated. Then, the last n draws from all MCMC repetitions were combined together as a set of posterior samples ($10 \times 5000 = 50,000$ draws in total). Figures A1 and A2 show the posterior distributions of the uncertain parameters and the results of the K-S test, respectively. Table A1 describes the corresponding 95% confidence interval of each uncertain parameter. The current implementation of the MCMC method requires 1e-8 aleatory samples. The Markov chain Monte Carlo-based Bayesian method also indicated that there is small linear correlation between the epistemic realizations.

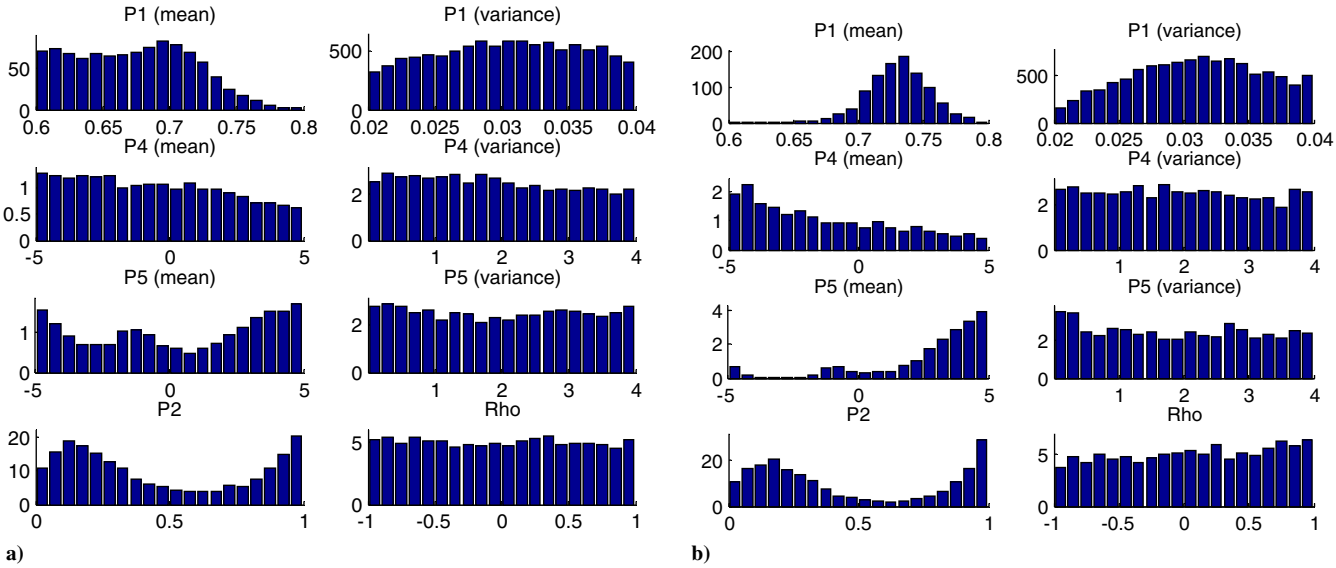


Fig. A1 Marginal posterior distributions of parameters by MCMC a) with the first 25 observations, and b) with all 50 obervations.

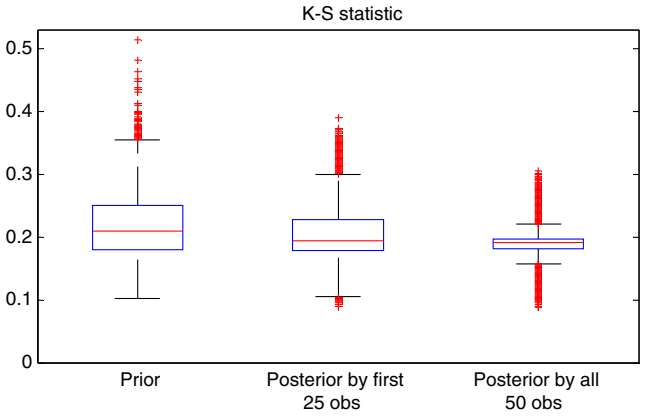


Fig. A2 Results of K-S test by all 50 observations.

Table A1 Refined epistemic bounds using MCMC

		Updated uncertainty model	
Symbol	Given prior	With first 25 observations	With all 50 observations
$E[p_1]$	[0.6, 0.8]	[0.6034, 0.7568]	[0.6783, 0.7762]
$V[p_1]$	[0.02, 0.04]	[0.0208, 0.0394]	[0.0215, 0.0395]
p_2	[0,1]	[0.0283, 0.9885]	[0.0279, 0.9906]
$E[p_4]$	[−5, 5]	[−4.8151, 4.6114]	[−4.8513, 4.3975]
$V[p_4]$	[0.0025, 4]	[0.1087, 3.8893]	[0.1119, 3.8975]
$E[p_5]$	[−5, 5]	[−4.8356, 4.850]	[−4.6209, 4.9289]
$V[p_5]$	[0.0025, 4]	[0.0842, 3.9047]	[0.0644, 3.8799]
ρ	[−1, 1]	[−0.9458, 0.9533]	[−0.9347, 0.9606]

Appendix B: Beta Subparameters

The expected value ($E[x]$) and variance ($V[x]$) of a beta distribution x can be converted to beta subparameters of a and b by using Eqs. (B1) and (B2), respectively, conditional on $V[x] < E[x](1 - E[x])$:

$$a = E[x]\left(\frac{E[x](1 - E[x])}{V[x]} - 1\right) \tag{B1}$$

$$b = (1 - E[x])\left(\frac{E[x](1 - E[x])}{V[x]} - 1\right) \tag{B2}$$

Appendix C: Results for Toy Problem

C.1. Results for Uncertainty Characterization of Toy Problem

The epistemic uncertainties are updated using a first set of five given observations of g_1 and a second set of 20 observations, generated using the known true values of epistemic uncertainty (Table 1) for the toy problem.

C.1.a. Update Uncertainty Models Using Various Observation Sets

The reduced median bounds of θ using the CDF matching UQ method for 50 repetitions using the first set of five observations and a larger set of 20 observations are given in Table C1. The random stream for generating 100 samples of p for a particular θ realization is fixed to reduce the noise in objective function calculation. To check the variation due to fixing the stream, the entire process is repeated 50 times.

Using five observations, the CDF matching UQ approach showed a maximum reduction in range for $E[p_2]$. The reduced bounds for $V[p_2]$ using the CDF matching method does not include the true value of the epistemic uncertainty and predicts it to be lower. This is due to the first five observations of g_1 being biased (Fig. C1). The other parameters mostly remain the same. A high standard deviation in the upper bound for $E[p_2]$ is seen because of the presence of a few outliers (Fig. C2, which gives a box plot of the variation in the lower and upper bounds of reduced epistemic uncertainties due to 50 repetitions). The robustness of the method is really good for the rest of the parameters.

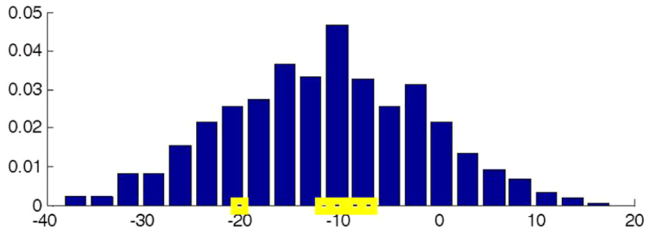
Using 20 observations, we get maximum reduction in range for $E[p_2]$ and $V[p_3] = 0.03$. The reduced bounds for p_1 obtained using the CDF matching method are also centered at the true value of the epistemic uncertainty. Except for the variation in the lower bound of $V[p_2]$, all the other higher standard deviations are found to be mostly due to outliers (as can be seen from the box plots in Fig. C3). Even after considering the standard deviations from 50 repetitions, the true values are still included in the reduced bounds for almost all p_1 (Fig. C3), indicating good robustness.

C.1.b. Effect of Number of Observations on Uncertainty Models

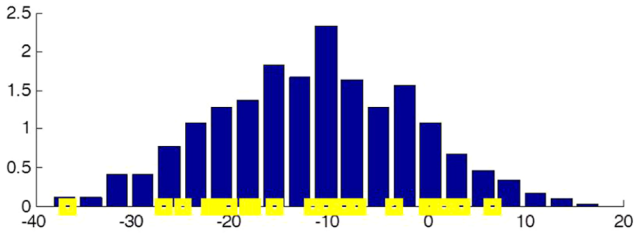
For the CDF matching approach, the K-S test is conducted by generating 7032 θ realizations in the updated median bounds from Table C1 (7000 using Latin hypercube sampling and 32 corner points in five dimensions). For each θ realization, 1000 g_1 samples are generated using the aleatory uncertainty, and the ECDF from these generated samples is compared against the ECDF of 20 given observations of g_1. This process is repeated for all the θ realizations to obtain the variation in the K-S statistic. If the K-S statistics for the posterior distributions are smaller than those for the

Table C1 Reduced epistemic bounds using CDF matching UQ method for first set of five observations and a larger set of 20 observations of g_1 for the toy problem for 50 repetitions

			Using 5 observations		Using 20 observations	
Epistemic uncertainty parameter	True value	Given prior	Reduced bounds (median)	Reduction in median range (% of prior range), %	Reduced bounds (median)	Reduction in median range (% of prior range), %
Δ_{p1}	0.5	[0,1]	[0.0185, 0.9815]	3.7	[0.0185, 0.9938]	2.5
$E[p_2]$	0	[−2, 1]	[−1.1481, 0.5]	45.1	[−0.6111, 0.5]	63
$V[p_2]$	1	[0.5, 1.1]	[0.5037, 0.9333]	28.4	[0.9667, 1.0988]	78
$E[p_3]$	0.7	[0.6, 0.8]	[0.6037, 0.7963]	3.7	[0.6037, 0.7963]	3.7
$V[p_3]$	0.03	[0.02. 0.04]	[0.0210, 0.0396]	6.8	[0.0211, 0.0396]	7.4



a) First 5 observations



b) All 20 observations

Fig. C1 Distribution of g_1 based on the true parameters and the sets of observations (dotted squares): a) first 5 observations, and b) all 20 observations.

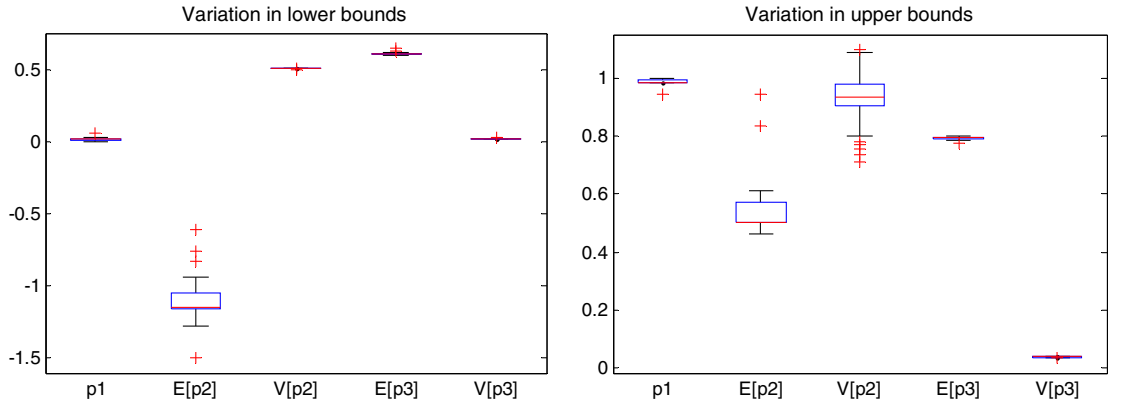


Fig. C2 Variation in reduced epistemic uncertainties using first set of five observations when CDF matching method is repeated 50 times for the toy problem.

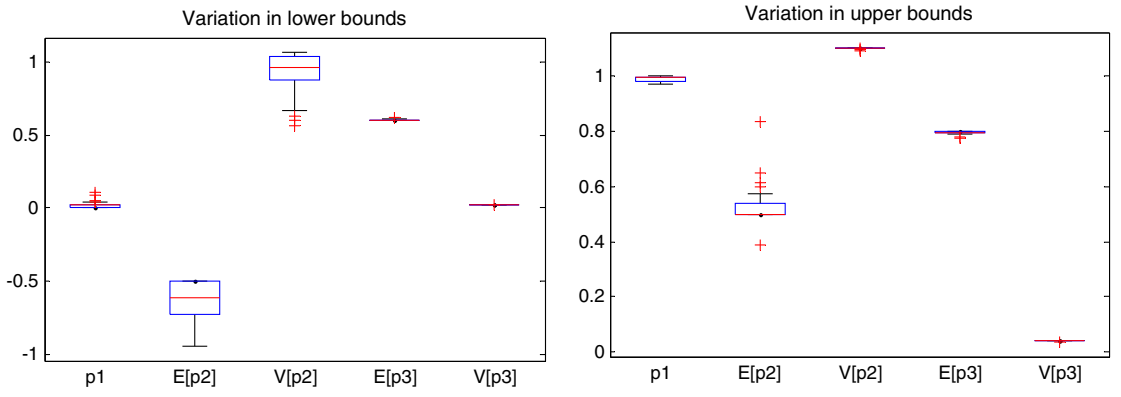


Fig. C3 Variation in reduced epistemic uncertainties using all the 20 observations when the CDF matching method is repeated 50 times for the toy problem.

prior distributions, the uncertainty can be considered to be reduced. Figure C4 compares the box plots of the K-S statistics tested against the ECDF of all the 20 observations. We see that the K-S statistics for the posterior by all 20 observations are substantially smaller than the others for both approaches, which indicates a successful uncertainty reduction for both approaches.

We also look at the percentage of ECDFs for which the null hypothesis of the K-S test (the ECDFs are identical) is rejected with $\alpha = 5\%$. The results for rejection percentages for different sample observation sizes are given in Table C2. The CDF matching method yielded a low rejection rate of 1.6% when all 20 observations were used.

Thus, the toy problem shows that the proposed CDF matching method works successfully. The repeatability of the CDF matching method also showed very good performance, giving us more confidence in the method.

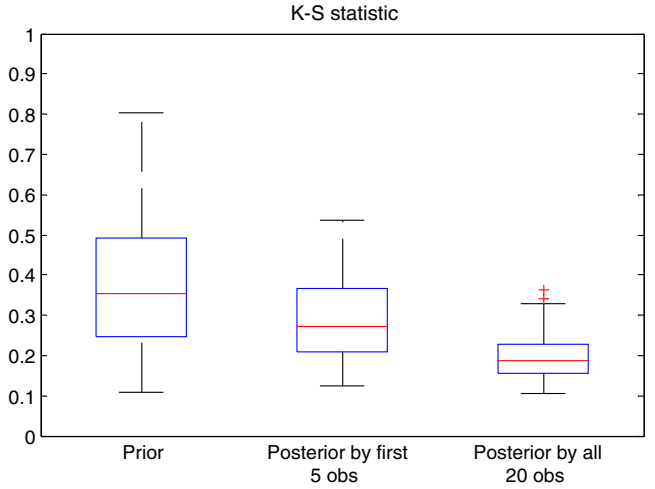


Fig. C4 Comparison of K-S statistics between prior distributions and posterior distributions tested by all 20 observations for the toy problem using and the CDF matching UQ approach.

C.2. Sensitivity Analysis for Toy Problem

We next tested our sensitivity analysis methods on our toy problem. Because there are no intermediate variables in this toy problem, we calculate the effect of the bounds of each subparameter on the p-box of our two constraints. Table C3 shows the final ranking of the effect of revising the uncertainty model for each toy problem parameter on the p-boxes of both g_1 and g_2.

We see that, as expected, p_2 and p_3 are the most influential factors for g_1 and g_2, respectively. We also see that the bounds of p_1 and p_2 have a nonzero influence on g_2, despite not entering the formulation of g_2. This represents the magnitude of numerical error in the area estimate due to limited samples.

Next, we attempt to fix the values of parameters 1, 2, and 3, such that the area of the p-box of constraint 1 is maximized. Table C4 shows the percent change of the p-box of constraint 1 after fixing each parameter at the value shown.

We next calculate the effect of a reduction in the bounds of our subparameters on the ranges of both J_1 and J_2 for our toy problem. Averaging the effect of each of the three cases of bound reductions and combining effects into their relevant parameters, we obtain the final ranking for each parameter in terms of the range of J_1 and J_2, displayed in Table C5.

Finally, we attempt to find fixed values of each parameter that maximize the range of one of our values of interest: J_1 or J_2. These results are shown in Table C6.

Table C2 K-S test rejection percentage for the prior range compared to refined epistemic uncertainty range from CDF matching method using first set of 5 observations and all 20 observations for toy problem

Parameter	Value, %
Prior range	64.1
Refined range by 5 observations	41.5
Refined range by 20 observations	1.6

Table C3 Combined expected change of constraint p-box from refined uncertainty model

Parameter	Percent change in g_1, %	Percent change in g_2, %
p_1	4	8
p_2	21	2
p_3	3	30

Table C4 Change in the p-box of g_1 with fixed parameters

Fixed parameter	Percent change, %	Optimal parameter value
p_1	3	0.5
p_2	91	−2.24
p_3	2	0.5

Table C5 Combined effect of revised uncertainty model on J_1 and J_2 range

Parameter	Percent change in J_1, %	Percent change in J_2, %
p_1	0.2	1.1
p_2	0.5	1.1
p_3	0.4	0.9

Table C6 Remaining J_1 and J_2 range with fixed parameters

	J_1			J_2		
Fixed parameter	Remaining J_1 range	Percent change, %	Fixed parameter value	Remaining J_2 range	Percent change, %	Fixed parameter value
1	5.32	26	0	0.74	16	0.019
2	6.06	16	3.09	0.80	9	1.14
3	7.18	0	0	0.88	0	0.7

Table C7 Toy Problem: Comparison of predicted ranges of J_1 and J_2 using DLS vs optimization with ERR

		Worst-case requirement metric (J_1)		Probability of failure (J_2)	
Method	Number of constraint function evaluations	Min	Max	Min	Max
DLS	1e − 4	−0.31	2.55	0.02	0.56
ERR	1e − 4	−0.35	4.65	0.02	0.65
MCS at ERR optima	1e − 6	−0.30	4.09	0.04	0.62
MCS at DLS extrema	1e − 6	−0.30	1.89	0.08	0.45

C.3. Extreme-Case Analysis for Toy Problem

Optimization for the toy problem was performed using ERR to find the epistemic realizations that resulted in extreme values of J_1 and J_2. ERR was performed using a sample size of 1e − 4 samples. The results of optimization with ERR were compared to a DLS of equivalent computational cost using $100\,\theta$ samples each, with 100 p samples. After finding the locations that corresponded to extreme values of ERR, a MCS was performed at those locations using 1e − 6 p samples. Similarly, a MCS with 1e − 6 p samples was also performed at the locations corresponding to the extrema found using DLS. As shown in Table C7, optimization with ERR found locations corresponding to more extreme values in all cases except for the minimum value of J_1, where both methods returned similar results. In addition, the estimated values from ERR were close to the MCS results at those locations.

Appendix D: Prioritized Observation Uncertainty Quantification Approach

Both performance metrics (J_1 and J_2) measure in some way the risk of failure, so it makes sense to refine the epistemic uncertainties in the random parameters p by prioritizing observations that lead to dangerous designs or higher probability of failure. In general, this ranking can be done based on any desired system performance metric. We fix each of the observed values of x_1 and generate N_{Pf} (here, 1000) sets of Θ using Latin hypercube sampling for epistemic parameters associated with $p_{6\text{–}21}$. For each Θ, multiple aleatory samples of p are sampled using the distribution types provided in Table 2. This gives us N_{pf} matrices of p and, using Eqs. (3–6), we get N_{pf} values of $x_{2\text{–}5}$. Then, we can get a distribution of probability of failure or J_2 for each $x_{1,\text{obs}}$ using Eqs. (7) and (9). The 95 percentile value of J_2 is used to rank the $x_{1,\text{obs}}$. The ones with a higher probability of failure are ranked higher (top), as these observations lead to dangerous system performance and we want the refined uncertainty model to capture them more precisely. Now, this ranking R for each observation is used to effectively weight the observations during the analysis.

We use a weighting strategy to give more weight to the higher-ranked observations of x_1. This leads to a weighted modified Kolmogorov–Smirnov statistic:

$$D_{W_{n,n'}} = \sum_{x_{1,\text{obs}}} (W_R * |F_n(x) - F_{n'}(x)|) \tag{D1}$$

where, W_R is the weight according to the ranking, R of observation x as given by Eq. (D2).

$$W_R = k - \frac{(k-1)(R-1)}{n-1} \tag{D2}$$

where the value of k is fixed at two to ensure that the maximum weight given to any observation is two (user-defined). This is a linear weighting function.

Then, the same process as the CDF matching method is used with $D_{W_{n,n'}}$ as the objective function to identify the optimum range of Θ using Eq. (16). To reduce the noise, we divided the range of $x_{1,\text{obs}}$ into three regions and randomly picked two observed values from each region to find their respective 95 percentile value of J_2. Then, the three regions are ranked from 1 to 3 (highest to lowest probability of failure) based on the picked observations and assigned three weights according to Eq. (D2). Although using the NASA black-box function for getting g, we could not get the results for this method because of the computational cost. The workaround would be to use ERR to find the weights, which we plan to explore in the future.

Appendix E: Strengths and Limitations of ERR-Based Approaches

Table E1 and E2 provide an overview of the strengths and limitations of various methods considered for Efficient Reliability Reanalysis and finding extreme values, respectively. These findings may assist others in assessing the best method to use based on their specific problem and goals.

Table E1 Strengths and limitations of different forms of the ERR method

Method	Strengths	Limitations
ERR	1) Black box functions are only evaluated once initially and then samples are reused for new epistemic realizations 2) Same set of initial samples can be used for any further analysis	1) A sufficient number of the computationally expensive function is still required 2) Predictions may contain some fixed bias, since initial samples are finite
Capped ERR	1) Black box functions are only evaluated once initially and then samples are reused for new epistemic realizations 2) Same set of initial samples can be used for any further analysis 3) Cap on maximum sample weight prevents single sample from dominating estimate of mean	1) A sufficient number of the computationally expensive function is still required 2) Predictions may contain some fixed bias since initial samples are finite 3) Selecting appropriate cap may be challenging
Capped self-normalizing ERR (CSN-ERR)	1) Black box functions are only evaluated once initially and then samples are reused for new epistemic realizations 2) Same set of initial samples can be used for any further analysis 3) Cap on maximum sample weight prevents single sample from dominating estimate of mean 4) Dividing by sum of weights helps compensate for finite sample size	1) A sufficient number of the computationally expensive function is still required 2) Predictions may contain some fixed bias, since initial samples are finite 3) Selecting appropriate cap may be challenging

Table E2 Strengths and limitations of different forms of the DLS and optimization with ERR methods

Method	Strengths	Limitations
DLS	1) All extreme values are obtained in one run 2) Simple	1) Very computationally expensive 2) Difficult to determine how many samples is sufficient 3) Probability of a random sample landing on each extreme value is likely to be small without performing optimization 4) If the range of epistemic uncertainty is changed, then a new DLS might be required
Optimization with CSN-ERR	1) Epistemic realizations are updated intelligently so less samples are needed than with random sampling 2) Cost of optimization is greatly reduced since black-box functions are not evaluated during optimization 3) Optimization can be run for longer due to reduced cost as it does not require any new computation of the expensive function 4) Even if the range of epistemic uncertainty is changed, the same set of initial samples can be used for CSN-ERR	1) A sufficient number of the computationally expensive function is still required 2) Predictions may contain some fixed bias, since initial samples are finite

References

[1] Crespo, L. G., Kenny, S. P., and Giesy, D. P., "The NASA Langley Multidisciplinary Uncertainty Quantification Challenge," 2013, http://uqtools.larc.nasa.gov/nda-uq-challenge-problem-2014/ [retrieved 2014].

[2] Oberkampf, W. L., DeLand, S. M., Rutherford, B. M., Diegert, K. V., and Alvin, K. F., "Estimation of Total Uncertainty in Modeling and Simulation," Sandia National Labs Rept. SAND2000-0824, Albuquerque, NM, 2000.

[3] Oberkampf, W., Helton, J. C., Joslyn, C. A., Wojtkiewicz, S. F., and Ferson, S., "Challenge Problems: Uncertainty in System Response Given Uncertain Parameters," *Reliability Engineering and System Safety*, Vol. 85, Nos. 1–3, 2004, pp. 11–19. doi:10.1016/j.ress.2004.03.002

[4] Shafer, G., *A Mathematical Theory of Evidence*, Princeton Univ. Press, Princeton, NJ, 1976.

[5] Gogu, C., Qiu, Y., Segonds, S., and Bes, C., "Optimization Based Algorithms for Uncertainty Propagation Through Functions with Multidimensional Output Within Evidence Theory," *Journal of Mechanical Design*, Vol. 134, No. 10, 2012, Paper 100914. doi:10.1115/1.4007393

[6] Nikolaidis, E., Chen, S., Cudney, H., Haftka, R. T., and Rosca, R., "Comparison of Probability and Possibility Design Against Catastrophic Failure Under Uncertainty," *Journal of Mechanical Design*, Vol. 126, No. 3, 2004, pp. 386–394. doi:10.1115/1.1701878

[7] Du, L., Choi, K. K., Youn, B. D., and Gorsich, D., "Possibility-Based Design Optimization Method Design Problems with Both Statistical and Fuzzy Input Data," *Journal of Mechanical Design*, Vol. 128, No. 4, 2006, pp. 928–935. doi:10.1115/1.2204972

[8] Youn, B. D., and Choi, K. K., "Selecting Probabilistic Approaches for Reliability-Based Design Optimization," *AIAA Journal*, Vol. 42, No. 1, 2004, pp. 124–131. doi:10.2514/1.9036

[9] Acar, E., Haftka, R. T., and Johnson, T. F., "Tradeoff of Uncertainty Reduction Mechanisms for Reducing Weight of Composite Laminates," *Journal of Mechanical Design*, Vol. 129, No. 3, 2007, pp. 266–274. doi:10.1115/1.2406097

[10] Farizal, F., and Nikolaidis, E., "Assessment of Imprecise Reliability Using Efficient Probabilistic Reanalysis," SAE TP 2007-01-0552, Warrendale, PA, 2007. doi:10.4271/2007-01-0552

[11] Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B., *Bayesian Data Analysis*, Chapman and Hall, New York, 2004.
[12] Gilks, W. R., *Markov Chain Monte Carlo*, John Wiley & Sons, 2005.
[13] Cowles, M. K., and Carlin, B. P., "Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review," *Journal of the American Statistical Association*, Vol. 91, No. 434, 1996, pp. 883–904.
doi:10.2307/2291683
[14] Finkel, D. E., "DIRECT Optimization Algorithm User Guide," 2004, http://www4.ncsu.edu/~ctk/Finkel_Direct/DirectUserGuide_PDF.PDF [retrieved 2014].
[15] Chaudhuri, A., Waycaster, G., Matsumura, T., Price, N. B., and Haftka, R. T., "Framework for Quantification and Risk Analysis for Layered Uncertainty Using Optimization: NASA UQ Challenge," *AIAA SciTech Conference*, AIAA Paper 2014-1498, Jan. 2014.
[16] Ghanem, R., Thimmisetty, C., Yadegaran, I., Keshavarzzadeh, V., Masri, S., Red-Horse, J., Moser, R., Oliver, T., Spanos, P., and Aldraihem, O. J., "A Probabilistic Approach to the NASA Langley Multidisciplinary Uncertainty Quantification Challenge Problem," *AIAA SciTech Conference*, AIAA Paper 2014-1348, Jan. 2014.
[17] Srivastava, A., Subramaniyan, A., and Wang, L., "A Hybrid Bayesian Solution to NASA Uncertainty Quantification Challenge," *AIAA SciTech Conference*, AIAA Paper 2014-1349, Jan. 2014.
[18] Safta, C., Sargsyan, K., Najm, H. N., Chowdhary, K., Debusschere, B., Swiler, L. P., and Eldred, M. S., "Uncertainty Quantification Methods for Model Calibration, Validation, and Risk Analysis," *AIAA SciTech Conference*, AIAA Paper 2014-1497, Jan. 2014.
[19] Liang, C., and Mahadevan, S., "Bayesian Framework for Multidisciplinary Uncertainty Quantification and Optimization," *AIAA SciTech Conference*, AIAA Paper 2014-1499, Jan. 2014.
[20] McFarland, J. M., Bichon, B. J., and Riha, D.S., "A Probabilistic Treatment of Multiple Uncertainty Types: NASA UQ Challenge," *AIAA SciTech Conference*, AIAA Paper 2014-1500, Jan. 2014.
[21] Patelli, E., Broggi, M., de Angelis, M., and Alvarez, D. A., "An Integrated and Efficient Numerical Framework for Uncertainty Quantification: Application to the NASA Langley Multidisciplinary Uncertainty Quantification Challenge," *AIAA SciTech Conference*, AIAA Paper 2014-1501, Jan. 2014.
[22] Bruns, M., and Paredis, C. J. J., "Numerical Methods for Propagating Imprecise Uncertainty," *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, DETC2006*, Sept. 2006.
[23] Cappé, O., Godsill, S. J., and Moulines, E., "An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo," *Proceedings of the IEEE*, Vol. 95, No. 5, 2007, pp. 899–924.
doi:10.1109/JPROC.2007.893250
[24] Douc, R., Guillin, A., Marin, J. M., and Robert, C. P., "Minimum Variance Importance Sampling via Population Monte Carlo," *ESAIM: Probability and Statistics*, Vol. 11, 2007, pp. 427–447.
doi:10.1051/ps:2007028

L. Crespo
Associate Editor