Sometimes, It Is Beneficial to Process Different Types of Uncertainty Separately

Chrysostomos D. Stylios[1], Andrzej Pownuk[2], and Vladik Kreinovich[2]

[1]Laboratory of Knowledge and Intelligent Computing
Department of Computer Engineering
Technological Educational Institute of Epirus
47100 Kostakioi, Arta, Greece, stylios@teiep.gr

[2]Computational Science Program
University of Texas at El Paso, 500 W. University
El Paso, Texas 79968, USA
ampownuk@utep.edu, vladik@utep.edu

1. Need for Data Processing

- One of the main objectives of science is to predict future values y of physical quantities:
  - in meteorology, we need to predict future weather;
  - in airplane control, we need to predict the location and the velocity of the plane under current control.
- To make this prediction:
  - we need to know the relation $y = f(x_1, \ldots, x_n)$ between y and related quantities $x_1, \ldots, x_n$;
  - then, we measure or estimate $x_1, \ldots, x_n$;
  - finally, we use the results $\widetilde{x}_i$ of measurement (or estimation) to compute an estimate

$$\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n).$$

- This computation of $\widetilde{y}$ is an important case of *data processing*.

2. Need to Take Uncertainty into Account

- Measurements are never absolutely accurate, and expert estimates are even less accurate.
- As a result, the estimates $\widetilde{x}_i$ are, in general, different from the actual (unknown) values x_i.
- Therefore, the estimate $\widetilde{y}$ is also only approximate.
- In practice, it is desirable to know how accurate is this estimate $\widetilde{y}$.
- To find this out, we need to take into account the accuracy of the estimates $\widetilde{x}_i$.
- For measurements, we usually know the upper bound Δ_i on the absolute value of the measurement error:

$$|\Delta x_i| \le \Delta_i, \text{ where } \Delta x_i \stackrel{\text{def}}{=} \widetilde{x}_i - x_i.$$

- The upper bound Δ_i is usually provided by the manufacturer of the measurement instrument.

3. Taking Uncertainty into Account (cont-d)

- Once we know Δ_i and $\widetilde{x}_i$, then we know that the actual value x_i is located in the interval $[\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i]$.
- To gauge the accuracy of expert estimates, it is reasonable to use fuzzy techniques, i.e., to describe:
  - for each possible value x_i,
  - the degree $\mu_i(x_i)$ to which x_i is possible.
- Sometimes, we also know the probabilities of different $\Delta x_i \in [-\Delta_i, \Delta_i]$; we plan to analyze this in the future.
- The prediction model is often approximate:

$$y = f(x_1, \ldots, x_n) + \Delta m, \text{ with } \Delta m \neq 0.$$

- Sometimes, we know the upper bound Δ_m on the model inaccuracy Δm: $|\Delta m| \le \Delta_m$.
- In other cases, we know a membership function $\mu_m(\Delta m)$ that describes Δm.

4. Measurement and Estimation Inaccuracies Are Usually Small

- In many practical situations, the measurement and estimation inaccuracies Δx_i are relatively small.
- Then, we can safely ignore terms which are quadratic (or of higher order) in terms of Δx_i:

$$\Delta y = \widetilde{y} - y = f(\widetilde{x}_1, \ldots, \widetilde{x}_n) - f(\widetilde{x}_1 - \Delta x_1, \ldots, \widetilde{x}_n - \Delta x_n) - \Delta m =$$

$$\sum_{i=1}^n c_i \cdot \Delta x_i - \Delta m, \text{ where } c_i = \frac{\partial f}{\partial x_i}.$$

- If needed, the derivative can be estimated by numerical differentiation

$$c_i \approx \frac{f(\widetilde{x}_1, \ldots, \widetilde{x}_{i-1}, \widetilde{x}_i + h, \widetilde{x}_{i+1}, \ldots, \widetilde{x}_n) - \widetilde{y}}{h}.$$

5. Estimating Accuracy of Data Processing

- The value $\Delta y = \sum\limits_{i=1}^{n} c_i \cdot \Delta x_i - \Delta m$ is the largest when each term is the largest, so $\Delta = \sum\limits_{i=1}^{n} |c_i| \cdot \Delta_i + \Delta_m$.
- In the fuzzy case, the similar formula holds for the α-cuts, for every α: ${}^{\alpha}\Delta = \sum\limits_{i=1}^{n} |c_i| \cdot {}^{\alpha}\Delta_i + {}^{\alpha}\Delta_m$.
- Experts cannot describe their degrees of confidence α with too much accuracy.
- Usually, it is sufficient to consider only eleven values $\alpha = 0.0$, $\alpha = 0.1$, $\alpha = 0.2$, ..., $\alpha = 0.9$, and $\alpha = 1.0$.
- Thus, we need to apply the above formula eleven times.
- This is in line with the fact we usually divide each quantity into 7 ± 2 categories (Miller's "7 ± 2 Law").
- So, it is sufficient to have at least 9 different categories.

6. Cases for Which Simplification Is Possible

- Sometimes, all membership functions are "of the same type": $\mu(z) = \mu_0(k \cdot z)$ for some symmetric $\mu_0(z)$.
- *Example:* for triangular functions,

$$\mu_0(z) = \max(1 - |z|, 0).$$

- In this case, $\mu(z) \geq \alpha$ is equivalent to $\mu_0(k \cdot z) \geq \alpha$, so ${}^{\alpha}\Delta_0 = k \cdot {}^{\alpha}\Delta$ and ${}^{0}\Delta_0 = k \cdot {}^{0}\Delta$.
- Thus, ${}^{\alpha}\Delta = f(\alpha) \cdot {}^{0}\Delta$, where $f(\alpha) \stackrel{\text{def}}{=} \dfrac{{}^{\alpha}\Delta_0}{{}^{0}\Delta_0}$.
- For example, for a triangular membership function, we have $f(\alpha) = 1 - \alpha$.
- So, if we know the type μ_0 (hence $f(\alpha)$), and we know the 0-cut, we can compute all α-cuts as ${}^{\alpha}\Delta = f(\alpha) \cdot {}^{0}\Delta$.
- So, if $\mu_i(\Delta x_i)$ and $\mu_m(\Delta m)$ are of the same type, then ${}^{\alpha}\Delta_i = f(\alpha) \cdot {}^{0}\Delta_i$ and ${}^{\alpha}\Delta_m = f(\alpha) \cdot {}^{0}\Delta_m$ for all α.

7. When Simplification Is Possible (cont-d)

- We know that ${}^\alpha\Delta = \sum\limits_{i=1}^n |c_i| \cdot {}^\alpha\Delta_i + {}^\alpha\Delta_m$.
- For ${}^\alpha\Delta_i = f(\alpha) \cdot {}^0\Delta_i$ and ${}^\alpha\Delta_m = f(\alpha) \cdot {}^0\Delta_m$, we get

$$ {}^\alpha\Delta = \sum_{i=1}^n |c_i| \cdot f(\alpha) \cdot {}^0\Delta_i + f(\alpha) \cdot {}^0\Delta_m. $$

- So, ${}^\alpha\Delta = f(\alpha) \cdot \left(\sum\limits_{i=1}^n |c_i| \cdot {}^0\Delta_i + {}^0\Delta_m\right) = f(\alpha) \cdot {}^0\Delta$.
- Thus, if all $\mu(x)$ are of the same type $\mu_0(z)$, there is no need to compute ${}^\alpha\Delta$ eleven times:
  - it is sufficient to compute ${}^0\Delta$;
  - to find all other values ${}^\alpha\Delta$, we simply multiply ${}^0\Delta$ by the factors $f(\alpha)$ corresponding to $\mu_0(z)$.

8. A More General Case

- A more general case is:
  - when we have a list of T different types of uncertainty – i.e., types of membership functions, and
  - each approximation error Δx_i consists of $\le T$ components of the corresponding type t:

$$\Delta x_i = \sum_{t=1}^{T} \Delta x_{i,t} \text{ and } \Delta m = \sum_{t=1}^{T} \Delta m_t.$$

- For example:
  - type $t = 1$ may correspond to intervals (which are, of course, a particular case of fuzzy uncertainty),
  - type $t = 2$ may correspond to triangular membership functions, etc.

9. How This Case Is Processed Now

- *First stage:*
  - we use the known membership functions $\mu_{i,t}(\Delta x_{i,t})$ and $\mu_{m,t}(\Delta m_t)$
  - to find the memberships functions $\mu_i(\Delta x_i)$ and $\mu_m(\Delta m)$ that correspond to the sums Δx_i and Δm.
- *Second stage:* we use $\mu_i(\Delta x_i)$ and $\mu_m(\Delta m)$ to compute the desired membership function $\mu(\Delta y)$.
- *Problem:* on the second stage, we apply the above formula eleven times:

$$^{\alpha}\Delta = \sum_{i=1}^{n} |c_i| \cdot {}^{\alpha}\Delta_i + {}^{\alpha}\Delta_m.$$

10. Our Main Idea

- We have $\Delta y = \sum\limits_{i=1}^{n} c_i \cdot \Delta x_i - \Delta m$, where

$$\Delta x_i = \sum_{t=1}^{T} \Delta x_{i,t} \text{ and } \Delta m = \sum_{t=1}^{T} \Delta m_t.$$

- Thus, $\Delta y = \sum\limits_{i=1}^{n} c_i \cdot \left(\sum\limits_{t=1}^{T} \Delta x_{i,t} \right) - \left(\sum\limits_{t=1}^{T} \Delta m_t \right).$

- Grouping together all the terms corr. to type t, we get $\Delta y = \sum\limits_{t=1}^{T} \Delta y_t$, where $\Delta y_t \stackrel{\text{def}}{=} \sum\limits_{i=1}^{n} c_i \cdot \Delta x_{i,t} - \Delta m_t$.

- For each t, we are combining membership functions of the same type, so it is enough to compute ${}^0\Delta_t$.

- Then, we add the resulting membership functions – by adding the corresponding α-cuts.

11. Resulting Algorithm

- Let $[-{}^0\Delta_{i,t}, {}^0\Delta_{i,t}]$ and $[-{}^0\Delta_{m,t}, {}^0\Delta_{m,t}]$ be 0-cuts of the membership functions $\mu_{i,t}(\Delta x_{i,t})$ and $\mu_{m,t}(\Delta m_t)$.
- Based on these 0-cuts, we compute, for each type t, the values ${}^0\Delta = \sum\limits_{i=1}^{n} |c_i| \cdot {}^0\Delta_{i,t} + {}^0\Delta_{m,t}$.
- Then, for $\alpha = 0$, $\alpha = 0.1$, $\ldots$, and for $\alpha = 1.0$, we compute the values ${}^\alpha\Delta_t = f_t(\alpha) \cdot {}^0\Delta_t$.
- Finally, we add up α-cuts corresponding to different types t, to come up with the expression ${}^\alpha\Delta = \sum\limits_{t=1}^{T} {}^\alpha\Delta_t$.
- *Comment.* We can combine the last two steps into a single step: ${}^\alpha\Delta = \sum\limits_{t=1}^{T} f_t(\alpha) \cdot {}^0\Delta_t$.

12. The New Algorithm Is Much Faster

- The original algorithm computed the above formula eleven times:

$$^\alpha\Delta = \sum_{i=1}^{n} |c_i| \cdot {}^\alpha\Delta_i + {}^\alpha\Delta_m.$$

- The new algorithm uses the corresponding formula T times, i.e., as many times as there are types.
- All the other computations are much faster, since they do not grow with the input size n.
- Thus, if the number T of different types is smaller than eleven, the new methods is much faster.
- *Example:* for $T = 2$ types (e.g., intervals and triangular $\mu(x)$), we get a $\dfrac{11}{2} = 5.5$ times speedup.

13. Conclusions and Future Work

- We can therefore conclude that sometimes, it is beneficial to process different types of uncertainty separately.
- Namely, it is beneficial when we have ten or fewer different types of uncertainty.
- The fewer types of uncertainty we have, the faster the resulting algorithm.
- We plan to test this idea of several actual data processing examples.
- We also plan to extend this idea to other types of uncertainty, in particular, to probabilistic uncertainty.

14. Acknowledgment

- This work was supported in part by the National Science Foundation grants:
  - HRD-0734825 and HRD-1242122 (Cyber-ShARE Center of Excellence) and
  - DUE-0926721.
- This work was performed when C. Stylios was a Visiting Researcher at the University of Texas at El Paso.