Monte-Carlo-Type Techniques for Processing Interval Uncertainty, and Their 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@cs.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

Typically, in engineering applications, we need to make decisions under uncertainty. In addition to measurement errors, some uncertainty comes from the fact that we do not know how exactly the engineering devices that we produced will be used: e.g., 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 $l_i \le L_i$.

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 – 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 is complex, this leads to a lot of computation time. To speed up the processing, 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 – so for large n, these methods are much faster.

In real life, we often do not know the exact probability distribution of measurement errors; we also do not know the distribution of user loads – and if we knew, 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. 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 $[0, L_i]$.

In such situations, we can use sensitivity analysis, we can use interval techniques – but for large n, this takes too long. To speed up, we developed a new Monte-Carlo-type technique

for processing interval uncertainty (Trejo and Kreinovich, 2001; Kreinovich and Ferson, 2004).

In this paper, we will describe this new technique, discuss its applications to engineering problems, describe its limitations, and explain how these limitations can be overcome.

2. Formulation of the Problem

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 this 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 of 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 values x_i of the measured quantities may somewhat differ from the measured values. 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 value $y = f(x_1, \ldots, x_n)$ of the desired quantity. In other words, uncertainty in the inputs leads to the uncertainty in the output as well. It is therefore desirable to estimate this output uncertainty. So, we arrive at the following problem:

- *We know:*
  - the algorithm $f(x_1, \ldots, x_n)$;
  - the measured values $\widetilde{x}_1, \ldots, \widetilde{x}_n$; and
  - the information about the uncertainty $\Delta x_i \stackrel{\text{def}}{=} \widetilde{x}_i - x_i$ of each direct measurement.
- *We must estimate:* uncertainty $\Delta y = \widetilde{y} - y$ of the algorithm's output.

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 measurement error Δx_i.

We do not know the exact values of the measurement errors Δx_i; as a result, in real life, we may have (and often we do have) several situations in which we get exactly exactly the same measurement results $\widetilde{x}_1, \ldots, \widetilde{x}_n$, but the actual values $x_1, \ldots, x_n$ of the measured quantity are different. Thus, to describe the uncertainty, we need to know:

- what are the possible values of Δx_i, and
- how often can different possible values occur.

In the ideal case, when we have a complete description of uncertainty, we know the exact frequency (probability) of all possible error combinations $(\Delta x_1, \ldots, \Delta x_n)$. In other words,

we know the exact probability distribution of the set of all n-dimensional vectors $\Delta x = (\Delta x_1, \ldots, \Delta x_n)$. Often, the measurement errors corresponding to different measurements are independent, so it is sufficient to know the distribution of each variable x_i. This distribution can be described, e.g., by a cumulative density function (cdf) $F_i(t) \stackrel{\rm def}{=} {\rm Prob}(x_i \le t)$.

Most traditional methods of processing uncertainty in science and engineering (see, e.g., (Wadsworth, 1990)) are based on the assumption that we have a *probabilistic uncertainty*, i.e., that the error distributions are independent, and that we know the probability distribution $F_i(t)$ for each of the variables x_i. However, in real life, we often do not have all this information.

In some real-life situations, we do not have any information about the frequency of different measurement error Δx_i; all we know is the range $[\Delta_i^-, \Delta_i^+]$ of possible values of this error. In this case, the only information that we have about the actual measured 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 $\underline{x}_i \stackrel{\rm def}{=} \widetilde{x}_i - \Delta_i^+$ and $\overline{x}_i \stackrel{\rm def}{=} \widetilde{x}_i - \Delta_i^-$. The corresponding uncertainty is called *interval uncertainty*; see, e.g., (Moore, 1979; Kearfott, 1996; Kearfott and Kreinovich, 1996; Jaulin et al., 2001).

So far, we have describe two extreme situations:

- in the case of probabilistic uncertainty, we have a complete information on which values Δx_i are possible, and what are the frequencies of different possible values;
- in the case of interval uncertainty, we only know the range of possible values of Δx_i, we do not have any information about the frequencies at all.

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

How can we describe such situations? To describe the complete information about the probabilities of different values of 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) (Ferson, 2002).

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

So, p-boxes are the most general way of representing these types of uncertainty.

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 intervals $[\underline{x}_i^{(k)}, \overline{x}_i^{(k)}]$ with probabilities $p_i^{(k)}$ attached to each such interval (so that for every i, $p_1^{(k)} + p_2^{(k)} + \ldots = 1$). For example, we may have several experts who provide us with different intervals $[\underline{x}_i^{(k)}, \overline{x}_i^{(k)}]$, and $p_i^{(k)}$ is the probability that k-th expert is right. The collection of intervals with probabilities attached to different intervals constitutes a DS knowledge base.

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;
- we know the DS knowledge bases

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

  that characterize the distribution of each variable x_i, we know that the corresponding distributions are independent, and we must find the DS 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 reasonable to consider the formulations in which the corresponding distributions may be dependent.

There exist efficient methods for solving these problems; see, e.g., (Ferson, 2002) and references therein (in particular, for interval uncertainty, see (Moore, 1979; Kearfott, 1996; Kearfott and Kreinovich, 1996; Jaulin et al., 2001)). Many of these 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.

In several practical situations, however, 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. It is worth mentioning that 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 model $f_{\rm approx}(x_1, \ldots, x_n) \approx f(x_1, \ldots, x_n)$, and

– then, we use this approximate 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 $f_{\rm approx}$.

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

Probabilistic uncertainty: Monte-Carlo techniques. Let us first consider the 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 the 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 $y^{(k)} = f(x_1^{(1)}, \ldots, x_n^{(k)})$.

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.

Interval uncertainty: case of linearization. Let us now consider the case of interval uncertainty.

In the interval case, we have intervals $[\underline{x}_i, \overline{x}_i]$ of possible values of each input x_i, and we are interested in finding the corresponding interval $[\underline{y}, \overline{y}]$ of possible values of y.

It is convenient to represent each interval $[\underline{x}_i, \overline{x}_i]$ by its midpoint $x_i^{\rm mid} \stackrel{\rm def}{=} \dfrac{\underline{x}_i + \overline{x}_i}{2}$ and by its half-width $\Delta_i \stackrel{\rm def}{=} \dfrac{\overline{x}_i - \overline{x}_i}{2}$, so that each such interval takes the form $[x_i^{\rm mid} - \Delta_i, x_i^{\rm mid} + \Delta_i]$. In this representation, instead of the original variables x_i that take values from $\underline{x}_i$ to $\overline{x}_i$, it is often convenient to consider auxiliary variables $\delta x_i \stackrel{\rm def}{=} x_i - x_i^{\rm mid}$ that take values from $-\Delta_i$ to Δ_i.

When the function $f(x_1, \ldots, x_n)$ is reasonable 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:

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

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

Interval uncertainty: sensitivity analysis. For small n, we can use the following *sensitivity analysis* method – a method that is applicable not only for approximately linear functions $f(x_1, \ldots, x_n)$, but also for all functions that are monotonic (increasing or decreasing) with respect of each of its variables. Specifically, in the sensitivity analysis method:

- First, we apply f to the results $\widetilde{x}_1, \ldots, \widetilde{x}_n$ of direct measurements, 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 $x_i^+ = \underline{x}_i$.

The main disadvantage 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 is complex, this leads to a lot of computation time.

Interval case: Cauchy deviates method. One way to speed up computations is to use the following Cauchy deviate method. This method works when the function $f(x_1, \ldots, x_n)$ is reasonable 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.

This method uses 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 we 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.

Simulation 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 $[0, 1]$.

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 $\left[0, \max\left|\delta y^{(k)}\right|\right]$.

How many iterations do we need for the ideal estimate. In (Trejo and Kreinovich, 2001; Kreinovich and Ferson, 2004), we found the number of iterations N that would provide the desired accuracy (usually, 20% accuracy in estimating Δ). The difference between the actual value Δ and its estimate $\widetilde{\Delta}$ is distributed, for large N, according to normal distribution, with 0 mean and standard deviation $\sigma_e = \Delta \cdot \sqrt{2/N}$. Thus, e.g., to get a 20% accuracy $0.2 \cdot \Delta$ with 95% certainty (corresponding to $2\sigma_e$), we need $N = 200$ runs.

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

Thus, 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.

4. Applications: Brief Overview

We have applied the Cauchy deviate techniques to the following engineering examples:

- Environmental and power engineering: safety analysis of complex systems (Kreinovich and Ferson, 2004). 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.

- Civil engineering: building safety. This example is similar to the models considered in (Muhanna and Mullen, 2001; Muhanna and Mullen, 2001a) 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.

- Petroleum and geotechnical engineering: estimating the uncertainty of the solution to the inverse problem caused by the measurement errors (Doser et al., 1998). 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.

In all these cases, we got reasonable estimates:

- In the environmental and civil engineering applications, we got the same results as sensitivity analysis, but much faster.

– In geotechnical engineering, 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.

5. Limitations of the Existing Cauchy Deviate Techniques and How These Limitations Can Be Overcome

5.1. Limitations

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 this section, we describe how we can modify the Cauchy deviate technique so as to overcome these limitations.

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

The possibility of such experiments is caused by the fact that, 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 Cauchy-distributed random variable with parameter $\Delta_0 = 1$ by the number Δ. Thus, it is sufficient to test the method on 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 $\left[0, \max_i \left|\delta y_i^{(m)}\right|\right]$.

– After that, we sort the values $\widetilde{\Delta}_m$ into an increasing sequence

$$\widetilde{\Delta}_{(1)} \le \ldots \le \widetilde{\Delta}_{(M)}.$$

– 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 $k_{\text{norm}} \approx 1 + 2\cdot\sqrt{2/20} \approx 1.7$.

– For $N = 10$, we get $k \approx 2.1$, which is slightly higher than the Gaussian-based formula $k_{\text{norm}} \approx 1 + 2\cdot\sqrt{2/10} \approx 1.9$.

– For $N = 5$, we get $k \approx 5$, which is already much higher than the Gaussian-based value $k_{\rm norm} \approx 1 + 2 \cdot \sqrt{2/5} \approx 2.3$.

5.3. p-Boxes and Dempster-Shafer Knowledge Bases: An Idea

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?

Why this problem is difficult. One reason why this problem is difficult is that it is not even clear how we can represent the DS knowledge base corresponding to the output.

Indeed, a DS knowledge base for each input variable x_i means that we may have different intervals $[\underline{x}_i^{(k)}, \overline{x}_i^{(k)}]$, with different probabilities $p_i^{(k)}$. For each combination of intervals, $[\underline{x}_1^{(k_1)}, \overline{x}_1^{(k_1)}], \ldots, [\underline{x}_n^{(k_n)}, \overline{x}_n^{(k_n)}]$, we can use the known techniques to find the corresponding interval $[\underline{y}^{(k_1,\ldots,k_n)}, \overline{y}^{(k_1,\ldots,k_n)}]$ for the output. Since we know the probability $p_i^{(k_i)}$ of each interval $[\underline{x}_i^{(k_i)}, \overline{x}_i^{(k_i)}]$, and we assume that these probabilities are independent, we can compute the probability $p^{(k_1,\ldots,k_n)}$ of the corresponding output interval as the product $p^{(k_1,\ldots,k_n)} = p_1^{(k_1)} \cdot \ldots \cdot p_n^{(k_n)}$.

At first glance, this may sound like a reasonable solution to our problem, but in reality, this solution is not practical at all: 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 $[\underline{y}^{(k_1,\ldots,k_n)}, \overline{y}^{(k_1,\ldots,k_n)}]$.

Thus, although the resulting uncertainty is still a DS uncertainty, we can no longer represent it as we represented the uncertainty for each input: by listing all the intervals and the corresponding probabilities.

Thus, not only it is not clear how to compute the resulting uncertainty, it is not even clear what exactly we want to compute.

Can we use the fact that DS 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 possible intervals $[\underline{x}_i^{(k)}, \overline{x}_i^{(k)}]$, with different probabilities $p_i^{(k)}$. For the interval uncertainty, in a realistic case when 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. The fact that the black-box function is linearizable means that we have $f(x_1, \ldots, x_n) = \widetilde{y} + \sum_{i=1}^{n} c_i \cdot (x_i - \widetilde{x}_i)$, where $\widetilde{y} \stackrel{\rm def}{=} f(\widetilde{x}_n, \ldots, \widetilde{x}_n)$ 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.

If we know the exact values $x_1, \ldots, x_n$ of all the inputs, then we can simply plug in the values x_i and get the desired value.

If for each i, we know the interval $[x_i^{\rm mid} - \Delta_i, x_i^{\rm mid} + \Delta x_i]$, 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:

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

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

In the Dempster-Shafer case, for each i, instead of a single pair $(y_i^{\rm mid}, \Delta_i)$, we have different pairs with different probabilities. Due to the formulas (2) and (3), the vector $(y^{\rm mid}, \Delta)$ is a linear combination of the vectors $(y_i^{\rm mid}, \Delta_i)$ corresponding to different inputs x_i.

If one of these vectors was prevailing, then we would have a single input (or a few dominating inputs), and there would be no need to consider the uncertainty in all n inputs. Thus, the only case when this problem makes sense is when the contributions of all n vectors is approximately of the same size (or at least the same order of magnitude). In this case, the vector $(y^{\rm mid}, \Delta)$ is a linear combination of n independent vectors of approximately the same size.

This situation is exactly the case covered by the Central Limit Theorem, the case when in the limit $n \to \infty$, we have a normal 2-D distribution and hence, for sufficient large n, with a good approximation, we can assume that the pair $(y^{\rm mid}, \Delta)$ 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 $(y^{\rm mid}, \Delta)$ 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.

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 $[y^{\rm mid} - \Delta, y^{\rm mid} + \Delta]$, and the probability of an interval is described by a 2-D normal distribution on the $(y^{\rm mid}, \Delta)$ plane.

To describe a 2-D normal distribution, it is sufficient to know 5 parameters: the means and standard deviations of both variables and the covariance (that describes their dependence).

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?

How can we determine the parameters of this model? If we simply take the midpoints $x_i^{(k){\rm mid}}$ of the corresponding intervals in our simulations, then the resulting value $y^{(k)}$ are normally distributed, with the distribution corresponding to $y^{\rm mid}$. We can therefore estimate the mean and standard deviation of $y^{\rm mid}$ as simply the sample mean and the sample variance of the values $y^{(1)}, y^{(2)}, \ldots$

For Δ, from the formula (3), we conclude that

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

and

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

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

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

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 $\delta y^{(k)}$.

We thus know how to estimate 4 of 5 parameters that describe the desired uncertainty. The only remaining problem is how to estimate the covariance between $y^{\rm mid}$ and Δ. For this, we propose the following idea.

The non-zero covariance means, in particular, that the conditional average $E[\Delta \,|\, y^{\rm mid} \le E[y^{\rm mid}]]$ of Δ over the cases when $y^{\rm mid}$ is smaller than its average $E[y^{\rm mid}]$ is different from the conditional average $E[\Delta \,|\, y^{\rm mid} \ge E[y^{\rm mid}]]$ of Δ over the cases when $y^{\rm mid}$ is larger than its average $E[y^{\rm mid}]$. From the difference between these two conditional averages, we can determine the desired value of the covariance.

To compute the conditional averages, we can use the Cauchy deviates idea. Namely, at each simulation, for each variable x_i, we select one of the intervals $[\underline{x}_i^{(k)}, \overline{x}_i^{(k)}]$ with the corresponding probability $p_i^{(k)}$, and we apply the black box function f to the centers of the corresponding intervals, to get the result $y^{\rm mid}$. 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 separate the results of Cauchy simulation into two different samples:
  - a sample containing all the cases in which $y^{\rm mid} \le E[y^{\rm mid}]$, and
  - a sample containing all the cases in which $y^{\rm mid} \ge E[y^{\rm mid}]$.

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.

For each sample, we can safely assume that the distribution of the width Δ is a Gaussian distribution, with mean μ and standard deviation σ. Thus, our 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:

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

By separating the full square in the integrated expression, one can show that 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 $\omega_1,\ldots,\omega_k>0$, compute $l(\omega_k)\stackrel{\text{def}}{=}-\ln(c(\omega_k))$, where $c(\omega_k)\stackrel{\text{def}}{=}\frac{1}{N}\cdot\sum\limits_{k=1}^{N}\cos(\omega\cdot y^{(k)})$;
- use the Least Squares Method to find the values μ and σ for which

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

The resulting value μ is the average Δ.

Thus, when we repeat this algorithm for both samples, we get the two desired conditional averages of Δ – from which we can then compute the covariance.

What about p-boxes? It is known that a p-box can be described as a DS 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 $p=0,\Delta p,2\cdot\Delta p,\ldots,1.0$ (e.g., for $p=0,0.1,0.2,\ldots,0.9.1.0$), 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 (Ferson, 2002).

Because of this description, we can interpret the p-box as the DS 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 DS 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 DS knowledge base, and apply the above DS-based method to describe th uncertainty of the output. The result, however, is a DS knowledge base. How can we describe the corresponding "Gaussian" DS knowledge base as a p-box?

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

- The probability $F(t) = \mathrm{Prob}(X \le t)$ attains its largest possible value $\overline{F}(t)$ if for each interval, we take the smallest possible value $\underline{x}$.
- Similarly, the probability $F(t) = \mathrm{Prob}(X \le t)$ 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 $y^{\min} - \Delta$, and
- $\underline{F}(t)$ is a probability distribution for the upper endpoints $y^{\min} + \Delta$ of the corresponding intervals.

Since the 2-D distribution of the pairs $(y^{\mathrm{mid}}, \Delta)$ is Gaussian, the distributions of both linear combinations $y^{\min} - \Delta$ and $y^{\min} + \Delta$ 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 $\underline{F}(t) \le \overline{F}(t)$, because, due to monotonicity of $F_0(t)$, this would be equivalent to $\dfrac{t-\underline{\mu}}{\underline{\sigma}} \le \dfrac{t-\overline{\mu}}{\overline{\sigma}}$ 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 DS 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 $\underline{F}(t) \le \overline{F}(t)$ 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 (Ferson, 2002) – hence, to DS knowledge bases – and use a similar approach.

5.4. 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

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

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

In general, computing the exact bound for a quadratic function of n variables in case of interval uncertainty is an NP-hard problem (Vavasis, 1991; Kreinovich et al., 1997). Luckily, in many practical case, the dependence of f on x_i is monotonic (see, e.g., (Lakeyev and Kreinovich, 1995)), 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 when 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 $\delta y_{\rm min}$ of δy is attained when $\delta x_i = -\varepsilon_i \cdot \Delta_i$, hence $\Delta^- \stackrel{\rm def}{=} |\delta y_{\rm min}|$ 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:

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

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

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

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

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

and similarly,

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

hence

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

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 $c_i+2\cdot\sum\limits_{j=1}^{n}c_{ij}\cdot z_j$ at the point

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

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

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

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 $g(0)=\sum\limits_{i=1}^{n}|c_i|\cdot\Delta_i$.

¿From (12) and (14), we conclude that

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

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

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

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

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 $\dfrac{\partial f}{\partial x_i}$ has the same sign at all the points from the box. Hence, the sign of the derivative $c_i + \sum\limits_{j=1}^{n} c_{ij} \cdot \varepsilon_j \cdot \Delta_j$ 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 $x^{\text{mid}} = (x_1^{\text{mid}}, \ldots, x_n^{\text{mid}})$ of the box. Since $|E| = \text{sign}(E) \cdot E$ for every expression E, we thus conclude that

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

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 $t \to \dfrac{1}{2} \cdot (f(x^{\text{mid}} + z + t) - f(x^{\text{mid}} + z - t))$ 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 $h(z) = \dfrac{1}{2} \cdot (g(z) - g(0) + f(x^{\text{mid}} + z) - f(x^{\text{mid}} - z))$; 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.

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 the anonymous referees for helpful suggestions.

References

Doser, D. I., K. D. Crain, M. R. Baker, V. Kreinovich, and M. C. Gerstenberger Estimating uncertainties for geophysical tomography, *Reliable Computing*, 1998, 4(3)241–268.

Ferson, S. *RAMAS Risk Calc 4.0: Risk Assessment with Uncertain Numbers*, CRC Press, Boca Raton, Florida, 2002.

Kreinovich, V., and S. Ferson, A New Cauchy-Based Black-Box Technique for Uncertainty in Risk Analysis, *Reliability Engineering and Systems Safety*, 2004, 85(1–3): 267-279.

Kreinovich, V., A. Lakeyev, J. Rohn, and P. Kahl, *Computational Complexity and Feasibility of Data Processing and Interval Computations*, Kluwer, Dordrecht, 1997.

Lakeyev, A. V., and V. Kreinovich, 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.

Jaulin L., M. Keiffer, O. Didrit, and E. Walter, *Applied Interval Analysis*, Springer-Verlag, Berlin, 2001.

Kearfott, R. B., *Rigorous Global Search: Continuous Problems*, Kluwer, Dordrecht, 1996.

Kearfott R. B., and V. Kreinovich, editors, *Applications of Interval Computations*, Kluwer, Dordrecht, 1996.

Moore, R. E., *Methods and Applications of Interval Analysis*, SIAM, Philadelphia, 1979.

Muhanna, R. L., and R. L. Mullen, Uncertainty in mechanics problems – interval-based approach, *ASCE Journal of Engineering Mechanics*, 2001, 127(6):557–566.

Muhanna, R. L., and R. L. Mullen, Treatment in geometric tolerances in finite element analysis, *International Journal of Advanced Manufacturing Systems*, 2001, 127(6):557–566.

Trejo, R., and V. Kreinovich, Error Estimations for Indirect Measurements, In: S. Rajasekaran et al., editors, *Handbook on Randomized Computing*, Kluwer, 2001, pp. 673–729.

Vavasis, S. A., *Nonlinear Optimization: Complexity Issues*, Oxford University Press, N.Y., 1991.

Wadsworth, H. M. Jr., editor, *Handbook of statistical methods for engineers and scientists*, McGraw-Hill Publishing Co., N.Y., 1990.