Monte-Carlo-Type Techniques for Processing Interval Uncertainty, and Their Potential Engineering Applications

V. Kreinovich, J. Beck, C. Ferregut, A. Sanchez, G. R. Keller, M. Averill and S. A. Starks
College of Engineering and NASA Pan-American Center for Earth and Environmental Studies (PACES), University of Texas, El Paso, TX 79968, USA (vladik@utep.edu)

Abstract. In engineering applications, we need to make decisions under uncertainty. Traditionally, in engineering, statistical methods are used, methods assuming that we know the probability distribution of different uncertain parameters. Usually, we can safely linearize the dependence of the desired quantities y (e.g., stress at different structural points) on the uncertain parameters x_i – thus enabling sensitivity analysis. Often, the number n of uncertain parameters is huge, so sensitivity analysis leads to a lot of computation time. To speed up the processing, we propose to use special Monte-Carlo-type simulations.

Keywords: interval uncertainty, Monte-Carlo techniques, engineering applications

1. Introduction

1.1. In Engineering Applications, It Is Important to Process Uncertainty

Two types of uncertainty. Typically, in engineering applications, we need to make decisions under uncertainty. This uncertainty can be of two different types (see, e.g., [1, 5, 8, 23, 24]):

– We may have uncertainty due to lack of knowledge (*epistemic* uncertainty). An example of such uncertainty is measurement-related uncertainty: there exists a single actual value x of the measured quantity; because of the measurement inaccuracy, we only know the approximate value $\widetilde{x}$ of the measured quantity. If we know the upper bound Δ on the absolute value $|\Delta x|$ of the measurement error $\Delta x \stackrel{\text{def}}{=} \widetilde{x} - x$, then the only information that we have about the actual (unknown) value x is that this value belongs to the interval $[\widetilde{x} - \Delta, \widetilde{x} + \Delta]$.

– We may also have uncertainty due to variability (*aleatoric* uncertainty). For example, we have limits L_i on the loads l_i in different rooms i, but we do not know how exactly these loads will be distributed – and we want to make sure that our design is safe for all possible $l_i \leq L_i$.

Comment. It is important to emphasize that while epistemic and aleatoric uncertainty are very different concepts, the problems related to these two types of uncertainty are closely related and often, described by similar mathematical formulas. For example, one the traditional ways to characterize the epistemic uncertainty related to a measurements is to describe the probability distribution on the set of all possible values of the measurement error $\Delta x_i = \widetilde{x}_i - x_i$.

– From the viewpoint of the quantity x_i, there is only one (unknown) actual value of x_i, so we are in the situation of epistemic uncertainty. In this sense, the probability distribution of Δx_i describes the epistemic uncertainty.

– On the other hand, where does this distribution of Δx_i come from? It usually comes from the process of *calibration* of the measuring instrument. For example, to find this distribution, we repeatedly measure some quantity by using this instrument and by using a *standard* measuring instrument (whose measurement error is negligible). We then view the differences $\Delta x_i \stackrel{\text{def}}{=} \widetilde{x}_i - x_i$ between the results of these two measurements as the sample values from the desired distribution. From this viewpoint, this distribution is *aleatoric* because it represents variability of measurement errors produced by the used measuring instrument.

  Similarly, in the case of the upper bound Δ on the measurement error, the value Δ is a characteristic of an epistemic uncertainty, but this value comes from the variability of the measurement errors produced by the given measuring instrument and from the variability of different measuring instruments from the same manufacturer.

1.2. Statistical Methods for Processing Uncertainty

Traditionally, statistical methods are used to process uncertainty. Traditionally, in engineering, statistical methods are used, methods assuming that we know the probability distribution of different uncertain parameters.

Sensitivity analysis: main idea, limitations. Usually, we can safely linearize the dependence of the desired quantities y (e.g., stress at different structural points) on the uncertain parameters x_i – thus enabling sensitivity analysis; see, e.g., [13, 25, 26, 27]. The main idea behind the use of sensitivity analysis in processing uncertainty is that for each of n uncertain parameters x_i, we determine the rate with which the desired quantity changes with the change in x_i, and then we use these rates to estimate the resulting change; see Section 3 for a more detailed description.

What are the limitations of sensitivity analysis? Due to the very nature of this method, a limitation is that the dependence of y on x_i may be highly non-linear. In this case, even when we know how exactly y changes when we change just one of parameters x_i, it is difficult to predict how y will change if we change the values of all uncertain parameters x_i.

This limitation is crucial in many *scientific* applications. In these applications, we often know the values of the unknown parameters with such a low accuracy, that within the corresponding range, we cannot ignore quadratic and higher order terms in the dependence of y on x_i. Thus, linear approximation does not work well and the results of sensitivity analysis are not correct.

In *engineering* applications, usually, the dependence can be safely linearized. In this case, the sensitivity analysis method always produces correct results. However, it may take too long to produce these results.

Indeed, to use sensitivity analysis, we must call the model as many times as there are uncertain parameters. Often, the number n of uncertain parameters is huge – e.g., in ultrasonic testing, we record (= measure) signal values at thousands moments of time. To use sensitivity analysis, we must call the model n times. Thus, if we denote by T_f the time needed to call a model, then to use sensitivity analysis, we need computation time $n \cdot T_f$. If the model is complex, i.e., if n is large and T_f is large, this leads to a lot of computation time.

Monte-Carlo simulations: another way of handling probabilistic uncertainty. To speed up the processing of uncertainty, we can use Monte-Carlo simulations. Their main advantage is that for Monte-Carlo techniques, the required number of calls to a model depends only on the desired accuracy ε and not on n. As we will see, usually, we need c/ε^2 calls, so the overall computation time of a Monte-Carlo method is $(c/\varepsilon^2) \cdot T_f$. Hence, for large n, when $n \gg c/\varepsilon^2$, the Monte-Carlo methods are much faster than methods of sensitivity analysis.

1.3. Interval Methods for Processing Uncertainty

Sometimes, we only know the parameters with interval uncertainty. The above methods are applicable when we know the exact probability distribution of the corresponding parameters. In real life, we often do not know the exact probability distribution of the parameters.

For example, in many practical situations, we do not know the probability of different values of the measurement errors $\Delta x_i \stackrel{\text{def}}{=} \widetilde{x}_i - x_i$, we only know the upper bound Δ_i on the (absolute value of) the measurement error. In this case, the only information we have about the actual (unknown) value x_i is that x_i belongs to the interval $[\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i]$.

In case of aleatoric uncertainty, often, we also do not know, e.g., the distribution of user loads. In such cases, usually, all we know is the *intervals* of possible values of the corresponding parameters: e.g., we know that the load l_i is in $[0, L_i]$.

Comment. For aleatoric uncertainty, even if we knew the corresponding probability distributions, it would be a disaster to, e.g., design a building that is stable against random loads, but could fall down with a rare (but allowable) combination of loads.

Known techniques for processing interval uncertainty. In case of interval uncertainty, we can also use sensitivity analysis; see details in Section 3. The limitation to this technique is that it requires too much computation time for large n.

An alternative idea is to use interval techniques; see, e.g., [16, 17, 18, 20] and references therein; however, for large n, these techniques also take too long.

To speed up, we developed a new Monte-Carlo-type technique for processing interval uncertainty [13, 27].

1.4. What We Do in This Paper: Roadmap

In this paper, we:

- describe a new Monte-Carlo type techniques for processing interval uncertainty,
- discuss potential applications of this technique to engineering problems,
- describe the limitations of this technique, and
- explain how these limitations can be overcome.

We start with a detailed formulation of the problem of processing different types of uncertainty (Section 2). In Section 3, we describe sensitivity analysis, a possible way to solve the problem of processing probabilistic and interval uncertainty. In this section, we also describe the main limitation of sensitivity analysis – that it often requires too much computation time. To overcome this limitation, for *probabilistic* uncertainty, we can use Monte-Carlo techniques. In Section 4, we show how these techniques can be modified to handle *interval* uncertainty as well. The corresponding Cauchy deviates technique is based on the following assumptions:

- that the measurement errors are small, so we can safely linearize the problem;
- that we only have interval information about the uncertainty, and
- that we can actually call the program f the required number of times (≈ 200).

In real-life engineering problems, these assumptions may not be satisfied. In the following sections, we describe how we can modify the Cauchy deviate technique so as to overcome these limitations:

- in Section 5, we describe how to modify the Cauchy deviate technique if we cannot perform many iterations;
- in Sections 6 and 7, we show how this technique can be extended to different types of uncertainty: namely, to Dempster-Shafer knowledge bases and p-boxes (interval-valued cumulative distribution functions);
- finally, in Section 8, we show how the Cauchy deviates method can be extended to strongly non-linear data processing functions.

Comment. For readers' convenience, the list of notations used in this paper is given in the Appendix.

2. Formulation of the Problem

2.1. Case of Epistemic Uncertainty

In many real-life situations, we are interested in the value of a quantity y that is difficult (or even impossible) to measure directly. In these cases, a natural idea is to measure easier-to-measure quantities $x_1, \ldots, x_n$ that are related to the desired quantity y, and try to estimate y based on the results $\widetilde{x}_1, \ldots, \widetilde{x}_n$ of these measurements. To be able to produce such an estimate, we need to have an algorithm $f(x_1, \ldots, x_n)$ that, based on the values $x_1, \ldots, x_n$ of the directly measured quantities, reconstructs the value y of the desired quantity as $y = f(x_1, \ldots, x_n)$. Once we have such an algorithm, we plug in the measured values $\widetilde{x}_i$ of the quantities x_i into this algorithm f, and get the following estimate for y: $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$.

Measurements are never 100% accurate; as a result, the actual (unknown) values x_i of the measured quantities may somewhat differ from the measured values $\widetilde{x}_i$. In other words, we know the inputs to the algorithm f only with some (measurement-related) uncertainty. Because of this input uncertainty $\widetilde{x}_i \neq x_i$, our estimate $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$ is, in general, different from the actual (unknown) value $y = f(x_1, \ldots, x_n)$ of the desired quantity. In other words, uncertainty $\Delta x_i = \widetilde{x}_i - x_i$ in the inputs leads to the uncertainty $\Delta y = \widetilde{y} - y$ in the output as well.

It is therefore desirable to "estimate" this output uncertainty, i.e., to use the known information about the (unknown) values Δx_i to deduce as much information as possible about Δy.

Comment. In some practical situations, we only know an approximate dependence $y \approx \widetilde{f}(x_1, \ldots, x_n)$ between the actual values of the quantities x_i and y. In such situations, even if we knew the actual values of x_i, we still would not be able to reconstruct the exact value of y – there is an additional source of uncertainty caused by the difference between the actual (unknown) dependence $f(x_1, \ldots, x_n)$ and the known approximate dependence $\widetilde{f}(x_1, \ldots, x_n)$.

In this paper, we only consider the case where we know the exact dependence; the additional error bound caused by the difference between f and $\widetilde{f}$ can be usually simply added to the error bound caused by the uncertainty of the input data.

2.2. Case of Aleatoric Uncertainty

A similar problem occurs in the case of aleatoric uncertainty. In this case, we are interested in a certain characteristic y of the engineering design. We know how the desired quantity y depends on the parameters $x_1, \ldots, x_n$ which describe the design, i.e., we know the algorithm $y = f(x_1, \ldots, x_n)$ that, based on the values of these parameters, predicts the value y of the desired characteristic as $y = f(x_1, \ldots, x_n)$.

We also know the approximate values $\widetilde{x}_i$ of the parameters x_i: e.g., nominal values for the parameters which describe the actual design (such as the length of a certain beam) and the approximate value of the parameters which describe the medium (e.g., the temperature).

We can plug in the approximate values $\widetilde{x}_i$ of the parameters x_i into the known algorithm f, and get the following estimate for y: $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$.

Since the actual (unknown) values x_i of the parameters may somewhat differ from the approximate values $\widetilde{x}_i$, our estimate $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$ is, in general, different from the actual (unknown) value $y = f(x_1, \ldots, x_n)$ of the desired characteristic y. So, here too, uncertainty in the inputs leads to the uncertainty in the output as well. It is therefore desirable to estimate this output uncertainty.

2.3. Formulation of the Practical Problem: General Case

In both cases, we arrive at the following problem:

- *We know:*
  - the algorithm $f(x_1, \ldots, x_n)$;
  - the approximate values $\widetilde{x}_1, \ldots, \widetilde{x}_n$; and
  - some information about the (unknown) uncertainty Δx_i of each input.
- *We must:* provide information about the uncertainty Δy of the algorithm's output.

Comment. It is worth repeating that we *do not* know the actual value x_i of the i-th input and we therefore, do not know the actual value of the difference $\Delta x_i = \widetilde{x}_i - x_i$. However, since there is a relation $y = f(x_1, \ldots, x_n)$ between the (unknown) actual values of the inputs x_i and the (unknown) actual value of the desired characteristic y, any partial information that we have about x_i – e.g., the bounds on Δx_i – can be transformed into the (partial) information about y. The main objective of this paper is to describe methods that transform partial information about x_i into the (partial) information about y.

In order to solve this problem, we must know what are the possible types of information that we can have about the uncertainty of each input x_i.

2.4. Types of Uncertainty

We do not know the exact values of the inputs x_i. For example, in the case of measurements, we do not know the exact values of the measurement errors Δx_i; thus, while we know the measurement result $\widetilde{x}_i$, we do not know the actual value x_i of the corresponding quantity.

As a result, in real life, we may have (and often we do have) several situations in which we get exactly the same (known) approximate values $\widetilde{x}_1, \ldots, \widetilde{x}_n$, but the (unknown) actual values $x_1, \ldots, x_n$ of the inputs are different. Thus, to describe the uncertainty, we need to have some information about x_i, namely:

- some information about which values of x_i are possible and which are not, and
- (if possible) some information about the frequencies with which different values can occur:
  - in case of aleatoric uncertainty, some information about how often different possible values occur,
  - in case of epistemic uncertainty, how often different deviations between the (known) approximate value and the (unknown) actual value occur.

We usually know the interval of possible values. In most real-life situations, we do not know the *exact* value of x_i, but we usually know the *set* of possible values of x_i, i.e., the set which is guaranteed to contain the (unknown) actual value x_i. For example, the manufacturer of the measuring instrument usually provides us with the upper bound Δ_i on the possible values of the measurement error $|\Delta x_i|$, where $\Delta x_i = \widetilde{x}_i - x_i$. In this case, after we get the measurement result $\widetilde{x}_i$, we can conclude that the actual (unknown) value of x_i is guaranteed to be in the interval $[\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i]$. This interval is thus the set of possible values of x_i.

On the other hand, if we do not know any upper bound on $|\Delta x_i|$, this means that the difference between the actual (unknown) value x_i and the result $\widetilde{x}_i$ of the measurement can be arbitrarily large. For example, we may get the measurement result 1.2, but the actual value can be in the millions or can be in the negative billions. This situation is possible, but in this case, engineers and scientists would not call the value $\widetilde{x}_i$ a measurement result, it is, at best, an expert estimate.

Traditional data processing techniques assume that we also know probabilities. Most traditional methods of processing uncertainty in science and engineering (see, e.g., [29]) are based on the assumption that we know the exact frequency (probability) of all possible combinations $(x_1, \ldots, x_n)$. In other words, we know the exact probability distribution of the set of all n-dimensional vectors $x = (x_1, \ldots, x_n)$.

- In the case of aleatoric uncertainty, it is usually assumed that we know the probability distribution of the values x.

- In the case of epistemic uncertainty, usually, what we assume is a probability distribution on the set of all n-dimensional vectors $\Delta x = (\Delta x_1, \ldots, \Delta x_n)$, i.e., the distribution of the measurement errors. Since we know the measurement results $\widetilde{x}_1, \ldots \widetilde{x}_n$, we can take into account that $x_i = \widetilde{x}_i - \Delta x_i$ and thus find the corresponding probability distribution of the set of all possible values of $x_1, \ldots, x_n$.

Often, the measurement errors corresponding to different measurements are independent, so it is sufficient to know the distribution of each variable x_i. Each of these distributions can be described, e.g., by a cumulative distribution function (cdf) $F_i(t) \stackrel{\text{def}}{=} \text{Prob}(x_i \le t)$.

Comment. In many real-life situations, there is correlation between the variables. However, while considerable work has gone into developing practicable methods for incorporating correlations between variables (see, e.g., [11, 12]), in engineering practice, most frequently, the variables are usually assumed to be independent.

Case of interval uncertainty. In real life, we often do not have the complete information about the probabilities of different possible values x_i. In some real-life situations, we do not have any information about the frequency of different values x_i or about the frequency of different measurement errors $\Delta x_i = \widetilde{x}_i - x_i$. In situation, all we know is the corresponding range:

- In the case of aleatoric uncertainty, we know the range $[\underline{x}_i, \overline{x}_i]$ of possible values of x_i.
- In case of epistemic uncertainty, we know the range $[\Delta_i^-, \Delta_i^+]$ of possible values of the measurement error (or, in case of expert estimates, the range of the possible expert estimation error $\Delta x_i = \widetilde{x}_i - x_i$). In this case, the only information that we have about the actual value $x_i = \widetilde{x}_i - \Delta x_i$ of i-th quantity is that x_i must be in the interval $[\underline{x}_i, \overline{x}_i]$, where we denoted $\underline{x}_i \stackrel{\text{def}}{=} \widetilde{x}_i - \Delta_i^+$ and $\overline{x}_i \stackrel{\text{def}}{=} \widetilde{x}_i - \Delta_i^-$.

The corresponding uncertainty is called *interval uncertainty*; see, e.g., [16, 17, 18, 20].

Additional cases. So far, we have described two extreme situations:

- the case where we have a complete information on which values x_i (or Δx_i) are possible, and what the frequencies of different possible values are; in this text, we call this case *probabilistic uncertainty*;

– the case where we only know the range of possible values of x_i (or Δx_i), and we do not have any information about the frequencies at all; we call this case *interval uncertainty*.

In many real-life cases, we have an intermediate situation: we have some (partial) information about the frequencies (probabilities) of different values of x_i (or Δx_i), but we do not have the complete information about these frequencies.

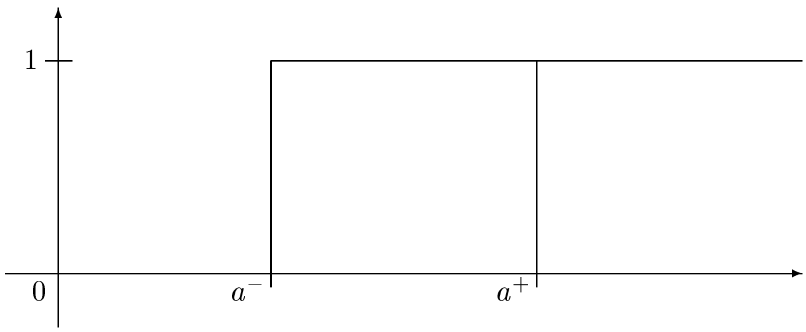
p-boxes. To describe the complete information about the probabilities of different values of x_i (or Δx_i), we must describe, for every real number t, the value $F_i(t)$ of the corresponding cdf. Thus, when we have a partial information about these probabilities, it means that, instead of the exact value $F_i(t)$, we only have the range $[\underline{F}_i(t), \overline{F}_i(t)]$ of possible values of $F_i(t)$. Thus, to describe such an intermediate situation, we must describe the *bounds* $\underline{F}_i(t)$ and $\overline{F}_i(t)$ for the cdf. These bounds are called *probability boxes* (or *p-boxes*, for short) [4].

Both probability distributions and intervals can be described as a particular case of p-boxes:

– a probability distribution $F_i(t)$ can be described as a degenerate p-box $[F_i(t), F_i(t)]$; and

– an interval $[a^-, a^+]$ can be described as a p-box $[\underline{F}_i(t), \overline{F}_i(t)]$ in which:

  • $\underline{F}_i(t) = 0$ for $t < a^+$ and $\underline{F}_i(t) = 1$ for $t \geq a^+$;
  • $\overline{F}_i(t) = 0$ for $t < a^-$ and $\overline{F}_i(t) = 1$ for $t \geq a^-$.



So, p-boxes are a general way of representing both probabilistic and interval uncertainty.

Comment. Of course, intervals and probabilities are not the only tools for modelling uncertainty. For example, we may have linguistic information also, e.g., "x_1 is low". The extension of the current techniques to such information is one of the important directions of future work.

Dempster-Shafer approach. Another way to describe partial information about the uncertainty is by using the Dempster-Shafer approach. In this approach, for each variable x_i, instead of a single interval $[\underline{x}_i, \overline{x}_i]$, we have several ($N_i$) intervals $[\underline{x}_i^{(1)}, \overline{x}_i^{(1)}]$, ..., $[\underline{x}_i^{(N_i)}, \overline{x}_i^{(N_i)}]$ (called *focal elements*) with *masses* (also known as *Basic Probability Assignments* (BPAs) $p_i^{(s)}$, $s = 1, \ldots, N_i$, attached to each such interval (so that for every i, $p_i^{(1)} + p_i^{(2)} + \ldots + p_i^{(N_i)} = 1$).

For example, we may have several (N_e) experts who provide us with intervals. Some experts provide the same interval, some experts provide different intervals. Here, $[\underline{x}_i^{(s)}, \overline{x}_i^{(s)}]$, $1 \le s \le n_e < N_e$, are different intervals, and the mass $p_i^{(s)}$ attached to s-th interval can be evaluated, e.g., as the proportion of experts who provide us with this interval, i.e., as the ratio $N_e^{(s)}/N_e$, where $N_e^{(s)}$ is the overall number of experts whose interval estimate is exactly $[\underline{x}_i^{(s)}, \overline{x}_i^{(s)}]$.

The collection of intervals with masses attached to different intervals constitutes a *D-S body of evidence*, also called *D-S knowledge base*.

The implied interpretation of a D-S knowledge base is that in the portion $p_i^{(s)}$ of the cases, the actual (unknown) value of the estimated quantity x_i belongs to the s-th interval $[\underline{x}_i^{(s)}, \overline{x}_i^{(s)}]$.

Comment. To avoid confusion, it is worth mentioning that the *mass* $p_i^{(s)}$ associated with the expert's interval may be different from the *probability* that the s-th expert is right. Let us consider an example in which we have two experts ($N = 2$), the first expert provides an interval $[0, 1]$, and the second expert provides a wider interval $[0, 2]$. In this case, the mass attached to the wider interval $[0, 2]$ is exactly $1/2$. On the other hand, for such situation, the implied interpretation of the D-S knowledge base is that:

- in half of such situations, the actual (unknown) value x_i is in the interval $[0, 1]$, and
- in the other half of the situations, the only information that we have about the unknown value x_i is that it is somewhere in the interval $[0, 2]$.

Since $[0, 1] \subseteq [0, 2]$, in all the cases, $x_i \in [0, 2]$, so the probability that the second expert is right is 1 (hence different from $1/2$).

Summary of possible cases. Thus, depending on the information that we have about the uncertainty in x_i, we can have five different formulations of the above problem:

- we know the probability distribution $F_i(t)$ for each variable x_i, we know that these distributions are independent, and we must find the distribution $F(t)$ for $y = f(x_1, \ldots, x_n)$;

- we know the interval $[\underline{x}_i, \overline{x}_i]$ of possible values of each variable x_i, and we must find the interval $[\underline{y}, \overline{y}]$ of possible values of y;

- we know the p-boxes $[\underline{F}_i(t), \overline{F}_i(t)]$ that characterize the distribution of each variable x_i, we know that the corresponding distributions are independent, and we must find the p-box $[\underline{F}(t), \overline{F}(t)]$ that describe the variable y;

- for each input i, we have the D-S body of evidence consisting of the focal elements $[\underline{x}_i^{(s)}, \overline{x}_i^{(s)}]$, $1 \le s \le N_i$, and their masses $p_i^{(s)}$:

  $$\langle[\underline{x}_i^{(1)}, \overline{x}_i^{(1)}], p_i^{(1)}\rangle, \langle[\underline{x}_i^{(2)}, \overline{x}_i^{(2)}], p_i^{(2)}\rangle, \ldots$$

  that characterize the distribution of each variable x_i, we know that the corresponding distributions are independent, and we must find the D-S knowledge base that describe the variable y;

- we may also have different types of uncertainty for different variables x_i: e.g., we may have probabilistic uncertainty or x_1 and interval uncertainty for x_2.

It is also important to consider the formulations in which the corresponding distributions may be dependent.

2.5. Known Methods for Processing Uncertainty, with an Emphasis on Interval-Type Methods

General situation. There exist numerous methods that efficiently solve several important practical classes of such problems; see, e.g., [4] and references therein (in particular, for interval uncertainty, see [20, 17, 18, 16]). In spite of these successes, there are still many open problems: for example, there is no presently known efficient method that would allow us to propagate a non-probabilistic uncertainty through a generic computationally demanding model, e.g., practice-motivated models that involve the numerical solution of large systems of nonlinear partial differential equations.

Main idea behind interval-type methods. Many of the existing methods are based on the fact that we know the algorithm f; so, instead of applying this algorithm step-by-step to the measured values $\widetilde{x}_1, \ldots, \widetilde{x}_n$, we apply this same algorithm step-by-step to the corresponding "uncertain numbers": probability distributions, intervals, and/or p-boxes.

Example: interval uncertainty. For example, in the case of interval uncertainty, for arithmetic operations $f(x_1, x_2)$, we have explicit formulas for the interval range of $f(x_1, x_2)$ when x_1 and $x - 2$ belong to the corresponding intervals. Namely, when $x_1 \in \mathbf{x}_1 = [\underline{x}_1, \overline{x}_1]$ and $x_2 \in \mathbf{x}_2 = [\underline{x}_2, \overline{x}_2]$, then:

- The range $\mathbf{x}_1 + \mathbf{x}_2$ for $x_1 + x_2$ is $[\underline{x}_1 + \underline{x}_2, \overline{x}_1 + \overline{x}_2]$.
- The range $\mathbf{x}_1 - \mathbf{x}_2$ for $x_1 - x_2$ is $[\underline{x}_1 - \overline{x}_2, \overline{x}_1 - \underline{x}_2]$.
- The range $\mathbf{x}_1 \cdot \mathbf{x}_2$ for $x_1 \cdot x_2$ is $[\underline{y}, \overline{y}]$, where

$$\underline{y} = \min(\underline{x}_1 \cdot \underline{x}_2, \underline{x}_1 \cdot \overline{x}_2, \overline{x}_1 \cdot \underline{x}_2, \overline{x}_1 \cdot \overline{x}_2);$$

$$\overline{y} = \max(\underline{x}_1 \cdot \underline{x}_2, \underline{x}_1 \cdot \overline{x}_2, \overline{x}_1 \cdot \underline{x}_2, \overline{x}_1 \cdot \overline{x}_2).$$

The range $1/\mathbf{x}_1$ for $1/x_1$ is $[1/\overline{x}_1, 1/\underline{x}_1]$ (if $0 \notin \mathbf{x}_1$).

We can apply these *interval arithmetic* operations step-by-step to find enclosures for the ranges of more general functions.

Interval uncertainty: numerical example. Let us illustrate this idea on the example of estimating the range of the function $f(x_1) = (x_1 - 2) \cdot (x_1 + 2)$ when $x_1 \in [1, 2]$. Based on this expression, the compiler will produce the following algorithm for computing the value of $y = f(x_1)$ (here r_i denote intermediate computation results):

- $\mathrm{r}_1 := x_1 - 2$;
- $\mathrm{r}_2 := x_1 + 2$;
- $y := \mathrm{r}_1 \cdot \mathrm{r}_2$.

To get an interval range, we perform the same operations (in the same order), but with *intervals* instead of *numbers*:

- $\mathbf{r}_1 := [1, 2] - [2, 2] = [-1, 0]$;
- $\mathbf{r}_2 := [1, 2] + [2, 2] = [3, 4]$;
- $\mathbf{y} := [-1, 0] \cdot [3, 4] = [-4, 0]$.

Comments.

- It is worth mentioning that here, the actual range is $f(\mathbf{x}_1) = [-3, 0]$, so our estimate is indeed a valid enclosure for this range: $[-4, 0] \supseteq [-3, 0]$.
- This is just a toy example, there are more efficient ways of computing enclosures.

Limitations. Such methods, that rely on tracking uncertainty through individual algebraic operations, work well for simple models but they are not practically workable for general computationally demanding models. Moreover, in several practical situations, we cannot use such methods at all because the algorithm is given as a *black box*: we do not know the sequence of steps forming this algorithm; we can only plug in different values into this algorithm and see the results. This situation is reasonably frequent:

- with commercial software, where the software's owners try to prevent competitors from using their algorithms, and
- with classified security-related software, where efficient security-related algorithms are kept classified to prevent the adversary from using them.

In some practical cases, the situation is made even more difficult by the fact that the software $f(x_1, \ldots, x_n)$ is so complex and requires so much time to run that it is only possible to run it a few times. This complex black-box situation is what we will analyze in this text.

Comment. Even for a black-box function, it may be possible to run more simulations if we do the following:

- first, we use the actual black-box function $f(x_1, \ldots, x_n)$ to provide an approximating easier-to-compute "meta"-model $\widetilde{f}(x_1, \ldots, x_n) \approx f(x_1, \ldots, x_n)$, and
- then, we use this approximate "meta"-model to estimate the uncertainty of the results.

So, if our preliminary computations show that we need more simulations that the black-box function can give us, it does not necessarily mean that the corresponding uncertainty estimation method cannot be applied to our case: we may still be able to apply it to the approximate function $\widetilde{f}$.

It is worth mentioning that while the development of an approximate (e.g., *response surface*) model is sometimes possible, many real-life situations involve dozens and hundreds of inputs x_i (corresponding to values of different quantities at different locations at different moments of time). In such situations, it is often not easy – and even impracticable – to develop an approximate model.

3. Sensitivity Analysis: A Possible Way to Solve the Problem of Processing Probabilistic and Interval Uncertainty

3.1. Linearization

Possibility of linearization. As we have mentioned, in many practical situations, the dependence $y = f(x_1, \ldots, x_n)$ of the desired quantities y on the uncertain parameters x_i is reasonably smooth, and the uncertainty in x_i is relatively small. In such cases, we can safely linearize the dependence of y on x_i.

Specifically, the smallness of uncertainty means that for each i, we know the approximate value $x_i^{\rm approx}$ of i-th input x_i, and the difference $\delta x_i \stackrel{\rm def}{=} x_i - x_i^{\rm approx}$ between the the actual (unknown) value x_i and the (known) approximate value and of the i-th input quantity is small. Since the function $f(x_1, \ldots, x_n)$ is reasonably smooth, and the inputs $x_i = x^{\rm approx_i} + \delta x_i$ differ only slightly from the known value $x_i^{\rm approx}$, we can thus ignore quadratic and higher order terms in the expansion of f and approximate the function f, in the vicinity of the approximate values $(x_1^{\rm approx}, \ldots, x_n^{\rm approx})$, by its linear terms:

$$f(x_1, \ldots, x_n) = f(x_1^{\rm approx} + \delta x_1, \ldots, x_n^{\rm approx} + \delta x_n) \approx y^{\rm approx} + \delta y,$$

where $\delta y \stackrel{\rm def}{=} c_1 \cdot \delta x_1 + \ldots + c_n \cdot \delta x_n$, $y^{\rm approx} \stackrel{\rm def}{=} f(x_1^{\rm approx}, \ldots, x_n^{\rm approx})$, and $c_i \stackrel{\rm def}{=} \dfrac{\partial f}{\partial x_i}$.

Linearization makes it possible to use special *sensitivity analysis* techniques to estimate the uncertainty of y; see, e.g., [13, 25, 26, 27]. Let us show how this can be done on the examples of probabilistic and interval uncertainty.

3.2. Sensitivity Analysis for Probabilistic Uncertainty

Probabilistic uncertainty: case of independent measurement errors. In many engineering applications, it is often assumed:

– that the systematic errors have been corrected, i.e., the mean value of each measurement error $\Delta x_i = \widetilde{x}_i - x_i$ is 0;

– that we know the standard deviation σ_i of each measurement error, and distributed with 0 mean and

– that the measurement errors Δx_i and Δx_j corresponding to different measurements $i \neq j$ are statistically independent.

(It is also often assumed that each measurement errors is normally distributed.)

In this case, we can take the measurement results $\widetilde{x}_i$ as the desired approximate values x_i^{approx} of the inputs. The differences $\delta x_i = x_i - x_i^{\text{approx}}$ thus take the form $\delta x_i = -\Delta x_i$, and similarly, $\delta y = -\Delta y$. The desired quantity $\delta y = c_1 \cdot \delta x_1 + \ldots + c_n \cdot \delta x_n$ is a linear combination of n independent random variables with 0 mean and known standard deviations. Thus, the mean value of the variable δy is also equal to 0, and the standard deviation σ of δy is equal to

$$\sigma = \sqrt{c_1^2 \cdot \sigma_1^2 + \ldots + c_n^2 \cdot \sigma_n^2}.$$

Probabilistic case: towards sensitivity analysis. We know the standard deviations σ_i, but we often do not know the values c_i of the partial derivatives, especially when the function $f(x_1, \ldots, x_n)$ is given as a black box.

To estimate the values of the partial derivatives, we can use *numerical differentiation.* By definition, a partial derivative is a limit

$$\frac{\partial f}{\partial x_i} = \lim_{h_i \to 0} \frac{f(\widetilde{x}_1, \ldots, \widetilde{x}_{i-1}, \widetilde{x}_i + h_i, \widetilde{x}_{i+1}, \ldots, \widetilde{x}_n) - \widetilde{y}}{h_i},$$

where $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_{i-1}, \widetilde{x}_i, \widetilde{x}_{i+1}, \ldots, \widetilde{x}_n)$. This means that when the value h_i is small, the ratio in the right-hand side is close to the desired derivative – and that the smaller h_i we take, the closer is the resulting ratio to the actual partial derivative.

In the linear approximation, the ratio is exactly equal to the desired derivative, so we can take any value h_i for which we can still ignore the quadratic terms in the expansion of f. Thus, we can estimate c_i as $c_i = \dfrac{\Delta y^{(i)}}{h_i}$, where

$$\Delta y^{(i)} \stackrel{\text{def}}{=} f(\widetilde{x}_1, \ldots, \widetilde{x}_{i-1}, \widetilde{x}_i + h_i, \widetilde{x}_{i+1}, \ldots, \widetilde{x}_n) - \widetilde{y}.$$

For these values of c_i, the expression for c takes the form

$$\sigma = \sqrt{\frac{(\Delta y^{(1)})^2}{h_1^2} \cdot \sigma_1^2 + \ldots + \frac{(\Delta y^{(n)})^2}{h_n^2} \cdot \sigma_n^2}.$$

For all choices of h_i, we need n additional calls to f (to compute the values $\Delta y^{(i)}$, $1 \le i \le n$). After these calls, the smallest number of arithmetic operations is when we take $h_i = \sigma_i$, because for this choice of h_i, there is no need to multiply by c_i and divide by h_i.

Probabilistic case: algorithm of sensitivity analysis. Thus, we are arrive at the following algorithm:

- First, we apply f to the approximate values $\widetilde{x}_1, \ldots, \widetilde{x}_n$, resulting in the value $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$.
- Then, for each of n inputs x_i, we modify this input to $x'_i = \widetilde{x}_i + \sigma_i$ and, leaving other inputs intact, apply f again, resulting in $y^{(i)} \stackrel{\text{def}}{=} f(\widetilde{x}_1, \ldots, \widetilde{x}_i, x'_i, \widetilde{x}_{i+1}, \ldots, \widetilde{x}_n)$.
- Finally, we get the desired value as follows:

$$\sigma = \sqrt{(y^{(1)} - \widetilde{y})^2 + \ldots + (y^{(n)} - \widetilde{y})^2}.$$

3.3. Interval Case: Sensitivity Analysis

Interval case: towards sensitivity analysis. When the function $f(x_1, \ldots, x_n)$ is reasonably smooth and the box $[\underline{x}_1, \overline{x}_1] \times \ldots \times [\underline{x}_n, \overline{x}_n]$ is reasonably small, then on this box, we can reasonably approximate the function f by its linear terms. Let us denote $x_i^{\text{mid}} \stackrel{\text{def}}{=} (\underline{x}_i + \overline{x}_i)/2$ and $\delta x_i \stackrel{\text{def}}{=} x_i - x_i^{\text{mid}}$. Then, linearization leads to the following formula:

$$f(x_1^{\text{mid}} + \delta x_1, \ldots, x_n^{\text{mid}} + \delta x_n) \approx y^{\text{mid}} + \delta y,$$

where $\delta y \stackrel{\text{def}}{=} c_1 \cdot \delta x_1 + \ldots + c_n \cdot \delta x_n$, $y^{\text{mid}} \stackrel{\text{def}}{=} f(x_1^{\text{mid}}, \ldots, x_n^{\text{mid}})$, and $c_i \stackrel{\text{def}}{=} \dfrac{\partial f}{\partial x_i}$. One can easily show that when each of the variables δx_i takes possible values from the interval $[-\Delta_i, \Delta_i]$, then the largest possible value of the linear combination δy is

$$\Delta = |c_1| \cdot \Delta_1 + \ldots + |c_n| \cdot \Delta_n, \tag{1}$$

and the smallest possible value of δy is $-\Delta$. Thus, in this approximation, the interval of possible values of δy is $[-\Delta, \Delta]$, and the desired interval of possible values of y is $[y^{\text{mid}} - \Delta, y^{\text{mid}} + \Delta]$.

Similarly to the probabilistic case, we can estimate c_i by using numerical differentiation and thus get the following expression for Δ:

$$\Delta = \frac{|\Delta y^{(1)}|}{h_1} \cdot \Delta_1 + \ldots + \frac{|\Delta y^{(n)}|}{h_n} \cdot \Delta_n.$$

Again, similarly to the probabilistic case, to minimize the number of computational steps, we can take $h_i = \Delta_i$.

Interval case: algorithm of sensitivity analysis. Thus, we are arrive at the following algorithm:

– First, we apply f to the midpoints $x_1^{\rm mid}, \ldots, x_n^{\rm mid}$ of input intervals, resulting in the value $y^{\rm mid} = f(x_1^{\rm mid}, \ldots, x_n^{\rm mid})$.

– Then, for each of n inputs x_i, we modify this input to $x'_i = x_i^{\rm mid} + \Delta_i$ and, leaving other inputs intact, apply f again, resulting in

$$y^{(i)} \stackrel{\rm def}{=} f(x_1^{\rm mid}, \ldots, x_{i-1}^{\rm mid}, x'_i, x_{i+1}^{\rm mid}, \ldots, x_n^{\rm mid}).$$

– Finally, we get the desired interval as $[y^{\rm mid} - \Delta, y^{\rm mid} + \Delta]$, where:

$$\Delta = |y^{(1)} - y^{\rm mid}| + \ldots + |y^{(n)} - y^{\rm mid}|.$$

Interval uncertainty: algorithm of sensitivity analysis for the more general case of a monotonic function f. For interval uncertainty, the above algorithm can be generalized to the case where the function f is not necessarily linearizable on the box $[\underline{x}_1, \overline{x}_1] \times \ldots \times [\underline{x}_n, \overline{x}_n]$, but we are sure that on this box, the data processing function $f(x_1, \ldots, x_n)$ is monotonic (increasing or decreasing) in each of its variables x_i. In this case, we can use the following natural algorithm:

– First, we apply f to the estimate $\widetilde{x}_1, \ldots, \widetilde{x}_n$ of the inputs, resulting in the value $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$.

– Then, for each of n inputs x_i, we modify this input to $x'_i \neq \widetilde{x}_i$ and, leaving other inputs, apply f again. By comparing the values $f(\widetilde{x}_1, \ldots, \widetilde{x}_i, x'_i, \widetilde{x}_{i+1}, \ldots, \widetilde{x}_n)$ and $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$, we decide whether f in increasing or decreasing in x_i.

– Finally, we apply f two more times to get the desired bounds for y as follows: $\underline{y} = f(x_1^-, \ldots, x_n^-)$ and $\overline{y} = f(x_1^+, \ldots, x_n^+)$, where:

  - for the variables x_i for which f increases with x_i, we take $x_i^- = \underline{x}_i$ and $x_i^+ = \overline{x}_i$, and
  - for the variables x_i for which f decreases with x_i, we take $x_i^- = \overline{x}_i$ and $x_i^+ = \underline{x}_i$.

Main limitation of sensitivity analysis. The main limitation of this method is that it requires n calls to the program f. Often, the number n of uncertain parameters is huge – e.g., in ultrasonic testing, we record (= measure) signal values at thousands moments of time. To use sensitivity analysis, we must call the model n times – and if the model f is complex, this leads to a lot of computation time.

The problem is even worse for situations mentioned earlier, situations that involve dozens and hundreds of inputs x_i, where it is often not easy – and even impracticable – to develop an approximate model; see, e.g., analysis reported in [2, 9].

It is therefore desirable to decrease the number of calls to the model f. Let us describe how we can do it.

4. From Traditional Monte-Carlo Techniques for Probabilistic Uncertainty to Monte-Carlo-Type Techniques for Interval Uncertainty: What Was Previously Known

4.1. Probabilistic Uncertainty: Monte-Carlo Techniques

Let us first consider the traditional case of the probabilistic uncertainty, when we know that the values Δx_i are distributed according to the cdf $F_i(t)$, and that the corresponding random variables Δx_i are independent. In this case, we are interested to know the distribution $F(t)$ of Δy.

In this traditional probabilistic case, a natural idea is to use Monte-Carlo simulations. Specifically, on each iteration k:

- for each input variable x_i, we simulate the values $x_i^{(k)}$ distributed according to the known distribution $F_i(t)$;
- then, we plug the simulated values $x_i^{(k)}$ the algorithm f, and thus get the value $y^{(k)} = f(x_1^{(k)}, \ldots, x_n^{(k)})$.

After N iterations, we get N values $y^{(k)}$.

Since the inputs $x_i^{(k)}$ are independently distributed according to the corresponding input distributions $F_i(t)$, the outputs $y^{(k)}$ are distributed according to the desired distribution $F(t)$. Thus, the N values $y^{(k)}$ are a sample from the unknown distribution $F(t)$. It is therefore necessary to extract information about $F(t)$ from this sample.

Comment. It is worth mentioning that Latin hypercube sampling is often used to improve the computational efficiency of the analysis; see, e.g., [6, 10, 19].

4.2. A Natural Idea: Monte-Carlo Method for Interval Uncertainty

In the case of probabilistic uncertainty (when we know the exact probability distribution for each input x_i), we can speed up computations by using a Monte-Carlo approach. Let us analyze whether a similar approach can work in interval case as well.

In the probabilistic case, we know the probability distribution for each input x_i, and this is exactly the distribution that we use to simulate the difference between the (unknown) actual value x_i and the (known) approximate value $\widetilde{x}_i$.

In the interval case, we do not know the exact probability distribution of x_i, we only know that this distribution is located on the interval $[x_i^{\rm mid} - \Delta_i, x_i^{\rm mid} + \Delta_i]$. In order to apply Monte-Carlo techniques, we must select a single distribution for each input x_i.

It is sufficient to be able to simulate the variable $\delta x_i = x_i - x_i^{\rm mid}$ which is located on the interval $[-\Delta_i, \Delta_i]$; then, the sum $x_i = x_i^{\rm mid} + \delta x_i$ will be located in the desired interval.

If a variable χ is located on the interval $[-1, 1]$, then for each $\Delta_i > 0$, the product $\Delta_i \cdot \chi$ is located on the interval $[-\Delta_i, \Delta_i]$. Thus, to produce a Monte-Carlo type method for interval uncertainty, it is sufficient to select a probability distribution d corresponding to the variable located in the interval $[-1, 1]$. Once this distribution is selected, we can do the following:

- Apply f to the midpoint values: $y^{\rm mid} := f(x_1^{\rm mid}, \ldots, x_n^{\rm mid})$;
- For $k = 1, 2, \ldots, N$, repeat the following:
  - use the random number generator to compute n numbers $r_i^{(k)}$, $i = 1, 2, \ldots, n$, that are distributed according to the distribution d;
  - compute the simulated approximation errors $\delta x_i^{(k)} := \Delta_i \cdot r_i^{(k)}$;
  - compute the simulated "actual values"
  $$x_i^{(k)} := x_i^{\rm mid} + \delta x_i^{(k)};$$
  - apply the program f to the simulated measurement results and compute the simulated error of the indirect measurement:
  $$\delta y^{(k)} := f\left(x_1^{(k)}, \ldots, x_n^{(k)}\right) - y^{\rm mid};$$
- Hopefully, the resulting sample $\delta y^{(1)}, \ldots, \delta y^{(N)}$ of simulated value δy will be distributed according to the distribution d multiplied by

Δ. Thus, we will be able to use statistical techniques to find the desired parameter Δ.

The only remaining question is how to select the sampling distribution d.

4.3. Monte-Carlo Method for Interval Uncertainty: From Idea to Implementation

First (seemingly natural) try: a distribution d located on the interval $[-1, 1]$. Since our objective is to provide a distribution that corresponds to a variable located on the interval $[-1, 1]$, a natural idea is to select a probability distribution d which is located on this interval (with probability 1).

However, as we will show, this seemingly natural choice does not allow us to compute the desired quantity (1) – even when the function function $f(x_1, \ldots, x_n)$ is reasonably smooth and the box $[\underline{x}_1, \overline{x}_1] \times \ldots \times [\underline{x}_n, \overline{x}_n]$ is reasonably small, so that on this box, we can reasonably approximate the function f by its linear terms $\delta y = \sum\limits_{i=1}^{n} c_i \cdot \delta x_i$.

Analysis of the first try. Due to our selection of $\delta x_i = \Delta \cdot r_i$, we have $\delta y = \sum\limits_{i=1}^{n} c_i \cdot \Delta_i \cdot r_i$, i.e., δy is a linear combination of n independent identically distributed random variables r_i. These variables are located on an interval $[-1, 1]$ and thus, have finite mean $E[r]$ and finite variance $V[r]$. Thus, for δy, we get $E[\delta y] = \left(\sum\limits_{i=1}^{n} c_i \cdot \Delta_i\right) \cdot E[r]$ and $V[\delta y] = \left(\sum\limits_{i=1}^{n} c_i^2 \cdot \Delta_i^2\right) \cdot V[r]$.

When n is large and all the values c_i are of the same order of magnitude, then the Central Limit Theorem applies, according to which δy is (approximately) normally distributed. A normal distribution is uniquely determined by its mean and variance; thus, the only information we can extract from observing the simulated values $c^{(k)}$ of δy are the sums $\sum c_i \cdot \Delta_i$ and $\sum c_i^2 \cdot \Delta_i^2$.

If this is the only information we know, we cannot reconstruct the desired sum $\sum |c_i| \cdot \Delta_i$.

Conclusion. So, by using a distribution d which is located on the interval $[-1, 1]$, we cannot estimate the interval uncertainty.

Cauchy deviates method. It turns out that it is possible to estimate the interval uncertainty if we use a distribution d which is *not* located on the interval $[-1,1]$ – namely, the *basic Cauchy* distribution with the probability density function $\rho(x) = \dfrac{1}{\pi\cdot(x^2+1)}$. The resulting Cauchy deviate method works in the linearized case – when the function $f(x_1,\ldots,x_n)$ is reasonably smooth and the box $[\underline{x}_1,\overline{x}_1]\times\ldots\times[\underline{x}_n,\overline{x}_n]$ is reasonably small, so that on this box, we can reasonably approximate the function f by its linear terms.

If we multiply a random variable distributed according to the above basic Cauchy distribution d by a value Δ, then we get a Cauchy distribution with a parameter Δ, i.e., a distribution described by the following density function: $\rho(x) = \dfrac{\Delta}{\pi\cdot(x^2+\Delta^2)}$. It is known that if $\xi_1,\ldots,\xi_n$ are independent variables distributed according to Cauchy distributions with parameters Δ_i, then, for every n real numbers $c_1,\ldots,c_n$, the corresponding linear combination $c_1\cdot\xi_1+\ldots+c_n\cdot\xi_n$ is also Cauchy distributed, with the parameter Δ described by the formula (1).

Thus, if for some number of iterations N, we simulate $\delta x_i^{(k)}$ ($1\le k\le N$) as Cauchy distributed with parameter Δ_i, then, in the linear approximation, the corresponding differences

$$\delta y^{(k)} \stackrel{\text{def}}{=} f(x_1^{\text{mid}}+\delta x_1^{(k)},\ldots,x_n^{\text{mid}}+\delta x_n^{(k)}) - y^{\text{mid}}$$

are distributed according to the Cauchy distribution with the parameter Δ. The resulting values $\delta y^{(1)},\ldots,\delta y^{(N)}$ are therefore a sample from the Cauchy distribution with the unknown parameter Δ. Based on this sample, we can estimate the value Δ.

Cauchy deviates method: technical details. In order to estimate Δ, we can apply the Maximum Likelihood Method which leads to the following equation:

$$\frac{1}{1+\left(\dfrac{\delta y^{(1)}}{\Delta}\right)^2}+\ldots+\frac{1}{1+\left(\dfrac{\delta y^{(N)}}{\Delta}\right)^2}=\frac{N}{2}.$$

The left-hand side of this equation is an increasing function that is equal to $0(<N/2)$ for $\Delta=0$ and $>N/2$ for $\Delta=\max\left|\delta y^{(k)}\right|$; therefore the solution to this equation can be found by applying a bisection method to the interval $\left[0,\max\left|\delta y^{(k)}\right|\right]$.

Simulation of Cauchy distribution with parameter Δ_i can be based on the functional transformation of uniformly distributed sample values: $\delta x_i^{(k)}=\Delta_i\cdot\tan(\pi\cdot(r_i-0.5))$, where r_i is uniformly distributed on the interval $[0,1]$.

Finding the number of iterations for given required accuracy. In [27, 13], we found the number of iterations N that would provide the desired (relative) accuracy ε in estimating Δ, i.e., the number of iterations that are needed to guarantee that $(1-\varepsilon)\cdot\widetilde{\Delta} \le \Delta \le (1+\varepsilon)\cdot\widetilde{\Delta}$ with a given certainty. In practice, it is reasonable to get a certainty 95% and accuracy $\varepsilon = 0.2$ (20%).

The desired number of iterations can be computed from the fact that the difference between the actual value Δ and its estimate $\widetilde{\Delta}$ is normally distributed, for large N, with 0 mean and standard deviation $\sigma_e = \Delta\cdot\sqrt{2/N}$. The certainty 95% corresponds to $2\sigma_e$, i.e., with this certainty, we can guarantee that $\widetilde{\Delta} - 2\sigma_e \le \Delta \le \widetilde{\Delta} + 2\sigma_e$. Thus, for a given $\varepsilon > 0$, to get an accuracy ε with 95% certainty, we must pick n for which $\varepsilon\cdot\widetilde{\Delta} = 2\sigma_e = 2\Delta\cdot\sqrt{2/N}$. Since $\widetilde{\Delta}\approx\Delta$, this means that $\varepsilon \approx 2\cdot\sqrt{2/N}$, hence $N = 8/\Delta^2$.

In particular, to get a 20% accuracy $(0.2\cdot\Delta)$ with 95% certainty, i.e., to guarantee that $0.8\cdot\widetilde{\Delta} \le \Delta \le 1.2\cdot\widetilde{\Delta}$, we need $N = 8/(0.2)^2 = 200$ runs.

After 200 runs, we can thus conclude that $0.8\cdot\widetilde{\Delta} \le \Delta \le 1.2\cdot\widetilde{\Delta}$ with certainty 95%.

In general, the required number of calls to a model depends only on the desired accuracy ε and not on n – so for large n, these methods are much faster.

Comment. It is important to mention that we assumed that the function f is reasonably linear within the box

$$[x_1^{\mathrm{mid}} - \Delta_1, x_1^{\mathrm{mid}} + \Delta_1] \times \ldots \times [x_n^{\mathrm{mid}} - \Delta_n, x_n^{\mathrm{mid}} + \Delta_n].$$

However, the simulated values δ_i may be outside the box. When we get such values, we do not use the function f for them, we use a normalized function that is equal to f within the box, and that is extended linearly for all other values (we will see, in the description of an algorithm, how this is done).

As a result, we arrive at the following algorithm [27, 13].

4.4. Cauchy Deviates Algorithm: Description

– Apply f to the midpoints: $y^{\mathrm{mid}} := f(x_1^{\mathrm{mid}}, \ldots, x_n^{\mathrm{mid}})$;

– For $k = 1, 2, \ldots, N$, repeat the following:

  - use the standard random number generator to compute n numbers $r_i^{(k)}$, $i = 1, 2, \ldots, n$, that are uniformly distributed on the interval $[0, 1]$;

- compute Cauchy distributed values $c_i^{(k)} := \tan(\pi \cdot (r_i^{(k)} - 0.5))$;
- compute the largest value of $|c_i^{(k)}|$ so that we will be able to normalize the simulated approximation errors and apply f to the values that are within the box of possible values: $K := \max_i |c_i^{(k)}|$;
- compute the simulated approximation errors

$$\delta x_i^{(k)} := \Delta_i \cdot c_i^{(k)} / K;$$

- compute the simulated "actual values"

$$x_i^{(k)} := x_i^{\text{mid}} + \delta x_i^{(k)};$$

- apply the program f to the simulated measurement results and compute the simulated approximation error for y:

$$\Delta y^{(k)} := K \cdot \left(f\left(x_1^{(k)}, \ldots, x_n^{(k)} \right) - y^{\text{mid}} \right);$$

– Compute Δ by applying the bisection method to solve the equation

$$\frac{1}{1 + \left(\frac{\Delta y^{(1)}}{\Delta} \right)^2} + \ldots + \frac{1}{1 + \left(\frac{\Delta y^{(N)}}{\Delta} \right)^2} = \frac{N}{2}.$$

4.5. Why Cauchy Distribution

As we have mentioned, if, instead of the Cauchy distribution, we use a different sampling distribution d, e.g., a distribution located on an interval $[-1, 1]$, then by running a similar Monte-Carlo simulations, we will not get the correct value Δ. Let us show that, moreover, Cauchy distribution is the *only* distribution d with the desired property.

Indeed, the above Monte-Carlo method is based on the fact that if $\xi_1, \ldots, \xi_n$ are independent variables distributed according to the distribution d, then, for every $2n$ real numbers $c_1, \ldots, c_n$, $\Delta_1, \ldots, \Delta_n$, the corresponding linear combination $c_1 \cdot \Delta_1 \cdot \xi_1 + \ldots + c_n \cdot \Delta_n \cdot \xi_n$ is also distributed as $\Delta \cdot \xi$, where the parameter $\Delta = |c_1| \cdot \Delta_1 + \ldots + |c_n| \cdot \Delta_n$ is described by the formula (1), and ξ is distributed according to the same distribution d.

It is sufficient to use this result for $n = 1, 2$ and $\Delta_1 = \ldots = \Delta_n = 1$, in which case $\Delta = |c_1|$ and $\Delta = |c_1| + |c_2|$. To prove the result, let us describe the desired property in terms of the characteristic function $\chi(\omega) \stackrel{\text{def}}{=} E[\exp(\mathrm{i} \cdot \omega \cdot \xi)] = \int \exp(\mathrm{i} \cdot \omega \cdot t) \cdot \rho(t)\, dt$ of the distribution d, where

$\rho(t)$ is the corresponding probability density function. By definition, $\chi(\omega)$ is the Fourier transform of $\rho(t)$; thus, if we know $\chi(\omega)$, we can uniquely reconstruct $\rho(t)$ by applying the inverse Fourier transform.

On the one hand, the characteristic function for the sum $\sum c_i \cdot \xi_i$ should be equal to $\chi(\Delta \cdot \omega)$. On the other hand, it is easy to check that the characteristic function of the sum of several independent variables is equal top the product of their characteristic functions. Thus, the characteristic function for the sum $\sum c_i \cdot \xi_i$ is equal to

$$\chi(c_1 \cdot \omega) \cdot \ldots \cdot \chi(c_n \cdot \omega).$$

For $n = 1$, the equality between these two expressions means that $\chi(c_1 \cdot \omega) = \chi(|c_1| \cdot \omega)$. For $\omega = 1$ and $c_1 < 0$, we conclude that $\chi(c_1) = \chi(|c_1|) = \chi(-c_1)$, i.e., that $\chi(\omega)$ is an even function.

For $n = 2$, $\omega = 1$, and $c_i \geq 0$, we have $\chi(c_1 + c_2) = \chi(c_1) \cdot \chi(c_2)$, i.e., that $\chi(\omega)$ is a *multiplicative* function; thus, its logarithm $L(\omega) \stackrel{\text{def}}{=} \ln(\chi(\omega))$ is an additive function: $L(c_1+c_2) = L(c_1)+L(c_2)$. It is known that every Lebesgue measurable additive function is linear, i.e., $L(c) = k \cdot c$, hence $\chi(\omega) = \exp(-k \cdot |\omega|)$. The inverse Fourier transform of this expression leads to a Cauchy distribution. Thus, we have proven that Cauchy distribution is the only one for which the above Monte-Carlo method produces correct values of Δ.

4.6. Applications: Brief Overview

We have applied the Cauchy deviate techniques to the several engineering examples. For example, petroleum and geotechnical engineering, we estimated the uncertainty of the solution to the inverse problem caused by the measurement errors [3]. In this example, $x_1, \ldots, x_n$ are the traveltimes of the seismic signals between the source and the sensor (and possibly other measurement results). The program $f(x_1, \ldots, x_n)$ solves the inverse problem, i.e., uses the traveltimes x_i to estimate the density y at different locations and at different depths. To be more accurate, the program reconstructs the speed of sound at different locations and at different depths, and then uses the known (approximate) relationship between the speed of sound and the density to reconstruct the desired density. As a result, the dependence of the accuracy on the location and depth fits much better with the geophysicists' understanding than the previous accuracy results obtained under the assumption that all the measurement errors are independent and normally distributed.

Another application is to environmental and power engineering: namely, to the safety analysis of complex systems [13]. In this example, $x_1, \ldots, x_n$ are the parameters of the system that are only known with interval uncertainty such as the thickness of the wall of the drum that contains radioactive waste. The program $f(x_1, \ldots, x_n)$ (usually given as a black box) describes how the desired parameters such as the radioactivity level at different places depend on x_i.

We also applied this technique to simplified building safety models (similar to the models considered in [21, 22] and references therein). In this example, $x_1, \ldots, x_n$ are the loads on a structure for each of which we only know the tolerance intervals, and the elastic parameters of this structure which are only known with interval uncertainty. The program $f(x_1, \ldots, x_n)$ (often commercial and thus, given as a black box) is a finite-element model that describes how the stresses in the corresponding structure (e.g., building) depend on x_i. By using the Cauchy deviates technique, we got the same results as sensitivity analysis, but much faster.

Other simplified application-related examples are given in [13].

4.7. Limitations of the Existing Cauchy Deviate Techniques

Cauchy deviate technique is based on the following assumptions:

- that the measurement errors are small, so we can safely linearize the problem;
- that we only have interval information about the uncertainty, and
- that we can actually call the program f 200 times.

In real-life engineering problems, these assumptions may not be satisfied. In the following sections, we describe how we can modify the Cauchy deviate technique so as to overcome these limitations.

5. What If We Cannot Perform Many Iterations

Problem. In many real-life engineering problems, we do not have the possibility to run the program f 200 times. In this case, we can still use the Cauchy deviates estimates with the available amount of N iterations, but we need to come up with new formulas that translate the numerical estimate into the enclosure for Δ.

Case where N is large enough. In this case, the difference $\widetilde{\Delta} - \Delta$ is still Gaussian, we can conclude that $\widetilde{\Delta} \cdot \left(1 - k_0 \cdot \sqrt{\dfrac{2}{N}}\right) \le \Delta \le$

$\widetilde{\Delta}\cdot\left(1+k_0\cdot\sqrt{\dfrac{2}{N}}\right)$ hence $\Delta\le\widetilde{\Delta}\cdot\left(1+k_0\cdot\sqrt{\dfrac{2}{N}}\right)$ (where $k_0=2$), with certainty 95%. (If we want, e.g., 99.9% certainty, which corresponds to 3 sigma, then we should take $k_0=3$.)

Thus, e.g., for $N=50$, we conclude that $\Delta\le 1.4\cdot\widetilde{\Delta}$. This is not such a bad estimate.

Case of very small number of iterations: idea. When the number of iterations is even smaller, then we can no longer assume that the distribution of the error $\widetilde{\Delta}-\Delta$ is Gaussian. In this case, to find the bounds on Δ with, e.g., 95% certainty, we must perform numerical experiments.

As we have mentioned in the above description of the Cauchy deviates method, the distribution of the results $\delta y^{(k)}$ always follows the Cauchy distribution, no matter how small N is.

So, to find out the confidence bounds on the Cauchy deviate estimates, it is sufficient to make experiments with the Cauchy distribution. The Cauchy distribution with a parameter Δ can be obtained by multiplying the basic Cauchy-distributed random variable (with parameter $\Delta_0=1$) by the number Δ. Thus, it is sufficient to test the method on basic Cauchy deviates, with parameter 1.

For each N and α, we want to find $k(N,\alpha)$ for which $\Delta\le k(N,\alpha)\cdot\widetilde{\Delta}$ with certainty $1-\alpha$, i.e., for which $\widetilde{\Delta}\ge(1/k(N,\alpha))\cdot\Delta$ with probability $1-\alpha$. Since we will be using Cauchy distribution with $\Delta=1$, we must thus find $k(N,\alpha)$ for which $\widetilde{\Delta}\ge 1/k(N,\alpha)$ with probability $1-\alpha$.

To find such value, we do the following. We pick a large number of iterations M (the relative accuracy of our estimate of $k(N,\alpha)$ will be $\approx 1/\sqrt{M}$). Then:

- For each m from 1 to M:
  - we simulate Cauchy distribution (with parameter $\Delta_0=1$) N times, producing N numbers

    $$\delta y_1^{(m)}=\tan(\pi\cdot(r_1^{(m)}-0.5)),\ldots,\delta y_N^{(m)}=\tan(\pi\cdot(r_N^{(m)}-0.5));$$
  - we then apply the above Maximum Likelihood Method to find $\widetilde{\Delta}_m$ as the solution to the following equation:

    $$\frac{1}{1+\left(\dfrac{\delta y_1^{(m)}}{\widetilde{\Delta}_m}\right)^2}+\ldots+\frac{1}{1+\left(\dfrac{\delta y_N^{(m)}}{\widetilde{\Delta}_m}\right)^2}=\frac{N}{2};$$

    we solve this equation by applying a bisection method to the interval $\left[0,\max\limits_i\left|\delta y_i^{(m)}\right|\right]$.

- After that, we sort the values $\widetilde{\Delta}_m$ into an increasing sequence
$$\widetilde{\Delta}_{(1)} \leq \ldots \leq \widetilde{\Delta}_{(M)}.$$
- We take the value $\widetilde{\Delta}_{(\alpha \cdot M)}$ for which the probability to be greater than this number is exactly $1-\alpha$, and estimate $k(N, \alpha)$ as $1/\widetilde{\Delta}_{(\alpha \cdot M)}$.

Simulation results. We wrote a C program that implements this algorithm. For $\alpha = 0.05$, the results of applying this program are:

- For $N = 20$, we get $k \approx 1.7$, which fits very well with the above Gaussian-based formula $k_{\rm norm} \approx 1 + 2 \cdot \sqrt{2/20} \approx 1.7$.
- For $N = 10$, we get $k \approx 2.1$, which is slightly higher than the Gaussian-based formula $k_{\rm norm} \approx 1 + 2 \cdot \sqrt{2/10} \approx 1.9$.
- For $N = 5$, we get $k \approx 5$, which is already much higher than the Gaussian-based value $k_{\rm norm} \approx 1 + 2 \cdot \sqrt{2/5} \approx 2.3$.

6. Dempster-Shafer Knowledge Bases

6.1. Formulation of the Problem

Formulation of the problem. In the previous sections, we described and analyzed different methods for estimating uncertainty in the cases when we have probabilistic or interval uncertainty in the inputs. What if the uncertainty in each input x_i is characterized, e.g., by the Dempster-Shafer knowledge bases?

Example. Let us consider a simple example, where f is a simple linear function of two variables $f(x_1, x_2) = x_1 + x_2$, the measured values are $\widetilde{x}_1 = \widetilde{x}_2 = 1.5$ – so that $\widetilde{y} = \widetilde{x}_1 + \widetilde{x}_2 = 3$ – and the uncertainty in each of the two variables is characterized by the Dempster-Shafer knowledge base:

- for x_1, we have two intervals $[\underline{x}_1^{(1)}, \overline{x}_1^{(1)}] = [0, 2]$ and $[\underline{x}_1^{(2)}, \overline{x}_1^{(2)}] = [1, 3]$, with masses $m_1^{(1)} = m_2^{(1)} = 0.5$;
- for x_2, we have two intervals $[\underline{x}_2^{(1)}, \overline{x}_2^{(1)}] = [0, 3]$ and $[\underline{x}_2^{(2)}, \overline{x}_2^{(2)}] = [1, 2]$, with masses $m_1^{(2)} = 0.7$ and $m_2^{(2)} = 0.5$;

D-S approach: general case. A D-S knowledge base for each input variable x_i means that we may have different intervals $[\underline{x}_i^{(s_i)}, \overline{x}_i^{(s_i)}]$, with different masses $p_i^{(s_i)}$. For each combination of intervals, $[\underline{x}_1^{(s_1)}, \overline{x}_1^{(s_1)}], \ldots, [\underline{x}_n^{(s_n)}, \overline{x}_n^{(s_n)}]$, we can use the known techniques to find the corresponding interval $[\underline{y}^{(s_1,\ldots,s_n)}, \overline{y}^{(s_1,\ldots,s_n)}]$ for the output. Since we know the mass (BPA) $p_i^{(s_i)}$ of each interval $[\underline{x}_i^{(s_i)}, \overline{x}_i^{(s_i)}]$, and we assume that the corresponding probability distributions are independent, we can compute the BPA $p^{(s_1,\ldots,s_n)}$ of the corresponding output interval as the product $p^{(s_1,\ldots,s_n)} = p_1^{(s_1)} \cdot \ldots \cdot p_n^{(s_n)}$.

Example. In particular, for the above simple example, the D-S knowledge base corresponding to y consists of the following $2 \times 2 = 4$ intervals:

- combining $[\underline{x}_1^{(1)}, \overline{x}_1^{(1)}] = [0,2]$ and $[\underline{x}_2^{(1)}, \overline{x}_2^{(1)}] = [0,3]$, we get an interval $[\underline{y}^{(1,1)}, \overline{y}^{(1,1)}] = [0,2] + [0,3] = [0,5]$ with the mass $p^{(1,1)} = p_1^{(1)} \cdot p_2^{(1)} = 0.5 \cdot 0.7 = 0.35$;
- combining $[\underline{x}_1^{(1)}, \overline{x}_1^{(1)}] = [0,2]$ and $[\underline{x}_2^{(2)}, \overline{x}_2^{(2)}] = [1,2]$, we get an interval $[\underline{y}^{(1,2)}, \overline{y}^{(1,2)}] = [0,2] + [1,2] = [1,4]$ with the mass $p^{(1,2)} = p_1^{(1)} \cdot p_2^{(2)} = 0.5 \cdot 0.3 = 0.15$;
- combining $[\underline{x}_1^{(2)}, \overline{x}_1^{(2)}] = [1,3]$ and $[\underline{x}_2^{(1)}, \overline{x}_2^{(1)}] = [0,3]$, we get an interval $[\underline{y}^{(2,1)}, \overline{y}^{(2,1)}] = [1,3] + [0,3] = [1,6]$ with the mass $p^{(2,1)} = p_1^{(2)} \cdot p_2^{(1)} = 0.5 \cdot 0.7 = 0.35$;
- combining $[\underline{x}_1^{(2)}, \overline{x}_1^{(2)}] = [1,3]$ and $[\underline{x}_2^{(2)}, \overline{x}_2^{(2)}] = [0,3]$, we get an interval $[\underline{y}^{(2,2)}, \overline{y}^{(2,2)}] = [1,3] + [1,2] = [2,5]$ with the mass $p^{(2,2)} = p_1^{(2)} \cdot p_2^{(2)} = 0.5 \cdot 0.3 = 0.15$.

Why this problem is difficult. One reason why this problem is difficult is that while, in principle, it is clear how we can represent the D-S knowledge base corresponding to the output, the corresponding representation may not always be practically possible; see, e.g., [7].

Indeed, at first glance, the above method may sound like a reasonable solution to our problem, but in reality, this solution is not always practical: even in the simplest case, when for each variable, we have two possible intervals, for $n = 50$ inputs, we will have an astronomical number of $2^{50} \approx 10^{15}$ output intervals $[\underline{y}^{(s_1,\ldots,s_n)}, \overline{y}^{(s_1,\ldots,s_n)}]$. Thus,

- *in principle*, the resulting uncertainty in y is still a D-S uncertainty, in the sense that there exists a list of intervals with corresponding basic probability assignments (BPAs),

- however, *in practice*, we can no longer represent it as we represented the uncertainty for each input: by listing all the intervals and the corresponding masses.

6.2. Main Idea

Our main idea: instead of listing all the output intervals and their BPAs, it is sufficient to describe the BPA distribution on the set of all possible intervals. As we have just mentioned, there are so many output intervals that it is physically impossible to describe the set of all such intervals by listing these intervals one by one. The situation is similar to numerous other practical situations when we need to describe a large collection of objects with different properties; e.g.:

- we cannot describe all the atoms in a gas by listing all their locations and velocities;
- we cannot easily describe the world population of mice by listing individual height and weight and color of each individual mouse.

Since we are not interested in individual atoms, all we want to know is what are the velocities v of different atoms. This desired knowledge can be described as a *discrete* probability distribution describing the probability that a randomly selected atom has a given velocity. For velocities that actually occur within a given gas, the probability is equal to $1/N$, where N is the overall number of atoms. For all other velocities, the probability is 0.

The resulting discrete probability distribution is located on a large number of values. Theoretically, it is possible to sort the velocities and get an explicit description of this discrete probability distribution. In practice, when N is large, the difference between the neighboring velocities is so small that it is below the measurement accuracy. As a result, for each measured value of velocity $\widetilde{v}$, we have a large number of atoms whose actual velocity values are (within the measurement accuracy) equal to $\widetilde{v}$. In effect, we now have a *continuous* probability distribution on the set of all possible velocity values, a distribution which accurately describes the empirical distribution of atoms' velocities. According to statistical physics, the velocity distribution follows Maxwell's law with probability density $c \cdot v^2 \cdot \exp\left(-\dfrac{m \cdot v^2}{2k \cdot T}\right)$, where T is the gas' temperature, m is the atom's mass, k is the Boltzmann's constant, and c is the normalization constant (which makes the overall probability equal to 1).

Once we know this distribution, we can determine, with a very good accuracy, how many atoms have a velocity above or below a given threshold – and many other characteristics of the gas.

Similarly, in the case of propagating D-S uncertainty, we have a large number $N > 2^{50}$ of output intervals with different BPAs. In other words, we have a *discrete* BPA distribution in which the overall mass 1 is distributed between N output intervals. Different output intervals $[\underline{y}, \overline{y}]$ can be characterized by different values of $\underline{y}$ and $\overline{y}$.

In practice, we can only determine the endpoints of the original intervals (focal elements) with a certain accuracy; as a result, we only know the endpoints $\underline{y}$ and $\overline{y}$ with a certain accuracy. If we take this accuracy into account, then for each pair of numbers $\underline{y} \le \overline{y}$, we will, in general, have many output intervals which (within this accuracy) coincide with $[\underline{y}, \overline{y}]$. Since these intervals are – within this accuracy – indistinguishable, it does not make any practical sense to list individual BPAs of all these intervals, it is sufficient to know the overall BPA of all these (practically coinciding) intervals.

In other words, to describe all the output intervals and their BPAs, we need a *continuous* BPA distribution on the set of all possible intervals.

This is how we will describe the propagation of a D-S structure:

- by explicitly describing the corresponding BPA distribution on the set of all intervals, and

- by using this PBA distribution (on the set of all focal elements) as a description of the uncertainty in y.

Let us analyze how we can do it.

In many practical problems, the contributions of different inputs are of the same order of magnitude. In many practical situations, the uncertainties which we know different inputs is approximately of the same size (or at least the same order of magnitude); as a result, in such situations, the contributions of different inputs to the resulting output uncertainty are of the same order of magnitude.

Examples of such situation include the inverse problems in geosciences (see, e.g., [3]), where the uncertain inputs are the measured traveltimes t_i of a seismic signal from the source to the sensor. These traveltimes are measured with approximately the same uncertainty, and their effect on the output is of the same order of magnitude. As a result, the influences of different inputs on the output uncertainty are also of the same order of magnitude.

These are the situations for which we will be developing a new algorithm.

Comment: the new algorithm will also be useful when we have a small number of dominating inputs among a large total number of uncertain inputs, and when expert provide us with a list of such inputs. In practice, the resulting output uncertainty is often dominated by only a few uncertain inputs. In many such situations, experts can tell us which of the inputs are dominant.

We will show that in such situations, the description of the resulting D-S uncertainty can be reduced to the above case – of inputs with approximately equal influence.

6.3. From Interval to Dempster-Shafer Uncertainty

Can we use the fact that D-S uncertainty is a generalization of interval uncertainty? Our idea comes from the fact that the Dempster-Shafer uncertainty is a generalization of interval uncertainty, a generalization in which, for each inputs x_i, instead of a single interval $[\underline{x}_i, \overline{x}_i]$, we have several ($N_i$) possible intervals $[\underline{x}_i^{(s)}, \overline{x}_i^{(s)}]$ ($s = 1, \ldots, N_i$), with different masses $p_i^{(s)}$. For the interval uncertainty, in a realistic case where the black-box function is linearizable, we can use the Cauchy deviates method to estimate the interval uncertainty of the output. Let us see whether it is possible – at least, under some reasonable assumptions – to extend the Cauchy deviates method to the more general Dempster-Shafer case.

Analysis of the problem. The fact that the black-box function is linearizable means that we have $f(x_1, \ldots, x_n) = \widetilde{y} + \sum\limits_{i=1}^{n} c_i \cdot (x_i - \widetilde{x}_i)$, where $\widetilde{y} \stackrel{\text{def}}{=} f(\widetilde{x}_n, \ldots, \widetilde{x}_n)$ and for every i, c_i denotes the (unknown) value of the partial derivative $\partial f / \partial x_i$ of the black-box function $f(x_1, \ldots, x_n)$ with respect to i-th input x_i.

As we have mentioned (when we described the drawbacks of sensitivity analysis), in many real-life situations, obtaining the indicated partial derivatives can be difficult (i.e., computationally expensive). It is therefore desirable to come up with an algorithm that would not require that we actually compute these values. Let us analyze how we can come up with such an algorithm.

If we knew the exact values $x_1, \ldots, x_n$ of all the inputs, then we could simply plug in the values x_i and get the desired value – but, of course, the whole problem is that we do not know the exact values x_i.

If for each i, we know the interval $[x_i^{\rm mid} - \Delta_i, x_i^{\rm mid} + \Delta_i]$, then, in the linearized case described above, the corresponding range of y can be described by the interval $[y^{\rm mid} - \Delta, y^{\rm mid} + \Delta]$, where:

$$y^{\rm mid} = \widetilde{y} + \sum_{i=1}^{n} c_i \cdot (x_i^{\rm mid} - \widetilde{x}_i); \tag{2}$$

$$\Delta = \sum_{i=1}^{n} |c_i| \cdot \Delta_i. \tag{3}$$

In the Dempster-Shafer case, for each i, instead of a single pair $(x_i^{\rm mid}, \Delta_i)$, we have different pairs with different BPAs.

In the above simple example, for x_1, we have two possible intervals $[0, 2]$ and $[1, 3]$ with BPA 0.5 for both. Computing the midpoint and half-width of each of these intervals, we conclude that:

- we have $(x_1^{\rm mid(1)}, \Delta_1^{(1)}) = (1, 1)$ with BPA 0.5, and
- we have $(x_1^{\rm mid(2)}, \Delta_1^{(2)}) = (2, 1)$ with BPA 0.5.

For x_2, we have two possible intervals: $[0, 3]$ with BPA 0.7 and $[1, 2]$ with BPA 0.3. Thus:

- we have $(x_2^{\rm mid(1)}, \Delta_2^{(1)}) = (1.5, 1.5)$ with BPA 0.7, and
- we have $(x_2^{\rm mid(2)}, \Delta_2^{(2)}) = (1.5, 0.5)$ with BPA 0.3.

Due to the formulas (2) and (3), the vector $(y^{{\rm mid}(s_1,s_2)}, \Delta^{(s_1,s_2)})$ is a linear combination of the vectors $(x_i^{{\rm mid}(s_i)}, \Delta_i^{(s_i)})$ corresponding to different inputs x_i.

In the above simple example, $\widetilde{y} = 3$, $\widetilde{x}_1 = \widetilde{x}_2 = 1.5$, and $c_1 = c_2 = 1$, hence the resulting vector $(y^{\rm mid}, \Delta)$ has the following form:

- with BPA $0.5 \cdot 0.7 = 0.35$, we have
$$y^{\rm mid(1,1)} = 3 + 1 \cdot (1 - 1.5) + 1 \cdot (1.5 - 1.5) = 2.5$$
and $\Delta^{(1,1)} = |1| \cdot 1 + |1| \cdot 1.5 = 2.5$;
- with BPA $0.5 \cdot 0.3 = 0.15$, we have
$$y^{\rm mid(1,2)} = 3 + 1 \cdot (1 - 1.5) + 1 \cdot (1.5 - 1.5) = 2.5$$
and $\Delta^{(1,2)} = |1| \cdot 1 + |1| \cdot 0.5 = 1.5$;
- with BPA $0.5 \cdot 0.7 = 0.35$, we have
$$y^{\rm mid(2,1)} = 3 + 1 \cdot (2 - 1.5) + 1 \cdot (1.5 - 1.5) = 3.5$$
and $\Delta^{(2,1)} = |1| \cdot 1 + |1| \cdot 1.5 = 2.5$;

– with BPA $0.5 \cdot 0.3 = 0.15$, we have
$$y^{\mathrm{mid}(2,2)} = 3 + 1 \cdot (2 - 1.5) + 1 \cdot (1.5 - 1.5) = 3.5$$
and $\Delta^{(2,2)} = |1| \cdot 1 + |1| \cdot 1.5 = 2.5$.

We consider the situations in which the contributions of all n inputs is approximately of the same size (or at least the same order of magnitude). In this case, the vector $(y^{\mathrm{mid}}, \Delta)$ is a linear combination of n independent vectors of approximately the same size.

Comment. In many practical situations, we have several dominant inputs, i.e., inputs whose contribution to the overall uncertainty is much larger than the contribution of other inputs – to the extent that the overall uncertainty can be, in effect, accounted for by these inputs only. In most such cases, experts can provide us with a list of such dominant inputs.

Our algorithm can also be applied to such cases where there are a few dominant inputs: for that, we simply restrict our analysis to these dominant inputs.

Analysis of the problem (cont-d). The above situation – when the contributions of different inputs are of the same order of magnitude – is exactly the case covered by the Central Limit Theorem. According to this theorem, in the limit $n \to \infty$, we have a normal 2-D distribution. Hence, for sufficient large n, with a good approximation, we can assume that the pair $(y^{\mathrm{mid}}, \Delta)$ is normally distributed.

Comment: strictly speaking, the distribution is almost normal but not exactly normal. From the purely *theoretical* viewpoint, the distribution of the pairs $(y^{\mathrm{mid}}, \Delta)$ cannot be exactly normal, because:

– the interval half-width Δ is always non-negative, while

– for every normally distributed random variable, there is a non-zero probability that this value attains negative values.

However, *in practice*, every normal distribution with mean μ and standard deviation σ is located within the interval $[\mu - k \cdot \sigma, \mu + k \cdot \sigma]$ with practically a certainty, i.e., with probability ≈ 1:

– for $k = 3$, the probability to be outside the 3 sigma interval is $\approx 0.1\%$;

– for $k = 6$, the probability to be outside the 3 sigma interval is $\approx 10^{-6}\%$; etc.

Thus, if $\mu \geq k \cdot \sigma$, then, for all practical purposes, the half-width Δ is indeed always non-negative.

6.4. Resulting Technical Idea

Resulting idea. It is therefore reasonable to conclude that for large n, the uncertainty in y can be characterized as follows: we have different intervals $[y^{\text{mid}} - \Delta, y^{\text{mid}} + \Delta]$, and the mass (BPA) of an interval is described by a 2-D normal distribution on the (y^{mid}, Δ) plane.

To describe a 2-D normal distribution, it is sufficient to know 5 parameters: the means $E[y^{\text{mid}}]$, $E[\Delta]$, and standard deviations $\sigma[y^{\text{mid}}]$, $\sigma[\Delta]$ of both variables and the covariance $C = C(y^{\text{mid}}, \Delta)$ (that describes their dependence).

The number of these intervals can be very large, as mentioned earlier. It is therefore impractical to find these characteristics by directly processing the actual midpoint and width of each interval. Instead, we must find a practical (indirect) method of computing these characteristics. Such a method (based on Monte-Carlo techniques) is described in this paper.

Example. The above simple example may not be a very good illustration of this idea since we only get Gaussian distribution for a large number of variables; however, for illustrative purposes, let us still estimate the corresponding 5 parameters of this distribution. We have:

- $(y^{\text{mid}}, \Delta) = (2.5, 2.5)$ with BPA 0.35;
- $(y^{\text{mid}}, \Delta) = (2.5, 1.5)$ with BPA 0.15;
- $(y^{\text{mid}}, \Delta) = (3.5, 2.5)$ with BPA 0.35;
- $(y^{\text{mid}}, \Delta) = (3.5, 2.5)$ with BPA 0.15.

Thus,

$$E[y^{\text{mid}}] = 0.35 \cdot 2.5 + 0.15 \cdot 2.5 + 0.35 \cdot 3.5 + 0.15 \cdot 3.5 = 3;$$

$$V[y^{\text{mid}}] =$$

$$0.35 \cdot (2.5 - 3)^2 + 0.15 \cdot (2.5 - 3)^2 + 0.35 \cdot (3.5 - 3)^2 + 0.15 \cdot (3.5 - 3)^2 =$$

$$0.25;$$

$$E[\Delta] = 0.35 \cdot 2.5 + 0.15 \cdot 1.5 + 0.35 \cdot 2.5 + 0.15 \cdot 1.5 = 2.2;$$

$$V[\Delta] =$$

$$0.35\cdot(2.5-2.2)^2+0.15\cdot(1.5-2.2)^2+0.35\cdot(2.5-2.2)^2+0.15\cdot(1.5-2.2)^2 = 0.21;$$

$$C = 0.35\cdot(2.5-3)\cdot(2.5-2.2)+0.15\cdot(2.5-3)\cdot(1.5-2.2)+$$
$$0.35\cdot(3.5-3)\cdot(2.5-2.2)+0.15\cdot(3.5-3)\cdot(1.5-2.2)=0.$$

Discussion: are we abandoning the idea of non-parametric estimates? At first glance, it may seem like we are abandoning our approach: we started with the idea of having non-parametric estimates, and we ended up with a 5-parametric family.

However, realistically, to exactly describe a generic distribution, we must use infinitely many parameters. In reality, we only have finitely many runs of the black-box function f with reasonable accuracy, and based on their results, we can only estimate finitely many parameters anyway.

Even in the ideal case of Monte-Carlo tests, we need N experiments to get a value of each parameter with an accuracy of $1/\sqrt{N}$. Thus, to get a reasonably low accuracy of 30% (everything worse makes it order-of-magnitude qualitative estimate), we need ≈ 10 runs.

With 50 runs, we can therefore determine the values of no more than 5 parameters anyway. The above 5-parametric family is reasonable; its justification is very similar to the justification of the Gaussian distribution – the main workhorse of statistics – so why not use it?

6.5. How Can We Actually Determine the Five Parameters of the 2-Dimensional Normal Distribution of $(y^{\mathrm{mid}}, \Delta)$

How can we determine the parameters of the normal BPA distribution: estimating the mean and variance of y^{mid}. If we simply take the midpoints $x_i^{(\mathrm{mid}(s_i)}$ of the corresponding intervals in our simulations, then the resulting values $y^{\mathrm{mid}(s_1,\ldots,s_n)}$ are normally distributed, with the distribution corresponding to y^{mid}. We can therefore estimate the mean and standard deviation of y^{mid} as simply the sample mean and the sample variance of the corresponding values $y^{\mathrm{mid}(s_1,\ldots,s_n)} = f(x_1^{(1)s_1},\ldots,x_n^{(n)s_n})$.

How can we determine the parameters of the normal BPA distribution: estimating the mean and variance of Δ. For Δ, from the formula (3), we conclude that

$$E[\Delta] = \sum_{i=1}^{n} |c_i|\cdot E[\Delta_i] \tag{4}$$

and

$$\sigma[\Delta] = \sqrt{\sum_{i=1}^{n} |c_i|^2 \cdot \sigma^2[\Delta_i]}. \qquad (5)$$

(One can easily check that for the above simple example, these formulas are indeed true.)

Due to the formula (4), we can use the Cauchy deviates technique to estimate $E[\Delta]$ if for each input x_i, we use the average half-width

$$E[\Delta_i] = p_i^{(1)} \cdot \Delta_i^{(1)} + p_i^{(2)} \cdot \Delta_i^{(2)} + \ldots$$

of the corresponding interval.

Due to the fact that $|c_i|^2 = c_i^2$, the formula (5) means that we can compute $\sigma[\Delta]$ by using the standard Monte-Carlo simulation technique: namely, we simulate δx_i to be normally distributed with 0 mean and standard deviation $\sigma[\Delta_i]$, then the resulting value of $\delta y = \sum c_i \cdot \delta x_i$ is also normally distributed, with the standard deviation equal to (5). We can thus estimate (5) as a sample variance of the corresponding simulated values $\delta y^{(k)}$.

Estimating the covariance between $y^{\rm mid}$ and Δ: an idea. We know how to estimate 4 of 5 parameters that describe the desired uncertainty: the means $E[y^{\rm mid}]$, $E\Delta]$ and the standard deviations $\sigma[y^{\rm mid}] = \sqrt{V[y^{\rm mid}]}$, $\sigma[\Delta] = \sqrt{V[\Delta]}$ of $y^{\rm mid}$ and Δ, where for each variable X, its variance VX is defined as $V[X] \stackrel{\rm def}{=} E[(X - E[X])^2]$. The only remaining problem is how to estimate the covariance

$$C \stackrel{\rm def}{=} C(y^{\rm mid}, \Delta) = E[(y^{\rm mid} - E[y^{\rm mid}]) \cdot (\Delta - E[\Delta])]$$

between $y^{\rm mid}$ and Δ.

Non-zero covariance means, in particular, that the conditional average $E[\Delta \,|\, y^{\rm mid} \geq E[y^{\rm mid}]]$ of Δ over the cases where $y^{\rm mid}$ is smaller than its average $E[y^{\rm mid}]$ is different from the average $E[\Delta]$ of Δ over all the cases. Let us show how from the difference between these two averages, we can determine the desired value of the covariance.

To simplify the formulas, let us denote the difference $y^{\rm mid} - E[y^{\rm mid}]$ by Y, and the difference $\Delta - E[\Delta]$ by X. In terms of these new variables, $E[X] = E[Y] = 0$, the formula for the desired covariance has the simplified form $C = E[X \cdot Y]$, and the conditional average takes the simplified form $E[\Delta + X \,|\, Y \geq 0] = E[\Delta] + E[X \,|\, Y \geq 0]$.

If $y^{\rm mid}$ and Δ were independent, then X and Y would be independent too: $C = E[X \cdot Y] = 0 = E[X] \cdot E[Y]$. Since a covariance may be non-zero, the variables X and Y may be dependent. To describe the conditional average in terms of the covariance, let us reduce this problem to the easier-to-handle case of independent normally distributed random variables. Specifically, let us find a real number λ for which the linear combination $X' \stackrel{\rm def}{=} X - \lambda \cdot Y$ is independent from Y. For this variable, $E[X'] = E[X] - \lambda \cdot E[Y] = 0$.

It is well known that a linear combination of normally distributed random variables is also normally distributed. Two variables X' and Y with a joint normal distribution and 0 mean are independent if and only if their covariance is 0, i.e., if $E[X' \cdot Y] = 0$. By definition of X', this condition means that

$$0 = E[X' \cdot Y] = E[X \cdot Y] - \lambda \cdot E[Y^2] = C - \lambda \cdot V[y^{\rm mid}].$$

Thus, to get the independent variables, we can take $\lambda = C/V[y^{\rm mid}]$.

Vice versa, if we know λ, we can reconstruct the covariance C as $\lambda \cdot V[y^{\rm mid}]$.

By definition of X', we have $X = X' + \lambda \cdot Y$, so the conditional average $E[X \,|\, Y \geq 0]$ takes the form

$$E[X \,|\, Y \geq 0] = E[X' \,|\, Y \geq 0] + \lambda \cdot E[Y \,|\, Y \geq 0].$$

Since X' and Y are independent, the first term is equal to $E[X' \,|\, Y \geq 0] = E[X'] = 0$. Since Y is normally distributed with 0 mean and standard deviation $\sigma = \sigma[y^{\rm mid}]$, the probability of $Y \geq 0$ is $1/2$, and the term $E[Y \,|\, Y \geq 0]$ is equal to

$$E[Y \,|\, Y \geq 0] = \frac{1}{1/2} \cdot \frac{1}{\sqrt{2\pi} \cdot \sigma} \cdot \int_0^\infty y \cdot \exp\left(-\frac{y^2}{2\sigma^2}\right)\, dy.$$

By introducing a new variable $z = \dfrac{y^2}{2\sigma^2}$, we get $dz = \dfrac{y\, dy}{\sigma^2}$, hence

$$E[Y \,|\, Y \geq 0] = \frac{2}{\sqrt{2\pi} \cdot \sigma} \cdot \int_0^\infty \sigma^2 \cdot \exp(-z)\, dz = \frac{2\sigma^2}{\sqrt{2\pi} \cdot \sigma} = \sqrt{\frac{2}{\pi}} \cdot \sigma.$$

Thus, $E[X \,|\, Y \geq 0] = \lambda \cdot \sigma[y^{\rm mid}]$, and $E[\Delta \,|\, y^{\rm mid} \geq E[y^{\rm mid}]] = E[\Delta] + \lambda \cdot \sigma[y^{\rm mid}]$. So, once we know the conditional average, we can find λ as $\lambda = \dfrac{E[\Delta \,|\, y^{\rm mid} \geq E[y^{\rm mid}]] - E[\Delta]}{\sigma[y^{\rm mid}]}$, and thus,

$$C = \alpha \cdot V[y^{\rm mid}] = (E[\Delta \,|\, y^{\rm mid} \geq E[y^{\rm mid}]] - E[\Delta]) \cdot \sigma[y^{\rm mid}].$$

Estimating the covariance between $y^{\rm mid}$ and Δ: how to actually compute the conditional average. We have just shown that to find the covariance, it is sufficient to compute the conditional average

$$E[\Delta \,|\, y^{\rm mid} \geq E[y^{\rm mid}]].$$

To compute this conditional averages, we can use the Cauchy deviates idea. Namely, at each simulation, for each variable x_i, we select one of the intervals $[\underline{x}_i^{(s_i)}, \overline{x}_i^{(s_i)}]$ with the corresponding probability $p_i^{(s_i)}$, and we apply the black box function f to the centers of the corresponding intervals, to get the result y^{mid}. We then apply the Cauchy techniques with the corresponding intervals and get the value distributed according to the Cauchy distribution with the width corresponding to selected intervals for x_i.

The main difference between what we propose to do here and the previously described Cauchy deviates methods is the following:

- in the previously described Cauchy deviates method, we combine all the results of Cauchy simulation into a single sample, and we then compute the parameter Δ based on this sample;

- in the proposed methods, we only consider the results of Cauchy simulation in which $y^{\text{mid}} \geq E[y^{\text{mid}}]$.

In the previous described approach, in all simulations, we had *the same* interval width, so the results of the simulation belong to the same Cauchy distribution. In the new method, we have *different* widths with different probabilities, so the resulting distribution is a combination of different Cauchy distributions, with different probabilities.

We can safely assume that the distribution of the width Δ is a Gaussian distribution, with mean μ and standard deviation σ. Thus, our new (conditional) sample corresponds to the combination in which the Cauchy distribution with parameter Δ occurs with the Gaussian probability density $\dfrac{1}{\sqrt{2\cdot\pi}\cdot\sigma}\cdot\exp\left(-\dfrac{(\Delta-\mu)^2}{2\sigma^2}\right)$. Cauchy-distributed random variable ξ with the parameter Δ can be described by its characteristic function $E[\exp(\mathrm{i}\cdot\omega\xi)] = \exp(-|\omega|\cdot\Delta)$. Thus, the above-described probabilistic combination of Cauchy distributions can be described by the corresponding probabilistic combination of these characteristic functions:

$$E[\exp(\mathrm{i}\cdot\omega\cdot\xi)] = \int \frac{1}{\sqrt{2\cdot\pi}\cdot\sigma}\cdot\exp\left(-\frac{(\Delta-\mu)^2}{2\sigma^2}\right)\cdot\exp(-|\omega|\cdot\Delta)\,\mathrm{d}\Delta. \quad (6)$$

Since $\exp(a+b)\cdot\exp(b) = \exp(a+b)$, this integral can be represented as

$$\int \frac{1}{\sqrt{2\cdot\pi}\cdot\sigma}\cdot\exp\left(-\frac{S}{2\sigma^2}\right)\,\mathrm{d}\Delta,$$

where

$$S \stackrel{\text{def}}{=} \Delta^2 - 2\cdot\Delta\cdot\mu + \mu^2 + |\omega|\cdot 2\cdot\sigma^2 = \Delta^2 - 2\cdot\Delta\cdot(\mu - |\omega|\cdot\sigma^2) + \mu^2.$$

To get the full square, we add and subtract $(\mu - |\omega| \cdot \sigma^2)^2$ to S, leading to

$$S = \Delta^2 - 2 \cdot \Delta \cdot (\mu - |\omega| \cdot \sigma^2) + (\mu - |\omega| \cdot \sigma^2)^2 - (\mu - |\omega| \cdot \sigma^2)^2 + \mu^2 =$$
$$(\Delta')^2 - (\mu - |\omega| \cdot \sigma^2)^2 + \mu^2,$$

where we denoted $\Delta' \stackrel{\text{def}}{=} \Delta - (\mu - |\omega| \cdot \sigma^2)$. Thus,

$$S = (\Delta')^2 - \mu^2 + 2 \cdot \mu \cdot |\omega| \cdot \sigma^2 - |\omega|^2 \cdot \sigma^4 + \mu^2 = (\Delta')^2 + 2 \cdot \mu \cdot |\omega| \cdot \sigma^2 - |\omega|^2 \cdot \sigma^4.$$

Thus,

$$\frac{S}{2\sigma^2} = \frac{(\Delta')^2}{2\sigma^2} + \mu \cdot |\omega| - \frac{1}{2} \cdot |\omega|^2 \cdot \sigma^2.$$

Therefore, the desired integral takes the form

$$\int \frac{1}{\sqrt{2 \cdot \pi} \cdot \sigma} \cdot \exp\left(-\frac{(\Delta')^2}{2\sigma^2}\right) \cdot \exp\left(\frac{1}{2} \cdot \sigma^2 \cdot \omega^2 - \mu \cdot |\omega|\right) \, d\Delta',$$

Moving the term that does not depend on Δ' outside the integral, we conclude that this integral is equal to

$$\exp\left(\frac{1}{2} \cdot \sigma^2 \cdot \omega^2 - \mu \cdot |\omega|\right) \cdot \int \frac{1}{\sqrt{2 \cdot \pi} \cdot \sigma} \cdot \exp\left(-\frac{(\Delta')^2}{2\sigma^2}\right) \, d\Delta'.$$

The remaining integral is the total probability of a normal distribution, i.e., 1, hence this integral is equal to:

$$\exp\left(\frac{1}{2} \cdot \sigma^2 \cdot \omega^2 - \mu \cdot |\omega|\right). \tag{7}$$

We can estimate the characteristic function by its sample value

$$E[\exp(\mathrm{i} \cdot \omega \cdot \xi)] \approx \frac{1}{N} \cdot \sum_{k=1}^{N} \cos(\omega \cdot y^{(k)})$$

(Since the expression (7) is real, it makes sense to only consider the real part of $\exp(\mathrm{i} \cdot \omega \cdot \xi)$, i.e., $\cos(\omega \cdot \xi)$.)

So, we arrive at the following algorithm for computing μ and σ from the sample values $y^{(1)}, \ldots, y^{(N)}$:

– for different real values $\omega_1, \ldots, \omega_t > 0$, compute $l(\omega_j) \stackrel{\text{def}}{=} -\ln(c(\omega_j))$, where $c(\omega_j) \stackrel{\text{def}}{=} \frac{1}{N} \cdot \sum\limits_{k=1}^{N} \cos(\omega_j \cdot y^{(k)})$;

– use the Least Squares Method to find the values μ and σ for which

$$\mu \cdot \omega_j - \frac{1}{2}\sigma^2 \cdot \omega_j^2 \approx l(\omega_j).$$

The resulting value μ is the desired conditional average Δ.

By putting together all the steps, we thus arrive at the following algorithm for computing the covariance C between $y^{\rm mid}$ and Δ.

Estimating the covariance between $y^{\rm mid}$ and Δ: the resulting algorithm.

– For $k = 1, 2, \ldots, 2N$, repeat the following:

  - for each variable x_i, select one of the intervals $[\underline{x}_i^{(s_i^{(k)})}, \overline{x}_i^{(s_i^{(k)})}]$ with the probability $p_i^{(s_i^{(k)})}$, and compute the corresponding half-width and midpoint $\Delta_i^{(s_i^{(k)})} := \dfrac{\overline{x}_i^{(s_i^{(k)})} - \underline{x}_i^{(s_i^{(k)})}}{2}$ and $x_i^{{\rm mid}(s_i^{(k)})} := \dfrac{\overline{x}_i^{(s_i^{(k)})} + \underline{x}_i^{(s_i^{(k)})}}{2}$;
  - compute the value $y^{{\rm mid}(k)} := f(x_1^{{\rm mid}(s_1^{(k)})}, \ldots, x_n^{{\rm mid}(s_n^{(k)})})$;
  - use the standard random number generator to compute n numbers $r_i^{(k)}$, $i = 1, 2, \ldots, n$, that are uniformly distributed on the interval $[0, 1]$;
  - compute Cauchy distributed values $c_i^{(k)} := \tan(\pi \cdot (r_i^{(k)} - 0.5))$;
  - compute the largest value of $|c_i^{(k)}|$ so that we will be able to normalize the simulated measurement errors and apply f to the values that are within the box of possible values: $K := \max_i |c_i^{(k)}|$;
  - compute the simulated approximation errors
$$\delta x_i^{(k)} := \Delta_i^{(k)} \cdot c_i^{(k)} / K;$$
  - compute the simulated "actual values"
$$x_i^{(k)} := x_i^{{\rm mid}(s_i^{(k)})} + \delta x_i^{(k)};$$
  - apply the program f to the simulated measurement results and compute the simulated result of the indirect measurement:
$$y^{(k)} := y^{{\rm mid}(k)} + K \cdot \left(f\left(x_1^{(k)}, \ldots, x_n^{(k)}\right) - y^{{\rm mid}(k)}\right);$$
dismiss this value if it is smaller than $E[y^{\rm mid}]$ and keep it if it is $\geq E[y^{\rm mid}]$.

– As a result, we get a sample $y^{(1)}, \dots, y^{(M)}$ of size $M \approx N$. Based on this sample, we compute the conditional average

$$E[\Delta \mid y^{\text{mid}} \geq E[y^{\text{mid}}]]$$

as follows:

- for different real values $\omega_1, \dots, \omega_t > 0$, compute $l(\omega_j) \stackrel{\text{def}}{=} -\ln(c(\omega_j))$, where $c(\omega_j) \stackrel{\text{def}}{=} \frac{1}{M} \cdot \sum\limits_{k=1}^{M} \cos(\omega_j \cdot y^{(k)})$;
- use the Least Squares Method to find the values μ and σ for which
$$\mu \cdot \omega_j - \frac{1}{2}\sigma^2 \cdot \omega_j^2 \approx l(\omega_j).$$

The resulting value μ is the desired conditional average $E[\Delta \mid y^{\text{mid}} \geq E[y^{\text{mid}}]]$.

– Finally, we estimate the covariance as

$$C(y^{\text{mid}}, \Delta) := (E[\Delta \mid y^{\text{mid}} \geq E[y^{\text{mid}}]] - E[\Delta]) \cdot \sigma[y^{\text{mid}}].$$

7. p-Boxes

Idea. It is known that a p-box can be described as a D-S knowledge base. Namely, a p-box $[\underline{F}(t), \overline{F}(t)]$ is a generalization of a cdf function $F(t)$. A cdf function can be represented by an explicit formula, or it can be represented if we list, for uniformly spaced levels $p = 0, \Delta p, 2 \cdot \Delta p, \dots, 1.0$ (e.g., for $p = 0, 0.1, 0.2, \dots, 0.9, 1.0$), the corresponding quantiles, i.e., values t for which $F(t) = p$. In mathematical terms, quantiles are the values of the inverse function $f(p) = F^{-1}(t)$ at equally spaced values p.

The variable with a probability distribution $F(t)$ can be approximately described as follows: we have the values $f(0)$, $f(\Delta p)$, etc., with equal probability Δp.

Similarly, a p-box can be alternatively represented by listing, for each p, the interval $[\underline{f}(p), \overline{f}(p)]$ of the possible quantile values. Here:

– the function $\underline{f}(p)$ is an inverse function to $\overline{F}(t)$, and

– the function $\overline{f}(p)$ is an inverse function to $\underline{F}(t)$.

This description, in effect, underlies some algorithms for processing p-boxes that are implement in RAMAS software [4].

Because of this description, we can interpret the p-box as the D-S knowledge base, in which, with equal probability Δp, we can have intervals $[\underline{f}(0), \overline{f}(0)]$, $[\underline{f}(\Delta p), \overline{f}(\Delta p)]$, etc.

Thus, whatever method we have for D-S knowledge bases, we can apply it to p-boxes as well.

How can we describe the resulting p-boxes? We have just mentioned that, in principle, we can interpret each p-box as a D-S knowledge base, and apply the above D-S-based method to describe th uncertainty of the output. The result, however, is a D-S knowledge base. How can we describe the corresponding "Gaussian" D-S knowledge base as a p-box?

It is known that for a D-S knowledge base, i.e., for a probabilistic distribution on the set of intervals $[\underline{x}, \overline{x}]$:

- The probability $F(t) = \text{Prob}(X \le t)$ attains its largest possible value $\overline{F}(t)$ if for each interval, we take the smallest possible value $\underline{x}$.
- Similarly, the probability $F(t) = \text{Prob}(X \le t)$ attains its smallest possible value $\underline{F}(t)$ if for each interval, we take the largest possible value $\overline{x}$.

Thus:

- $\overline{F}(t)$ is a probability distribution for the lower endpoints $y^{\text{mid}} - \Delta$, and
- $\underline{F}(t)$ is a probability distribution for the upper endpoints $y^{\text{mid}} + \Delta$ of the corresponding intervals.

Since the 2-D distribution of the pairs (y^{mid}, Δ) is Gaussian, the distributions of both linear combinations $y^{\text{mid}} - \Delta$ and $y^{\text{mid}} + \Delta$ are Gaussian as well.

Therefore, as a result of this procedure, we get a p-box $[\underline{F}(t), \overline{F}(t)]$ for which both bounds $\underline{F}(t)$ and $\overline{F}(t)$ correspond to Gaussian distributions.

Comment: strictly speaking, the distributions are almost normal but not exactly normal. Let us denote the cdf of the standard Gaussian distribution, with 0 mean and standard deviation 1 by $F_0(t)$. Then, an arbitrary Gaussian distribution, with mean μ and standard deviation σ, can be described as $F(t) = F_0((t-\mu)/\sigma)$. In particular, if we denote:

- the mean and the standard deviations of the Gaussian distribution $\underline{F}(t)$ by $\underline{\mu}$ and $\underline{\sigma}$, and

– the mean and the standard deviations of the Gaussian distribution $\overline{F}(t)$ by $\overline{\mu}$ and $\overline{\sigma}$,

then we conclude that $\underline{F}(t) = F_0((t-\underline{\mu})/\underline{\sigma})$ and $\overline{F}(t) = F_0((t-\overline{\mu})/\overline{\sigma})$.

From the theoretical viewpoint, for thus defined functions $\underline{F}(t)$ and $\overline{F}(t)$, we cannot always have $\underline{F}(t) \leq \overline{F}(t)$, because, due to monotonicity of $F_0(t)$, this would be equivalent to $\dfrac{t-\underline{\mu}}{\underline{\sigma}} \leq \dfrac{t-\overline{\mu}}{\overline{\sigma}}$ for all t, i.e., to one straight line being always below the other – but this is only possible when they are parallel.

However, as we have mentioned while describing the similar situation with the D-S knowledge bases, in practice, we can have this inequality if we ignore the values t for which $F_0(t)$ is very small – and thus, not practically possible.

Alternatively, we can assume that the inequality $\underline{F}(t) \leq \overline{F}(t)$ holds for all t – but the distributions $\underline{F}(t)$ and $\overline{F}(t)$ are only approximately – but not exactly – normal.

What if we have different types of uncertainty for different inputs? If we have different types of uncertainty for different inputs, we can transform them to p-boxes [4] – hence, to D-S knowledge bases – and use a similar approach.

8. Cauchy Deviates Methods for Non-Linear Functions $f(x_1, \ldots, x_n)$

Case of weak non-linearity. In some cases, we cannot reasonably approximate f by a linear expression on the entire box, but we can divide the box into a few subboxes on each of which f is approximately linear. For example, if the dependence of f on one of the variables x_i is strongly non-linear, then we can divide the interval $[\underline{x}_i, \overline{x}_i]$ of possible values of this variable into two (or more) subintervals, e.g., $[\underline{x}_i, x_i^{\rm mid}]$ and $[x_i^{\rm mid}, \overline{x}_i]$, and consider the corresponding subboxes

$$[\underline{x}_1, \overline{x}_1] \times \ldots \times [\underline{x}_{i-1}, \overline{x}_{i-1}] \times [\underline{x}_i, x_i^{\rm mid}] \times [\underline{x}_{i+1}, \overline{x}_{i+1}] \times \ldots \times [\underline{x}_n, \overline{x}_n]$$

and

$$[\underline{x}_1, \overline{x}_1] \times \ldots \times [\underline{x}_{i-1}, \overline{x}_{i-1}] \times [x_i^{\rm mid}, \overline{x}_i] \times [\underline{x}_{i+1}, \overline{x}_{i+1}] \times \ldots \times [\underline{x}_n, \overline{x}_n].$$

By using the Cauchy deviates methods, we compute the range of f over each of these subboxes, and then take the union of the resulting range intervals.

Quadratic case. Linearization technique is based on the assumption that the measurement errors Δx_i and/or uncertainties are so small that we can safely ignore terms that are quadratic (or of higher order) in Δx_i. If the measurement errors are larger, so that we can no longer reasonably approximate f by a linear expression, a natural next step is to take quadratic terms into consideration while still ignoring cubic and higher-order terms: $f(x_1^{\rm mid}+\delta x_1,\ldots,x_n^{\rm mid}+\delta x_n)\approx y^{\rm mid}+\delta y$, where

$$\delta y \stackrel{\rm def}{=} \sum_{\imath=1}^{n} c_i\cdot\delta x_i + \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}\cdot\delta x_i\cdot\delta x_j, \tag{8}$$

where c_i are the same as for the linearized case and $c_{ij}\stackrel{\rm def}{=}\dfrac{1}{2}\cdot\dfrac{\partial^2 f}{\partial x_i\partial x_j}$.

In general, computing the exact bound for a quadratic function of n variables in case of interval uncertainty is an NP-hard problem [28, 14]. Luckily, in many practical cases, the dependence of f on x_i is monotonic (see, e.g., [15]), so we can use, e.g., the above-described sensitivity analysis technique.

The problem with the sensitivity analysis technique, as we have mentioned, is that this technique requires n calls to the program f, which for large n may be too long. It is therefore desirable to modify the Cauchy deviate technique so that it can be used for quadratic functions as well.

Analysis of the problem. We consider the case where the function $f(x_1,\ldots,x_n)$ is monotonic in each variable x_i.

If the function f is increasing in x_i, then the derivative $\dfrac{\partial f}{\partial x_i}$ is always positive; in particular, it is positive at the central point $(x_1^{\rm mid},\ldots,x_n^{\rm mid})$, so $c_i>0$. In this case, the maximum of f is attained when $\delta x_i=\Delta_i$ and $x_i=\overline{x}_i=x_i^{\rm mid}+\Delta_i$.

Similarly, when the function f is decreasing in f, then $c_i<0$ and the maximum is attained when $\delta x_i=-\Delta_i$ and $x_i=x_i^{\rm mid}-\Delta_i$. In both cases, the largest possible value Δ^+ of the difference δy is attained when for every i, we have $\delta x_i=\varepsilon_i\cdot\Delta_i$, where $\varepsilon_i\stackrel{\rm def}{=}{\rm sign}(c_i)$. Substituting this expression for δx_i into the above formula for δy, we conclude that

$$\Delta^+ = \sum_{i=1}^{n} c_i\cdot\varepsilon_i\cdot\Delta_i + \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}\cdot\varepsilon_i\cdot\varepsilon_j\cdot\Delta_i\cdot\Delta_j =$$

$$\sum_{i=1}^{n}|c_i|\cdot\Delta_i + \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}\cdot\varepsilon_i\cdot\varepsilon_j\cdot\Delta_i\cdot\Delta_j. \tag{9}$$

Similarly, the smallest possible value $\delta y_{\min}$ of δy is attained when $\delta x_i = -\varepsilon_i \cdot \Delta_i$, hence $\Delta^- \stackrel{\text{def}}{=} |\delta y_{\min}|$ is equal to:

$$\Delta^- = \sum_{i=1}^n |c_i| \cdot \Delta_i - \sum_{i=1}^n \sum_{j=1}^n c_{ij} \cdot \varepsilon_i \cdot \varepsilon_j \cdot \Delta_i \cdot \Delta_j. \tag{10}$$

We would like to use a Cauchy-type method to find the bounds (9) and (10). For this, we consider, for every pairs of vectors $z = (z_1, \ldots, z_n)$ and $t = (t_1, \ldots, t_n)$, the following auxiliary expression:

$$\frac{f(x^{\text{mid}} + z + t) - f(x^{\text{mid}} + z - t)}{2} =$$

$$\frac{1}{2} \cdot f(x_1^{\text{mid}} + z_1 + t_1, \ldots, x_n^{\text{mid}} + z_n + t_n) -$$
$$\frac{1}{2} \cdot f(x_1^{\text{mid}} + z_1 - t_1, \ldots, x_n^{\text{mid}} + z_n - t_n). \tag{11}$$

Substituting $\delta x_i = z_i + t_i$ into the formula (8), we conclude that

$$f(x^{\text{mid}}+z+t) = y^{\text{mid}} + \sum_{\imath=1}^n c_i \cdot (z_i+t_i) + \sum_{i=1}^n \sum_{j=1}^n c_{ij} \cdot (z_i+t_i) \cdot (z_j+t_j), \tag{12}$$

and similarly,

$$f(x^{\text{mid}}+z-t) = y^{\text{mid}} + \sum_{\imath=1}^n c_i \cdot (z_i-t_i) + \sum_{i=1}^n \sum_{j=1}^n c_{ij} \cdot (z_i-t_i) \cdot (z_j-t_j), \tag{13}$$

hence

$$\frac{1}{2} \cdot (f(x^{\text{mid}}+z+t) - f(x^{\text{mid}}+z-t)) = \sum_{i=1}^n \left(c_i + 2 \cdot \sum_{j=1}^n c_{ij} \cdot z_j \right) \cdot t_i. \tag{14}$$

This expression is linear with respect to $t_1, \ldots, t_n$. Therefore, we can use the existing linear Cauchy algorithm in order to find bounds for this expression as a function of t_i when $|t_i| \le \Delta_i$.

Let $g(z) = g(z_1, \ldots, z_n)$ denote the result of applying the linear Cauchy method to the expression (14) considered as as a function of t; then,

$$g(z) = \sum_{i=1}^n \left| c_i + 2 \cdot \sum_{j=1}^n c_{ij} \cdot z_j \right| \cdot \Delta_i.$$

Since the function f is monotonic on the box, its derivative $\dfrac{\partial f}{\partial x_i}$ has the same sign at all the points from the box. Hence, the sign of the derivative $c_i + 2 \cdot \sum_{j=1}^{n} c_{ij} \cdot z_j$ at the point

$$x^{\text{mid}} + z = (x_1^{\text{mid}} + z_1, \ldots, x_n^{\text{mid}} + z_n)$$

is the same as the sign ε_i of the derivative c_i at the midpoint $x^{\text{mid}} = (x_1^{\text{mid}}, \ldots, x_n^{\text{mid}})$ of the box. Since $|E| = \text{sign}(E) \cdot E$ for every expression E, we thus conclude that

$$\left| c_i + 2 \cdot \sum_{j=1}^{n} c_{ij} \cdot z_j \right| = \varepsilon_i \cdot \left(c_i + 2 \cdot \sum_{j=1}^{n} c_{ij} \cdot z_j \right),$$

hence

$$g(z) = \sum_{i=1}^{n} |c_i| \cdot \Delta_i + 2 \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \cdot \varepsilon_i \cdot \Delta_i \cdot z_j. \qquad (15)$$

In particular, for $z = 0 = (0, \ldots, 0)$, we get $g(0) = \sum_{i=1}^{n} |c_i| \cdot \Delta_i$.

From (12) and (14), we conclude that

$$f(x^{\text{mid}} + z) - f(x^{\text{mid}} - z) = 2 \cdot \sum_{i=1}^{n} c_i \cdot z_i.$$

We can therefore construct a new function $h(z)$ as follows:

$$h(z) \stackrel{\text{def}}{=} \frac{1}{2} \cdot (g(z) - g(0) + f(x^{\text{mid}} + z) - f(x^{\text{mid}} - z)) =$$

$$\sum_{i=1}^{n} c_i \cdot z_i + \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \cdot \varepsilon_j \cdot \Delta_j \cdot z_i. \qquad (16)$$

This expression is linear with respect to $z_1, \ldots, z_n$. Therefore, we can use the existing linear Cauchy algorithm in order to find bounds for this expression as a function of z_i when $|z_i| \le \Delta_i$. As a result, we get the estimate

$$H \stackrel{\text{def}}{=} \sum_{i=1}^{n} \left| c_i + \sum_{j=1}^{n} c_{ij} \cdot \varepsilon_j \cdot \Delta_j \right| \cdot \Delta_i.$$

Since the function f is monotonic on the box, its derivative $\frac{\partial f}{\partial x_i}$ has the same sign at all the points from the box. Hence, the sign of the derivative $c_i + \sum_{j=1}^{n} c_{ij} \cdot \varepsilon_j \cdot \Delta_j$ at the point

$$(x_1^{\text{mid}} + \frac{1}{2} \cdot \varepsilon_1 \cdot \Delta_1, \ldots, x_n^{\text{mid}} + \frac{1}{2} \cdot \varepsilon_n \cdot \Delta_n)$$

is the same as the sign ε_i of the derivative c_i at the midpoint $x^{\rm mid} = (x_1^{\rm mid}, \ldots, x_n^{\rm mid})$ of the box. Since $|E| = \mathrm{sign}(E) \cdot E$ for every expression E, we thus conclude that

$$\left| c_i + \sum_{j=1}^{n} c_{ij} \cdot \varepsilon_j \cdot \Delta_j \right| = \varepsilon_i \cdot \left(c_i + \sum_{j=1}^{n} c_{ij} \cdot \varepsilon_j \cdot \Delta_j \right),$$

hence

$$H = \sum_{i=1}^{n} |c_i| \cdot \Delta_i + \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \cdot \varepsilon_i \cdot \Delta_i \cdot \varepsilon_j \cdot \Delta_j,$$

which is exactly the above expression for Δ^+. The value Δ^- can now be computed as $2g(0) - \Delta^+$.

We thus arrive at the following algorithm for computing Δ^+ and Δ^-.

Algorithm. As an auxiliary step, we first design an algorithm that, given a vector $z = (z_1, \ldots, x_n)$, computes $g(z)$. This algorithm consists of applying the linear Cauchy deviate method to the auxiliary function $t \to \dfrac{1}{2} \cdot (f(x^{\rm mid} + z + t) - f(x^{\rm mid} + z - t))$ and the values $t_i \in [-\Delta_i, \Delta_i]$. The linear Cauchy methods requires N calls to the auxiliary function (where N depends on the desired accuracy), and each call to the auxiliary function means 2 calls to the program f; so, overall, we need $2N$ calls to f.

The algorithm itself works as follows:

- First, we apply the algorithm $g(z)$ to the vector $0 = (0, \ldots, 0)$, thus computing the value $g(0)$.
- Second, we apply the linear Cauchy deviate method to the auxiliary function $h(z) = \dfrac{1}{2} \cdot (g(z) - g(0) + f(x^{\rm mid} + z) - f(x^{\rm mid} - z))$; the result is the desired value Δ^+.
- Finally, we compute Δ^- as $2g(0) - \Delta^+$.

What is the computational complexity of this algorithm? How many calls to the program f did we make?

- In the first stage, we made a single call to g, so this stage requires $2N$ calls to f.
- The second stage requires N calls to h. Each call to h means 2 calls to f and 1 call to g; each call to g, as we have mentioned, requires $2N$ calls to f. Thus, overall, each call to h requires $2 + 2N$ calls to f; in total, the second stage requires $N \cdot (2 + 2N)$ calls to f.

- On the final stage, there are no calls to f.

So, overall, this algorithm requires $2N + n \cdot (2 + 2N) = 2N \cdot (N + 2)$ calls to f.

For example, if we want the 20% accuracy on average, we need $N = 50$, so this algorithm would require ≈ 5000 calls to f. Thus, when we have $n \ll 5000$ variables, it is faster to use the sensitivity analysis method, but when we have $n \gg 5000$ variables, this Monte-Carlo-type method is faster.

If we want 20% accuracy with certainty 95%, then we need $N = 200$. In this case, the above quadratic method requires ≈ 80000 calls to f, so this method is faster only if we have $n \gg 80000$ variables.

9. Conclusions

The paper presents a method for solving the problem of estimating the uncertainty in the output parameter $y = f(x_1, \ldots, x_n)$ of a system, given the uncertainty in the input parameters $x_1, \ldots, x_n$ and a tool f for finding the value of the output parameter when the values of the input parameters x_i are known. Uncertainty can be described by a single interval, by a Dempster-Shafer body of evidence (a set of intervals with their associated masses), or by probability boxes. This can be viewed as an optimization problem in which the minimum and the maximum values of function f are to be found. In realistic situations when there are many input variables ($n > 50$), this problem requires a large amount of computation time. In this paper, we have shown that we can speed up these computations if we use Monte-Carlo techniques in which the generated sample values of the input variables are drawn from Cauchy probability distribution.

Acknowledgements

This work was supported in part by NASA under cooperative agreement NCC5-209, by the Future Aerospace Science and Technology Program (FAST) Center for Structural Integrity of Aerospace Systems, effort sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant F49620-00-1-0365, by NSF grants EAR-0112968, EAR-0225670, and EIA-0321328, by the Army Research Laboratories grant DATM-05-02-C-0046, and by a research grant from Sandia National Laboratories as part of the Department of Energy Accelerated Strategic Computing Initiative (ASCI).

The authors are greatly thankful to all the participants of the International Workshop on Reliable Engineering Computing REC'04 (Savannah, Georgia, September 15–17, 2004), especially to Arnold Neumaier, for valuable comments, and to the anonymous referees for their very useful suggestions.

References

1. Apostolakis, G.: The Concept of Probability in Safety Assessments of Technological Systems, *Science*, 1990, 250:1359–1364.
2. Breeding, R. J., Helton, J. C., Gorham, E. D., and Harper, F. T.: Summary Description of the Methods Used in the Probabilistic Risk Assessments for NUREG-1150, *Nuclear Engineering and Design*, 1992, 135:1–27.
3. Doser, D. I., Crain, K. D., Baker, M. R., Kreinovich, V., and Gerstenberger, M. C.: Estimating uncertainties for geophysical tomography, *Reliable Computing*, 1998, 4(3)241–268.
4. Ferson, S.: *RAMAS Risk Calc 4.0: Risk Assessment with Uncertain Numbers*, CRC Press, Boca Raton, Florida, 2002.
5. Helton, J. C.: Uncertainty and Sensitivity Analysis in the Presence of Stochastic and Subjective Uncertainty, *Journal of Statistical Computation and Simulation*, 1997, 57:3–76.
6. Helton, J. C., and Davis, F. J.: Latin Hypercube Sampling and the Propagation of Uncertainty in Analyses of Complex Systems, *Reliability Engineering and System Safety*, 2003, 81:23–69.
7. Helton, J. C., Johnson, J. D., and Oberkampf, W. L.: An Exploration of Alternative Approaches to the Representation of Uncertainty in Model Predictions, *Reliability Engineering and System Safety*, 2004, 85:39–71.
8. Hoffman, F. O., and Hammonds, J. S.: Propagation of Uncertainty in Risk Assessments: The Need to Distinguish Between Uncertainty Due to Lack of Knowledge and Uncertainty Due to Variability, *Risk Analysis*, 1994, 14:707–712.
9. Helton, J. C., and Marietta, M. G.: Special Issue: The 1996 Performance Assessment for the Waste Isolation Pilot Plant, *Reliability Engineering and System Safety*, 2000, 69:1–451.
10. Iman, R. L.: Uncertainty and Sensitivity Analysis for Computer Modeling Applications, *Proceedings of Reliability Technology'1992, The Winter Annual Meeting of the American Society of Mechanical Engineers (ASME)* (Anaheim, California, November 8–13, 1992), ASME Press, New York, NY, 1992, Vol. 28, pp. 153–168.
11. Iman, R. L. and Conover, W. J.: A Distribution-Free Approach to Inducing Rank Correlation Among Input Variables, *Communications in Statistics: Simulation and Computation*, 1982, B11:311–334.
12. Iman, R. L. and Davenport, J. M.: Rank Correlation Plots for Use with Correlated Input Variables, *Communications in Statistics: Simulation and Computation*, 1982, Bll:335–360.
13. Kreinovich, V. and Ferson, S.: A New Cauchy-Based Black-Box Technique for Uncertainty in Risk Analysis, *Reliability Engineering and Systems Safety*, 2004, 85(1–3): 267-279.

14. Kreinovich, V., Lakeyev, A., Rohn, J., and Kahl, P.: *Computational Complexity and Feasibility of Data Processing and Interval Computations*, Kluwer, Dordrecht, 1997.
15. Lakeyev, A. V. and Kreinovich, V.: If Input Intervals Are Small Enough, Then Interval Computations Are Almost Always Easy, *Reliable Computing*, 1995, Supplement (Extended Abstracts of APIC'95: International Workshop on Applications of Interval Computations, El Paso, TX, Febr. 23–25, 1995), pp. 134–139.
16. Jaulin L., Keiffer, M., Didrit, O., and Walter, E.: *Applied Interval Analysis*, Springer-Verlag, Berlin, 2001.
17. Kearfott, R. B.: *Rigorous Global Search: Continuous Problems*, Kluwer, Dordrecht, 1996.
18. Kearfott R. B. and Kreinovich, V. (eds.): *Applications of Interval Computations*, Kluwer, Dordrecht, 1996.
19. McKay, M. D., Beckman, R. J., and Conover, W. J.: A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code, *Technometrics*, 1979, 21:239–245.
20. Moore, R. E.: *Methods and Applications of Interval Analysis*, SIAM, Philadelphia, 1979.
21. Muhanna, R. L. and Mullen, R. L.: Uncertainty in mechanics problems – interval-based approach, *ASCE Journal of Engineering Mechanics*, 2001, 127(6):557–566.
22. Muhanna, R. L. and Mullen, R. L.: Treatment in geometric tolerances in finite element analysis, *International Journal of Advanced Manufacturing Systems*, 2001, 127(6):557–566.
23. Parry, G. W., and Winter, P. W., "Characterization and Evaluation of Uncertainty in Probabilistic Risk Analysis," *Nuclear Safety*, 1981, 22:28–42.
24. Pate-Cornell, M. E.: Uncertainties in Risk Analysis: Six Levels of Treatment, *Reliability Engineering and System Safety*, 1996, 54:95–111.
25. Saltelli, A., Chan, K., and Scott, E. M.: *Sensitivity Analysis*, Wiley, New York, N.Y., 2000.
26. Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M.: *Sensitivity Analysis in Practice*, Wiley, New York, N.Y., 2004.
27. Trejo, R. and Kreinovich, V.: Error Estimations for Indirect Measurements, In: S. Rajasekaran et al., editors, *Handbook on Randomized Computing*, Kluwer, 2001, pp. 673–729.
28. Vavasis, S. A.: *Nonlinear Optimization: Complexity Issues*, Oxford University Press, N.Y., 1991.
29. Wadsworth, H. M. Jr. (ed.): *Handbook of statistical methods for engineers and scientists*, McGraw-Hill Publishing Co., N.Y., 1990.

Appendix

A. List of Notations

Main problem: y – difficult-to-estimate quantity of interest; the actual (unknown) value of this quantity

$x_1, \ldots, x_n$ – easier-to-estimate quantities; actual (unknown) values of these quantities;

n – overall number of easier-to-estimate quantities;

i – an index used to describe number of the quantity, $i = 1, 2, \ldots, n$;

$f(x_1, \ldots, x_n)$ – an algorithm relating the actual values of x_i and y, so that $y = f(x_1, \ldots, x_n)$; this algorithm is also called the *data processing* algorithm;

$\widetilde{f}(x_1, \ldots, x_n)$ – a known approximation to the actual function $f(x_1, \ldots, x_n)$ – in case the expression for f is unknown;

$\widetilde{x}_i$ or x_i^{approx} – a known approximate value of i-th quantity; these values are also called *inputs* to the data processing algorithm;

$\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$ – approximate value of y resulting from the approximate values for x_i; it is also called the *result* (*output*) of data processing;

$\Delta x_i = \widetilde{x}_i - x_i$ – approximation error of i-th input quantity;

$\Delta y = \widetilde{y} - y$ – approximation error of the output quantity;

ε – relative accuracy with which we want to find the desired uncertainty characteristics.

Linearized case: $c_i = \dfrac{\partial f}{\partial x_i}$;

$\varepsilon_i = \text{sign}(c_i)$;

Probabilistic uncertainty – general notations: $E[X]$ – expected value (mean) of the random variable X;

$V[X]$ – variance of X;

$\sigma[X]$ – standard deviation of X;

$C(X, Y)$ – covariance between X and Y;

α – degree of uncertainty; for example, $\alpha = 0.05$ means that the result is correct in 95% of the cases;

$\rho(t)$ – probability density function;

k_0 – for normal distribution, selecting values within distance $k_0 \cdot \sigma$ from the mean as possible values.

Probabilistic uncertainty – specific notations: $F_i(t)$ – cumulative distribution function (cdf) for the i-th input x_i;

$F(t)$ – desired cdf for the output y;

$\sigma_i = \sigma[x_i]$ – standard deviation of the i-th input x_i;

$\sigma = \sigma[y]$ – desired standard deviation of the output y;

$[\underline{F}_i(t), \overline{F}_i(t)]$ – known bounds on the cdf of x_i; also known as a *p-box* for x_i;

$[\underline{F}(t), \overline{F}(t)]$ – desired bounds on the cdf of y; also known as a *p-box* for y.

Interval uncertainty: $\mathbf{x}_i = [\underline{x}_i, \overline{x}_i]$ – known guaranteed bounds on the value of the i-th input x_i; also known as the i-th input interval;

$x_i^{\text{mid}} = \dfrac{\underline{x}_i + \overline{x}_i}{2}$ – the midpoint of the i-th input interval;

$\Delta_i = \dfrac{\overline{x}_i - \underline{x}_i}{2}$ – the half-width of the i-th input interval;

Δ_i^- and Δ_i^+ – bounds on the approximation error Δx_i:

$$\Delta_i^- \le \Delta x_i \le \Delta_i^+;$$

$\mathbf{y} = [\underline{y}, \overline{y}]$ – desired bounds on the value of the output y; also called the output interval;

$y^{\text{mid}} = \dfrac{\underline{y} + \overline{y}}{2}$ – the midpoint of the output interval;

$\Delta = \dfrac{\overline{y} - \underline{y}}{2}$ – the half-width of the output interval;

Δ^- and Δ^+ – bounds on the approximation error Δy:

$$\Delta^- \le \Delta y \le \Delta^+;$$

$\delta x_i = x_i - x_i^{\text{mid}}$;
$\delta y = y - y^{\text{mid}}$.

Dempster-Shafer uncertainty: s – an index used to describe focal elements;

N_e – overall number of experts;

n_e – overall number of opinions, i.e., overall number of experts with different opinions;

$N_e^{(s)}$ – number of experts who selected s-th focal element;

N_i – number of focal elements for i-th input;

s_i – an index used to describe one of the focal elements for the i-th input; $s_i = 1, 2, \ldots, N_i$;

$[\underline{x}_i^{(s)}, \overline{x}_i^{(s)}]$ – s-th focal element corresponding to i-th input;

$p_i^{(s)}$ – mass of the corresponding focal element;

$[\underline{y}^{(s_1,\ldots,s_n)}, \overline{y}^{(s_1,\ldots,s_n)}]$ – an output interval corresponding to $[\underline{x}_1^{(s_1)}, \overline{x}_1^{(s_1)}]$, ..., $[\underline{x}_n^{(s_n)}, \overline{x}_n^{(s_n)}]$;

$p^{(s_1,\ldots,s_n)}$ – mass of the focal element $[\underline{y}^{(s_1,\ldots,s_n)}, \overline{y}^{(s_1,\ldots,s_n)}]$.

Data processing algorithm: r_j – the j-th intermediate result.

Algorithms for computing uncertainty of the result of data processing:

N – overall number if iterations;

k – an index used to describe an iteration, $1 \le k \le N$;

$\Delta x_i^{(k)}$, $\delta y^{(k)}$, ...– the values of Δx_i, δy, ..., simulated on the k-th iteration;

M – overall number of auxiliary iterations;
h_i – small simulated change of the i-th input;
χ_i, r_i – auxiliary random variables used to simulate the i-th input.