Jie Hu
The State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body,
Hunan University,
Changsha, Hunan 410082, China

Yan Wang
Multiscale Systems Engineering Research Group,
School of Mechanical Engineering,
Georgia Institute of Technology,
Atlanta, GA 30332

Aiguo Cheng
The State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body,
Hunan University,
Changsha, Hunan 410082, China

Zhihua Zhong
The State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body,
Hunan University,
Changsha, Hunan 410082, China

Sensitivity Analysis in Quantified Interval Constraint Satisfaction Problems

Interval is an alternative to probability distribution in quantifying uncertainty for sensitivity analysis (SA) when there is a lack of data to fit a distribution with good confidence. It only requires the information of lower and upper bounds. Analytical relations among design parameters, design variables, and target performances under uncertainty can be modeled as interval-valued constraints. By incorporating logic quantifiers, quantified constraint satisfaction problems (QCSPs) can integrate semantics and engineering intent in mathematical relations for engineering design. In this paper, a global sensitivity analysis (GSA) method is developed for feasible design space searching problems that are formulated as QCSPs, where the effects of value variations and quantifier changes for design parameters on target performances are analyzed based on several proposed metrics, including the indeterminacy of target performances, information gain of parameter variations, and infeasibility of constraints. Three examples are used to demonstrate the proposed approach. [DOI: 10.1115/1.4029513]

1 Introduction

In engineering design, SA methods can be applied in various problems to study the effects of input uncertainty, such as design optimization where searching directions may have different sensitivities, reliability analysis where the most sensitive variables need to be recognized, robust design where robustness is associated with variations of design parameters, and design space searching where influential design parameters with respect to target performances need to be identified. This paper focuses on the feasible design space searching problems, where SA is based on mathematical relations among design parameters, design variables, and target performances. Feasible design space is formed by all acceptable values of design variables that can meet the target performances based on their mathematical relations. Searching feasible design is different from searching optimum design. Feasible design space consists of all design solutions that satisfy design requirements, whereas searching optimum design is to choose the best among the feasible solutions. Searching feasible design is important in real-world engineering practice, because searching a solution that can satisfy all of the complex and multidisciplinary requirements itself is already a daunting task. The main purpose of SA for design optimization is to check the robustness of optimality, whereas SA for feasible design space searching is to identify sensitive parameters or variables that have the most influence on the variations of performances and the size of feasibility space. Engineers can make informed decisions of which parameters or variables to modify in order to find feasible solutions if current requirements are not satisfied, or to improve the existing design.

The uncertainty associated with design parameters or variables is typically quantified by probability distributions. Sampling-based approaches for statistical SA, such as one-factor-at-a-time, scattering, regression analysis, and variance-based methods, have been developed. Based on different variation ranges of design parameters, SA methods can be classified as either local or global. Local SA methods, such as the derivative based approach [1–4], response surface modeling [5], design of experiments [6], fractional factorial design [7], and elementary effect method [8], examine the impacts on target performances from small variations of design parameters. Local SA is general and easy to implement. Yet some issues associated with the derivative based approach include how to differentiate small variations from large ones, and the cost of computing derivatives. Constructing response surfaces by design of experiments also becomes expensive for high-dimensional problems. GSA [9,10] considers the entire possible ranges of parameter variations and can be conducted in several ways, such as scatter plots [11,12], screening [13], regression analysis [14], variance-based methods [15–18], probabilistic SA [4,19], and local variation approximation [4,20]. Most of them are statistical methods, which use probabilistic information of design variables and design parameters. Statistical GSA methods provide a comprehensive view of sensitivity. The main shortcoming is that the essential information of probabilistic distributions may not be easily obtainable in practice. When there is a lack of data and data fitting is not reliable, assumptions of distributions are needed, including the types of distributions as well as the lower and upper bounds of distributions. In addition, Monte Carlo sampling involved in some of the statistical methods implies high computational costs associated with functional or model evaluations ($\gg$1000) for an exhaustive search, especially for nonlinear systems.

The alternative approach of quantifying uncertainty is using intervals, in which no assumption of distributions is made. Only the lower and upper bounds of variation ranges are needed, and no sampling is required. In contrast to statistical SA, few research efforts have been taken for the sensitivity of interval-valued models. Early research focused on applying interval analysis to rigorously bound the sensitivity estimate of real-valued models, where the worst-case and best-case estimations of Jacobians for nonlinear equations are computed [21–24]. More recently, a local SA method was proposed to study the impacts of the width and midpoint value changes for interval variables [25]. The maximum impact is searched by varying the design parameter values with small steps. This method however did not address the interaction between design parameters, and only the main effect was analyzed. A hybrid approach [26,27] was developed to analyze the

Contributed by the Design for Manufacturing Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received October 9, 2013; final manuscript received December 17, 2014; published online February 6, 2015. Assoc. Editor: Rikard Söderberg.

sensitivity of parameterized interval variables via multi-objective optimization problems, where the variation of an output with a given interval input was estimated by optimization. The searching procedure for optimization can be computationally expensive for highly nonlinear and coupled systems.

In this paper, a different approach for SA is taken for feasible design space searching problems. In engineering design, numerous actions are performed and decisions are made to ensure that the final design meets various requirements and specifications, such as functionalities, physical properties of materials, correlated behaviors of subsystems, manufacturing capabilities, financial budgets, and many others. The mathematical relations among design variables, design parameters, and target performances can be generally regarded as constraints. Design parameters are known but varying within some ranges because of noises or other sources of uncertainty. Design variables are unknown, but their values are typically between some lower and upper bounds. Target performances are expected performances the design can achieve. An important task of design process is searching the feasible design alternatives or solutions in the allowable design space that satisfy the constraints. SA in design space searching problems is to check how the variations of design parameters and variables within the permissible ranges, characterized as intervals in our approach, affect target performances. Design parameters are not precisely known, and their variations are bounded by intervals. As a result of the mathematical relations, target performances are also uncertain. The goal of SA is to estimate the expected variation ranges of target performances.

Searching in the feasible design space for all possible values of design variables that satisfy the design constraints can be generally formulated as solving constraint satisfaction problems (CSPs). A CSP can also be extended to a QCSP which allows for universal ($\forall$) and existential ($\exists$) quantifiers associated with variables [28,29]. The variables with their quantifiers generate the first-order logic interpretation [28] of the mathematical relations. QCSP is a generalization of CSP, and CSP is a special case of QCSP where all variables are associated with $\exists$. In the QCSP formulation of feasible design problems, design intent of controllability, materials properties, process sequences, and others can be captured by assigning appropriate quantifiers to variables so that logic interpretations of quantified constraints become available [30]. The interval-based SA approach can also be applied to the compromise programming (CP) formulation [31] in set-based design [32,33], as well as its extensions (e.g., Refs. [34] and [35]). The set-based design method searches and constructs the feasible design space defined by constraints, which is equivalent to solving CSPs. The interval-based SA method can be used in solving CP problems, such as studying the importance of imprecise parameters with respect to the final solution, and fine-tuning the subjective weights associated with criteria based on their significance toward the objective function.

Here, we propose a global SA method for quantified constraints in feasible design space searching problems. Note that solving CSPs or QCSPs is to find feasible values of design variables that can achieve the specified target performances, whereas conducting SA is to tell which parameters are more influential to target performances than the other. In this paper, a GSA approach is proposed to analyze the sensitivity of design parameters with respect to target performances in the feasible design space searching problems that are formulated as QCSPs. Again, the QCSP formulation is general enough for feasible design space searching problems, because QCSP is a generalization of CSP and CSP is a generic formulation for feasible design space searching. In the proposed GSA approach, interval ranges of target performances are efficiently estimated by applying Kaucher interval arithmetic. The sensitivity is measured by two metrics. The first metric, *indeterminacy*, is a generalization of *Hartley like measure* [36] and is defined to measure the change of information about a target performance as a design parameter varies. The second metric, *infeasibility*, qualitatively measures whether a quantified interval constraint is satisfiable or not. Instead of the costly statistical sampling, three representative values for each design parameter with interval uncertainty are used to calculate the metrics to generate *sensitivity zones*, which are used to provide the ranking of sensitivity among design parameters.

Different from other interval based SA approaches [25–27], our approach considers both the main and interaction effects for feasible design problems with continuous design parameters. Our SA approach for interval-valued variables is global. No assumption of probability distributions is needed, and no time-consuming sampling is required. Interval-valued constraints are considered directly in contrast to Refs. [21–24]. In this paper, the proposed approach is illustrated by the examples with systems of equations. But it is general enough to be extended to inequality. The novelty of the proposed GSA approach is our unique way to assess the individual and joint effects of interval uncertainty in CSPs. Additionally, the sensitivities of logic quantifiers associated with design parameters are also evaluated in QCSPs. The sensitivities of both quantitative values and qualitative semantics of design parameters are analyzed with the representation of generalized interval. With this information, the design parameters can be adjusted for feasible solutions by considering design intent in applications and practice.

In the remainder of the paper, the background of CSP and QCSP is introduced in Sec. 2. The proposed GSA approach and sensitivity metrics are described in Sec. 3. Three examples are presented to demonstrate the proposed method in Sec. 4.

2 CSP and QCSP

A CSP is a system of constraints where the variables are within certain domains. Formally, a CSP is constructed by a set of constraints $C(x) = \{C_1(x), \ldots, C_m(x)\}$, which denotes mathematical relations among a set of variables $V = \{x_1, \ldots, x_n\}$. Each variable x_i has an associated domain D_i. The complete searching space for the CSP is a Cartesian product $D = D_1 \times D_2 \times \cdots \times D_n$. CSP is related to but different from optimization. The standard form of an optimization problem is typically defined as the minimization of an objective function $f(x)$ subject to inequality constraints $g_i(x) \le 0$ $(i = 1,\ldots,m)$ and equality constraints $h_j(x) = 0$ $(j = 1,\ldots,p)$. The feasible domain of x in optimization is defined by constraints g_i's and h_j's. Solving the optimization problem is to find the best solution among the feasible ones within the feasible domain defined by the constraints. In contrast, solving a CSP is to find *all* of the feasible solutions that satisfy the constraints. A constraint $C_i(x)$ in a CSP can be either $g_i(x) \le 0$ or $h_j(x) = 0$. In this paper, we only consider equality constraints. An inequality constraint can be easily converted to an equality one by introducing slack interval variables. In the context of design, the equality constraint $h(x) = 0$ is transformed to an equivalent form $F(a, x) = b$, where a is the design parameter, b is the target performance, and x is the design variable. Given some known values of a and b, solving the CSP is to find the possible values of x such that $F(a, x) = b$. The solution of a CSP in design thus is the feasible design space in which the values of design variables satisfy all design constraints.

The CSP formulation has been applied in floor plan design [37,38], geometric modeling [39], conceptual design [42,43], embodiment design [42], collaborative design [43–45], design space searching, and others.

The CSP formulation can only express limited semantics. All of the variables in CSPs are existentially quantified ($\exists$) in determining whether the statement is true in the sense of first-order logic. In contrast, a QCSP allows universal and existential quantifiers to be associated with variables [28,29]. A QCSP is a general problem with its solution satisfying all constraints in the form of both mathematical and logic expressions. In engineering design, QCSP can be used to integrate design intent of engineers into calculations by associating quantifiers with variables. Those variables that are not controllable by the designer can be associated with universal quantifier $\forall$. They usually correspond to the external disturbance or bias of a system. The variables that can be controlled and modified within some prescribed ranges by the designer are associated with existential quantifier $\exists$. They are controllable and adjustable internally. For a universally quantified variable, all values in its domain must satisfy the constraints. For an existentially quantified variable, at least one value in its domain satisfies the constraints. By incorporating quantifiers, the QCSP formulation can capture design intent as well as material properties, process sequences, and other semantics. With the advantage of quantified variables, QCSPs have recently been applied in mechanical design [46–50], control [51–55], scheduling [56,57], and planning [58,59]. Notice that CSP can be regarded a special case of QCSP, in which all variables are existentially quantified. Here, we use generalized interval [60–62], an algebraic and semantic extension of the classical interval [63], to incorporate quantifiers in QCSP formulation. More information of how to formulate QCSPs from a given design problem and how to solving QCSPs can be found in Ref. [30].

Instead of solving QCSPs to find feasible values of design variable x, in this paper, we use the QCSP formulation to study the sensitivity of design parameters. The variational ranges of design parameter a and design variable x are represented as intervals $\mathbf{a}$ and $\mathbf{x}$, respectively. The impact on the target performance b thus is estimated as an interval $\mathbf{b}$, which is a straightforward evaluation of constraint $F(\mathbf{a},\mathbf{x}) = \mathbf{b}$ based on interval arithmetic.

The SA with the QCSP formulation can provide the information of which design parameter is the most influential one that affects the target performance and which parameters are more robust than the other. For over-constrained problems in searching feasible design, SA also provides the information of how to modify design requirements to find feasible solutions. For instance, when constraints are overly restrictive on interval value ranges, or when all variables in a constraint are universally quantified and interpretation is not possible (i.e., over-constrained in logic), we need to know which design parameter to adjust in order to receive feasible solutions. SA is necessary and useful to gain such information.

3 The Proposed GSA Approach

The main idea of the proposed GSA approach is summarized as follows. The sensitivity of a design parameter with respect to each constraint is estimated both quantitatively and qualitatively. The new metric, *indeterminacy*, is a generalization of Hartley like measure [36] that considers both proper and improper intervals. Three *representative values*, lower bound, midpoint, and upper bound, of an interval-valued design parameter are chosen as the references. In engineering applications, these three values are typically the most important ones for analysis. When the value of a design parameter changes, it is called a *variation*. The difference between the original indeterminacy of a target performance and the new one after a variation quantifies *information gain*. In order to differentiate the impact of each design parameter on a target performance, the GSA is implemented in a framework of making the variations of design parameters one at a time. The interaction of two design parameters is estimated by varying the two design parameters simultaneously. The sensitivities of the design parameters are ranked based on some *sensitivity zones*, which are generated by computing the total information gains for the three different representative values of a design parameter. When two design parameters have the same *total information gain* which include information gains of individual parameters and their interactions, qualitative metrics of *infeasibility* are compared.

Figure 1 provides an overview of the proposed concepts in our GSA method, in which the relationships among the concepts and the calculation sequences are indicated as arrows with solid lines. When total information gain is not computable, the additional relationship between total information gain and quantifier mutation gain is used and indicated by the arrow with a dashed line. The directions of the arrows show the sequence of calculation. For each of design parameter, the quantities enclosed in the box with the dotted line in Fig. 1 are calculated three times for the three representative values to construct sensitivity zones. The final output, sensitivity ranking, is generated based on the total infeasibility and sensitivity zones. As an extension of our previous work [64,65], here high-order interaction is introduced in calculating the total information gain. In addition, rules of suggested ranking are provided when sensitivity zones overlap and there is a lack of further information to fully decide the ranking. The details of the proposed GSA method are described in Subsections 3.1–3.3.

3.1 Basic Definitions. The proposed SA of constraints is based on the variation of interval ranges. A generalized interval $\mathbf{x} = [\underline{x}, \bar{x}] \in \mathbb{KR}$ is called *proper* when $\underline{x} \le \bar{x}$ and denoted as $\mathbf{x} \in \mathbb{IR}$. $\mathbf{x}$ is called *improper* when $\underline{x} \ge \bar{x}$ and denoted as $\mathbf{x} \in \overline{\mathbb{IR}}$. Proper or improper is called the *modality* of an interval. The *width*

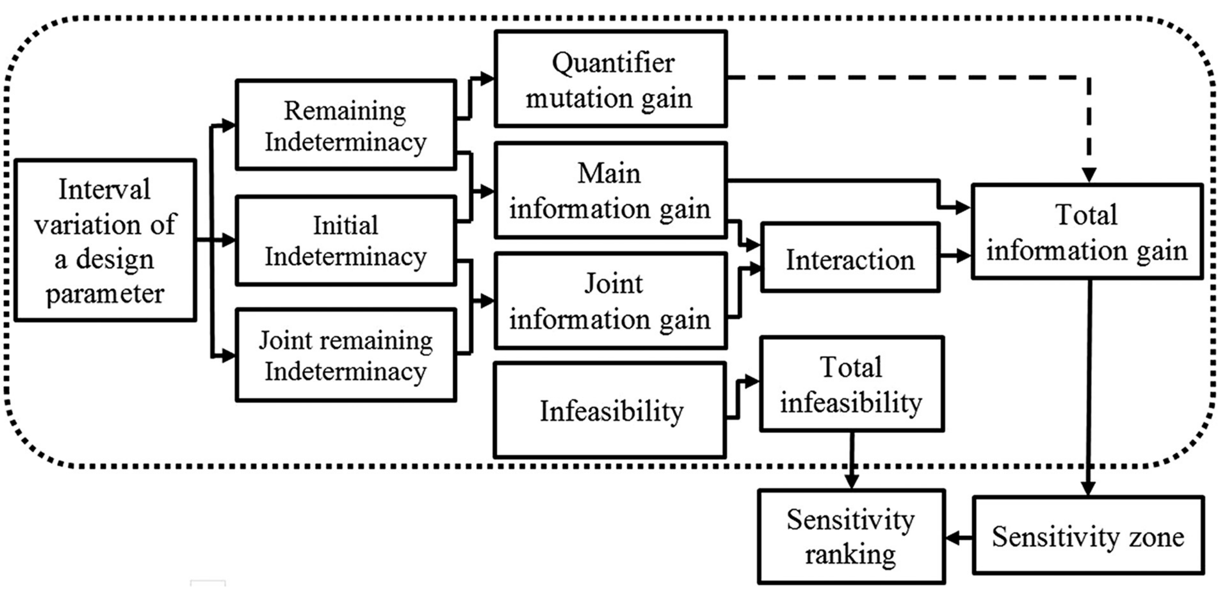

Fig. 1 The relations among the concepts developed in the proposed method

of $\mathbf{x}$ is defined by $\mathrm{wid}(\mathbf{x}) := |\bar{x} - \underline{x}|$. The center is found by $\mathrm{mid}(\mathbf{x}) := (\bar{x} + \underline{x})/2$. The radius is defined by $\mathrm{rad}(\mathbf{x}) := (\bar{x} - \underline{x})/2$, which is positive when $\mathbf{x}$ is proper and negative when it is improper. Functions $\inf([\underline{x}, \bar{x}]) = \underline{x}$ and $\sup([\underline{x}, \bar{x}]) = \bar{x}$ return the lower and upper bounds of $\mathbf{x}$, respectively. The relationship between proper and improper intervals is established by an operator *dual*, defined as $\mathrm{dual}([\underline{x}, \bar{x}]) := [\bar{x}, \underline{x}]$. Furthermore, $\mathrm{pro}([\underline{x}, \bar{x}]) := [\min(\underline{x}, \bar{x}), \max(\underline{x}, \bar{x})]$ and $\mathrm{imp}([\underline{x}, \bar{x}]) := [\max(\underline{x}, \bar{x}), \min(\underline{x}, \bar{x})]$. Intersection is defined as $[\underline{x}, \bar{x}] \cap [\underline{y}, \bar{y}] = [\max(\underline{x}, \underline{y}), \min(\bar{x}, \bar{y})]$.

The *variation* of $\mathbf{x}$, denoted by $v_{\mathbf{x}}$, is defined as a value change from $\mathbf{x} = [\underline{x}, \bar{x}]$ to $\mathbf{x}' = [\underline{x}', \bar{x}']$ with $\underline{x}' = \underline{x} + \delta_1$ and $\bar{x}' = \bar{x} - \delta_2$ where $0 < \delta_1 < 2\mathrm{rad}(\mathbf{x})$ and $0 < \delta_2 < 2\mathrm{rad}(\mathbf{x})$ if $x \in \mathbb{IR}$, and $2\mathrm{rad}(\mathbf{x}) < \delta_1 < 0$ and $2\mathrm{rad}(\mathbf{x}) < \delta_2 < 0$ if $x \in \overline{\mathbb{IR}}$. When $\mathbf{x}'$ has the same modality as $\mathbf{x}$, the variation is called *local*. When the modality is changed, the variation is called *global*. For instance, if interval [1,2] is changed to [1.1,1.9], the variation is local. If it is changed to [1.5,1.5] or [2,1], the variation is global.

Note that the term, global, used in traditional SA means target performance variance is evaluated within the entire value ranges of all design parameters. Here, global indicates that the variation causes not only the value change within the entire design parameter range but also the interval modality change, which has an even more significant effect than the value change only.

Consider a constraint system with m constraints and l variables. The first GSA metric is the *indeterminacy* measure, denoted as $\mathfrak{M}(\bullet)$. It quantifies the uncertainty associated with a generalized interval. The indeterminacy measure for a generalized interval $\mathbf{x} \in \mathbb{KR}$ is defined as

$$\mathfrak{M}(\mathbf{x}) = \begin{cases} \log_2(1 + \mathrm{wid}(\mathbf{x})), & \text{if } \mathbf{x} \in \mathbb{IR} \\ -\log_2(1 + \mathrm{wid}(\mathbf{x})), & \text{if } \mathbf{x} \in \overline{\mathbb{IR}} \end{cases} \tag{3.1}$$

A negative $\mathfrak{M}(\bullet)$ is associated with an improper interval. Therefore, the indeterminacies for two intervals that have equal widths but different quantifiers can be differentiated.

The indeterminacy of a generalized interval vector $\mathbf{x} \in \mathbb{KR}^l$ is defined as a vector $\mathfrak{M}(\mathbf{x}) = [\mathfrak{M}(\mathbf{x}_1), \ldots, \mathfrak{M}(\mathbf{x}_l)]$, which measures the indeterminacy of each element in $\mathbf{x}$ separately.

The second metric is *infeasibility*, defined as

$$u_j(\mathbf{a}, \mathbf{x}) = \begin{cases} 1, & \text{if } c_j \text{ is unsatisfiable} \\ 0, & \text{otherwise} \end{cases}$$

which indicates whether the jth constraint c_j can be satisfied or not with the values of variables $\mathbf{a}$ and $\mathbf{x}$. The *total infeasibility* of the constraint system with m constraints is defined as

$$U(\mathbf{a}, \mathbf{x}) = \sum_{j=1}^{m} u_j(\mathbf{a}, \mathbf{x})$$

3.2 Sensitivity Quantified by Indeterminacy of Target Performance. Given a constraint system $F(\mathbf{a}, \mathbf{x}) = \mathbf{b}$, with $F : \mathbb{KR}^n \times \mathbb{KR}^l \to \mathbb{KR}^m$ in which $\mathbf{x} \in \mathbb{KR}^l$ is the design variable, $\mathbf{a} \in \mathbb{KR}^n$ is the design parameter, and $\mathbf{b} \in \mathbb{KR}^m$ is the target performance, the indeterminacy $\mathfrak{M}(\mathbf{b})$ of $\mathbf{b}$ is called *initial indeterminacy* when design parameter $\mathbf{a}$ takes its initially given interval range. The indeterminacy $\mathfrak{M}(\mathbf{b}|a_i)$ of $\mathbf{b}$ is called *remaining indeterminacy*, where $\mathbf{b} = \boldsymbol{F}(\mathbf{a}_{\sim i}, a_i, \mathbf{x})$ is estimated by choosing a_i as a representative value, i.e., lower bound $\inf(\mathbf{a}_i)$, midpoint $\mathrm{mid}(\mathbf{a}_i)$, or upper bound $\sup(\mathbf{a}_i)$, while other design parameters remain as the original interval values. Here, $\sim i$ denotes the rest of elements in a vector except the ith one.

Similarly, $\mathfrak{M}(\mathbf{b}|a_i, a_k)$ is called *joint remaining indeterminacy*, where $\mathbf{b} = \boldsymbol{F}(\mathbf{a}_{\sim i \sim k}, a_i, a_k, \mathbf{x})$ is estimated by choosing a_i and a_k as the representative values of $\mathbf{a}_i$ and $\mathbf{a}_k$, respectively, with the same correspondence of lower bound, midpoint, or upper bound.

The *main information gain* by knowing $\mathbf{a}_i$ with certainty with respect to the jth constraint is quantified as

$$I_j^m(a_i) = \frac{\mathfrak{M}(\mathbf{b}_j) - \mathfrak{M}(\mathbf{b}_j|a_i)}{\mathfrak{M}(\mathbf{b}_j)} \tag{3.2}$$

For example, in constraint $\mathbf{b} = \mathbf{a}_1\mathbf{x} + \mathbf{a}_2\mathbf{x}^2$, $\mathbf{a}_1 = [1,2]$, $\mathbf{a}_2 = [6,4]$, $\mathbf{x} = [-1,3]$, and target performance $\mathbf{b} = [-2,42]$ are all generalized intervals. When $a_1 = \mathrm{mid}(\mathbf{a}_1)$, the target performance is updated to $\mathbf{b} = [-1.5,40.5]$. The main information gain by knowing $\mathbf{a}_1$ is $I^m(a_1) = (\mathfrak{M}(\mathbf{b}) - \mathfrak{M}(\mathbf{b}|a_1))/\mathfrak{M}(\mathbf{b}) = (5.49 - 5.42)/5.49 = 0.01$. It means that the indeterminacy of target performance $\mathbf{b}$ in the constraint is reduced by 0.01 when $\mathbf{a}_1$ becomes certain as a_1.

We say $I_j^m(\bullet)$ is computable if

$$\mathfrak{M}(\mathbf{b}_j) \times \mathfrak{M}(\mathbf{b}_j|a_i) \geq 0 \tag{3.3}$$

and

$$\mathfrak{M}(\mathbf{b}_j) \neq 0 \tag{3.4}$$

The two computable conditions in Eqs. (3.3) and (3.4) ensure that the definition of $I_j^m(a_i)$ in Eq. (3.2) holds. The computable condition in Eq. (3.3) requires that $\mathfrak{M}(\mathbf{b}_j)$ and $\mathfrak{M}(\mathbf{b}_j|a_i)$ have the same sign, which indicates that only the numerical value of indeterminacy is changed by knowing a_i. $I_j^m(a_i)$ reveals the numerical change of indeterminacy of the jth target performance when $\mathbf{a}_i$ is changed to a_i.

When $\mathfrak{M}(\mathbf{b}_j) \times \mathfrak{M}(\mathbf{b}_j|a_i) < 0$, the quantifier of the jth target performance is also changed besides its value change for the variation from $\mathbf{a}_i$ to a_i. In this scenario, the quantifier change is measured by *quantifier mutation gain*, defined as

$$Q_j(a_i) = \mathfrak{M}(\mathbf{b}_j|a_i)/\mathfrak{M}(\mathbf{a}_i) \tag{3.5}$$

where $\mathbf{b}_j = \boldsymbol{F}_j(\mathbf{a}_{\sim i}, a_i, \mathbf{x})$, $i = 1, \ldots, n$, and $j = 1, \ldots, m$.

The computable condition in Eq. (3.4) requires that the denominator $\mathfrak{M}(\mathbf{b}_j)$ should be nonzero. When $\mathfrak{M}(\mathbf{b}_j) = 0$, the target performance b_j is a real number with zero interval width. It implies that the interval uncertainties associated with the strongly correlated design parameters are canceled by each other and the target performance becomes a precisely known value. Any change of the design parameters may introduce uncertainty back into the target performance. Since the pointwise interval can be treated as either proper or improper, the quantifier can be seen as either changed or not when an interval width is zero. In this paper, we treat it as a change so that the indeterminacy of target performance in this scenario can also be quantified by Eq. (3.5).

The indeterminacy of a target performance has two levels. One is a numerical value change with the same quantifier, and the other is a quantifier change. With the same amount of variation for design parameters, a target performance with a quantifier change is seen as more sensitive than the one with only numerical value changes.

The *joint information gain* $I_j^{jt}(a_i, a_k)$ quantifies the uncertainty reduction with respect to the jth constraint by simultaneously knowing two design parameters $\mathbf{a}_i$ and $\mathbf{a}_k$ with certainty. It is calculated by

$$I_j^{jt}(a_i, a_k) = \frac{\mathfrak{M}(\mathbf{b}_j) - \mathfrak{M}(\mathbf{b}_j|a_i, a_k)}{\mathfrak{M}(\mathbf{b}_j)} \tag{3.6}$$

under the computable conditions $\mathfrak{M}(\mathbf{b}_j) \times \mathfrak{M}(\mathbf{b}_j|a_i, a_k) \geqslant 0$ and $\mathfrak{M}(\mathbf{b}_j) \neq 0$, where $i = 1, \ldots, n$ and $j = 1, \ldots, m$. In the previous example, $\mathbf{b} = [-1.5,49.5]$ when $a_1 = \mathrm{mid}(\mathbf{a}_1)$ and $a_2 = \mathrm{mid}(\mathbf{a}_2)$. Then $I^{jt}(a_1, a_2) = (\mathfrak{M}(\mathbf{b}) - \mathfrak{M}(\mathbf{b}|a_1, a_2))/\mathfrak{M}(\mathbf{b}) = (5.49 - 5.7)/5.49 = -0.03$. The negative value means that the indeterminacy of target performance $\mathbf{b}$ in the constraint is increased by 0.03 when $\mathbf{a}_1$ and $\mathbf{a}_2$ become

a_1 and a_2 simultaneously. Because $\mathbf{a}_2$ is an improper interval in the constraint, the reduction of uncertainty for design parameter $\mathbf{a}_2$ actually increases the level of uncertainty for target performance $\mathbf{b}$. Improper intervals play the role of controllable parameters and can reduce the variations of target performances. When controllable parameters are fixed, uncontrollable parameters become more influential on the target performance.

When $I_j^{jt}(a_i,a_k)$, $I_j^m(a_i)$, and $I_j^m(a_k)$ are all computable, the difference between $I_j^{jt}(a_i,a_k)$ and the sum of $I_j^m(a_i)$ and $I_j^m(a_k)$ is the extra information gained by the interaction between $\mathbf{a}_i$ and $\mathbf{a}_k$ when they become certain simultaneously. Assuming that a_i and a_k are independent to each other, the interaction between $\mathbf{a}_i$ and $\mathbf{a}_k$, as an indicator of the strength of correlation between the two, is quantified by

$$I_j^{\mathrm{in}}(a_i, a_k) = I_j^{jt}(a_i, a_k) - I_j^m(a_i) - I_j^m(a_k) - \alpha\varepsilon \quad (i \neq k) \tag{3.7}$$

where $\alpha\varepsilon$ is a compensation term. The compensation term is introduced because there could be a numerical difference in calculating the linear relationship used in Eq. (3.7), whereas indeterminacy is defined with the logarithm function. In other words, $\alpha\varepsilon$ is introduced such that $I_j^{\mathrm{in}}(a_i,a_k)$ becomes zero when the linear combination of the effects of individual design parameters $I_j^m(a_i)$ and $I_j^m(a_k)$ is comparable to $I_j^{jt}(a_i,a_k)$ if there is no interaction between the two design parameters. Indicator α is defined as 1 if $2^{\mathfrak{M}(\mathbf{b}_j)} + 2^{\mathfrak{M}(\mathbf{b}_j|a_i,a_k)} = 2^{\mathfrak{M}(\mathbf{b}_j|a_i)} + 2^{\mathfrak{M}(\mathbf{b}_j|a_k)}$. Otherwise, $\alpha = 0$. ε is defined as $\varepsilon = (\mathfrak{M}(\mathbf{b}_j|a_i) + \mathfrak{M}(\mathbf{b}_j|a_k) - \mathfrak{M}(\mathbf{b}_j) - \mathfrak{M}(\mathbf{b}_j|a_i,a_k))/\mathfrak{M}(\mathbf{b}_j)$.

Higher-order interaction among multiple design parameters can be evaluated in a similar way as in Eq. (3.7), $I_j^{\mathrm{in}}(a_i,a_k,a_l,\ldots) = I_j^{jt}(a_i,a_k,a_l,\ldots) - I_j^m(a_i) - I_j^m(a_k) - I_j^m(a_l) - \cdots - \alpha'\varepsilon'$, in which ε' is defined as $(\mathfrak{M}(\mathbf{b}_j|a_i) + \mathfrak{M}(\mathbf{b}_j|a_k) + \mathfrak{M}(\mathbf{b}_j|a_l) + \cdots - \mathfrak{M}(\mathbf{b}_j) - \mathfrak{M}(\mathbf{b}_j|a_i,a_k,a_l,\ldots))/\mathfrak{M}(\mathbf{b}_j)$. Similarly, $\alpha' = 1$ if $2^{\mathfrak{M}(\mathbf{b}_j)} + 2^{\mathfrak{M}(\mathbf{b}_j|a_i,a_k,a_l,\cdots)} = 2^{\mathfrak{M}(\mathbf{b}_j|a_i)} + 2^{\mathfrak{M}(\mathbf{b}_j|a_k)} + 2^{\mathfrak{M}(\mathbf{b}_j|a_l)} + \cdots$. Otherwise, $\alpha' = 0$.

The *total information gain* $I_j(a_i)$ with respect to the jth target performance by knowing $\mathbf{a}_i$ with certainty is defined as

$$I_j(a_i) = I_j^m(a_i) + \sum_{k \neq i} I_j^{in}(a_i, a_k) + \sum_{k,l \neq i} I_j^{in}(a_i, a_k, a_l) + \sum_{k,l,\cdots \neq i} I_j^{in}(a_i, a_k, a_l, \ldots) + \cdots \tag{3.8}$$

in which high order interaction terms are applied if the fine granularity is desirable.

Note that the total information gain in Eq. (3.8), the main information gain in Eq. (3.2), and the extra information gain by interaction in Eq. (3.7) correspond to the total, main, and interaction effects respectively in the statistical SA.

With the above definitions of information gains and quantifier change, we can now perform the SA. A higher total information gain indicates a higher sensitivity. We make the variation of design parameters one at a time by fixing its value and leaving the other as intervals. The lowest or highest values of the target performance typically occur when a design parameter has the largest variation if the design parameter and target performance are either positively or negatively correlated. Therefore, the lower and upper bounds are chosen as the representative values. The midpoint is chosen to exam the case when the design parameter and target performance are not linearly correlated. Thus, we use a *sensitivity zone* of $I_j(a_i)$ constructed from the three representative values to estimate the possible sensitivities of $\mathbf{a}_i$ with respect to the jth target performance. The lower and upper bounds of the sensitivity zone are computed from the minimum and maximum among the three values in $I_j(a_i) = \{I_j(a_i = \inf(\mathbf{a}_i)), I_j(a_i = \mathrm{mid}(\mathbf{a}_i)), I_j(a_i = \sup(\mathbf{a}_i))\}$. When $I_j(a_i)$ is not computable, the sensitivity zone is computed from the minimum and maximum values in $\mathcal{Q}_j(a_i) = \{Q_j(\inf(\mathbf{a}_i)), Q_j(\mathrm{mid}(\mathbf{a}_i)), Q_j(\sup(\mathbf{a}_i))\}$.

Here, the concept of sensitivity zone is developed to rank the sensitivities of design parameters. The sensitivity zone is defined to include the possible impact on the target performance as much as possible. The ranking based on the best-case and worst-case scenarios will be inherently robust. The introduction of sensitivity zone is intended to provide such scenarios. The rankings with sensitivity zones provide engineers more information to make decisions of how to adjust design parameters. In addition, when a design parameter is shared by several constraints, it may have opposite effects on those constraints. With the interaction effect included in the proposed approach, engineers can find tradeoffs among different target performances when adjusting design parameters.

Table 1 lists the sensitivity comparison rules for ranking, where $S_j(a_i)$ denotes the sensitivity of the ith design parameter with respect to the jth target performance. The lower and upper bounds of the sensitivity zone are defined as

$$\mathrm{mig}(\mathcal{I}_j(a_i)) = \min\{|I_j(\inf(\mathbf{a}_i))|, |I_j(\mathrm{mid}(\mathbf{a}_i))|, |I_j(\sup(\mathbf{a}_i))|\} \tag{3.9}$$

and

$$\mathrm{mag}(\mathcal{I}_j(a_i)) = \max\{|I_j(\inf(\mathbf{a}_i))|, |I_j(\mathrm{mid}(\mathbf{a}_i))|, |I_j(\sup(\mathbf{a}_i))|\} \tag{3.10}$$

respectively.

Quantified interval constraints have logic interpretations embedded in the mathematical expression. Therefore, the impact of design parameter variation includes not only the change of indeterminacy for the target performance but also the feasibility of the constraint. The feasibility of a quantified constraint can be verified by checking if the interval intersection between an initially given target performance $\mathbf{b}_j^0$ and the one after the variation $\mathbf{b}_j = \boldsymbol{F}_j(\mathbf{a}_{\sim i}, a_i, \mathbf{x})$, calculated as $\mathbf{b}_j^{\cap} = \mathrm{pro}(\boldsymbol{F}_j(\mathbf{a}_{\sim i}, a_i, \mathbf{x})) \cap \mathrm{pro}(\mathbf{b}_j^0)$, is empty. If the intersection is not empty, then the interpretation exists and constraint is satisfiable or feasible. Otherwise, the constraint is infeasible.

The indeterminacy of intersection $\mathfrak{M}(\mathbf{b}_j^{\cap}|a_i)$ can be calculated, where $\mathbf{b}_j^{\cap} = \mathrm{pro}(\boldsymbol{F}_j(\mathbf{a},\mathbf{x})) \cap \mathrm{pro}(\mathbf{b}_j^0)$ is estimated. If $\mathfrak{M}(\mathbf{b}_j^{\cap}|a_i) < 0$, the jth quantified interval constraint is infeasible. That is, the infeasibility of jth constraint after variation is

$$u_j(\mathbf{a}_{\sim i}, a_i, \mathbf{x}) = \begin{cases} 1 & \text{if } \mathfrak{M}(\mathbf{b}_j^{\cap}|a_i) < 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.11}$$

Similarly, the infeasibility $u_j(\mathbf{a}, \mathbf{x})$ with the original values of variables $\mathbf{a}$ and $\mathbf{x}$ can be estimated by Eq. (3.11).

The *infeasibility change* of the jth constraint with the variation $\mathbf{a}_i \mapsto a_i$ is defined as

$$\Delta u_j(\mathbf{a}_i \mapsto a_i) = u_j(\mathbf{a}_{\sim i}, a_i, \mathbf{x}) - u_j(\mathbf{a}, \mathbf{x}) \tag{3.12}$$

Table 1 Sensitivity comparison rules

$I_j(a_i)$	$I_j(a_k)$	Rules
Comp	Comp	$S_j(a_i) \geqslant S_j(a_k)$, if $\mathrm{mig}(\mathcal{I}_j(a_i)) \geqslant \mathrm{mag}(\mathcal{I}_j(a_k))$; $S_j(a_i) < S_j(a_k)$, if $\mathrm{mag}(\mathcal{I}_j(a_i)) < \mathrm{mig}(\mathcal{I}_j(a_k))$; cannot be decided, otherwise
Comp	Incomp	$S_j(a_i) < S_j(a_k)$
Incomp	Comp	$S_j(a_i) > S_j(a_k)$
Incomp	Incomp	$S_j(a_i) \geqslant S_j(a_k)$, if $\mathrm{mig}(\mathcal{Q}_j(a_i)) \geqslant \mathrm{mag}(\mathcal{Q}_j(a_k))$; $S_j(a_i) < S_j(a_k)$, if $\mathrm{mag}(\mathcal{Q}_j(a_i)) < \mathrm{mig}(\mathcal{Q}_j(a_k))$; cannot be decided, otherwise

Note: Comp, computable and Incomp, not computable.

For example, if we have three constraints and $u(\mathbf{a}, \mathbf{x}) = (1, 0, 1)$, the first and third constraints are infeasible. With a variation of design parameter $\mathbf{a}_i$, $u(\mathbf{a}_{\sim i}, a_i, \mathbf{x}) = (1,1,1)$. We will know that the second constraint becomes infeasible too.

The *total infeasibility change* with the variation $\mathbf{a}_i \mapsto a_i$, denoted by $\Delta U(\mathbf{a}_i \mapsto a_i)$, is computed as

$$\Delta U(\mathbf{a}_i \mapsto a_i) = \sum_{j=1}^{m} \Delta u_j(\mathbf{a}_i \mapsto a_i) \tag{3.13}$$

When $\Delta U(\mathbf{a}_i \mapsto a_i) > 0$, the variation $\mathbf{a}_i \mapsto a_i$ introduces infeasibility to the problem. A design parameter with $\Delta U(\bullet) > 0$ has more impact than the one with $\Delta U(\bullet) = 0$. For example, if $u(\mathbf{a}, \mathbf{x}) = (1, 0, 1)$, $u(\mathbf{a}_{\sim i}, a_i, \mathbf{x}) = (1,1,1)$ and $u(\mathbf{a}_{\sim k}, a_k, \mathbf{x}) = (1,0,1)$, we have $\Delta U(\mathbf{a}_i \mapsto a_i) > 0$ and $\Delta U(\mathbf{a}_k \mapsto a_k) = 0$. Then $\mathbf{a}_i$ has more impact than $\mathbf{a}_k$. Because a_i can be either one of the three representative values, three values of $\Delta\mathcal{U}(\bullet)$ need to be obtained in the comparison, similar to $\mathcal{I}_j(\bullet_i)$ and $\mathcal{Q}_j(\bullet)$. The lower and upper bounds of the total feasibility change are obtained as

$$\begin{aligned}&\mathrm{mig}(\Delta\mathcal{U}(\mathbf{a}_i \mapsto a_i)) \\ &= \min\{\Delta U(\mathbf{a}_i \mapsto \inf(\mathbf{a}_i)),\ \Delta U(\mathbf{a}_i \mapsto \mathrm{mid}(\mathbf{a}_i)),\ \Delta U(\mathbf{a}_i \mapsto \sup(\mathbf{a}_i))\}\end{aligned} \tag{3.14}$$

and

$$\begin{aligned}&\mathrm{mag}(\Delta\mathcal{U}(\mathbf{a}_i \mapsto a_i)) \\ &= \max\{\Delta U(\mathbf{a}_i \mapsto \inf(\mathbf{a}_i)),\ \Delta U(\mathbf{a}_i \mapsto \mathrm{mid}(\mathbf{a}_i)),\ \Delta U(\mathbf{a}_i \mapsto \sup(\mathbf{a}_i))\}\end{aligned} \tag{3.15}$$

respectively.

SA includes two metrics. One is based on $I(\bullet)$ that specifies the total information gain of design parameters quantitatively. The other is based on $\Delta U(\bullet)$ that provides the infeasibility change for the problem qualitatively. The values in $\Delta U(\bullet)$ will be used if the sensitivity levels cannot be decided based on the rules in Table 1. In this case, if $\mathrm{mig}(\Delta\mathcal{U}(\mathbf{a}_i \mapsto a_i)) \geqslant \mathrm{mag}(\Delta\mathcal{U}(\mathbf{a}_j \mapsto a_j))$, then $S(a_i) \geqslant S(a_j)$. If $\mathrm{mag}(\Delta\mathcal{U}(\mathbf{a}_i \mapsto a_i)) \leq \mathrm{mig}(\Delta\mathcal{U}(\mathbf{a}_j \mapsto a_j))$, then $S(a_i) \leq S(a_j)$. Otherwise, the sensitivity is not comparable.

3.3 Procedure of Interval Based GSA With the QCSP Formulation. Suppose that functional relationships $f(a, x) = b$ exist in a design problem, with design parameters $a \in \mathbb{R}^n$, design variables $x \in \mathbb{R}^l$, and target performances $b \in \mathbb{R}^m$. The nominal or ideal value of the target performances are denoted as b^0. The ranges of possible values for design parameters and variables are given as intervals in the QCSP formulation. The procedure of the proposed GSA approach starts with the QCSPSP formulation and produces a ranked list of design parameters based their sensitivity zones with each target performance. The SA procedure is described as follows:

Step 1. Formulate the given problem as a QCSP with constraints $F(\mathbf{a},\mathbf{x}) = \mathbf{b}$, where interval-valued $\mathbf{x} \in \mathbb{KR}^l$, $\mathbf{a} \in \mathbb{KR}^n$, and $\mathbf{b} \in \mathbb{KR}^m$ are the variational value ranges. $\mathbf{a}$ and $\mathbf{x}$ are assigned to be proper when they are not controllable by the designer, otherwise improper when they can be controlled and modified. $\mathbf{b}$ is calculated from $\mathbf{a}$ and $\mathbf{x}$.

Step 2. Select a design parameter a_i to be studied, and calculate the main information gain $I^m(a_i)$ with respect to each target performance for its variation from an interval to a representative value based on Eq. (3.2). The corresponding initial indeterminacy $\mathfrak{M}(\mathbf{b})$ and remaining indeterminacy $\mathfrak{M}(\mathbf{b}|a_i)$ are calculated based on Eq. (3.1). If the computable conditions in Eqs. (3.3) and (3.4) are not satisfied, compute quantifier mutation gain $Q(a_i)$ based on Eq. (3.5).

Step 3. Calculate the joint information gain $I^{jt}(a_i,a_k)$ between the selected design parameter a_i and another one a_k in the rest of the parameters by Eq. (3.6). Then calculate the interaction between the two parameters $I^{\mathrm{in}}(a_i,a_k)$ by Eq. (3.7) from $I_j^m(a_i)$, $I_j^m(a_k)$, and $I^{jt}(a_i,a_k)$. Repeat until the interactions between the selected parameter a_i and all other parameters are obtained. Here, only the interactions between two parameters are demonstrated.

Step 4. Calculate total information gains from the selected design parameter a_i by Eq. (3.8) for all three representative values. Generate the sensitivity zone based on the minimum and maximum of total information gains by Eqs. (3.9) and (3.10). If the main information gain of a parameter is not available, steps 3 and 4 are omitted for this parameter.

Step 5. Repeat steps (2)–(4) for all design parameters and rank the sensitivity of the parameters based on the rules in Table 1. If the ranking cannot be decided, continue to step 6 and rank them based on $\Delta U(\bullet)$.

Step 6. Calculate the infeasibility for each constraint with a given variation for the selected parameter by Eq. (3.11). It is a variation when a_i is chosen as one of the representative values of $\mathbf{a}_i$. Obtain the infeasibility change by Eq. (3.12), the total infeasibility change for the whole constraint system by Eq. (3.13). Repeat step 5 for all three representative values of the selected parameter a_i and generate the lower bound $\mathrm{mig}(\Delta\mathcal{U}(\mathbf{a}_i \mapsto a_i))$ and upper bound $\mathrm{mag}(\Delta\mathcal{U}(\mathbf{a}_i \mapsto a_i))$ by Eqs. (3.14) and (3.15). Rank parameters based on $\mathrm{mig}(\Delta\mathcal{U}(\mathbf{a}_i \mapsto a_i))$ and $\mathrm{mag}(\Delta\mathcal{U}(\mathbf{a}_i \mapsto a_i))$. If the ranking cannot be decided, continue to step 7.

Step 7. Provide a *suggested ranking* based on Table 2. A suggested ranking is the one that is likely to occur given the overlaps between sensitivity zones and the lack of further information.

Table 2 The rules for suggested ranking

Condition	Rules
Both $I_j(a_i)$ and $I_j(a_k)$ are computable; and steps 5 and 6 fail to decide the ranking	$S_j(a_i) \gtrsim S_j(a_k)$, if $\mathrm{mig}(\mathcal{I}_j(a_i)) \geqslant \mathrm{mig}(\mathcal{I}_j(a_k))$ and $\mathrm{mag}(\mathcal{I}_j(a_i)) \geqslant \mathrm{mag}(\mathcal{I}_j(a_k))$; or if $[\mathrm{mig}(\mathcal{I}_j(a_i)) + \mathrm{mag}(\mathcal{I}_j(a_i))]/2 \geqslant [\mathrm{mig}(\mathcal{I}_j(a_k)) + \mathrm{mag}(\mathcal{I}_j(a_k))]/2$ and $\mathrm{mag}(\mathcal{I}_j(a_i)) \geqslant \mathrm{mag}\ (\mathcal{I}_j(a_k))$
	$S_j(a_i) \lesssim S_j(a_k)$, if $\mathrm{mag}(\mathcal{I}_j(a_i)) < \mathrm{mag}(\mathcal{I}_j(a_k))$ and $\mathrm{mig}(\mathcal{I}_j(a_i)) < \mathrm{mig}(\mathcal{I}_j(a_k))$; or if $[\mathrm{mig}(\mathcal{I}_j(a_i)) + \mathrm{mag}(\mathcal{I}_j(a_i))]/2 < [\mathrm{mig}(\mathcal{I}_j(a_k)) + \mathrm{mag}(\mathcal{I}_j(a_k))]/2$ and $\mathrm{mag}(\mathcal{I}_j(a_i)) < \mathrm{mag}(\mathcal{I}_j(a_k))$
	Cannot be decided, otherwise
Neither $I_j(a_i)$ nor $I_j(a_k)$ is computable; and steps 5 and 6 fail to decide the ranking	$S_j(a_i) \gtrsim S_j(a_k)$, if $\mathrm{mig}(\mathcal{Q}_j(a_i)) \geqslant \mathrm{mig}(\mathcal{Q}_j(a_k))$ and $\mathrm{mag}(\mathcal{Q}_j(a_i)) \geqslant \mathrm{mag}(\mathcal{Q}_j(a_k))$; or if $[\mathrm{mig}(\mathcal{Q}_j(a_i)) + \mathrm{mag}(\mathcal{Q}_j(a_i))]/2 \geqslant [\mathrm{mig}(\mathcal{Q}_j(a_k)) + \mathrm{mag}(\mathcal{Q}_j(a_k))]/2$ and $\mathrm{mag}(\mathcal{Q}_j(a_i)) \geqslant \mathrm{mag}(\mathcal{Q}(a_k))$
	$S_j(a_i) \lesssim S_j(a_k)$, if $\mathrm{mag}(\mathcal{Q}_j(a_i)) < \mathrm{mag}(\mathcal{Q}_j(a_k))$ and $\mathrm{mig}(\mathcal{Q}_j(a_i)) < \mathrm{mig}(\mathcal{Q}_j(a_k))$; or if $[\mathrm{mig}(\mathcal{Q}_j(a_i)) + \mathrm{mag}(\mathcal{Q}_j(a_i))]/2 < [\mathrm{mig}(\mathcal{Q}_j(a_k)) + \mathrm{mag}(\mathcal{Q}_j(a_k))]/2$, and $\mathrm{mag}(\mathcal{Q}_j(a_i)) < \mathrm{mag}(\mathcal{Q}_j(a_k))$
	Cannot be decided, otherwise

Note: $\gtrsim$, suggested to rank higher and $\lesssim$, suggested to rank lower.

Note that the cost of the above SA procedure depends on the number of design parameters n. Because the variation is made one variable at a time, and three representative values are needed for each design parameter, the number of calculations is bounded by $\mathcal{O}(3^n)$.

The obtained result includes two outputs, a parameter ranking and the respective sensitivity zones. The ranking gives the relative importance of design parameters with respect to target performances. It provides an overview of which parameter is more sensitive than the other in one problem. The sensitivity zones provide quantitative measures of sensitivity that can be compared in different problems. When sensitivity zones do not overlap, the ranking can be decided based on the rules in Table 1. When there are overlaps, the suggested rankings are given based on Table 2.

In the current method, total information gain $I_j(a_i)$ computed as in Eq. (3.8) is used to construct sensitivity zones. This can be easily modified based on specific needs from users. For example, if we would like to rank based on the main information gain, the rules in Table 1 can be applied similarly to $I_j^m(a_i)$. That is, if $I_j^m(a_i)$ and $I_j^m(a_k)$ are both computable, $\mathrm{mig}(I_j(a_i))$ and $\mathrm{mag}(I_j(a_k))$ can be computed as the minimum among $I_j^m(\inf(\mathbf{a}_i))$, $I_j^m(\mathrm{mid}(\mathbf{a}_i))$, and $I_j^m(\sup(\mathbf{a}_i))$ and the maximum among $I_j^m(\inf(\mathbf{a}_k))$, $I_j^m(\mathrm{mid}(\mathbf{a}_k))$, and $I_j^m(\sup(\mathbf{a}_k))$, respectively. The ranking then is for the main information gain.

4 Numerical Examples and Results

In this section, the proposed GSA method for interval-valued quantified constraints is demonstrated with three examples. Ishigami function [66] as an analytical example is used to compare our method with the variance based GSA. The dynamic performance analysis of a battery-electric vehicle (BEV) [67] is used to compare our method with the gradient based local SA. A third example of pump family design [68] is used to illustrate the SA with logic quantifier change that is unique in our method. Among the three examples, the first one is a numerical example, whereas the second and third ones are for specific design problems. The second example has two constraints. The third example is the most complex one with multilevel attributes. Our approach is generic enough to solve problems with multiple constraints.

4.1 Example 1: Ishigami function. Ishigami function is an analytical function, as

$$Y = \sin X_1 + a \sin^2 X_2 + b X_3^4 \sin X_1 \tag{4.1}$$

where X_1, X_2, and X_3 are design parameters with values varying within the range from $-\pi$ to π. $a = 7$ and $b = 0.1$ are constants. In our approach, no further assumptions of probability distributions about design parameters are made as in Ref. [66]. Because any numerical constraint can be treated as a quantified one, in which all variables are proper intervals. Therefore, Eq. (4.1) is formulated as quantified interval constraint $\mathbf{Y} = \sin\mathbf{X}_1 + 7\sin^2\mathbf{X}_2 + 0.1\mathbf{X}_3{}^4\sin\mathbf{X}_1$, where $\mathbf{X}_1 = \mathbf{X}_2 = \mathbf{X}_3 = [-\pi,\pi]$ are proper intervals.

For the variance based GSA approach, the total and partial variances can be obtained analytically from Eq. (4.1). Hence, the global sensitivities of design parameters are in a decreasing order when the values of a and b are plugged into the analytical expressions of the variances, as shown in the first row of Table 3.

Table 3 Sensitivity ranking in example 1

		Sensitivity ranking		
Output	Methods	highest←		→lowest
Y	Variance based approach [66]	X_1	X_2	X_3
	The proposed approach	X_1^*	X_3^*	X_2^*

In the proposed approach, the sensitivity zones of the three design parameters are calculated and shown as vertical bars in Fig. 2. The sensitivity zone consists of the minimum and maximum values of total information gain calculated based on Eqs. (3.9) and (3.10). The dots in Fig. 2 denote the total information gain calculated from the representative values. For this example, the computed lower and upper bounds of total information gain for X_1 are equal to each other, so as for X_3, i.e., $I(\inf(\mathbf{X}_1)) = I(\sup(\mathbf{X}_1))$ and $I(\inf(\mathbf{X}_3)) = I(\sup(\mathbf{X}_3))$. The values of total information gain for X_2 from the three representative values are exactly the same, which is about 0.1.

The sensitivity ranking from the proposed approach is shown in the second row of Table 3. The sensitivity of design parameters are ranked based on the sensitivity zones and infeasibility as stated in steps 5–7 in Sec. 3.3. From Fig. 2, it can be seen that Y is more sensitive with respect to X_1 than to X_2 because $\mathrm{mig}(\mathcal{I}(X_1)) \geqslant \mathrm{mag}(\mathcal{I}(X_2))$ based on the first rule in Table 1. Again, a higher total information gain indicates a higher sensitivity if sensitivity zones do not overlap. However, the ranking between X_1 and X_3, and the ranking between X_2 and X_3 cannot be decided based on Table 1 because there are overlaps between their sensitivity zones. Then, $\Delta\mathcal{U}(\bullet)$'s are used to rank as described in step 6. However, the ranking still cannot be decided, because the values of $\Delta\mathcal{U}(\bullet)$'s are equal. As a further step, a suggested ranking between X_1 and X_3 is provided in Table 3, which is marked by stars (*). That is, Y is likely to be more sensitive with respect to X_1 than to X_3, because of the first rule in Table 2. For X_2 and X_3, $\mathrm{mig}(\mathcal{I}(X_2)) \geqslant \mathrm{mig}(\mathcal{I}(X_3))$, $\mathrm{mag}(\mathcal{I}(X_2)) < \mathrm{mag}(\mathcal{I}(X_3))$, and the midpoint of $\mathcal{I}(X_3)$ is larger than the midpoint of $\mathcal{I}(X_2)$. Then Y is likely to be more sensitive with respect to X_3 than to X_2. The suggested ranking between X_2 and X_3 is also marked by * in Table 3.

It is seen that our method is more conservative than the variance based approach. Here, we only assert that X_1 is more sensitive than X_2, and suggest that X_1 is likely to be more sensitive than X_3 and X_3 is likely to be more sensitive than X_2, given that we use less information in the analysis, without probability distributions. Sensitivity zones intend to include all possible values of sensitivities for the parameters. Based on the information obtained from the proposed method, designers may choose to adjust X_3, between X_2 and X_3, to change the target performance as an optimistic strategy. Yet, as a conservative strategy, X_2 may be chosen instead. In other cases, designers may choose to adjust X_1 among X_1, X_2, and X_3 to test the robustness of the system in the presence of uncertainty. When a model simplification with dimensionality reduction is needed, designers may choose to fix the value of X_2 because it potentially has the smallest effect on Y.

4.2 Example 2: Application in BEV Dynamic Performance. Maximum speed $u_{\max}$ and acceleration time t are two primary indices of the dynamic performance for a vehicle, which are mainly influenced by the mass of the vehicle m, the coefficient of

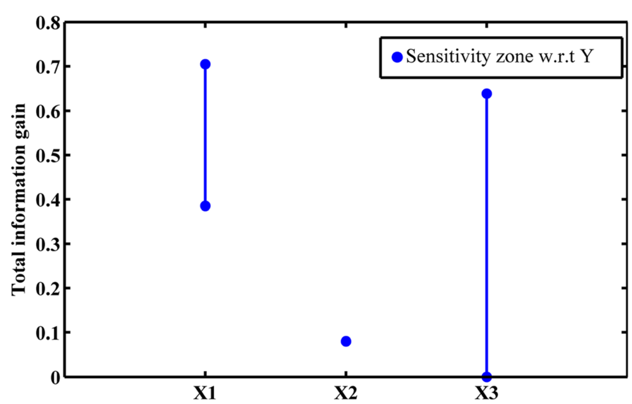

Fig. 2 Sensitivity zones of X_1–X_3 with respect to Y

rolling resistance f, the mechanical efficiency of the transmission system η_t, and the factor of air resistance C_dA. $u_{\max}$ is calculated as

$$u_{\max} = \left(-\frac{q}{2} + \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}\right)^{1/3} + \left(-\frac{q}{2} - \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}\right)^{1/3} \quad (4.2)$$

where $p = 21.15mgf/C_dA$, $q = -76140P_m\eta_t/C_dA$, g is the gravity coefficient, and P_m is the rated power of motor. The acceleration time is calculated by

$$t = \delta m \int_{u_1}^{u_2} (F_t - F_f - F_\omega)^{-1} du \quad (4.3)$$

where $F_t = T_m i\eta_t/r$, $F_f = mgf$, $F_w = C_dAu^2/21.15$, δ is the conversion coefficient of rotating mass, r is the wheel radius, and i is the transmission ratio.

The data used in this example are given in Table 4 in which m, f, η_t, and C_dA vary between $\pm 9\%$ of the initial values while others are fixed real values. Similar to example 1, two constraints in Eqs. (4.2) and (4.3) can be formulated as quantified interval-valued constraints while preserving the original design intent, which have a solution set with the interpretation $(\forall X\rho \epsilon \mathbf{X}\rho^{\Delta})(\forall A\rho \epsilon \mathbf{A}\rho^{\Delta})(\exists B\rho \epsilon \mathbf{B}\rho^{\Delta}) f(A\rho, X\rho) = B\rho$ with the target performance, design parameter, and design variable as $B\rho = (u_{\max}, t)$, $A\rho = (C_dA, f, m, \eta_t)$, and $X\rho = (T_m, P_m, r, i, \delta)$, respectively.

The comparison between the results from the proposed approach and the ones from the traditional local SA method in Ref. [67] is shown in Fig. 3. In this example, some sensitivity zones of design parameters are not clearly shown as the bars in Fig. 2, because the values of total information gain computed from the representative values are very close to each other. The sensitivity ranking between input parameters η_t and m with respect to constraint t cannot be decided in the proposed approach and a suggested ranking is provided and marked with * in Table 5, because their sensitivity zones overlap, as shown in Fig. 3(*b*). In the figure, three markers for each design parameter indicate min, max, and middle values, respectively. The sensitivity zone of η_t is slightly higher than that of m. It indicates that η_t could be more sensitive than m, which matches the result obtained by the traditional local SA method, as in the second row in Table 5. The similar situation also occurs for the rankings of parameters η_t and C_dA with respect to constraint $u_{\max}$. Moreover, for those design parameters where rankings cannot be decided by the proposed method, their sensitivities are also very close to each other in traditional local SA method. Thus, it is likely that their sensitivities are too close to decide when the uncertainty associated with the data is considered. The results in Fig. 3 show that the proposed approach works well for the problem and it is compatible to traditional local SA method.

Based on the results from the proposed method, engineers can find that η_t is the most sensitive parameter for both $u_{\max}$ and t. Thus, η_t can be adjusted when both performances need to be improved. When only $u_{\max}$ needs to be improved, C_dA can be adjusted. Similarly, when only t needs to be improved, m can be adjusted.

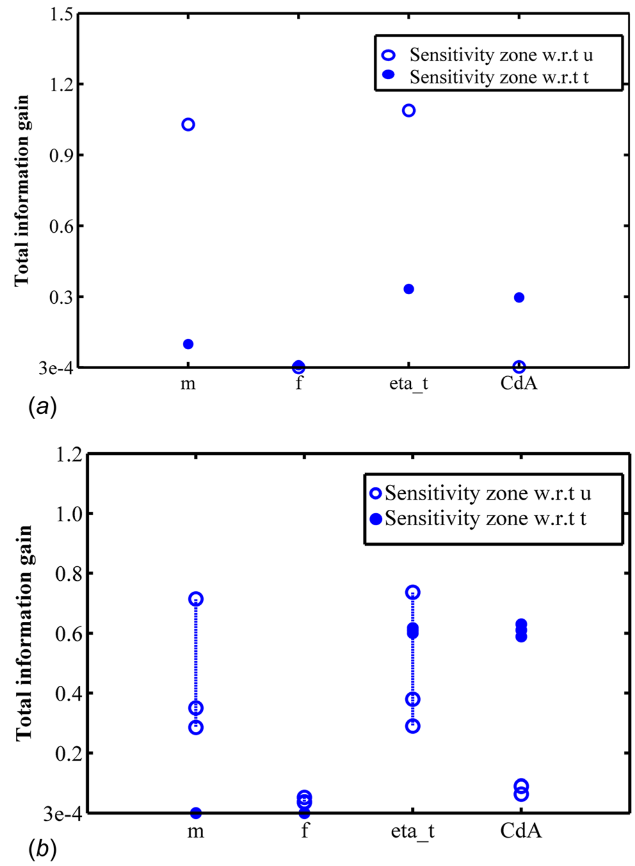

Fig. 3 Result comparison between the (*a*) proposed method and (*b*) traditional local SA method

4.3 Example 3: Application in Finger Pump Design. In the pump family design problem in Ref. [68], tube width T_w, the number of fingers N_f, and finger width F_w are chosen to be the design parameters for customization. Their commonality within the family is analyzed by conducting SA with respect to a multi-attribute utility function $U(x)$. Voltage v is a design variable and not considered for SA, because it is allowed to vary such that the desired flow rate is achievable.

The pump efficiency η and volume vol are two attributes to evaluate the overall performance of the pump family, but with conflicting goals. A tradeoff is formulated as the utility function

$$U(x) = k_\eta u(\eta) + k_{\text{vol}} u(\text{vol}) \quad (4.4)$$

where k_η and k_{vol} are weight constants (here $k_\eta = k_{\text{vol}} = 0.5$), and

$$u(\eta) = -16.54\eta^2 + 8.12\eta - 0.001,$$
$$u(\text{vol}) = 0.0002\text{vol}^2 - 0.01\text{vol} + 1.4508$$

The average efficiency η and volume vol of the product family are calculated as the average of the efficiencies and volumes for each product variant, which are

Table 4 Data used in example 2

Name	Value	Name	Value
C_dA	0.63 m^2	T_m	120 N M
f	0.013	P_m	16 KW
η_t	0.9	g	9.8 N/kg
m	1030 kg	δ	1.4
i	4.87	r	0.28 m
u_1	50 km/hr	u_2	70 km/h

Table 5 Sensitivity ranking in example 2

Methods	Constraints	Sensitivity ranking highest←			→lowest
Local SA	U	η_t	C_dA	m	f
	T	η_t	m	f	C_dA
Proposed	U	η_t^*	C_dA^*	m	f
	T	η_t^*	m^*	f	C_dA

$$\eta = \frac{1}{n}\sum_{i=1}^{n} F_{\mathrm{p}_i}/B_{\mathrm{p}_i}, \quad \mathrm{vol} = \frac{1}{n}\sum_{i=1}^{n} D_i \times W_i \times H_i \tag{4.5}$$

where n is the total number of variants. For each product variant, F_{p} is the fluid power, B_{p} is the brake power, and D, W, and H are the pump's depth, width, and height, respectively. The fluid power and brake power can be calculated as

$$F_{\mathrm{P}} = 0.785 \times S_{\mathrm{q}} \times F_{\mathrm{w}} \times N_{\mathrm{f}} \times N_{\mathrm{c}}, B_{\mathrm{p}} = I^2 \times U_{\mathrm{r}} \times S_{\mathrm{q}} \times C_{\mathrm{r}} \times L_{\mathrm{r}}$$

where N_{c} is the number of cycles (one cycle is from the first finger to the last finger compression), I is the battery current, U_{r} is the unit resistance of flexinol, C_{r} is the contraction ratio of flexinol, and L_{r} is the lever ratio. The pump depth D, width W, and height H are calculated by

$$D = N_{\mathrm{f}} \times F_{\mathrm{w}} + \alpha, W = 2 \times (S_{\mathrm{q}} + \beta), H = T_{\mathrm{w}} + \gamma$$

where S_{q} is the squeeze distance, α, β, and γ are the additional lengths due to the frame of the pump, which are predetermined when a particular setting is desired. Here, they are set to be zeroes. The numerical values are given in Table 6. Two scenarios are used to illustrate the proposed method.

Scenario 1: Only one type of quantifiers, either $\forall$ or $\exists$, is associated with interval-valued parameters. In order to compare with the result in Ref. [68], consider the solution set with the interpretation of $(\forall X_\rho \in \mathbf{X}_\rho^\Delta)(\forall A_\rho \in \mathbf{A}_\rho^\Delta)(\exists B_\rho \in \mathbf{B}_\rho^\Delta) f(A_\rho, X_\rho) = B_\rho$, where $B_\rho = (U(x))$ is target performance, $A_\rho = (T_{\mathrm{w}}, N_{\mathrm{f}}, F_{\mathrm{w}})$ is design parameter, and design variable $X\rho$ denotes all other interval variables used in the problem, such as S_{q}, C_{r}, L_{r}, etc. Note that universally quantified Eqs. (4.4) and (4.5) correspond to the classical interval constraints without quantifiers.

Scenario 2: Two types of quantifiers, $\forall$ and $\exists$, are associated with the interval-valued parameters so that the impact of quantifier change on the sensitivities can also be evaluated. Consider the solution set with the interpretation $(\forall X_\rho \in \mathbf{X}_\rho^\Delta)(\forall A_\rho \in \mathbf{A}_\rho^\Delta)(\exists B_\rho \in \mathbf{B}_\rho^\Delta)(\exists A_\iota \in \mathbf{A}_\iota^\Delta) f(A_\rho, A_\iota, X_\rho) = B_\rho$ where B_ρ and X_ρ are the same as the ones in scenario 1, $A_\rho = (T_{\mathrm{w}}, 0, F_{\mathrm{w}})$ is the proper set of parameters, and $A_\iota = (0, N_{\mathrm{f}}, 0)$ is the improper set.

The global sensitivity result in Ref. [68] and the suggested sensitivity rankings provided by the proposed approach are shown in Table 7. The total information gains when parameters are associated with different quantifiers are shown in Fig. 4. As marked by * in the second and third rows of Table 7, the sensitivity rankings of design parameters cannot be decided for sure because of their overlapped sensitivity zones, as shown in Fig. 4. In scenario 1 where all parameters are universally quantified, the upper bound of the sensitivity zone for F_{w} is higher than the one for T_{w}. It indicates that F_{w} could be more sensitive than T_{w}, which matches the result obtained in Ref. [68], as in the first row of Table 7. The similar situation also occurs to the rankings of T_{w} and N_{f}. In scenario 2 where the existential quantifier is associated with N_{f}, the upper bound of the sensitivity zone for N_{f} becomes higher than it is in scenario 1, as shown Fig. 4. In addition, a design parameter may have influences on the sensitivities of the other design parameters when it is associated with different quantifiers. For instance, because design parameters F_{w} and T_{w} are involved in the same constraint as N_{f}, their sensitivity rankings are changed when the quantifier of N_{f} is changed from universal to existential. The enlarged sensitivity zones indicate that N_{f} is more sensitive than F_{w}, and F_{w} is more sensitive than T_{w}.

Table 6 Data used in example 3

Name	Value	Name	Value
T^w (cm)	[1.0,2.5]	C^r	[0.011, 0.013]
N^f (discrete)	[5,15]	L^r	[10, 20]
F^w (cm)	[0.3,1]	S^q (cm)	[0.1, 1]
N^c (discrete)	[10,38]	I (A)	[0.1, 0.2]
N	3	U^r (Ω/cm)	0.5

Table 7 Sensitivity ranking comparison in example 3

		Sensitivity ranking		
Methods		highest←		→lowest
Ref. [68]		F_{w}	T_{w}	N_{f}
Proposed method	Scenario 1: All universal	F_{w}^*	T_{w}^*	N_{f}^*
	Scenario 2: N_f existential	N_{f}^*	F_{w}^*	T_{w}^*

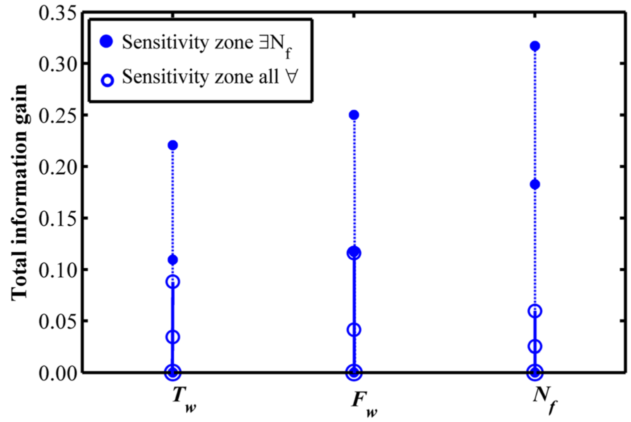

Fig. 4 Sensitivity zones of design parameters in scenario 1 (all universal) and scenario 2 (N_f existential)

Based on the results obtained in the SA, engineers can adjust the values of design parameters according to their sensitivities with respect to some particular target performances. In a product family, the commonality of product variants is a result of reuse and asset sharing of components, processes, technologies, interfaces, and infrastructure. The most sensitive design parameter which also has the least commonality within a product family is easy to be isolated for adjustment. In contrast, the least sensitive parameter which also has the most commonality provides the robustness baseline of design. In scenario 1, when all design parameters are universally quantified, any possible combination of their values as the variation is considered in the analysis. In order to ensure the compatibility of the product family, engineers need to be careful when adjusting F_{w}, which is the most sensitive parameter. In scenario 2, when the design requirement of N_{f} is flexible and can be modified as more information is available during the design process, it needs to be existentially quantified. Yet engineers need to be careful when specifying its values, because existentially quantified N_{f} is likely to become the most sensitive one among all parameters as a result of the change of design intent.

5 Conclusions

In engineering design, functional relationships among design parameters, design variables, and target performances are usually expressed as constraints. SA for design parameters in the constraints is a critical problem for engineers when they want to know which design parameter contributes the most to the variation of target performances when uncertainty is involved. In this paper, a new global SA method was developed for interval-valued quantified constraints, in which an interval specifies the range of variation. Different from statistical SA, only the lower and upper bounds of uncertain design parameters are needed in the proposed approach, without assuming probability distributions of the parameters. Generalized intervals with logical quantifiers are applied to represent the design parameters and variables, with the advantage of capturing more design intent than the classical interval. A generalized Hartley like measure is used to characterize the indeterminacy associated with the interval-valued design parameters. The effects of both quantifiers and numerical values on the constraints are analyzed qualitatively and quantitatively. Based on the proposed metrics of information gain and infeasibility, we can efficiently assess the effect of design parameter variations on the target performances of the constraints.

This approach provides an efficient alternative to the classical variance-based statistical SA with a much lower computational cost using interval arithmetic. The major limitation of this approach is that closed form analytical constraints need to be available. In addition, the sensitivity zone in the proposed method is intended to improve the robustness of SA without exhaustive computation. As a result, the estimation of sensitivity zones based on the three representative values may not be the actual lower and upper bounds of sensitivity. The lower, upper, and midpoint values of an interval-valued design parameter are chosen because of their practical meanings in engineering design. The midpoint is the nominal or ideal value of a design parameter, whereas the lower and upper bounds correspond to the limits of its value. Furthermore, the absolute rankings of sensitivities may not be available when the sensitivity zones overlap. In the future, we will investigate different options of choosing representative values and reducing the possibility of underestimating sensitivity zones while maintaining reasonable computational loads.

The focus of this paper is the SA for feasible design space searching problems. In the future, the proposed approach will be extended to problems with inequality constraints, as well as design optimization problems, such as the sensitivity of searching directions if analytical relations between the searching directions and variables are available, or the sensitivity of optimality if the exact lower or upper bounds of objective functions within the searching domain can be estimated.

Acknowledgment

This research was supported in part by China Scholarship Council with a scholarship (Grant No. 2011613059) to the first author.

References

[1] Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M., 2004, *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*, Wiley, New York.
[2] Millwater, H., and Feng, Y., 2011, "Probabilistic Sensitivity Analysis With Respect to Bounds of Truncated Distributions," ASME J. Mech. Des., **133**(6), p. 061001.
[3] Caro, S., Wenger, P., Bennis, F., and Chablat, D., 2005, "Sensitivity Analysis of the Orthoglide: A Three-DOF Translational Parallel Kinematic Machine," ASME J. Mech. Des., **128**(2), pp. 392–402.
[4] Youn, B. D., and Wang, P., 2008, "Bayesian Reliability-Based Design Optimization Using Eigenvector Dimension Reduction (EDR) Method," Struct. Multidiscip. Optim., **36**(2), pp. 107–123.
[5] Zou, T., Mourelatos, Z. P., Mahadevan, S., and Tu, J., 2008, "An Indicator Response Surface Method for Simulation-Based Reliability Analysis," ASME J. Mech. Des., **130**(7), p. 071401.
[6] Kleijnen, J. P., 1998, "Experimental Design for Sensitivity Analysis, Optimization, and Validation of Simulation Models," *Handbook of Simulation*, J. Banks, ed., Wiley, New York, pp. 173–223.
[7] Box, G. E., Hunter, W. G., and Hunter, J. S., 1978, *Statistics for Experimenters: An Introduction to Design, Data Analysis, and Model Building*, Wiley, New York.
[8] Morris, M. D., 1991, "Factorial Sampling Plans for Preliminary Computational Experiments," Technometrics, **33**(2), pp. 161–174.
[9] Chen, W., Jin, R., and Sudjianto, A., 2005, "Analytical Variance-Based Global Sensitivity Analysis in Simulation-Based Design Under Uncertainty," ASME J. Mech. Des., **127**(5), pp. 875–886.
[10] Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S., 2008, *Global Sensitivity Analysis: The Primer*, Wiley-Interscience, New York.
[11] Iman, R. L., and Helton, J. C., 1988, "An Investigation of Uncertainty and Sensitivity Analysis Techniques for Computer Models," Risk Anal., **8**(1), pp. 71–90.
[12] Helton, J., Garner, J., Marietta, M., Rechard, R., Rudeen, D., and Swift, P., 1993, "Uncertainty and Sensitivity Analysis Results Obtained in a Preliminary Performance Assessment for the Waste Isolation Pilot Plant," Nucl. Sci. Eng., **114**(4), pp. 286–331.
[13] Drignei, D., and Mourelatos, Z. P., 2012, "Parameter Screening in Statistical Dynamic Computer Model Calibration Using Global Sensitivities," ASME J. Mech. Des., **134**(8), p. 081001.
[14] Chatterjee, S., and Hadi, A. S., 2009, *Sensitivity Analysis in Linear Regression*, Wiley, New York.
[15] Liu, Y., Yin, X., Huang, H.-Z., Arendt, P., and Chen, W., 2010, "A Hierarchical Statistical Sensitivity Analysis Method for Multilevel Systems With Shared Variables," ASME J. Mech. Des., **132**(3), p. 031006.
[16] Yue, J., Camelio, J. A., Chin, M., and Cai, W., 2006, "Product-Oriented Sensitivity Analysis for Multistation Compliant Assemblies," ASME J. Mech. Des., **129**(8), pp. 844–851.
[17] Saltelli, A., Tarantola, S., and Chan, K.-S., 1999, "A Quantitative Model-Independent Method for Global Sensitivity Analysis of Model Output," Technometrics, **41**(1), pp. 39–56.
[18] Chan, K., Saltelli, A., and Tarantola, S., 1997, "Sensitivity Analysis of Model Output: Variance-Based Methods Make the Difference," 29th Conference on Winter Simulation, IEEE Computer Society, pp. 261–268.
[19] Liu, H., Sudjianto, A., and Chen, W., 2006, "Relative Entropy Based Method for Probabilistic Sensitivity Analysis in Engineering Design," ASME J. Mech. Des., **128**(2), pp. 326–336.
[20] Hutcheson, R. S., and McAdams, D. A., 2010, "A Hybrid Sensitivity Analysis for Use in Early Design," ASME J. Mech. Des., **132**(11), p. 111007.
[21] Neumaier, A., 1989, "Rigorous Sensitivity Analysis for Parameter-Dependent Systems of Equations," J. Math. Anal. Appl., **144**(1), pp. 16–25.
[22] Rump, S. M., 1990, "Rigorous Sensitivity Analysis for Systems of Linear and Nonlinear Equations," Math. Comput., **54**(190), pp. 721–736.
[23] Wallner, J., Schröcker, H. P., and Hu, S. M., 2005, "Tolerances in Geometric Constraint Problems," Reliab. Comput., **11**(3), pp. 235–251.
[24] Goldsztejn, A., 2008, "Sensitivity Analysis Using a Fixed Point Interval Iteration," Techncial Report Hal No. 00339377.
[25] Guo, J., and Du, X., 2009, "Reliability Sensitivity Analysis With Random and Interval Variables," Int. J. Numer. Eng., **78**(13), pp. 1585–1617.
[26] Li, M., and Williams, N., 2009, "Interval Uncertainty Reduction and Single-Disciplinary Sensitivity Analysis With Multi-Objective Optimization," ASME J. Mech. Des., **131**(3), pp. 1–11.
[27] Li, M., Hamel, J., and Azarm, S., 2010, "Optimal Uncertainty Reduction for Multi-Disciplinary Multi-Output Systems Using Sensitivity Analysis," Struct. Multidiscip. Optim., **40**(1), pp. 77–96.
[28] Börner, F., Bulatov, A., Jeavons, P., and Krokhin, A., 2003, "Quantified Constraints: Algorithms and Complexity," Computer Science Logic, M. Baaz and J. A. Makowsky eds., Springer Publishing, New York, pp. 58–70.
[29] Shary, S. P., 2002, "A New Technique in Systems Analysis Under Interval Uncertainty and Ambiguity," Reliab. Comput., **8**(5), pp. 321–418.
[30] Hu, J., Aminzadeh, M., and Wang, Y., 2014, "Searching Feasbile Design Space by Solving Quantified Constraint Satisfaction Problems," ASME J. Mech. Des., **136**(3), p. 031002.
[31] Zeleny, M., 1973, "Compromise Programming," *Multiple Criteria Decision Making*, J. L. C. A. M. Zeleny, ed., University of South Carolina Press, Columbia, SC, pp. 262–301.
[32] Sobek, D. K., Ward, A. C., and Liker, J. K., 1999, "Toyota's Principles of Set-Based Concurrent Engineering," Sloan Manage. Rev., **40**(2), pp. 67–84.
[33] Singer, D. J., Doerry, N., and Buckley, M., 2009, "What Is Set-Based Design?," Nav. Eng. J., **121**(4), pp. 31–43.
[34] Chen, W., Wiecek, M. M., and Zhang, J., 1999, "Quality Utility—A Compromise Programming Approach to Robust Design," ASME J. Mech. Des., **121**(2), pp. 179–187.
[35] Apt, K. R., 1999, "The Essence of Constraint Propagation," Theor. Comput. Sci., **221**(1–2), pp. 179–210.
[36] Klir, G. J., 2006, *Uncertainty and Information: Foundations of Generalized Information Theory*, Wiley-Interscience, Hoboken, NJ.
[37] Eastman, C. M., 1973, "Automated Space Planning," Artif. Intell., **4**(1), pp. 41–64.
[38] Medjdoub, B., and Yannou, B., 2000, "Separating Topology and Geometry in Space Planning," Comput. Aided Des., **32**(1), pp. 39–61.
[39] Dohmen, M., 1995, "A Survey of Constraint Satisfaction Techniques for Geometric Modeling," Comput. Graphics, **19**(6), pp. 831–845.
[40] Yannou, B., Moreno, F., Thevenot, H. J., and Simpson, T. W., "Faster Generation of Feasible Design Points," ASME Paper No. DETC2005-85449.
[41] Titus, N., and Ramani, K., 2005, "Design Space Exploration Using Constraint Satisfaction," Configuration Workshop at the 19th International Joint Conference on Artificial Intelligence (IJCAI'05), pp. 31–36.
[42] Sébastian, P., Chenouard, R., Nadeau, J. P., and Fischer, X., 2007, "The Embodiment Design Constraint Satisfaction Problem of the BOOTSTRAP Facing Interval Analysis and Genetic Algorithm Based Decision Support Tools," Int. J. Interact. Des. Manuf., **1**(2), pp. 99–106.
[43] Panchal, J. H., Gero Fernández, M., Paredis, C. J. J., Allen, J. K., and Mistree, F., 2007, "An Interval-Based Constraint Satisfaction (IBCS) Method for Decentralized, Collaborative Multifunctional Design," Concurrent Eng., **15**(3), pp. 309–323.
[44] Yvars, P. A., 2009, "A CSP Approach for the Network of Product Lifecycle Constraints Consistency in a Collaborative Design Context," Eng. Appl. Artif. Intell., **22**(6), pp. 961–970.
[45] Lottaz, C., Sam-Haroud, D., Faltings, B., and Smith, I., "Constraint Techniques for Collaborative Design," IEEE International Conference on Tools With Artificial Intelligence, Taipei, Nov. 10–12, pp. 34–41.

[46] Dantan, Y. J., 2005, "Tolerance Synthesis: Quantifier Notion and Virtual Boundary," Comput.-Aided Des., **37**(2), pp. 231–240.
[47] Qureshi, A. J., Dantan, J. Y., Bruyere, J., and Bigot, R., 2010, "Set Based Robust Design of Mechanical Systems Using the Quantifier Constraint Satisfaction Algorithm," Eng. Appl. Artif. Intell., **23**(7), pp. 1173–1186.
[48] Wang, Y., 2008, "Interpretable Interval Constraint Solvers in Semantic Tolerance Analysis," Comput. Aided Des. Appl., **5**(5), pp. 654–666.
[49] Wang, Y., 2008, "Closed-Loop Analysis in Semantic Tolerance Modeling," ASME J. Mech. Des., **130**(6), p. 061701.
[50] Dantan, J. Y., and Qureshi, A. J., 2009, "Worst-Case and Statistical Tolerance Analysis Based on Quantified Constraint Satisfaction Problems and Monte Carlo Simulation," Comput.-Aided Des., **41**(1), pp. 1–12.
[51] Benhamou, F., Goualard, F., Languenou, E., and Cheristie, M., 2004, "Interval Constraint Solving for Camera Control and Motion Planning," ACM Trans. Comput. Logic, **5**(4), pp. 732–767.
[52] Jirstrand, M., 1997, "Nonlinear Control System Design by Quantifier Elimination," J. Symbolic Comput., **24**(2), pp. 137–152.
[53] Herrero, P., Sainz, M. A., Vehi, J., and Jaulin, L., 2005, "Quantified Set Inversion Algorithm With Applications to Control," Reliab. Comput., **11**(5), pp. 369–382.
[54] Herrero, P., Sainz, M. Á., Vehí, J., and Jaulin, L., 2004, "Quantified Set Inversion With Applications to Control," IEEE International Symposium on Computer Aided Control Systems Design, Taipei, Sept. 4, pp. 179–183.
[55] Ratschan, S., and Vehı, J., "Robust Pole Clustering of Parametric Uncertain Systems Using Interval Methods," 4th IFAC Symposium on Robust Control Design, S. Bittanti, and P. Colaneri, eds., pp. 323–328.
[56] Benedetti, M., Lallouet, A., and Vautard, J., 2007, "Modeling Adversary Scheduling With QCSP+," Proceedings of the 23rd Annual ACM Symposium on Applied Computing, ACM Press, New York, pp. 151–155.
[57] Benedetti, M., Lallouet, A., and Vautard, J., "QCSP Made Practical by Virtue of Restricted Quantification," 20th International Joint Conference on Artificial Intelligence (IJCAI 2007), pp. 38–43.
[58] Sachenbacher, M., and Maier, P., 2008, "Test Strategy Generation Using Quantified CSPs," Proceedings of the 14th International Conference on Principles and Practice of Active of Constraint Programming (CP-08), P. J. Stuckey, ed., Springer, New York, pp. 566–570.
[59] Sachenbacher, M., and Schwoon, S., 2008, "Model-Based Testing Using Quantified CSPs: A Map," Workshop at the ECAI 2008 on Model-Based Systems, pp. 37–41.
[60] Gardeñes, E., Sainz, M. Á., Jorba, L., Calm, R., Estela, R., Mielgo, H., and Trepat, A., 2001, "Modal Intervals," Reliab. Comput., **7**(2), pp. 77–111.
[61] Dimitrova, N., Markov, S., and Popova, E., 1992, "Extended Interval Arithmetics: New Results and Applications," *Computer Arithmetics Enclosure Methods*, L. Atanasova, and J. Herzberger, eds., Elsevier, Amsterdam, The Netherlands, pp. 225–232.
[62] Kaucher, E., 1980, "Interval Analysis in the Extended Interval Space IR," Comput. Suppl., **2**, pp. 33–49.
[63] Moore, R. E., Kearfott, R. B., and Cloud, M. J., 2009, *Introduction to Interval Analysis*, Society for Industrial Mathematics, Philadelphia, PA.
[64] Rényi, A., 1970, *Introduction to Information Theory, Probability Theory*, North-Holland, Amsterdam, The Netherlands, pp. 540–616.
[65] Hu, J., and Wang, Y., 2014, "Sensitivity Analysis for Quantified Interval Constraints," *Safety, Reliability, Risk and Life-Cycle Performance of Structures and Infrastructures*, G. Deodatis, B. R. Ellingwood, and D. M. Frangopol, eds., CRC Press, Boca Raton, FL, pp. 2931–2938.
[66] Homma, T., and Saltelli, A., 1996, "Importance Measures in Global Sensitivity Analysis of Nonlinear Models," Reliab. Eng. Syst. Safety, **52**(1), pp. 1–17.
[67] Chen, M., and Guo, L., 2011, "The Parameters Sensitivity Analysis of Battery Electric Vehicle Dynamic," Appl. Mech. Mater., **80–81**, pp. 837–840.
[68] Chad, H., and Rosen, D. W., "Identification of Platform Variables in Product Family Design Using Sensitivity Analysis," ASME Paper No. DETC2012-71198.