Practical Applications of Interval Computations: an Overview with a Special Emphasis on Applications Involving Probabilities

Vladik Kreinovich
Department of Computer Science
University of Texas at El Paso, 500 W. University
El Paso, TX 79968, USA, vladik@utep.edu

http://www.cs.utep.edu/vladik
http://www.cs.utep.edu/interval-comp
1. General Problem of Data Processing under Uncertainty

- **Indirect measurements**: way to measure $y$ that are difficult (or even impossible) to measure directly.

- **Idea**: $y = f(x_1, \ldots, x_n)$

![Diagram]

- **Problem**: measurements are never 100% accurate: $\tilde{x}_i \neq x_i$ ($\Delta x_i \neq 0$) hence

$$\tilde{y} = f(\tilde{x}_1, \ldots, \tilde{x}_n) \neq y = f(x_1, \ldots, y_n).$$

What are bounds on $\Delta y \overset{\text{def}}{=} \tilde{y} - y$?
2. Probabilistic and Interval Uncertainty

- *Traditional approach:* we know probability distribution for $\Delta x_i$ (usually Gaussian).

- *Where it comes from:* calibration using standard MI.

- *Problem:* sometimes we do not know the distribution because no “standard” (more accurate) MI is available. Cases:
  - fundamental science
  - manufacturing

- *Solution:* we know upper bounds $\Delta_i$ on $|\Delta x_i|$ hence

$$x_i \in [\tilde{x}_i - \Delta_i, \tilde{x}_i + \Delta_i].$$
3. Interval Computations: A Problem

- **Given:**
  - an algorithm \( y = f(x_1, \ldots, x_n) \) that transforms \( n \) real numbers \( x_i \) into a number \( y \);
  - \( n \) intervals \( x_i = [x_i, x_i] \).

- **Compute:** the corresponding range of \( y \):

  \[
  [\underline{y}, \overline{y}] = \{ f(x_1, \ldots, x_n) \mid x_1 \in [\underline{x_1}, \overline{x_1}], \ldots, x_n \in [\underline{x_n}, \overline{x_n}] \}.
  \]

- **Fact:** even for quadratic \( f \), the problem of computing the exact range \( y \) is NP-hard.

- **Practical challenges:**
  - find classes of problems for which efficient algorithms are possible; and
  - for problems outside these classes, find efficient techniques for approximating uncertainty of \( y \).
4. Why Not Maximum Entropy?

- **Situation:** in many practical applications, it is very difficult to come up with the probabilities.

- **Traditional engineering approach:** use probabilistic techniques.

- **Problem:** many different probability distributions are consistent with the same observations.

- **Solution:** select one of these distributions – e.g., the one with the largest entropy.

- **Example – single variable:** if all we know is that \( x \in [x, \bar{x}] \), then MaxEnt leads to a uniform distribution on \([x, \bar{x}]\).

- **Example – multiple variables:** different variables are independently distributed.

- **Conclusion:** if \( \Delta y = \Delta x_1 + \ldots + \Delta x_n \), with \( \Delta x_i \in [-\Delta_i, \Delta_i] \), then due to Central Limit Theorem, \( \Delta y \) is almost normal, with \( \sigma = \frac{1}{\sqrt{3}} \cdot \sqrt{\sum_{i=1}^{n} \Delta_i^2} \).

- **Why this may be inadequate:** when \( \Delta_i = \Delta \), we get \( \Delta \sim \sqrt{n} \), but due to correlation, it is possible that \( \Delta = n \cdot \Delta_i \sim n \gg \sqrt{n} \).

- **Conclusion:** using a single distribution can be very misleading, especially if we want guaranteed results – e.g., in high-risk application areas such as space exploration or nuclear engineering.
5. Chip Design: Case Study When Intervals Are Not Enough

- **One of the main objectives**: decrease the chip’s clock cycle $D$.

- **Conclusion**: it is therefore important to estimate the clock cycle on the design stage.

- **Formula – idea**: $D$ is the maximum delay over all possible paths $D \overset{\text{def}}{=} \max(D_1, \ldots, D_N)$, where $D_i$ is the sum of the delays corresponding to the gates and wires along this path.

- **Formula – details**: each $D_i$ depends on factors $x_1, \ldots, x_n$ – variation caused by the current design practices, environmental design characteristics (e.g., variations in temperature and in in supply voltage), etc. –

$$D_i = a_i + \sum_{j=1}^{n} a_{ij} \cdot x_j, \text{ so } D = \max \left( a_i + \sum_{j=1}^{n} a_{ij} \cdot x_j \right).$$

- **Traditional approach to estimating $D$**: worst-case (interval) analysis.

- **Result**: over-estimation up to 30% above the observed clock time, so chips are over-designed and under-performing.

- **Reason**: factors $x_i$ are independent, so the probability that all these factors are at their worst is extremely small.

- **Challenge**: take into account the probabilistic character of the factor variations.
6. **General Approach: Interval-Type Step-by-Step Techniques**

- **Problem:**

- **Solution:** compute an enclosure $Y$ such that $y \subseteq Y$.

- **Interval arithmetic:** for arithmetic operations $f(x_1, x_2)$, we have explicit formulas for the range.

- **Examples:** when $x_1 \in x_1 = [x_1, \bar{x}_1]$ and $x_2 \in x_2 = [\underline{x}_2, \bar{x}_2]$, then:
  - The range $x_1 + x_2$ for $x_1 + x_2$ is $[\underline{x}_1 + \underline{x}_2, \bar{x}_1 + \bar{x}_2]$.
  - The range $x_1 - x_2$ for $x_1 - x_2$ is $[\underline{x}_1 - \bar{x}_2, \bar{x}_1 - \underline{x}_2]$.
  - The range $x_1 \cdot x_2$ for $x_1 \cdot x_2$ is $[y, \bar{y}]$, where
    
    $\underline{y} = \min(\underline{x}_1 \cdot \underline{x}_2, \underline{x}_1 \cdot \bar{x}_2, \bar{x}_1 \cdot \underline{x}_2, \bar{x}_1 \cdot \bar{x}_2)$;

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

- The range $1/x_1$ for $1/x_1$ is $[1/\bar{x}_1, 1/\underline{x}_1]$ (if $0 \not\in x_1$).
7. Interval Approach: Example

- *Example:* \( f(x) = (x - 2) \cdot (x + 2), \ x \in [1, 2] \).

- How will the computer compute it?
  - \( r_1 := x - 2 \);
  - \( r_2 := x + 2 \);
  - \( r_3 := r_1 \cdot r_2 \).

- *Main idea:* do the same operations, but with *intervals* instead of *numbers*:
  - \( r_1 := [1, 2] - [2, 2] = [-1, 0] \);
  - \( r_2 := [1, 2] + [2, 2] = [3, 4] \);
  - \( r_3 := [-1, 0] \cdot [3, 4] = [-4, 0] \).

- *Actual range:* \( f(x) = [-3, 0] \).

- *Comment:* this is just a toy example, there are more efficient ways of computing an enclosure \( Y \supseteq y \).
8. Extension of Interval Arithmetic to Probabilistic Case: Successes

- **Objective:** make decisions $E_x[u(x, a)] \rightarrow \max a$.

- For smooth $u(x)$, we have $u(x) = u(x_0) + (x - x_0) \cdot u'(x_0) + \ldots$, so we must know moments to estimate $E[u]$.

- For threshold-type $u(x)$, we need cdf $F(x) = \text{Prob}(\xi \leq x)$.

- **General solution:** parse to elementary operations $+,-,\cdot,1/x,\max,\min$.

- Explicit formulas for arithmetic operations known for intervals, for p-boxes $F(x) = [\underline{F}(x), \overline{F}(x)]$, for intervals + 1st moments $E_i \overset{\text{def}}{=} E[x_i]$:
9. Successes (cont-d)

- *Easy cases:* $+,-$, product of independent $x_i$.

- *Example of a non-trivial case:* multiplication $y = x_1 \cdot x_2$, when we have no information about the correlation:
  
  \[
  E = \max(p_1 + p_2 - 1, 0) \cdot \bar{x}_1 \cdot \bar{x}_2 + \min(p_1, 1 - p_2) \cdot \bar{x}_1 \cdot \bar{x}_2 + \\
  \min(1 - p_1, p_2) \cdot \bar{x}_1 \cdot \bar{x}_2 + \max(1 - p_1 - p_2, 0) \cdot \bar{x}_1 \cdot \bar{x}_2;
  \]

  \[
  \bar{E} = \min(p_1, p_2) \cdot \bar{x}_1 \cdot \bar{x}_2 + \max(p_1 - p_2, 0) \cdot \bar{x}_1 \cdot \bar{x}_2 + \\
  \max(p_2 - p_1, 0) \cdot \bar{x}_1 \cdot \bar{x}_2 + \min(1 - p_1, 1 - p_2) \cdot \bar{x}_1 \cdot \bar{x}_2,
  \]

  where $p_i \overset{\text{def}}{=} (E_i - \bar{x}_i)/(\bar{x}_i - \bar{x}_i)$. 
10. **Challenges**

- intervals + 2nd moments:
  \[ x_1, E_1, V_1 \]
  \[ x_2, E_2, V_2 \]
  \[ \ldots \]
  \[ x_n, E_n, V_n \]

- moments + p-boxes; e.g.:
  \[ E_1, F_1(x) \]
  \[ E_2, F_2(x) \]
  \[ \ldots \]
  \[ E_n, F_n(x) \]
11. Problem

- *Result of interval-type approach:* over-estimation practically as bad as with interval computations.

- *Good news:* for $D_i = a_i + \sum a_{ij} \cdot x_j$, we use independence of $x_i$ and get reasonable p-boxes.

- *Bad news:* the values $D_i$ depends on same factors, so they are not independent.

- *Analogy:* this is similar to dependence-caused excess width in interval computations.

- *In interval computations:* methods beyond straightforward interval computations – centroid, affine, bisection – decrease excess width.

- *What we have done so far:* extended interval arithmetic to the probabilistic case.

- *What we need:* extend state-of-the-art interval computations techniques to the probabilistic case.
12. Main Idea: Use Moments

- **What we want:** find $D_0$ s.t. $D \leq D_0$ with the probability $\geq 1 - \varepsilon$ (where $\varepsilon > 0$ is a given small probability).

- **Traditional statistical analysis:** compute moments $M_v \overset{\text{def}}{=} E[D^v]$, $v = 1, 2, \ldots$

- **From moments to p-boxes – guaranteed:** Chebyshev inequality

\[
\text{Prob}(|D - M_1| > k_0 \cdot \sigma) \leq 1/k_0^2,
\]

where $\sigma \overset{\text{def}}{=} \sqrt{V} = \sqrt{M_2 - M_1^2}$.

- **Example:** for $\varepsilon = 10^{-3}$, we need $D_0 = E + 30\sigma$.

- **Problem:** $D$ is often almost normal, so $D_0 \approx E + 3\sigma$ – excess width.

- **Idea:** higher moments $D_0 = M_1 + k_{2q} \cdot \sigma_{2q}$ with $\sigma_{2q} = C_{2q}^{1/q}$ and $k_{2q} = \varepsilon^{-1/(2q)}$.

- **Example:** for $\varepsilon = 10^{-3}$, $k_2 \approx 30$, $k_4 \approx 5.5$, $k_6 \approx 3$.

- **Central moment:** $C_4 = E[(D - M_1)^4] = M_4 - 4 \cdot M_3 \cdot M_1 + 6 \cdot M_2 \cdot M_1^2 - 3 \cdot M_1^4$.

- **Interval uncertainty:** $D_0 = \overline{M}_1 + k_{2q} \cdot \overline{(C_{2q})}^{1/q}$, where

\[
\overline{C}_4 = \overline{M}_4 - 4 \cdot \overline{M}_3 \cdot \overline{M}_1 + 6 \cdot \overline{M}_2 \cdot \overline{M}_1^2 - 3 \cdot \overline{M}_1^4.
\]
13. Formulation of the Problem: Convex Case

GIVEN:

- natural numbers $n$, $k \leq n$, and $v \geq 1$;
- a function $y = F(x_1, \ldots, x_n)$ (algorithmically defined) such that for every combination of values $x_{k+1}, \ldots, x_n$, the dependence of $y$ on $x_1, \ldots, x_k$ is convex;
- $n-k$ probability distributions $x_{k+1}, \ldots, x_n$—e.g., given in the form of cumulative distribution function (cdf) $F_j(x)$, $k + 1 \leq j \leq n$;
- $k$ intervals $x_1, \ldots, x_k$, and
- $k$ values $E_1, \ldots, E_k$.

such that for every $x_1 \in [x_1, \bar{x}_1], \ldots, x_k \in [x_k, \bar{x}_k]$, we have $F(x_1, \ldots, x_n) \geq 0$ with probability 1.

TAKE: all possible joint probability distributions on $R^n$ for which:

- all $n$ random variables are independent;
- for each $j$ from 1 to $k$, $x_j \in x_j$ with probability 1 and the mean value of $x_j$ is equal $E_j$;
- for $j > k$, the variable $x_j$ has a given distribution $F_j(x)$.

FIND: for the variable $y = F(x_1, \ldots, x_n)$, find the set $M_v = [M_v, \bar{M}_v]$ of all possible values of $M_v \overset{\text{def}}{=} E[y^v]$ for all such distributions.
14. Result

- The smallest possible value $M_v$ is attained when for each $j$ from 1 to $k$, we use a 1-point distribution in which $x_j = E_j$ with probability 1.

- The largest possible values $M_v$ is attained when for each $j$ from 1 to $k$, we use a 2-point distribution for $x_j$, in which:
  
  - $x_j = x_j$ with probability $p_j = \frac{x_j - E_j}{x_j - \bar{x}_j}$.
  
  - $x_j = \bar{x}_j$ with probability $\bar{p}_j = \frac{E_j - x_j}{x_j - \bar{x}_j}$.

- Main idea – transfer: $F$ is convex and $F \geq 0$, hence $F^v$ is convex.

- Algorithm: Monte-Carlo simulations.

- Results: much smaller excess width.

- Additional result: if we also know that each distribution is unimodal.
15. Case Study: Bioinformatics

- **Practical problem:** find genetic difference between cancer cells and healthy cells.

- **Ideal case:** we directly measure concentration $c$ of the gene in cancer cells and $h$ in healthy cells.

- **In reality:** difficult to separate, so we measure \( y_i \approx x_i \cdot c + (1 - x_i) \cdot h \), where \( x_i \) is the percentage of cancer cells in \( i \)-th sample.

- **Equivalent form:** \( a \cdot x_i + h \approx y_i \), where \( a \stackrel{\text{def}}{=} c - h \).

- **If we know \( x_i \) exactly:** Least Squares Method \( \sum_{i=1}^{n} (a \cdot x_i + h - y_i)^2 \to \min_{a,h} \)
  
  hence \( a = \frac{C(x,y)}{V(x)} \) and \( h = E(y) - a \cdot E(x) \), where \( E(x) = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i \),

  \[
  V(x) = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (x_i - E(x))^2, \quad C(x,y) = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (x_i - E(x)) \cdot (y_i - E(y)).
  \]

- **Interval uncertainty:** experts manually count \( x_i \), and only provide interval bounds \( x_i \), e.g., \( x_i \in [0.7, 0.8] \).

- **Fact:** different \( x_i \in x_i \) lead to different \( a \) and \( h \).

- **Problem:** find the range of \( a \) and \( h \) corresponding to all possible values \( x_i \in [\underline{x}_i, \overline{x}_i] \).
16. General Problem

- **General problem**: how to efficiently deduce the statistical information from, e.g., interval data.

- **Example**: we know intervals $x_1 = [\underline{x}_1, \bar{x}_1], \ldots, x_n = [\underline{x}_n, \bar{x}_n]$, we want to compute the ranges of possible values of the population mean $E(x) = \frac{1}{n} \sum_{i=1}^{n} x_i$, population variance $V = \frac{1}{n} \sum_{i=1}^{n} (x_i - E(x))^2$, etc.

- **Difficulty**: in general, this problem is NP-hard even for the variance.

- **Known**:
  - efficient algorithms for $V$,
  - efficient algorithms for $V$ for reasonable situations,
  - efficient algorithms for $C(x, y)$ when intervals comes from a partition, etc.

- **Bioinformatics case**: we find intervals for $C(x, y)$ and for $V(x)$ and divide.

- **Challenges**: finding the ranges of covariance, correlation, etc., in other situations
17. Case Study: Detecting Outliers

- In many application areas, it is important to detect outliers, i.e., unusual, abnormal values.
- In medicine, unusual values may indicate disease.
- In geophysics, abnormal values may indicate a mineral deposit (or an erroneous measurement result).
- In structural integrity testing, abnormal values may indicate faults in a structure.
- Traditional engineering approach: a new measurement result $x$ is classified as an outlier if $x \not\in [L, U]$, where
  \[
  L \overset{\text{def}}{=} E - k_0 \cdot \sigma, \quad U \overset{\text{def}}{=} E + k_0 \cdot \sigma,
  \]
  and $k_0 > 1$ is pre-selected.
- Comment: most frequently, $k_0 = 2, 3, \text{or } 6$. 
18. Outlier Detection Under Interval Uncertainty: A Problem

- In some practical situations, we only have intervals $x_i = [\underline{x}_i, \overline{x}_i]$.
- For different values $x_i \in x_i$, we get different $k_0$-sigma intervals $[L, U]$.
- A possible outlier is a value outside some $k_0$-sigma interval.
- Example: structural integrity – not to miss a fault.
- A guaranteed outlier is a value outside all $k_0$-sigma intervals.
- Example: before a surgery, we want to make sure that there is a microcalcification.
- A value $x$ is a possible outlier if $x \notin [\overline{L}, \overline{U}]$.
- A value $x$ is a guaranteed outlier if $x \notin [L, U]$.
- Conclusion: to detect outliers, we must know the ranges of $L = E - k_0 \cdot \sigma$ and $U = E + k_0 \cdot \sigma$. 

- **We need:** to detect outliers, we must compute the ranges of \( L = E - k_0 \cdot \sigma \) and \( U = E + k_0 \cdot \sigma \).

- **We know:** how to compute the ranges \( E \) and \([\sigma, \bar{\sigma}]\) for \( E \) and \( \sigma \).

- **Possibility:** use interval computations to conclude that \( L \in E - k_0 \cdot [\sigma, \bar{\sigma}] \) and \( L \in E + k_0 \cdot [\sigma, \bar{\sigma}] \).

- **Problem:** the resulting intervals for \( L \) and \( U \) are wider than the actual ranges.

- **Reason:** \( E \) and \( \sigma \) use the same inputs \( x_1, \ldots, x_n \) and are hence not independent from each other.

- **Practical consequence:** we miss some outliers.

- **Desirable:** compute exact ranges for \( L \) and \( U \).

- **What we do:** exactly this.

- **Application:** detecting outliers in gravity measurements.
20. Acknowledgments

This work was supported in part:

- by NASA under cooperative agreement NCC5-209,
- by NSF grant EAR-0225670,
- by NIH grant 3T34GM008048-20S1,
- by Army Research Lab grant DATM-05-02-C-0046,
- by Star Award from the University of Texas System,
- by Texas Department of Transportation grant No. 0-5453, and
- by the workshop organizers.
21. Detecting Possible Outliers: Idea

- To detect possible outliers, we need $\mathcal{L}$ and $\mathcal{U}$.
- The minimum $\mathcal{U}$ of a smooth function $U$ on an interval $[x_i, \bar{x}_i]$ is attained:
  - either inside, when $\frac{\partial U}{\partial x_i} = 0$ – i.e., when
    \[
    x_i = \mu \overset{\text{def}}{=} E - \alpha \cdot \sigma \text{ (where } \alpha \overset{\text{def}}{=} 1/k_0); \]
  - or at $x_i = \bar{x}_i$, when $\frac{\partial U}{\partial x_i} \geq 0$ – i.e., when $\mu \leq x_i$;
  - or at $x_i = x_i$, when $\frac{\partial U}{\partial x_i} \leq 0$ – i.e., when $\bar{x}_i \leq \mu$.

  Thus, once we know how $\mu$ is located w.r.t. all the intervals $x_i$, we can find the optimal values of $x_i$.

- Comment. the value $\mu$ can be obtained from the condition $E - \alpha \cdot \sigma = \mu$.

- Hence, to find min $U$, we analyze how the endpoints $x_i$ and $\bar{x}_i$ divide the real line, consider all the resulting sub-intervals, and take the smallest $U$. 
22. Computing $U$: Algorithm

- First, sort all $2n$ values $x_i$, $\bar{x}_i$ into a sequence $x(1) \leq x(2) \leq \ldots \leq x(2n)$; take $x(0) \overset{\text{def}}{=} -\infty$, $x(2n+1) \overset{\text{def}}{=} +\infty$.

- For each zone $[x(k), x(k+1)]$, we compute the values

$$e_k \overset{\text{def}}{=} \sum_{i : x_i \geq x(k+1)} x_i + \sum_{j : \bar{x}_j \leq x(k)} \bar{x}_j,$$

$$m_k \overset{\text{def}}{=} \sum_{i : x_i \geq x(k+1)} (x_i)^2 + \sum_{j : \bar{x}_j \leq x(k)} (\bar{x}_j)^2,$$

and $n_k$ = the total number of such $i$’s and $j$’s.

- Solve equation $A - B \cdot \mu + C \cdot \mu^2 = 0$, where

$$A \overset{\text{def}}{=} e_k^2 (1 + \alpha^2) - \alpha^2 m_k n,$$

$$B \overset{\text{def}}{=} 2e_k ((1 + \alpha^2) n_k - \alpha^2 n); \quad C \overset{\text{def}}{=} B \cdot \frac{n_k}{2e_k};$$

select $\mu \in$ zone for which $\mu \cdot n_k \leq e_k$.

- $E_k \overset{\text{def}}{=} \frac{e_k}{n} + \frac{n - n_k}{n} \cdot \mu$, $M_k \overset{\text{def}}{=} \frac{m_k}{n} + \frac{n - n_k}{n} \cdot \mu^2$, $U_k \overset{\text{def}}{=} E_k + k_0 \cdot \sqrt{M_k - (E_k)^2}$.

- $U$ is the smallest of these values $U_k$.
23. **Computing $\bar{L}$: Algorithm**

- First, sort all $2n$ values $x_i, \bar{x}_i$ into a sequence $x(1) \leq x(2) \leq \ldots \leq x(2n)$; take $x(0) \overset{\text{def}}{=} -\infty$, $x(2n+1) \overset{\text{def}}{=} +\infty$.

- For each zone $[x(k), x(k+1)]$, we compute the values

$$
e_k \overset{\text{def}}{=} \sum_{i: x_i \geq x(k+1)} x_i + \sum_{j: \bar{x}_j \leq x(k)} \bar{x}_j;$$

$$m_k \overset{\text{def}}{=} \sum_{i: x_i \geq x(k+1)} (x_i)^2 + \sum_{j: \bar{x}_j \leq x(k)} (\bar{x}_j)^2,$$

and $n_k = \text{the total number of such } i \text{'s and } j \text{'s}.$

- Solve equation $A - B \cdot \mu + C \cdot \mu^2 = 0$, where

$$A \overset{\text{def}}{=} e_k^2 \cdot (1 + \alpha^2) - \alpha^2 \cdot m_k \cdot n,$$

$$B \overset{\text{def}}{=} 2e_k \cdot ((1 + \alpha^2) \cdot n_k - \alpha^2 \cdot n); \quad C \overset{\text{def}}{=} B \cdot \frac{n_k}{2e_k};$$

select $\mu \in \text{zone for which } \mu \cdot n_k \geq e_k$.

- $E_k \overset{\text{def}}{=} \frac{e_k}{n} + \frac{n-n_k}{n} \cdot \mu$, $M_k \overset{\text{def}}{=} \frac{m_k}{n} + \frac{n-n_k}{n} \cdot \mu^2$,

$$L_k \overset{\text{def}}{=} E_k - k_0 \cdot \sqrt{M_k - (E_k)^2}.$$

- $\bar{L}$ is the largest of these values $L_k$. 
24. Computational Complexity of Outlier Detection

- **Detecting possible outliers:** The above algorithm $A_U$ always computes $U$ in quadratic time.

- **Detecting possible outliers:** The above algorithm $\overline{A}_L$ always computes $L$ in quadratic time.

- **Detecting guaranteed outliers:** For every $k_0 > 1$, computing the upper endpoint $\overline{U}$ of the interval $[\underline{U}, \overline{U}]$ of possible values of $U = E + k_0 \cdot \sigma$ is NP-hard.

- **Detecting guaranteed outliers:** For every $k_0 > 1$, computing the lower endpoint $\underline{L}$ of the interval $[\underline{L}, \overline{L}]$ of possible values of $L = E - k_0 \cdot \sigma$ is NP-hard.

- **Comment.** For interval data, the NP-hardness of computing the upper bound for $\sigma$ was known before.
25. How Can We Actually Detect Guaranteed Outliers?

- **1st result:** if \( 1 + (1/k_0)^2 < n \), then \( \max U \) and \( \min L \) are attained at endpoints of \( x_i \).

- **Example:** \( k_0 > 1 \) and \( n \geq 2 \).

- **Resulting algorithm:** test all \( 2^n \) combinations of values \( x_i \) and \( \bar{x}_i \).

- **Important case:** often, measured values \( \tilde{x}_i \) are definitely different from each other, in the sense that the “narrowed” intervals

\[
\left[ \tilde{x}_i - \frac{1 + \alpha^2}{n} \cdot \Delta_i, \tilde{x}_i + \frac{1 + \alpha^2}{n} \cdot \Delta_i \right]
\]

do not intersect with each other.

- **Slightly more general case:** for some \( C \), no more than \( C \) “narrowed” intervals can have a common point.
26. **Computing $\overline{U}$**

- Sort all endpoints of the narrowed intervals into a sequence $x(1) \leq x(2) \leq \ldots \leq x(2n)$, with $x(0) \defeq -\infty$, $x(2n+1) \defeq +\infty$.

- For each zone $[x(i), x(i+1)]$, for each $j$, pick $x_j$:
  
  - if $x(i+1) < \tilde{x}_j - \frac{1 + \alpha^2}{n} \Delta_j$, pick $x_j = \overline{x}_j$;
  
  - if $x(i+1) > \tilde{x}_j + \frac{1 + \alpha^2}{n} \Delta_j$, pick $x_j = \underline{x}_j$;
  
  - for all other $j$, consider both $x_j = \overline{x}_j$ and $x_j = \underline{x}_j$.

- We get $\leq 2^C$ sequences of $x_j$ for each zone.

- For each sequence $x_j$, check whether $E - \alpha \cdot \sigma$ is within the zone.

- If $E - \alpha \cdot \sigma \in$ zone, compute $U \defeq E + k_0 \cdot \sigma$.

- Finally, we return the largest of the computed values $U$ as $\overline{U}$.
27. Computing $L$

- Sort all endpoints of the narrowed intervals into a sequence $x(1) \leq x(2) \leq \ldots \leq x(2n)$, with $x(0) \overset{\text{def}}{=} -\infty$, $x(2n+1) \overset{\text{def}}{=} +\infty$.

- For each zone $[x(i), x(i+1)]$, for each $j$, pick $x_j$:
  
  - if $x(i+1) < \tilde{x}_j - \frac{1 + \alpha^2}{n} \cdot \Delta_j$, pick $x_j = \overline{x}_j$;
  
  - if $x(i+1) > \tilde{x}_j + \frac{1 + \alpha^2}{n} \cdot \Delta_j$, pick $x_j = \underline{x}_j$;

  - for all other $j$, consider both $x_j = \overline{x}_j$ and $x_j = \underline{x}_j$.

- We get $\leq 2C$ sequences of $x_j$ for each zone.

- For each sequence $x_j$, check whether $E + \alpha \cdot \sigma$ is within the zone.

- If $E + \alpha \cdot \sigma \in$ zone, compute $L \overset{\text{def}}{=} E - k_0 \cdot \sigma$.

- Finally, we return the smallest of the computed values $L$ as $L$. 