How to Efficiently Propagate P-Box Uncertainty

Olga Kosheleva and Vladik Kreinovich

University of Texas at El Paso
500 W. University
El Paso, Texas 79968, USA
olgak@utep.edu, vladik@utep.edu
1. Need for uncertainty propagation: a brief reminder

- In some cases, all we need is measurement results.
- However, in most cases, we are interested in something else.
- In this case:
  - we apply some algorithm \( f \) to the measurement results \( x_1, \ldots, x_n \), and
  - we get the desired estimates or recommended control values
    \[
    y = f(x_1, \ldots, x_n).
    \]
- The values \( x_i \) are only known with uncertainty.
- Therefore, the result \( y \) also comes with uncertainty.
- Determining this uncertainty based on uncertainties in \( x_i \) is known as uncertainty propagation.
2. Need for p-boxes

- In the ideal case, we know the probability distribution of each measurement error $\Delta x_i$.

- There are many ways to represent a probability distribution: by the probability density function (pdf), by moments, etc.

- Most of these representations are not universal; examples:
  - some distributions do not have finite moments – e.g., Cauchy distribution;
  - some distributions do not have the probability density function – e.g., distribution located at a single value with probability 1.

- The only universal representation is by using a cumulative distribution function (cdf)

\[ F_i(X_i) \equiv \text{Prob}(\Delta x_i \leq X_i). \]
3. Need for p-boxes (cont-d)

- In many real-life cases, we only have partial information about the probabilities.
- This means that for each $X_i$:
  - instead of knowing the exact value $F_i(X_i)$,
  - we only have partial information about $F_i(X_i)$.
- Usually, possible values of $F_i(X_i)$ form an interval $[F_i(X_i), \overline{F}_i(X_i)]$.
- So, a natural way to describe such cases is to have a function that assigns such interval to each $X_i$.
- This function is known as a *probability box*, or *p-box*, for short.
4. Uncertainty propagation under p-boxes: a challenge

- In the ideal case, when we know all the probability distributions, we can use the usual Monte-Carlo (MC) approach:
  - we simulate each input,
  - we plug in the simulation results into $f$, getting a sample of $y$’s;
  - based on this sample, we determine $y$’s cdf.

- In the case of p-box uncertainty, there are many possible distributions for each $x_i$.

- Even if we consider 2 values for each of $N$ points $X_1, \ldots, X_N$, this means $2^N$ options, which is not feasible.

- There exist feasible algorithms for propagating p-box uncertainty for many important cases.

- However, there is no general efficient algorithm for such propagation.
5. Analysis of the problem

- Probability estimates are usually reasonably accurate.
- Thus, terms which are quadratic (or of higher order) in terms of estimation errors $\Delta F(X) \overset{\text{def}}{=} F(X) - \tilde{F}(x)$ can be safely ignored.
- So, we can assume that the data processing algorithm is linear in terms of $\Delta F(X)$.
- Thus, $y$ is a linear function of the values $F(x)$.
- Instead of all infinitely many values $F(x)$, we can take values $F(X_i)$ corresponding to a dense grid $X_1 < X_2 < \ldots < X_N$.
- Then, for some $a_i$, we have:

$$y = a_0 + \sum_{i=1}^{N} a_i \cdot F(X_i).$$
6. What we propose

- For each $i = 0, 1, \ldots, N$, we form $F^{(i)}(X)$ for which:
  - we have $F^{(i)}(X_j) = F(X_j)$ for $j \leq i$, and
  - we have $F^{(i)}(X_j) = \overline{F}(X_j)$ for $j > i$.

- We use Monte-Carlo (or any other) method to find the value $y^{(i)}$ corresponding to $F^{(i)}(X)$.

- Because of linearity, we have $y^{(i)} - y^{(i-1)} = a_i \cdot (\overline{F}(X_i) - \overline{F}(X_i))$, so we can estimate $a_i$ as

$$a_i = \frac{y^{(i)} - y^{(i-1)}}{\overline{F}(X_i) - \overline{F}(X_i)}.$$

- After that, we use the estimate $y^{(0)}$ for $F^{(0)}(X)$ to estimate $a_0$ as

$$a_0 = y^{(0)} - \sum_{i=1}^{N} \overline{F}(X_i).$$
7. What we propose

- Now, we can estimate the range $[y, \bar{y}]$ of all possible values $y$ for the p-box by solving two linear programming problems:

\[
a_0 + \sum_{i=1}^{N} a_i \cdot F_i \rightarrow \min(\max)
\]

under the conditions

\[
\underline{F}(X_i) \leq F_i \leq \bar{F}(X_i) \text{ and } F_i \leq F_{i+1}.
\]

- This procedure requires $N + 1$ calls to estimating $y$, which is feasible.

- Linear programming is also feasible: it takes $O(N^{2+\varepsilon})$ computational steps, where $\varepsilon = 1/18$. 
8. What if we have several p-box inputs?

- In this case, the linear dependence is over all the values $F_j(X_i)$:

$$y = a_0 + \sum_j t_j, \text{ where } t_j \overset{\text{def}}{=} \sum_i a_{ij} \cdot F_j(X_i).$$

- Here, for each $j$, we have separate constraints – bounds on $F_j$ and monotonicity.

- Thus, to find $\min y$ and $\max \bar{y}$ of $y$, it is sufficient to:
  - use linear programming to find $\min t_j$ and $\max \bar{t}_j$ of each $t_j$, and
  - compute $\underline{y} = a_0 + \sum_j t_j$ and $\bar{y} = a_0 + \sum_j \bar{t}_j$. 
9. How many calls to $f$ do we need to reach given accuracy $\varepsilon$

- Let $\Delta$ denote the size of $\Delta F(X) = \overline{F}(X) - \underline{F}(X)$.
- So, in linear approximation, the difference $\overline{y} - \underline{y}$ is proportional to $\Delta$.
- Let $\varepsilon$ be the relative accuracy with which we want to estimate this difference.
- For example, we can take $\varepsilon = 20\%$:
  - remember, this is accuracy with which we determine accuracy;
  - measuring instrument can have accuracy 10%, but 11.6% accuracy does not make too much practical sense.
- This means that we need absolute accuracy $\varepsilon \cdot \Delta$.
- In general, if we use values at $N$ points, a monotonic function is represented with accuracy $\sim 1/N$.
- Thus, we need to have $N \sim 1/(\varepsilon \cdot \Delta)$. 

10. How many calls to $f$ do we need (cont-d)

- Let $\delta$ be the accuracy with which we determine each value $y^{(i)}$.
- Linear dependence can be described as $y = b_0 + \sum_i b_i \cdot (F_i - F_{i-1})$.
- Each term in this sum is close to $y^{(i)} - y^{(i-1)}$.
- Thus, the accuracy of each term is approximately equal to $\delta$.
- The standard deviation of the sum of $N$ independent terms grows as $\sqrt{N}$.
- So, the accuracy with which we determine $y$ is $\delta \cdot \sqrt{N}$.
- Thus, to reach accuracy $\varepsilon \cdot \Delta$, we need to select $\delta = \varepsilon \cdot \Delta / \sqrt{N}$.
- Let $M$ denote the number of calls to $f$ that we use to estimate each $y^{(i)}$.
- In general, $M$ iterations provide relative accuracy $\sim 1/\sqrt{M}$.
11. How many calls to $f$ do we need (cont-d)

- To get $1/\sqrt{M} \sim \delta = \varepsilon \cdot \Delta/\sqrt{N}$, we thus need:
  
  $$M \sim \varepsilon^{-2} \cdot \Delta^{-2} \cdot N \sim \varepsilon^{-3} \cdot \Delta^{-2} \text{ calls to } f.$$ 

- We need to compute $N \sim \varepsilon^{-1} \cdot \Delta^{-1}$ values $y^{(i)}$.

- Thus, overall, we need
  
  $$N \cdot M \sim (\varepsilon^{-1} \cdot \Delta^{-1}) \cdot (\varepsilon^{-3} \cdot \Delta^{-2}) = \varepsilon^{-4} \cdot \Delta^{-3} \text{ calls to } f.$$ 

- This is indeed feasible.
12. Acknowledgments

This work was supported in part by:

- National Science Foundation grants 1623190, HRD-1834620, HRD-2034030, and EAR-2225395;
- AT&T Fellowship in Information Technology;
- program of the development of the Scientific-Educational Mathematical Center of Volga Federal District No. 075-02-2020-1478, and
- a grant from the Hungarian National Research, Development and Innovation Office (NRDI).