2026 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
DIGITAL ENGINEERING/SYSTEMS ENGINEERING (DE/SE) TECHNICAL SESSION
AUGUST 11-13, 2026 - NOVI, MICHIGAN

A METHOD FOR EVALUATING SYSTEM ROBUSTNESS IN STOCHASTIC, HIGH-DIMENSIONAL, MULTI-FAILURE MODE MODELS

Edward Louis[1], Gregory Mocko[1], Evan Taylor[1]

[1]Clemson University, Clemson SC

ABSTRACT

This paper introduces a method of predicting system robustness using engineering models with aleatory uncertainty. The Stochastic Robustness Evaluation and Categorization (SREC) method is useful for the design of systems where performance along some dimension is limited by several failure modes. SREC integrates and extends interaction plots and Monte Carlo methods to complex engineering models. These complex models are often difficult to evaluate and visualize due to the curse of dimensionality. SREC is effective for non-linear, non-convex, non-monotonic, and discontinuous models due to its basis in Monte Carlo methods. The method is based on the identification of low-performing solutions, the construction of probability density functions and intervals from these solutions, and the categorization of the input space based on the likelihood of low-performing solutions occurring. Using SREC in the late stages of the design process provides insight to the designer about possible improvements in the system's robustness. A ground vehicle model based on a US Army test procedure is used to demonstrate the effectiveness of SREC on high-dimensional, multi-failure mode models.

Citation: E. Louis, G. Mocko, E. Taylor, "A Method for Evaluating System Robustness in Stochastic, High-Dimensional, Multi-Failure Mode Models," In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 11-13, 2026.

1. INTRODUCTION AND MOTIVATION

Simulating an engineered system involves mapping a set of design variables and environmental parameters to a set of performance metrics that a designer then uses to help inform a decision about the system [1]. Often, the designer is concerned with deciding among a set of design variables so that the system performance meets or exceeds a set of design requirements. Acknowledging that design variables and environmental parameters are subject to inherent variance means that guaranteeing that a system always performs well is challenging and often impossible. Assurance that a system performs well must be done probabilistically, and can only quantify how often the system is expected to perform well [2,3,4]. When inputs are represented as probability density functions (PDF), the set of inputs becomes an input space. For high-dimensional input spaces, it is often difficult to understand how combinations of inputs interact to influence the system performance when the number of inputs is large and subject to aleatory uncertainty [5].

Approaches like Response Surface Modeling (RSM) assist a designer in understanding how an output changes across the input space, but the curse of dimensionality makes visualizing the input space difficult [6]. Aleatory uncertainty in the form of input variance adds additional challenge since the relationship between input and output is no longer strictly deterministic. Does the system perform well at the nominal design point, but experience a sharp decrease in performance when a single input varies by a small amount? Are there emergent combinatorial effects when multiple inputs vary? These questions fundamentally underpin robust design methodologies, and are critical for designers seeking to develop robust, high-performing systems [2]. The Stochastic Robustness Evaluation and Categorization (SREC) method, developed in this research, is used to evaluate the robustness of a modeled system in the detailed design phase. At this stage of the systems design process, the nominal values of the design variables have been specified but may vary due to aleatory uncertainty. Further, environmental noise factors that are not controlled by the designer but influence the behavior of the system are introduced and may be represented as deterministic values, probability density functions, or as a range.

This research provides a formal method to model, analyze, and visualize predictive performance models and associated non-linear interactions of complex simulated systems. A method is developed using aspects from design of experiment (DOE) methods and the Monte Carlo method (MCM).

DOE methods are useful for analyzing predictive models, though these models are often ill-behaved, and Taguchi's foundational DOE technique assumes a linear and monotonic relationship between input and output [7]. In practice, real engineered systems often exhibit non-linear, non-convex, non-monotonic, and non-continuous behavior over their input spaces, and models of these systems must faithfully capture these ill-behaved phenomena [8]. Despite their limitations, interaction plots are a useful and widely adopted tool for exploring interactions among discrete, typically extreme values of a high-dimensional input space [7,9,10].

MCM does not stipulate that the relationship between model inputs and outputs must be linear, continuous, monotonic, or convex – MCM can operate on any input-output relationship expressible as a black box function that maps an input space to an output space [2]. MCM results in an output distribution generated from input values sampled from their respective probability density functions. However, a typical MCM does not explicitly connect the output values to their respective inputs, nor does it explicitly capture the interactions and combinatorial effects of the input space, which interaction plots do. MCM is output-focused, and an extension of the method is required to link outputs to a combination of input values.

The SREC methods is comprised of six steps, described below.

1. Execute a Monte Carlo simulation over the input space.
2. Identify solutions that perform outside of the thresholds set by the functional requirements.

3. Construct a subset of the input space from these low-performing solutions, the OTS.
4. Construct probability intervals around the inputs and their respective OTS.
5. Identify and categorize regions bounded by the probability intervals.
6. Quantify likelihood of poor performance within each region.

SREC involves running Monte Carlo simulations over the input space, then constructing subsets of the output space that do not meet specified thresholds. This subset is termed the Outside-of-Threshold Set, or OTS. The construction of the OTS is shown in Fig 1. Some models consider several physical phenomena that limit the performance. These limiting phenomena are called failure modes in this paper, and the identification, prediction, and reduction of failure modes is a crucial task in systems design [11]. For a model with several failure modes, an OTS of each failure mode can be constructed and analyzed separately from the other failure modes.

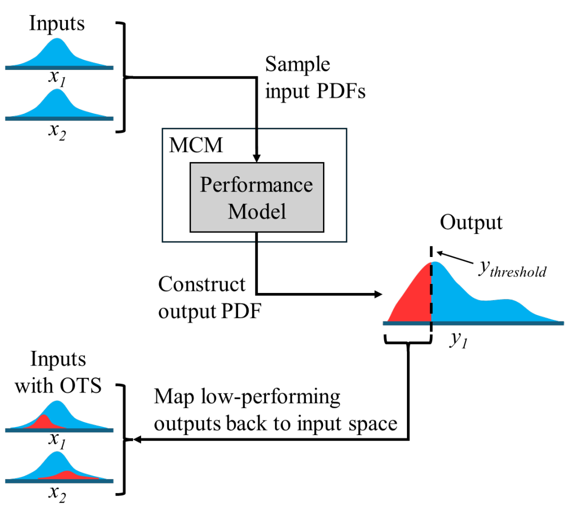


Figure 1: Construction of OTS distributions.

For each input sample distribution and its subset distributions, probability intervals on the input set and OTS are used to categorize the input sample distribution into four regions, shown in Fig 2. These regions correspond to four cases of an input's likelihood to occur, and given its occurrence, its likelihood of yielding performance that does not meet the design requirements.

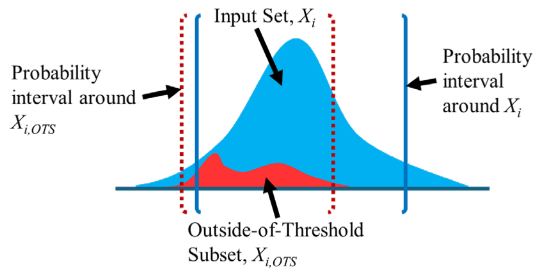


Figure 2: Input distribution with OTS distribution and probability intervals around the distributions.

The structure of the paper follows. Section 2 presents an overview of relevant literature. Section 3 introduces the steps of the SREC method, the categorization of regions within the input space, and the presentation of results. Section 4 applies SREC to a high-dimensional, multi-failure mode ground vehicle model based on US Army Test Operation Procedure, TOP 02-2-610 [12]. This testing procedure describes several failure modes that may limit performance when operating in steeply graded environments and demonstrates the utility of SREC in evaluating the sensitivity of a system to input variance over a large input space. Section 5 discusses the applicability and limitations of the method, which motivates the future work, and concludes with a summarization of the work.

2. LITERATURE REVIEW

Robust design engineering stems largely from the work by Taguchi in the automotive manufacturing industry [9]. Key to the development of robust systems is the classification of system inputs into factors a designer can control and noise that is costly or impossible to control or reduce [13,14]. Taguchi's approach to systems design is to select controllable parameters that yield both desired system performance at the selected design point and a minimum of performance drop-off due to noise and variance. This task is made difficult when many noise factors and many control variables make developing an understanding of system response difficult [10].

Central to grappling the task of mapping the input space to system response are design of experiments (DOE) methods. DOE methods systematically vary input parameters between high and low settings and observe the system response [7,9]. High-dimensional input spaces can be explored in full-factorial, half-factorial, quarter-factorial, and so on based on the size, computational resources, and sampling resolution needed by the designer [7]. These exploratory design experiments assist a designer in developing a high-dimensional response model of a complex system, though a foundational assumption is that system performance between the low and high settings is linear, monotonic, and continuous [15].

Extensions and methods have been developed to address the needs of complex engineering problems. Uncertainty, both quantifiable in the form of noise and unquantifiable in the form of missing information or poor information quality about the phenomena affecting a system have been addressed by [14,16,17,18,19]. Taguchi proposes robust engineering as a minimization of quality loss after product deployment [9]. Optimization methods have been developed and integrated with existing robustness methods to codify and facilitate robust, high-performing, high-quality systems and products [4,14,20,21,22]. Much of the literature frames systems design as a multi-objective optimization (MOO) problem, and the authors acknowledge that systems are evaluated on, ranked by, and ultimately selected on the basis of several (often) competing performance metrics [2,4,13,15,23,24,25]. However, the examples presented in this paper consider one performance metric at a time – though SREC may be applied to as many models of as many performance metrics as is required by the design problem.

Several challenges in systems design in stochastic contexts are prevalent in the literature. Many of these challenges stem from the types of uncertainty in simulation contexts, and what each type implies for the reduction and management of a source of uncertainty. Sinha et al categorize simulation uncertainty into variability (natural uncertainty), model parameter uncertainty (data uncertainty), and model structure uncertainty (model uncertainty) [26].

Variability generally aligns with the accepted definitions of aleatory uncertainty, which is the inherent, irreducible uncertainty of phenomena within a system [3,16]. Parameter uncertainty straddles both aleatory and epistemic uncertainty definitions. A designer may be unsure of what value a design variable or parameter takes on, but depending on the information available to the designer, may be able to quantify the uncertainty (aleatory). They may also have to accept that there are influencing factors yet unknown to the designer (epistemic), though they may be discovered through additional effort [16,26,27,28]. Model structure uncertainty is most aligned with accepted definitions of epistemic uncertainty, where the phenomena, inputs, control factors, and noise factors of a system may be unknown, or must be approximated or neglected via model assumptions [19,26,29,30]. SREC explicitly addresses quantifiable variability and model parameter uncertainty. Insights about possible areas for epistemic uncertainty reduction or model refinement can be gleaned from applying SREC, though only aleatory uncertainty is quantifiably addressed. Support methods for handling or reducing epistemic uncertainty have been developed in [29,30,31].

Another challenge in systems design is the inclusion, intentional or otherwise, of excess capability in a system [24, 25]. Margin refers to excess performance, robustness, and reliability – a factor of safety on the endurance strength of a component is a common example of margin in engineered systems [24]. Margin may be included as insurance against misuse, overuse, or premature failure, or it may be included as a means of "future-proofing" a system against changing requirements. Real-world systems are often designed under shifting use cases, requiring the development of "flexible" systems [32]. An example of margin for future adaptability is the B-52 Stratofortress, which has been in service since 1955, and is intended to remain in service for over a century. In this time, adaptations for fly-by-wire, larger operational range, larger payloads, and anti-aircraft detection and avoidance have been implemented to improve the B-52's suitability for ever-evolving missions [33]. Consideration of mission scenarios and the uncertainty inherent in carrying out real-world missions is both critical to and a challenge for simulation-based design processes [34].

In some cases, margin is included in a design accidentally, as a byproduct of either epistemic uncertainty or of poor understanding of the stack up of margin in the design variables [24]. The method presented in this paper helps a designer understand the individual and combinatorial effects of design variables on system performance to lessen the likelihood of unintentionally including excessive margin. SREC also assists a designer in selecting the right amount of margin for a system. We will demonstrate that the OTS may be unevenly distributed across an input space, so that a small adjustment in margin might drastically increase the likelihood of a system failing, and a large amount of intentionally included margin may not significantly improve the system's robustness.

3. STOCHASTIC ROBUSTNESS EVALUATION AND CATEGORIZATION

SREC utilizes the construction of sets and the construction of probability intervals around sets. Sets are constructed from the input space, and the OTS is a subset of the overall input space. The input space of each design variable is divided into regions based on the intersection of the probability intervals constructed in the prior steps. The interactions and combinatorial effects of the inputs can be visualized using probability density functions and quantified using probability tables. The flowchart in Fig. 3 below outlines the steps for applying SREC to a model of a system with outputs, y.

This section of the paper details each of the steps in Fig. 3 Steps 1, 2, and 3 are discussed in Section 3.1. Steps 4 and 5 are discussed in Section 3.2. Steps 6 and 7 are discussed in Section 3.3. Application of SREC to multi-failure mode models is addressed in Section 3.4.

3.1 SET CONSTRUCTION

From repeated Monte Carlo simulations, an understanding of output probability can be developed from individual input probability density functions. Designers typically place an acceptable threshold on system performance based on the design requirements. MCM can tell the designer what proportion of outputs fail to meet the requirements, but MCM alone does not inform the designer of what combinations of inputs cause an unacceptable output value. MCM also does not help assess the sensitivity of input values to yielding poor performance in combination with other inputs.

SREC requires tracking each input sample during the execution of MCM. An indexed set of each input variable, and an indexed set of outputs is required. The global input space, X, is represented as a matrix, with rows corresponding to individual input sets, X_i, and columns representing samples over

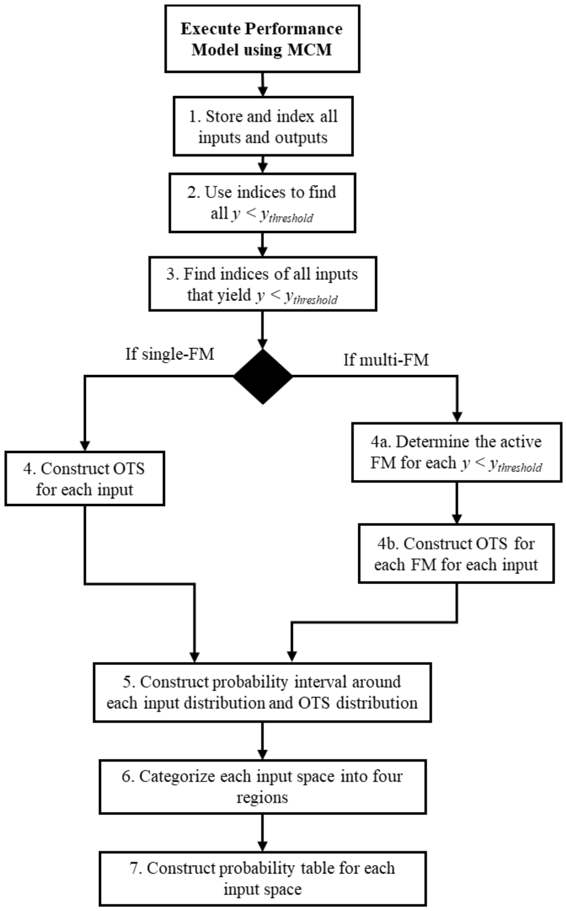


Figure 3: Steps of the SREC method.

successive simulations. For a model with n inputs sampled m times, the following representations of the input space are derived. The global input space, X, can be composed of individual input sets, X_i. This is shown as a column vector in Eqn. 1.

$$X = \begin{Bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_n \end{Bmatrix} \tag{1}$$

Each input set is a vector containing each individual input sample taken over the number of Monte Carlo simulations, m. This is shown in Eqn. 2 as a row vector.

$$X_i = \{x_{i,1} \quad \cdots \quad x_{i,j} \quad \cdots \quad x_{i,m}\} \tag{2}$$

The total input space can thus be represented as an n-by-m matrix shown in Eqn 3.

$$X = \begin{Bmatrix} x_{1,1} & \cdots & x_{1,j} & \cdots & x_{1,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i,1} & \cdots & x_{i,j} & \cdots & x_{i,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,j} & \cdots & x_{n,m} \end{Bmatrix} \tag{3}$$

A column selected from the input space matrix represents an instance of the system, subject to some aleatory uncertainty, expressed as a design vector, DV. Each design vector, DV_j, is an individual instance of the simulated system being perturbed by some uncontrollable noise, and has an associated performance y_j belonging to the output space Y.

$$DV_j = \begin{Bmatrix} x_{1,j} \\ \vdots \\ x_{n,j} \end{Bmatrix} \tag{4}$$

$$Y = \{y_1 \quad \cdots \quad y_j \quad \cdots \quad y_m\} \tag{5}$$

Each output, y_j, is therefore a function of some j^{th} design vector, DV_j. The deterministic model of the system is considered as the function mapping each DV to some y.

$$y_j = f(DV_j) \tag{6}$$

Each output is compared to the performance threshold level, $y_{threshold}$, set from the functional requirements of the system. If an output is below $y_{threshold}$, its index is mapped back to the input space, where those inputs are then stored in the OTS. A given input, $x_{i,j}$, is a member of that input's OTS, $X_{i,OTS}$, if it is part of a design vector that yields a below-threshold performance. Eqn. 7 outlines the construction of these subsets for all inputs making up the global input space. For some j^{th} output, the following holds:

$$X_{i,OTS} = \{x_{i,j} \in X_i | y_j(DV_j) \leq y_{threshold}\} \tag{7}$$

The next section details constructing probability density functions from an input set and its below-threshold set, and the construction of probability intervals around each set.

3.2 CONSTRUCTING PROBABILITY DENSITY FUNCTIONS

Each input set, X_i, is represented as an m-length vector, and each input set has a corresponding OTS, $X_{i,OTS}$, with length less than m. Given sufficiently large m, a histogram can be constructed from each input set that approximates the input's probability density function. The input distributions shown are Gaussian, though they may take on any PDF the designer chooses to represent an input's aleatory uncertainty.

Overlaying the PDF of the OTS over the PDF of the input set is the primary means of quantifying and visualizing how the likelihood of poor system performance changes over the input space in SREC. The proportion of the input space taken up by the OTS, informs the designer of how probable a poor-performing system is to occur. Parameters like the OTS variance, skewness, or bimodality suggest that other phenomena are affecting the system performance. This may prompt a designer to consider individual failure modes of a system – this is discussed in Section 3.4. In this way, SREC indicates to the designer that additional model refinement or epistemic uncertainty reduction could help evaluate the system of interest. The location of the OTS within the input space informs the designer of whether an input is particularly impactful to the output. An OTS that is widely distributed over the entire input space indicates that specific values of that input do not affect the output much, and that low performance is caused by other inputs. An OTS that is clustered densely within the tails indicates that the input has a strong impact on performance. The height at some "slice" along the input space indicates to the designer the likelihood of failure at that point. When the height of the OTS distribution at some point is high relative the the input distribution, then that design point has a high likelihood of poor performance relative to the likelihood of that design point occurring.

Fig. 2 shows the input distribution and the overlain OTS. Note that the OTS is not necessarily the same shape as the input set, nor must its mean, median, or mode coincide with that of the input set. Additionally, the shape of the OTS may offer insights to the designer about the system behavior that MCM and interaction plots alone cannot provide. Fig.2 also shows an OTS with bimodality, which may imply that there are several failure modes contributing to the OTS shape within the input space. Analysis of the parameters and shape of the OTS are discussed further in Section 3.3.

Having constructed PDFs of the input set and its out-of-threshold subset, probability intervals are constructed around the distributions. The choice of probability value is arbitrary; all examples in this paper use a two-tailed interval that contains 95% of the values. Probability intervals on the input set and OTS distributions are shown in Fig. 2.

The bounds on the input set's interval are denoted $\left[\underline{X_i}, \overline{X_i}\right]$, and denoted $\left[\underline{X_{i,OTS}}, \overline{X_{i,OTS}}\right]$ for the OTS. The bounds are defined by Eqn. 8 and Eqn 9.

$$P\left(\underline{X_i} < x_{i,j} < \overline{X_i}\right) = 0.95 \tag{8}$$

$$P\left(\underline{X_{i,OTS}} < x_{i,j} < \overline{X_{i,OTS}}\right) = 0.95 \tag{9}$$

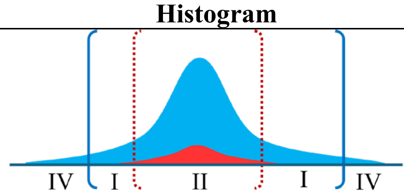
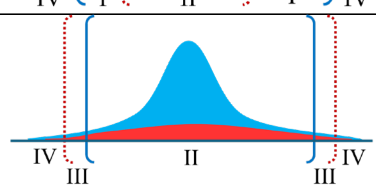
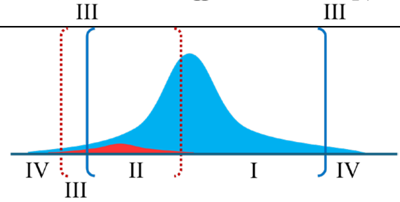
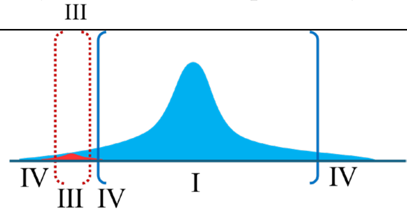
The next section details the categorization of regions of the input space based on the overlap of the intervals. This categorization, and the proportion of each set that lies within each region, ultimately works towards analysis and quantification of the robustness of a modeled system.

3.3 CATEGORIZATION OF INPUT SPACE

Constructing probability intervals around the input distribution and its OTS divides the input space into four regions, shown in Figs. 4a-d. Region I corresponds to the case where samples are within the probability interval of the input distribution, but outside of the interval of the OTS distribution. This paper refers to the regions of a distribution that lie outside of the interval as the tails of that distribution. Region II is made up of samples that fall within the probability intervals of both distributions. Region III samples are in the tails of the input distribution, but within the OTS probability interval. Region IV samples are outside of both distributions' probability intervals – it is composed of the tails of both input and OTS distributions.

The interval bounds can take one of four permutations, shown in Table 1. Case A covers when the input set probability interval fully encompasses the OTS probability interval, shown in Fig. 4a. Case B is when the OTS interval fully encompasses the input set interval, shown in Fig. 4b. Case C covers situations when the bounds on the OTS straddle either the upper or lower bound of the input set, shown in Fig. 4c. Case D is the case when the OTS probability interval is fully outside of the input set interval, shown in Fig. 4d. For cases C and D, the inverse is considered to be equivalent – SREC does not differentiate between an OTS clustered in the left tail or the right tail of the input PDF. What is significant to the evaluation of robustness and input interaction is whether the OTS is clustered to one end of the input space or distributed over the selected design point.

Table 1: Cases for probability interval order

Case	Histogram	Description	Boundary Order
Case A		**Figure 4a:** OTS interval contained within input interval	Lower: $\underline{X_i}$ Lower-Mid: $\underline{X_{i,OTS}}$ Upper-Mid: $\overline{X_{i,OTS}}$ Upper: $\overline{X_i}$
Case B		**Figure 4b:** Input interval contained within OTS interval	Lower: $\underline{X_{i,OTS}}$ Lower-Mid: $\underline{X_i}$ Upper-Mid: $\overline{X_i}$ Upper: $\overline{X_{i,OTS}}$
Case C		**Figure 4c:** Overlap of OTS and input intervals	Lower: $\underline{X_{i,OTS}}$ / $\underline{X_i}$ Lower-Mid: $\underline{X_i}$ / $\underline{X_{i,OTS}}$ Upper-Mid: $\overline{X_{i,OTS}}$ / $\overline{X_i}$ Upper: $\overline{X_i}$ / $\overline{X_{i,OTS}}$
Case D		**Figure 4d:** No overlap of OTS and input interval	Lower: $\underline{X_{i,OTS}}$ / $\underline{X_i}$ Lower-Mid: $\overline{X_{i,OTS}}$ / $\overline{X_i}$ Upper-Mid: $\underline{X_i}$ / $\underline{X_{i,OTS}}$ Upper: $\overline{X_i}$ / $\overline{X_{i,OTS}}$

From Table 1 it can be seen that in cases A and B, when the input set interval fully encompasses the OTS interval and vice versa, then some regions are not present. When the OTS interval is completely contained within the input set interval, then there are no solutions in Region III. When the OTS completely contains the input set – a case where the OTS has high variance and heavy tails, then there are no Region I solutions.

Having constructed probability intervals and established regions based on the intersection of the input and OTS intervals, criteria for evaluating the robustness of a design are established. It is preferred that the OTS is small overall – it is preferable for the system to have a low likelihood of experiencing failure over the entire input space [4]. However, discretizing the input space into individual regions reveals more information about the robustness of the system, especially in cases where the OTS is not small.

It is also preferable for the OTS to take up a small proportion of samples in Region I, and for Region I to contain the nominal design point. Region II is the region in which poor system performance is most expected, but it is still preferable for the proportion of failures in Region II to be small. There are cases in which Region I does not exist, or Region II has a very large interval. These cases are not a problem for the designer on their own – it is acceptable for Region II to cover a large region of the input space if the size of the OTS is very small. Likewise, it is acceptable for Region I not to exist if the OTS is very small.

Cases where the OTS lies in the tails of the input distribution are preferable, as this indicates that variation away from the nominal design point is unlikely to increase the likelihood of a poor performing system. The further into the tails the OTS is pushed, and the smaller this set is relative to the input set, the more assurance a designer has that the design has a low likelihood of experiencing adverse performance at the selected design point. An OTS that is clustered in the tails of a given input distribution indicates that this input is a strong contributor to the system performance. Alternatively, when the OTS is roughly centered within the input set, it indicates that other inputs are contributing to the system's failure to meet performance requirements.

Observing where the OTS exists within the input space can help a designer determine which inputs are especially critical in meeting system requirements, and inform the designer where to allocate resources in the design process. Inputs with an OTS that has a mode occurring at or near the input mode, and variance similar to the input variance indicates that the system is insensitive to changes in this input. The OTS may still be large in this case, indicating a high likelihood of failure, but reducing failure likelihood and ensuring system robustness cannot occur from changes in these insensitive design variables.

Based on the four cases of interval order and the size of the OTS relative to the input set, three criteria that indicate a robust, high-performing vehicle are established.

1. It is preferable for Region I to exist for a given input. An OTS with lower variance and lighter tails than the input set is preferred.
2. Region I should be large. The probability interval on the OTS should be as narrow as possible relative to the input set.
3. A designer should primarily prefer for the OTS within Region I to be small, and secondarily, prefer for the OTS to be small over the entire input space.

A selected design that fulfills these criteria indicates that the selected nominal design point ensures robustness to aleatory uncertainty. Categorizing the input space into regions also indicates the interaction of inputs, highlighting especially sensitive and insensitive regions of the input space.

SREC demonstrates that while Monte Carlo methods alone can assess the likelihood of a system failing to meet requirements across the total input space, and interaction plots can indicate system response for varying inputs at a low level of resolution, there is added value in understanding how each input, in combination with the other inputs, contributes to system performance probabilistically [23]. A low likelihood of poor system performance is always the preference of a designer. In the face of aleatory uncertainty and input noise, it is also valuable for a designer to assess how the likelihood of poor performance changes over the input space, and to ensure that values near the selected design point have a low likelihood of performing poorly.

3.4 MULTI-FAILURE MODE SYSTEMS

Engineered systems often have several failure modes that are active over different regions of the input space. The selection of a design point must consider the tradeoffs of choosing designs that are more susceptible to one failure mode over another, and SREC provides support in making such tradeoffs. The ground vehicle example presented in Section 4.2 presents a case where the mobility of the vehicle is determined by predicting the maximum slope a vehicle can ascend. The maximum slope is limited by one of three failure modes: tip-over, traction loss, or insufficient torque.

The SREC method may be applied to individual failure modes, if the model is extended to store and index the failure mode on performance. It should be noted that additional modeling effort may be needed to consider additional failure modes and extend the model to track and index their occurrence over repeated Monte Carlo sampling. The example presented in Section 4.2 uses a series of if-then checks to determine which failure mode was active when the system output failed to meet performance requirements.

In multi-failure mode models, SREC can be applied to all outside-of-threshold inputs, or applied individually to each failure mode. This is demonstrated in Fig. 5, where the total OTS is a two-humped distribution. Suppose that a closer look at the physics of the model reveals that there are two physical phenomena that cause the system to fail to meet a performance requirement, and the model is extended to track which phenomenon is at fault for causing a sample to fall in the OTS.

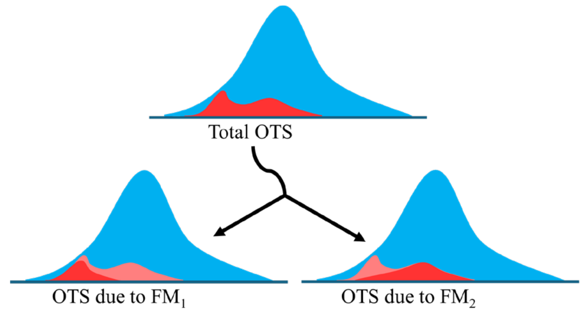


Figure 5: Decomposition of the total OTS into individual failure modes.

Probability intervals are constructed around the OTS of each failure mode, and the same procedure for constructing regions as the total OTS can be followed. The size of each failure mode's OTS relative to the other failure mode informs the designer of what failure mode is most often the limiting case for system performance. Identifying each failure mode's contribution to the total OTS in cases where the OTS has high variance is also helpful. There may be regions of the input space in which a failure mode is more active, and regions where another failure mode is more active.

Designers can use this knowledge of what failure modes limit performance, where within the input space they limit performance, and how often they occur, to better evaluate system robustness. For example, some failure modes may be especially detrimental to the system in terms of structural damage, financial loss, or danger to human users. Others may be relatively benign in context of the entire system life cycle, but still cause the system to underperform in the context of some scenario. For example, the gradeability example in Section 4.2 considers both tip-over and traction loss as physical phenomena that prevent a vehicle from ascending a grade. Vehicle tip-overs are catastrophic failures and are likely to render a vehicle totally inoperable in the field and cause injury to the occupants. Another failure mode, traction loss, is unlikely to render the vehicle inoperable, as the vehicle will most likely slide down the grade to a point where it regains traction. The vehicle fails to ascend the grade in that scenario, but the vehicle is not damaged to an extent that it cannot continue to operate in other scenarios and along other performance dimensions. In such a case, vehicle designs with a small tip-over OTS relegated in the tails of the input space are preferred, even if the traction-loss OTS is larger and the selected design point is encapsulated in the traction-loss OTS interval.

4. APPLICATION OF SREC TO GROUND VEHICLE MODEL

An example model based on a US Army Test Operation Procedure, TOP 02-2-610 Gradeability and Side Slope Performance is analyzed using SREC. This testing procedure prescribes testing methods to determine a ground vehicle's performance in graded environments. The specific test presented here is *4.1.3 Longitudinal Slope Speed* which determines a wheeled vehicle's ability to drive up a longitudinal grade when starting from rest on the grade. The test prescribes three conditions that indicate the vehicle's failure to ascend the grade: tip-over, traction loss, and insufficient torque.

Figure 6: HMMWV ascending a grade.

The vehicle is to be started from rest and driven up slopes of increasing grade, until a grade that the vehicle cannot ascend is found. When the scenario is conducted in real-life, the vehicle is instrumented and observed by the conductors of the test to determine if a failure mode occurs. In tip-over failures, the vehicle's front wheels lose contact with the ground, and the vehicle rolls over backwards. In traction failures, the tires have insufficient grip to hold the vehicle on the grade, and the vehicle slides down the grade. In cases where the vehicle has insufficient torque from the driveline, the vehicle cannot begin to move from a stop, even if the tires have sufficient grip.

A MATLAB model based on this testing standard uses a series of checks to determine if the vehicle meets or fails each failure mode at a given grade. Front and rear tire normal forces, tractive force, tire grip force, and the horizontal position of the center of gravity are determined. The vehicle experiences tip-over failure when the front wheel normal force drops to zero. The vehicle model experiences traction failure when the friction force – calculated using Coulomb friction assumptions, is less than the component of vehicle weight acting parallel to the grade. The vehicle model experiences torque failure when the tractive force is less than the component of vehicle weight acting parallel to the grade. The bisection method is used to converge to a maximum critical grade value. The active failure mode when the model converges is taken as the failure mode for that design vector.

The model takes seven design variables as input, where four are represented using probability density functions. The height of the center of gravity (CG), static coefficient of friction between wheel and ground, and driveline gear reduction are taken as deterministic inputs. The vehicle weight, horizontal distance from the CG to the rear wheel, horizontal distance from the CG to the front wheel, and peak torque from the motor are represented as Gaussian distributions.

The selected design points are based on the HMMWV, though standard deviations are exaggerated in this example to increase the size of the input space, and increase the likelihood of a *DV* failing to meet performance requirements. The results of this model are not indicative of actual HMMWV performance on grades, but is done so the OTS sets are large and easily visualized for the purposes of demonstrating SREC. Table 2 shows each of the stochastic inputs to the model, and their mean and standard deviation. The gradeability threshold for this model is 27.0 degrees.

Table 2: Parameters of gradeability model inputs.

Design Variable	Mean	Standard Deviation
Vehicle Weight [N]	60,000 N	5,000 N
CG to R. Wheel Distance [m]	1.8 m	1 m
CG to F. Wheel Distance [m]	1.5 m	0.5 m
Peak Motor Torque [Nm]	515 Nm	150 Nm

This model formulation, with four stochastic inputs and one output, yields a five-dimensional model, and a four-dimensional region of below-threshold values bounded by $y_{threshold}$. The high dimensionality of this problem makes visualizing the combinatorial effects of design variables difficult. Running Monte Carlo simulations alone yields the designer limited information about how variance in the design variables increases or decreases the likelihood of the system failing to meet requirements. To demonstrate this, the setup for the MCM model of the gradeability model is shown in Fig. 7, where the likelihood of the vehicle underperforming is found to be 22.9%.

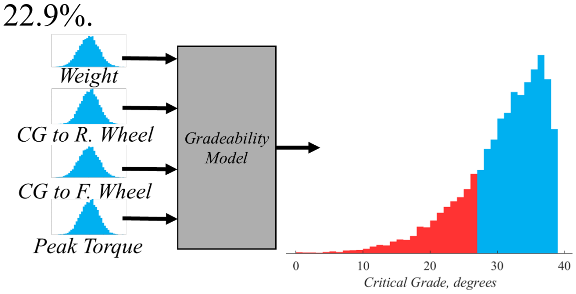


Figure 7: Monte Carlo simulation of gradeability model.

The model considers four inputs, each with an OTS for all below-threshold values, and an OTS for each of the three failure modes. This yields sixteen histogram plots and probability tables. Every input, failure mode, and probability table is not shown in this example – a few key illustrative examples are shown, though data for the entire model is available. A primary function of the gradeability example is to demonstrate the application of SREC to multi-failure mode models. Shown in Fig. 8 is the peak motor torque input and its OTS for all failure mode together and individually. It is shown here that similar to Fig. 5, the OTS due to all failure modes combined can be discretized into individual OTS histograms.

From these figures, it can be seen that each failure mode sums together to form the OTS of all below-threshold values. Failure to meet requirements due to traction failure results in an OTS that is centered within the input space of peak motor torque, while failure to ascend a grade due to insufficient torque is skewed heavily left within the input space. Relatively few *DV*'s result in a tip over failure, but those that do have a large variance over the input space, indicating that peak motor torque is not influential on tip-over failures.

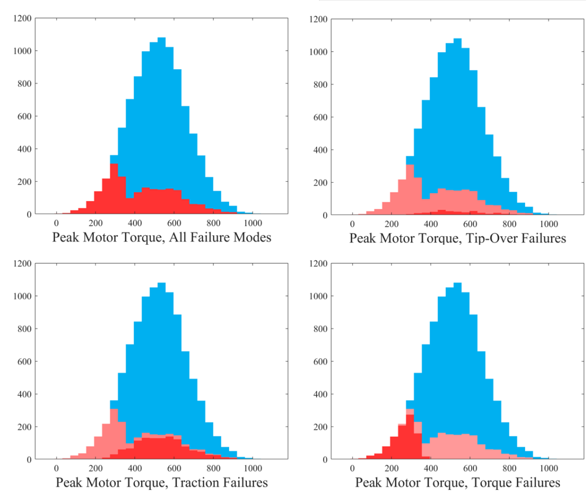


Figure 8: Multi-failure mode decomposition of peak motor torque input.

An input of interest is the CG to rear wheel distance. This is a case where the OTS of all failure modes has a higher variance than the input distribution itself, corresponding to case B from Table 1. This can be seen in Fig. 9, where the probability interval on the OTS, shown in dashed red, fully encompasses the probability interval of the input set, shown in solid blue. Considering each failure mode on its own reveals that no individual failure mode has a higher variance than the input set, but the summation of all failure modes yields a high overall variance.

This input shows a high OTS variance when considering all failures together. Considering each failure mode alone provides additional design decision-making affordance to a designer. Tip-over failures are catastrophic failure modes, posing a high risk of damage to the system and danger to the occupants of the vehicle. The OTS of the tip-over failure mode indicates that Region II of the tip-over plot takes up a small amount of the input space overall – only 6.8% of all values fall within this region, and the region itself has an 8.4% chance of failure. Region I takes up 88.2% of the input space, and contains the selected design point. The likelihood of experiencing a tip-over failure within Region I is 0.1%, indicating a high degree of robustness to tip-over failures at the selected design point.

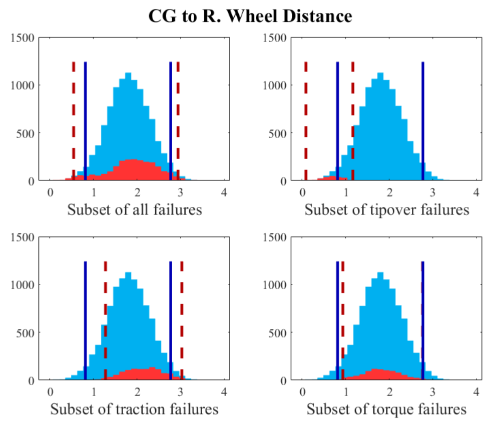


Figure 9: SREC output plots for CG to rear wheel distance input.

Table 3: Probability table for all failure modes of CG to rear wheel distance input.

	Reg. II	Reg. III	Reg. IV
Input Proportion	95.0%	3.4%	1.6%
OTS Proportion	20.1%	1.8%	1.2%
OT Likelihood	21.2%	53.8%	72.3%

Table 4: Probability table for tip-over failures of CG to rear wheel distance input.

	Reg. I	Reg. II	Reg. III	Reg. IV
Input Proportion	88.2%	6.8%	2.4%	2.5%
OTS Proportion	0.1%	0.6%	1.5%	0.0%
OT Likelihood	0.1%	8.4%	59.4%	1.6%

Another input of interest is the vehicle weight, where the variance of the total OTS is not much different from the variance of each failure mode OTS, shown in Fig 10. This indicates that the vehicle's weight is not a particularly strong contributor to the gradeability performance of the vehicle for any of the failure modes considered.

In this case, Region I of the total OTS is very small, and Region II takes up the majority of the input space. The tip-over failure OTS shows a similar pattern, where Region II occupies 93.2% of all inputs, but since the likelihood of failure within Reg. II is only 2.1%, a designer has some assurance that the likelihood of experiencing a tip-over failure is low. The other failure modes have

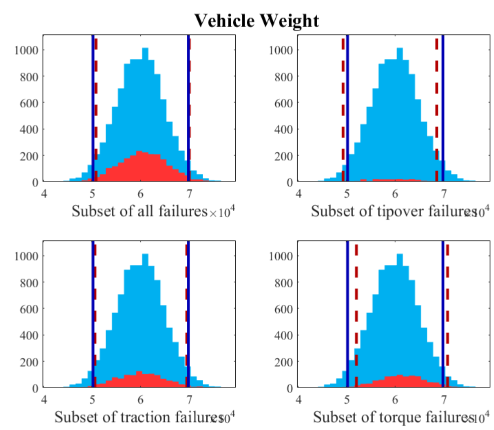


Figure 10: SREC output plots for vehicle weight input.

Table 5: Probability table for all failure modes of vehicle weight input.

	Reg. I	Reg. II	Reg. III	Reg. IV
Input Proportion	0.9%	94.1%	0.3%	4.7%
OTS Proportion	0.1%	21.9%	0.1%	1.0%
OT Likelihood	14.6%	23.2%	25.0%	21.7%

Table 6: Probability table for tip-over failures of vehicle weight input.

	Reg. I	Reg. II	Reg. III	Reg. IV
Input Proportion	1.8%	93.2%	0.9%	4.1%
OTS Proportion	0.0%	2.0%	0.0%	0.1%
OT Likelihood	1.1%	2.1%	4.3%	1.7%

similarly large variances, which indicates that design effort to control the variance in weight is not likely to yield much improvement in gradeability performance – design effort is best spent on other inputs.

5. Conclusion and Future Work

The work in this paper presents a method for mapping a system performance value to the design variables that yield that performance when the design variables are subject to uncertainty. Constructing a probability distribution from the OTS affords the designer several insights in the design process. Inputs at or near the mode of the OTS distribution indicates that there are many design variable combinations that yield a poor-performing system at that input value, indicating that that input value is particularly active in contributing to poor system performance.

The size and location of the OTS's probability interval within the input space also indicates how active of a contributor an input is to system performance. When the OTS is centered within the input distribution with roughly equal variance to the input set – take the HMMWV's weight input as an example – the OTS for each failure mode is approximately centered and has similar variance to the input distribution. This indicates that the input is not a strong contributor to the system's failure to meet the gradeability performance threshold. The likelihood of failing to meet requirements at some input value relative to the likelihood of that input value occurring is roughly even across the input space in such cases. In cases where the OTS is heavily shifted to the left or right of the input space, it indicates that the region of the input space where the OTS is clustered should be avoided, or design decisions to shift the OTS further into the tail of the input distribution should be made. This is especially valuable to the designer when considering several failure modes, as some failure modes are more likely in one region of the input space than another.

The consideration of where the OTS lies is codified and quantified using the region classification system and probability tables. It is preferable for Region I to be as large as possible, with as low of a likelihood as possible of outside-of-threshold values occurring in this region. Region II is preferred to be small, as this region contains designs with a high likelihood of occurrence and high likelihood of poor performance. Region III is also preferred to be small, though values in Region III do not occur often, as they are in the tails of the input distribution. Region IV is of least concern, where occurrence and poor performance are both unlikely.

A promising use case for SREC is in multi-failure mode systems, where a probabilistic assessment of several failure modes over the input space is a fundamental feature of the method. SREC can be used to assess the likelihood of individual failure modes, and where they exist relative to the selected design point. If a failure mode OTS lies to the left of the selected design point, for example, and this failure mode is a critical, catastrophic failure mode, then a designer knows that effort to control the variance in that region of the input space will have the largest impact on improving the robustness of the system. SREC is an additional tool to quantify both the likelihoods of poor performance due to all identified failure modes, but to qualify where they occur within the input space and what inputs are particularly strong contributors to those failure modes.

All examples demonstrated in this paper considered a single performance metric for evaluating system performance. Most, if not all, engineered systems are evaluated on several performance metrics. The gradeability example indicated that weight is not a strong contributor to HMMWV performance in graded environments, and that design effort is better spent on controlling the variance of other design variables. However, the HMMWV is evaluated on many different performance metrics, some of which are heavily influenced by vehicle weight. SREC seeks to support the systems design process, and given that systems design is often a multi-objective problem, SREC and its methods can be applied to individual performance metrics, though it only addresses the effects of variance at a single design point on a single performance metric in question.

Future work includes extending the SREC method to a range of design points, instead of evaluating a single selected design point. Doing so will allow this method to support design optimization methods in which the nominal performance, likelihood of poor performance due to random variations, and region in which poor performance is more likely can be simultaneously optimized.

6. Acknowledgement

DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited. OPSEC10392

This work was supported by the Virtual Prototyping Ground Systems (VIPR-GS) Center at Clemson University and the Automotive Research Center (ARC), a US Army Center of Excellence for modeling and simulation of ground vehicles, under Cooperative Agreement W56HZV-19-2-0001 with the US Army DEVCOM Ground Vehicle Systems Center (GVSC). All opinions, conclusions and findings wherein are those of the authors and may not be those of the affiliated institutions.

7. AUTHOR CONTACT

Gregory Mocko
Associate Professor
Clemson University
Department of Mechanical Engineering
216 S. Palmetto Dr
Clemson, SC
gmocko@clemson.edu

8. REFERENCES

[1] D. Wu, W. Gao, C. Song, and S. Tangaramvong, “Probabilistic interval stability assessment for structures with mixed uncertainty,” *Struct. Saf.*, vol. 58, pp. 105–118, Jan. 2016, doi: 10.1016/j.strusafe.2015.09.003.

[2] X. Du and W. Chen, “Towards a Better Understanding of Modeling Feasibility Robustness in Engineering Design,” *J. Mech. Des.*, vol. 122, pp. 385–394, Dec. 2000, doi: https://doi.org/10.1115/1.1290247.

[3] K. K. Benke, S. Norng, N. J. Robinson, L. R. Benke, and T. J. Peterson, “Error propagation in computer models: analytic approaches, advantages, disadvantages and constraints,” *Stoch. Environ. Res. Risk Assess.*, vol. 32, no. 10, pp. 2971–2985, Oct. 2018, doi: 10.1007/s00477-018-1555-8.

[4] D. Padmanabhan, H. Agarwal, J. E. Renaud, and S. M. Batill, “A study using Monte Carlo Simulation for failure probability calculation in Reliability-Based Optimization,” *Optim. Eng.*, vol. 7, no. 3, pp. 297–316, Sept. 2006, doi: 10.1007/s11081-006-9973-8.

[5] W. Berger, H. Piringer, P. Filzmoser, and E. Gröller, “Uncertainty-Aware Exploration of Continuous Parameter

Spaces Using Multivariate Prediction," *Comput. Graph. Forum*, vol. 30, no. 3, pp. 911–920, June 2011, doi: 10.1111/j.1467-8659.2011.01940.x.

[6] A. Dean, D. Voss, and D. Draguljić, *Design and Analysis of Experiments*. in Springer Texts in Statistics. Cham: Springer International Publishing, 2017. doi: 10.1007/978-3-319-52250-0.

[7] J. K. Telford, "A Brief Introduction to Design of Experiments," *JOhns HOpKins ApL Tech. Dig.*, vol. 27, no. 3, 2007.

[8] G. Mocko, R. Malak, C. Paredis, and R. Peak, "A knowledge repository for behavioral models in engineering design," in *Proceedings of the ASME Design Engineering Technical Conference*, Salt Lake City, Utah: American Society of Mechanical Engineers (ASME), 2004, pp. 943–952. doi: 10.1115/detc2004-57746.

[9] N. Ryan, *Taguchi Methods and QFD: Hows and Whys for Management: A Special Collection of Papers and Essays on Today's Quality Issues and New Quality Technologies*. Dearborn, Michigan: ASI Press, 1988.

[10] R. Splechtna, M. Elshehaly, D. Gračanin, M. Đuras, K. Bühler, and K. Matković, "Interactive interaction plot: Supporting parameter space exploration in a design phase," *Vis. Comput.*, vol. 31, no. 6–8, pp. 1055–1065, June 2015, doi: 10.1007/s00371-015-1095-x.

[11] I. Y. Tumer and R. B. Stone, "Mapping function to failure mode during component development," *Res. Eng. Des.*, vol. 14, no. 1, pp. 25–33, Feb. 2003, doi: 10.1007/s00163-002-0024-y.

[12] "Gradeability and Side Slope Performance," Automotive Instrumentation Division, Aberdeen, TOP 02-2-610, Dec. 2009.

[13] A. Naim, S. Ferguson, and K. Lewis, "Evaluating the Impact of Performance Uncertainty on Vehicle Demand and Design," in *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Victoria, British Columbia, Canada: American Institute of Aeronautics and Astronautics, Sept. 2008. doi: 10.2514/6.2008-6005.

[14] D. W. Apley, J. Liu, and W. Chen, "Understanding the Effects of Model Uncertainty in Robust Design With Computer Experiments," *J. Mech. Des.*, vol. 128, no. 4, pp. 945–958, July 2006, doi: 10.1115/1.2204974.

[15] P. Pandita, I. Bilionis, and J. Panchal, "Bayesian Optimal Design of Experiments For Inferring The Statistical Expectation Of A Black-Box Function," Jan. 15, 2019, *arXiv*: arXiv:1807.09979. doi: 10.48550/arXiv.1807.09979.

[16] N. W. Porter and V. A. Mousseau, "UNDERSTANDING ALEATORY AND EPISTEMIC PARAMETER UNCERTAINTY IN STATISTICAL MODELS," in *Best Estimate Plus Uncertainty International Conference*, Giardini Naxos, Sicily, Italy, pp. 1–10.

[17] J. M. Aughenbaugh and C. J. J. Paredis, "The value of using imprecise probabilities in engineering design," *J. Mech. Des.*, vol. 128, no. 4, pp. 969–979, 2006, doi: 10.1115/1.2204976.

[18] H. B. Liu, C. Jiang, and Z. Xiao, "Efficient uncertainty propagation for parameterized p-box using sparse-decomposition-based polynomial chaos expansion," *Mech. Syst. Signal Process.*, vol. 138, Apr. 2020, doi: 10.1016/j.ymssp.2019.106589.

[19] M. G. R. Faes, M. Daub, S. Marelli, E. Patelli, and M. Beer, "Engineering analysis with probability boxes: A review on computational methods," *Struct. Saf.*, vol. 93, Nov. 2021, doi: 10.1016/j.strusafe.2021.102092.

[20] Y. Huang, J. Sun, and Y. Tian, "A Bayesian Optimization Method for Finding the Worst-Case Scenarios of Autonomous Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 1, pp. 529–543, Jan. 2025, doi: 10.1109/TITS.2024.3490616.

[21] N. Rajesh, E. Van Hassel, and M. Alirezaei, "Unknown-Unsafe Scenario Generation for Verification and Validation of Automated Vehicles," in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, Bilbao, Spain: IEEE, Sept. 2023, pp. 1759–1764. doi: 10.1109/ITSC57777.2023.10421804.

[22] S. Chen, W. Chen, and S. Lee, "Level set based robust shape and topology optimization under random field uncertainties," *Struct. Multidiscip. Optim.*, vol. 41, no. 4, pp. 507–524, Apr. 2010, doi: 10.1007/s00158-009-0449-2.

[23] T. Hasenkamp, M. Arvidsson, and I. Gremyr, "A review of practices for robust design methodology," *J. Eng. Des.*, vol. 20, no. 6, pp. 645–657, Dec. 2009, doi: 10.1080/09544820802275557.

[24] S. Ferguson, A. Brahma, C. Eckert, and O. Isaksson, "Managing and modelling margins in design as a means for confronting the challenges of a disruptive world," *J. Eng. Des.*, vol. 35, no. 10, pp. 1185–1192, Oct. 2024, doi: 10.1080/09544828.2024.2414130.

[25] E. Z. Cansler, S. B. White, S. M. Ferguson, and C. A. Mattson, "Excess Identification and Mapping in Engineered Systems," *J. Mech. Des.*, vol. 138, no. 8, p. 081103, Aug. 2016, doi: 10.1115/1.4033884.

[26] A. Sinha, N. Bera, J. K. Allen, J. H. Panchal, and F. Mistree, "Uncertainty Management in the Design of Multiscale Systems," *J. Mech. Des.*, vol. 135, no. 1, p. 011008, Jan. 2013, doi: 10.1115/1.4006186.

[27] J. H. Panchal, C. J. J. Paredis, J. K. Allen, and F. Mistree, "A value-of-information based approach to simulation model refinement," *Eng. Optim.*, vol. 40, no. 3, pp. 223–251, Mar. 2008, doi: 10.1080/03052150701690764.

[28] M. Beer, S. Ferson, and V. Kreinovich, "Imprecise probabilities in engineering analyses," *Mech. Syst. Signal Process.*, vol. 37, no. 1–2, pp. 4–29, May 2013, doi: 10.1016/j.ymssp.2013.01.024.

[29] E. Taylor, E. Louis, G. Mocko, G. Li, and E. Hybl, "Characterizing Model Fidelity through a Set-Based Approach," in *Proceedings of the ASME 2024 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Washington DC: ASME, pp. 1–10.

[30] D. A. Alvarez, F. Uribe, and J. E. Hurtado, "Estimation of the lower and upper bounds on the probability of failure using subset simulation and random set theory," *Mech. Syst. Signal Process.*, vol. 100, pp. 782–801, Feb. 2018, doi: 10.1016/j.ymssp.2017.07.040.

[31] L. Warren, "Computing with Epistemic Uncertainty," Australian Government Department of Defence, DSTO-TN-1394, 2015.

[32] N. Suh, "Axiomatic Design Theory for Systems," *Res. Eng. Des.*, vol. 10, pp. 189–209, 1998.

[33] D. Long and S. Ferguson, "A Case Study of Evolvability and Excess on the B-52 Stratofortress and F/A-18 Hornet," in *Volume 4: 22nd Design for Manufacturing and the Life Cycle Conference; 11th International Conference on Micro- and Nanosystems*, Cleveland, Ohio, USA: American Society of Mechanical Engineers, Aug. 2017, p. V004T05A026. doi: 10.1115/DETC2017-68287.

[34] Y. Liu *et al.*, "Simulation-based mission mobility reliability analysis of off-road ground vehicles," *J. Mech. Des.*, vol. 143, no. 3, Mar. 2021, doi: 10.1115/1.4048314.

[35] T. P. Bagchi, *Taguchi methods explained: practical steps to robust design*. New Delhi: Prentice-Hall, 1993.