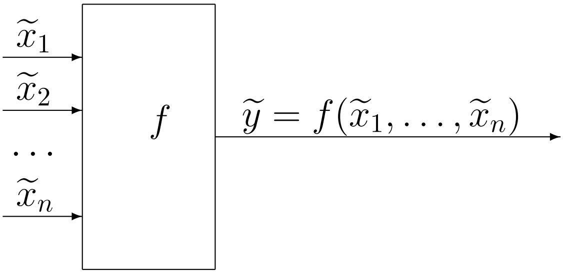
Interval Computations Technology in Mathematics Research: From Help in Theoretical Breakthroughs to Practically Useful Results About Numerical Methods

Roberto Araiza, Olga Kosheleva,
Vladik Kreinovich, and Pavel Šolín
University of Texas at El Paso
El Paso, Texas 79968, USA
contact email vladik@utep.edu

1. General Problem of Data Processing under Uncertainty

- *Indirect measurements:* way to measure y that are are difficult (or even impossible) to measure directly.
- *Idea:* $y = f(x_1, \ldots, x_n)$



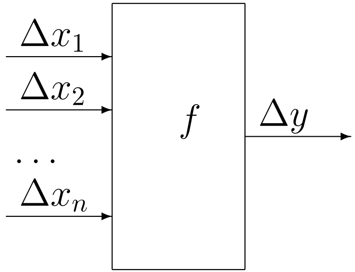
- *Problem:* measurements are never 100% accurate: $\widetilde{x}_i \neq x_i$ ($\Delta x_i \neq 0$) hence

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

What are bounds on $\Delta y \stackrel{\text{def}}{=} \widetilde{y} - y$?

Title Page
◀◀ ▶▶
◀ ▶
Page 2 of 34
Go Back
Full Screen
Close
Quit

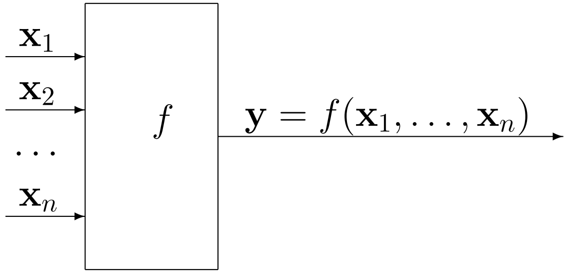
2. Probabilistic and Interval Uncertainty



- *Traditional approach:* we know probability distribution for Δx_i (usually Gaussian).
- *Where it comes from:* calibration using standard MI.
- *Problem:* calibration is not possible in:
  - fundamental science
  - manufacturing
- *Solution:* we know upper bounds Δ_i on $|\Delta x_i|$ hence

$$x_i \in [\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i].$$

3. Interval Computations: A Problem



- *Given:* an algorithm $y = f(x_1, \dots, x_n)$ and n intervals $\mathbf{x}_i = [\underline{x}_i, \overline{x}_i]$.
- *Compute:* the corresponding range of y:

$$[\underline{y}, \overline{y}] = \{f(x_1, \dots, x_n) \mid x_1 \in [\underline{x}_1, \overline{x}_1], \dots, x_n \in [\underline{x}_n, \overline{x}_n]\}.$$

- *Fact:* NP-hard even for quadratic f.
- *Challenge:* when are feasible algorithm possible?
- *Challenge:* when computing $\mathbf{y} = [\underline{y}, \overline{y}]$ is not feasible, find a good approximation $\mathbf{Y} \supseteq \mathbf{y}$.

4. Interval Computations: A Brief History

- *Origins:* Archimedes (Ancient Greece)
- *Modern pioneers:* Warmus (Poland), Sunaga (Japan), Moore (USA), 1956–59
- *First boom:* early 1960s.
- *First challenge:* taking interval uncertainty into account when planning spaceflights to the Moon.
- *Current applications* (sample):
  - design of elementary particle colliders: Berz, Kyoko (USA)
  - will a comet hit the Earth: Berz, Moore (USA)
  - robotics: Jaulin (France), Neumaier (Austria)
  - chemical engineering: Stadtherr (USA)

5. Alternative Approach: 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 – 1 variable:* if all we know is that $x \in [\underline{x}, \overline{x}]$, then MaxEnt leads to a uniform distribution on $[\underline{x}, \overline{x}]$.
- *Example – multiple variables:* different variables are independently distributed.

6. Limitations of Maximum Entropy Approach

- *Example:* simplest algorithm $y = x_1 + \ldots + x_n$.
- *Measurement errors:* $\Delta x_i \in [-\Delta, \Delta]$.
- *Analysis:* $\Delta y = \Delta x_1 + \ldots + \Delta x_n$.
- *Worst case situation:* $\Delta y = n \cdot \Delta$.
- *Maximum Entropy approach:* due to Central Limit Theorem, Δy is $\approx$ normal, with $\sigma = \Delta \cdot \dfrac{\sqrt{n}}{\sqrt{3}}$.
- *Why this may be inadequate:* we get $\Delta \sim \sqrt{n}$, but due to correlation, it is possible that $\Delta = n \cdot \Delta \sim n \gg \sqrt{n}$.
- *Conclusion:* using a single distribution can be very misleading, especially if we want guaranteed results.
- Examples: high-risk application areas such as space exploration or nuclear engineering.

7. Interval Arithmetic: Foundations of Interval Techniques

- *Problem:* compute the range
  $$[\underline{y}, \overline{y}] = \{f(x_1, \ldots, x_n) \mid x_1 \in [\underline{x}_1, \overline{x}_1], \ldots, x_n \in [\underline{x}_n, \overline{x}_n]\}.$$
- *Interval arithmetic:* for arithmetic operations $f(x_1, x_2)$ (and for elementary functions), we have explicit formulas for the range.
- *Examples:* when $x_1 \in \mathbf{x}_1 = [\underline{x}_1, \overline{x}_1]$ and $x_2 \in \mathbf{x}_2 = [\underline{x}_2, \overline{x}_2]$, then:
  - The range $\mathbf{x}_1 + \mathbf{x}_2$ for $x_1 + x_2$ is $[\underline{x}_1 + \underline{x}_2, \overline{x}_1 + \overline{x}_2]$.
  - The range $\mathbf{x}_1 - \mathbf{x}_2$ for $x_1 - x_2$ is $[\underline{x}_1 - \overline{x}_2, \overline{x}_1 - \underline{x}_2]$.
  - The range $\mathbf{x}_1 \cdot \mathbf{x}_2$ for $x_1 \cdot x_2$ is $[\underline{y}, \overline{y}]$, where
  $$\underline{y} = \min(\underline{x}_1 \cdot \underline{x}_2, \underline{x}_1 \cdot \overline{x}_2, \overline{x}_1 \cdot \underline{x}_2, \overline{x}_1 \cdot \overline{x}_2);$$
  $$\overline{y} = \max(\underline{x}_1 \cdot \underline{x}_2, \underline{x}_1 \cdot \overline{x}_2, \overline{x}_1 \cdot \underline{x}_2, \overline{x}_1 \cdot \overline{x}_2).$$
- The range $1/\mathbf{x}_1$ for $1/x_1$ is $[1/\overline{x}_1, 1/\underline{x}_1]$ (if $0 \notin \mathbf{x}_1$).

8. Straightforward Interval Computations: 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:* perform the same operations, but with *intervals* instead of *numbers*:
  - $\mathbf{r}_1 := [1,2] - [2,2] = [-1,0];$
  - $\mathbf{r}_2 := [1,2] + [2,2] = [3,4];$
  - $\mathbf{r}_3 := [-1,0] \cdot [3,4] = [-4,0].$
- *Actual range:* $f(\mathbf{x}) = [-3,0]$.
- *Comment:* this is just a toy example, there are more efficient ways of computing an enclosure $\mathbf{Y} \supseteq \mathbf{y}$.

9. First Idea: Use of Monotonicity

- *Reminder:* for arithmetic, we had exact ranges.
- *Reason:* $+$, $-$, $\cdot$ are monotonic in each variable.
- *How monotonicity helps:* if $f(x_1, \ldots, x_n)$ is (non-strictly) increasing ($f \uparrow$) in each x_i, then
$$f(\mathbf{x}_1, \ldots, \mathbf{x}_n) = [f(\underline{x}_1, \ldots, \underline{x}_n), f(\overline{x}_1, \ldots, \overline{x}_n)].$$
- *Similarly:* if $f \uparrow$ for some x_i and $f \downarrow$ for other x_j $(-)$.
- *Fact:* $f \uparrow$ in x_i if $\dfrac{\partial f}{\partial x_i} \geq 0$.
- *Checking monotonicity:* check that the range $[\underline{r}_i, \overline{r}_i]$ of $\dfrac{\partial f}{\partial x_i}$ on $\mathbf{x}_i$ has $\underline{r}_i \geq 0$.
- *Differentiation:* by Automatic Differentiation (AD) tools.
- *Estimating ranges of* $\dfrac{\partial f}{\partial x_i}$*:* straightforward interval comp.

10. Monotonicity: Example

- *Idea:* if the range $[\underline{r}_i, \overline{r}_i]$ of each $\dfrac{\partial f}{\partial x_i}$ on $\mathbf{x}_i$ has $\underline{r}_i \geq 0$, then

$$f(\mathbf{x}_1, \ldots, \mathbf{x}_n) = [f(\underline{x}_1, \ldots, \underline{x}_n), f(\overline{x}_1, \ldots, \overline{x}_n)].$$

- *Example:* $f(x) = (x - 2) \cdot (x + 2)$, $\mathbf{x} = [1, 2]$.
- *Case $n = 1$:* if the range $[\underline{r}, \overline{r}]$ of $\dfrac{df}{dx}$ on $\mathbf{x}$ has $\underline{r} \geq 0$, then

$$f(\mathbf{x}) = [f(\underline{x}), f(\overline{x})].$$

- *AD:* $\dfrac{df}{dx} = 1 \cdot (x + 2) + (x - 2) \cdot 1 = 2x.$
- *Checking:* $[\underline{r}, \overline{r}] = [2, 4]$, with $2 \geq 0$.
- *Result:* $f([1, 2]) = [f(1), f(2)] = [-3, 0]$.
- *Comparison:* this is the exact range.

11. Non-Monotonic Example

- *Example:* $f(x) = x \cdot (1 - x)$, $x \in [0, 1]$.
- How will the computer compute it?
  - $r_1 := 1 - x$;
  - $r_2 := x \cdot r_1$.
- *Straightforward interval computations:*
  - $\mathbf{r}_1 := [1, 1] - [0, 1] = [0, 1]$;
  - $\mathbf{r}_2 := [0, 1] \cdot [0, 1] = [0, 1]$.
- *Actual range:* min, max of f at $\underline{x}$, $\overline{x}$, or when $\dfrac{df}{dx} = 0$.
- Here, $\dfrac{df}{dx} = 1 - 2x = 0$ for $x = 0.5$, so
  - compute $f(0) = 0$, $f(0.5) = 0.25$, and $f(1) = 0$.
  - $\underline{y} = \min(0, 0.25, 0) = 0$, $\overline{y} = \max(0, 0.25, 0) = 0.25$.
- *Resulting range:* $f(\mathbf{x}) = [0, 0.25]$.

12. Second Idea: Centered Form

- *Main idea:* Intermediate Value Theorem

$$f(x_1, \ldots, x_n) = f(\widetilde{x}_1, \ldots, \widetilde{x}_n) + \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(\chi) \cdot (x_i - \widetilde{x}_i)$$

  for some $\chi_i \in \mathbf{x}_i$.

- *Corollary:* $f(x_1, \ldots, x_n) \in \mathbf{Y}$, where

$$\mathbf{Y} = \widetilde{y} + \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(\mathbf{x}_1, \ldots, \mathbf{x}_n) \cdot [-\Delta_i, \Delta_i].$$

- *Differentiation:* by Automatic Differentiation (AD) tools.
- *Estimating the ranges of derivatives:*
  - if appropriate, by monotonicity, or
  - by straightforward interval computations, or
  - by centered form (more time but more accurate).

13. Centered Form: Example

- *General formula:*

$$\mathbf{Y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n) + \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(\mathbf{x}_1, \ldots, \mathbf{x}_n) \cdot [-\Delta_i, \Delta_i].$$

- *Example:* $f(x) = x \cdot (1 - x)$, $\mathbf{x} = [0, 1]$.
- Here, $\mathbf{x} = [\widetilde{x} - \Delta, \widetilde{x} + \Delta]$, with $\widetilde{x} = 0.5$ and $\Delta = 0.5$.
- *Case* $n = 1$*:* $\mathbf{Y} = f(\widetilde{x}) + \dfrac{df}{dx}(\mathbf{x}) \cdot [-\Delta, \Delta]$.
- *AD:* $\dfrac{df}{dx} = 1 \cdot (1 - x) + x \cdot (-1) = 1 - 2x$.
- *Estimation:* we have $\dfrac{df}{dx}(\mathbf{x}) = 1 - 2 \cdot [0, 1] = [-1, 1]$.
- *Result:* $\mathbf{Y} = 0.5 \cdot (1 - 0.5) + [-1, 1] \cdot [-0.5, 0.5] = 0.25 + [-0.5, 0.5] = [-0.25, 0.75]$.
- *Comparison:* actual range $[0, 0.25]$, straightforward $[0, 1]$.

14. Third Idea: Bisection

- *Known:* accuracy $O(\Delta_i^2)$ of first order formula

$$f(x_1,\ldots,x_n) = f(\widetilde{x}_1,\ldots,\widetilde{x}_n) + \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(\chi)\cdot(x_i-\widetilde{x}_i).$$

- *Idea:* if the intervals are too wide, we:
  - split one of them in half ($\Delta_i^2 \to \Delta_i^2/4$); and
  - take the union of the resulting ranges.
- *Example:* $f(x) = x\cdot(1-x)$, where $x \in \mathbf{x} = [0,1]$.
- *Split:* take $\mathbf{x}' = [0, 0.5]$ and $\mathbf{x}'' = [0.5, 1]$.
- *1st range:* $1-2\cdot\mathbf{x} = 1-2\cdot[0,0.5] = [0,1]$, so $f\uparrow$ and $f(\mathbf{x}') = [f(0), f(0.5)] = [0, 0.25]$.
- *2nd range:* $1-2\cdot\mathbf{x} = 1-2\cdot[0.5,1] = [-1,0]$, so $f\downarrow$ and $f(\mathbf{x}'') = [f(1), f(0.5)] = [0, 0.25]$.
- *Result:* $f(\mathbf{x}')\cup f(\mathbf{x}'') = [0, 0.25]$ – exact.

15. Use of Interval Computations in Pure Mathematics: Computer-Aided Proofs

- *Main idea:* to prove (guarantee) that certain inequalities hold for all the values within given intervals.
- *Case study:* Lorenz equations – a simple model of atmospheric circulation.
- *Empirical fact:* solutions of Lorentz equations behave chaotically ("have a strange attractor").
- *Theoretical confirmation:* by Warwick Tucker using interval computations (Moore prize 2002).
- *Case study:* Kepler conjecture – that the standard layer-by-layer packing of spheres in the densest.
- *Solution:* Thomas Hales (Moore prize 2004).
- *Case study:* "double bubble" iso-perimetric problem (R. Schlafly, 1995).

16. Practical Case Study: Description

- We are interested in *indirect* measurements:
  - *we measure:* $f(\vec{x})$, resulting in $[\underline{f}(\vec{x}), \overline{f}(\vec{x})]$;
  - *we know:* that $L(u) = f$ for a known operator L;
  - *we want:* the range $[\underline{u}(\vec{x}), \overline{u}(\vec{x})]$ of possible values of $u(\vec{x})$.
- *"Monotonic" case:* if $f_1(\vec{x}) \leq f_2(\vec{x})$ for all $\vec{x}$, then $u_1(\vec{x}) \leq u_2(\vec{x})$.
- In the "monotonic" case, the range is easier to compute: $u(\vec{x}) \in [\underline{u}, \overline{u}] = [L^{-1}(\underline{f}), L^{-1}(\overline{f})]$.
- For linear operators L, "monotonic" means *non-negativity-preserving*: if $u \geq 0$, then $L(u) = f \geq 0$.
- We look for situations where L^{-1} is non-negativity-preserving.
- *Example:* find u s.t. $-u'' = f$ and $u(-1) = u(1) = 0$.

17. Finite Element Methods (FEM) and Discrete Non-Negativity Conservation Principles

- *Problem:* find u s.t. $-u'' = f$ and $u(-1) = u(1) = 0$.
- *FEM – main idea:* use (piece-wise) polynomial approximations $f_{h,p}(x)$ and $u_{h,p}(x)$ of given order p.
- *Problem:* we cannot have $z(x) \stackrel{\text{def}}{=} u''_{h,p}(x) + f_{h,p}(x) = 0$ for all x: e.g., for $p = 1$, we have $u''_{h,p} \equiv 0$.
- *In practice:* we never measure point values $z(x)$, only (regional) averages $\int z(x) \cdot w(x)\, dx$.
- *Solution:* require that $\int z(x) \cdot w_{h,p}(x)\, dx = 0$ for all polynomials $w_{h,p}$ which vanish at endpoints ($x = \pm 1$).
- *So:* $\int_{-1}^{1} u'_{h,p}(x) \cdot w'_{h,p}(x)\, dx = \int_{-1}^{1} f_{h,p}(x) \cdot w_{h,p}(x)\, dx$.
- *Question:* can we guarantee $u_{h,p}(x) \geq 0$ if $f_{h,p}(x) \geq 0$?

18. Proof of Non-Negativity Preservation: Main Ideas

- *Basis:* Lobatto shape functions $l_i(x)$ s.t. $l_i'(x)$ are orthonormal polynomials.
- *Resulting formula:* $u_{h,p}(x) = \int_{-1}^{1} f_{h,p}(z) \cdot \Phi_p(x,z)\,dz$, where $\Phi_p(x,z) \stackrel{\text{def}}{=} \sum_{i=1}^{p-1} l_{i+1}(x) \cdot l_{i+1}(z)$.
- *We want to prove:* if $f_{h,p}(x) \geq 0$, then $u_{h,p}(x) \geq 0$.
- *Idea of the proof:*
  - identify a subdomain Ω_p^+ of $(-1,1)^2$ where $\Phi_p \geq 0$;
  - find a quadrature rule of the order of accuracy $2p$ with positive weights and points lying in Ω_p^+.
- *What was known:* subdomains Ω_p^+ and quadrature rules for $p = 4, \ldots, 10$.
- *Remained to be proven:* $\Phi_p(x,z) \geq 0$ for all $(x,z) \in \Omega_p^+$.

19. Application of Interval Arithmetic, Case $p = 4$

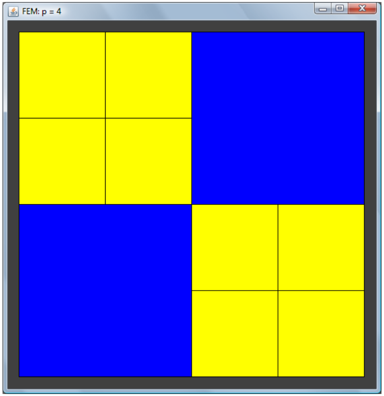
- *We need to show:* $\Phi_4(x,z) = \sum_{i=1}^{3} l_{i+1}(x) \cdot l_{i+1}(z) \geq 0.$
- $l_{i+1}(x)$ vanishes at ± 1, so $l_{i+1}(x) = (x^2 - 1) \cdot m_{i+1}(x)$.
- So, $\Phi_4(x,z) = (x^2 - 1) \cdot (z^2 - 1) \cdot \Psi_4(x,z)$, where $\Psi_4(x,z) \stackrel{\text{def}}{=} \sum_{i=1}^{3} m_{i+1}(x) \cdot m_{i+1}(z)$.
- *Consequence:* $\Phi_4(x,z) \geq 0$ for all $x, z \in (-1,1)$ if and only if $\Psi_4(x,z) \geq 0$ for all $x, z \in (-1,1)$.
- Here, $\Psi_4(x,z) = \dfrac{3}{8} + \dfrac{5}{8} \cdot x \cdot z + \dfrac{7}{128} \cdot (5x^2 - 1) \cdot (5z^2 - 1)$.
- *Idea:*
  - use interval arithmetic to compute the enclosure $[\underline{\Psi}_4, \overline{\Psi}_4]$ for the range of the function $\Psi_4(x,z)$;
  - if $\underline{\Psi}_4 \geq 0$, then $\Psi_4(x,z) \geq 0$ for all x, z.

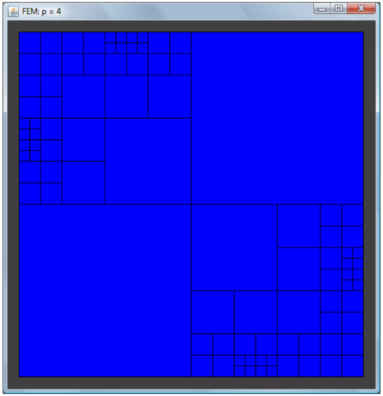
20. Application of Interval Arithmetic (cont-d)

- *Reminder:* we want to find the range of the function
$$\Psi_4(x,z)=\frac{3}{8}+\frac{5}{8}\cdot x\cdot z+\frac{7}{128}\cdot(5x^2-1)\cdot(5z^2-1).$$
- For $[-1,1]\times[-1,1]$, we get $[\underline{\Psi}_4,\overline{\Psi}_4]=\left[-\frac{25}{16},\frac{95}{32}\right]$.
- Since $\underline{\Psi}_4<0$, we subdivide the domain and evaluate the range of each sub-domain.
- For $[0,1]\times[0,1]$, we get $[\underline{\Psi}_4,\overline{\Psi}_4]=\left[\frac{5}{32},\frac{15}{8}\right]$, so $\underline{\Psi}_4\ge 0$.
- For $[-1,0]\times[-1,0]$, we similarly get $\underline{\Psi}_4=\frac{5}{32}\ge 0$.
- For $[0,1]\times[-1,0]$, we get $[\underline{\Psi}_4,\overline{\Psi}_4]=\left[-\frac{15}{32},\frac{5}{4}\right]$, so $\underline{\Psi}_4<0$.
- Thus, we need to further subdivide this sub-domain.

21. Result for the Case $p = 4$

- We subdivide the domains until (after 6 iterations) we get a partition of $[-1,1]^2$ where $[\underline{\Psi}_4, \overline{\Psi}_4] > 0$.





- *Conclusion:* $\Psi_4(x, z) \geq 0$.
- *Conclusion:* the operator $f_{h,p} \to u_{h,p}$ is non-negativity-preserving.
- Similar results are available for $p = 5, \ldots, 10$.

22. Acknowledgments

This work was supported in part:

- by NSF grant HRD-0734825 and
- by Grant 1 T36 GM078000-01 from the National Institutes of Health.

Title Page
◀◀ ▶▶
◀ ▶

Go Back
Full Screen
Close
Quit

23. Alternative Approach: Affine Arithmetic

- *So far:* we compute the range of $x \cdot (1 - x)$ by multiplying ranges of x and $1 - x$.
- *We ignore:* that both factors depend on x and are, thus, dependent.
- *Idea:* for each intermediate result a, keep an explicit dependence on $\Delta x_i = \widetilde{x}_i - x_i$ (at least its linear terms).
- *Implementation:*

$$a = a_0 + \sum_{i=1}^{n} a_i \cdot \Delta x_i + [\underline{a}, \overline{a}].$$

- *We start:* with $x_i = \widetilde{x}_i - \Delta x_i$, i.e.,

$$\widetilde{x}_i + 0 \cdot \Delta x_1 + \ldots + 0 \cdot \Delta x_{i-1} + (-1) \cdot \Delta x_i + 0 \cdot \Delta x_{i+1} + \ldots + 0 \cdot \Delta x_n + [0, 0].$$

- *Description:* $a_0 = \widetilde{x}_i$, $a_i = -1$, $a_j = 0$ for $j \neq i$, and $[\underline{a}, \overline{a}] = [0, 0]$.

24. Affine Arithmetic: Operations

- *Representation:* $a = a_0 + \sum\limits_{i=1}^{n} a_i \cdot \Delta x_i + [\underline{a}, \overline{a}]$.
- *Input:* $a = a_0 + \sum\limits_{i=1}^{n} a_i \cdot \Delta x_i + \mathbf{a}$ and $b = b_0 + \sum\limits_{i=1}^{n} b_i \cdot \Delta x_i + \mathbf{b}$.
- *Operations:* $c = a \otimes b$.
- *Addition:* $c_0 = a_0 + b_0$, $c_i = a_i + b_i$, $\mathbf{c} = \mathbf{a} + \mathbf{b}$.
- *Subtraction:* $c_0 = a_0 - b_0$, $c_i = a_i - b_i$, $\mathbf{c} = \mathbf{a} - \mathbf{b}$.
- *Multiplication:* $c_0 = a_0 \cdot b_0$, $c_i = a_0 \cdot b_i + b_0 \cdot a_i$,

$$\mathbf{c} = a_0 \cdot \mathbf{b} + b_0 \cdot \mathbf{a} + \sum_{i \neq j} a_i \cdot b_j \cdot [-\Delta_i, \Delta_i] \cdot [-\Delta_j, \Delta_j] +$$

$$\sum_i a_i \cdot b_i \cdot [-\Delta_i, \Delta_i]^2 +$$

$$\left(\sum_i a_i \cdot [-\Delta_i, \Delta_i] \right) \cdot \mathbf{b} + \left(\sum_i b_i \cdot [-\Delta_i, \Delta_i] \right) \cdot \mathbf{a} + \mathbf{a} \cdot \mathbf{b}.$$

25. Affine Arithmetic: Example

- *Example:* $f(x) = x \cdot (1 - x)$, $x \in [0, 1]$.
- Here, $n = 1$, $\widetilde{x} = 0.5$, and $\Delta = 0.5$.
- How will the computer compute it?
  - $r_1 := 1 - x$;
  - $r_2 := x \cdot r_1$.
- *Affine arithmetic:* we start with $x = 0.5 - \Delta x + [0, 0]$;
  - $\mathbf{r}_1 := 1 - (0.5 - \Delta) = 0.5 + \Delta x$;
  - $\mathbf{r}_2 := (0.5 - \Delta x) \cdot (0.5 + \Delta x)$, i.e.,

$$\mathbf{r}_2 = 0.25 + 0 \cdot \Delta x - [-\Delta, \Delta]^2 = 0.25 + [-\Delta^2, 0].$$

- *Resulting range:* $\mathbf{y} = 0.25 + [-0.25, 0] = [0, 0.25]$.
- *Comparison:* this is the exact range.

26. Affine Arithmetic: Towards More Accurate Estimates

- *In our simple example:* we got the exact range.
- *In general:* range estimation is NP-hard.
- *Meaning:* a feasible (polynomial-time) algorithm will sometimes lead to excess width: $\mathbf{Y} \supset \mathbf{y}$.
- *Conclusion:* affine arithmetic may lead to excess width.
- *Question:* how to get more accurate estimates?
- *First idea:* bisection.
- *Second idea* (Taylor arithmetic):
  - *affine arithmetic:* $a = a_0 + \sum a_i \cdot \Delta x_i + \mathbf{a}$;
  - *meaning:* we keep linear terms in Δx_i;
  - *idea:* keep, e.g., quadratic terms

$$a = a_0 + \sum a_i \cdot \Delta x_i + \sum a_{ij} \cdot \Delta x_i \cdot \Delta x_j + \mathbf{a}.$$

27. Interval Computations vs. Affine Arithmetic: Comparative Analysis

- *Objective:* we want a method that computes a reasonable estimate for the range in reasonable time.
- *Conclusion – how to compare different methods:*
  - how accurate are the estimates, and
  - how fast we can compute them.
- *Accuracy:* affine arithmetic leads to more accurate ranges.
- *Computation time:*
  - *Interval arithmetic:* for each intermediate result a, we compute two values: endpoints $\underline{a}$ and $\overline{a}$ of $[\underline{a}, \overline{a}]$.
  - *Affine arithmetic:* for each a, we compute $n + 3$ values:

$$a_0 \quad a_1, \ldots, a_n \quad \underline{a}, \overline{a}.$$

- *Conclusion:* affine arithmetic is $\sim n$ times slower.

28. Solving Systems of Equations: Extending Known Algorithms to Situations with Interval Uncertainty

- *We have:* a system of equations $g_i(y_1, \ldots, y_n) = a_i$ with unknowns y_i;
- *We know:* a_i with interval uncertainty: $a_i \in [\underline{a}_i, \overline{a}_i]$;
- *We want:* to find the corresponding ranges of y_j.
- *First case:* for exactly known a_i, we have an algorithm $y_j = f_j(a_1, \ldots, a_n)$ for solving the system.
- *Example:* system of linear equations.
- *Solution:* apply interval computations techniques to find the range $f_j([\underline{a}_1, \overline{a}_1], \ldots, [\underline{a}_n, \overline{a}_n])$.
- *Better solution:* for specific equations, we often already know which ideas work best.
- *Example:* linear equations $Ay = b$; y is monotonic in b.

29. Solving Systems of Equations When No Algorithm Is Known

- *Idea:*
  - parse each equation into elementary constraints, and
  - use interval computations to improve original ranges until we get a narrow range (= solution).
- *First example:* $x - x^2 = 0.5$, $x \in [0,1]$ (no solution).
- *Parsing:* $r_1 = x^2$, $0.5\ (= r_2)\ = x - r_1$.
- *Rules:* from $r_1 = x^2$, we extract two rules:

$$(1)\ x \to r_1 = x^2; \quad (2)\ r_1 \to x = \sqrt{r_1};$$

from $0.5 = x - r_1$, we extract two more rules:

$$(3)\ x \to r_1 = x - 0.5; \quad (4)\ r_1 \to x = r_1 + 0.5.$$

30. Solving Systems of Equations When No Algorithm Is Known: Example

- (1) $r = x^2$; (2) $x = \sqrt{r}$; (3) $r = x - 0.5$; (4) $x = r + 0.5$.
- *We start with:* $\mathbf{x} = [0, 1]$, $\mathbf{r} = (-\infty, \infty)$.

(1) $\mathbf{r} = [0, 1]^2 = [0, 1]$, so $\mathbf{r}_{\text{new}} = (-\infty, \infty) \cap [0, 1] = [0, 1]$.

(2) $\mathbf{x}_{\text{new}} = \sqrt{[0, 1]} \cap [0, 1] = [0, 1]$ – no change.

(3) $\mathbf{r}_{\text{new}} = ([0, 1] - 0.5) \cap [0, 1] = [-0.5, 0.5] \cap [0, 1] = [0, 0.5]$.

(4) $\mathbf{x}_{\text{new}} = ([0, 0.5] + 0.5) \cap [0, 1] = [0.5, 1] \cap [0, 1] = [0.5, 1]$.

(1) $\mathbf{r}_{\text{new}} = [0.5, 1]^2 \cap [0, 0.5] = [0.25, 0.5]$.

(2) $\mathbf{x}_{\text{new}} = \sqrt{[0.25, 0.5]} \cap [0.5, 1] = [0.5, 0.71]$;
round $\underline{a}$ down $\downarrow$ and $\overline{a}$ up $\uparrow$, to guarantee enclosure.

(3) $\mathbf{r}_{\text{new}} = ([0.5, 0.71] - 0.5) \cap [0.25, 5] = [0.0.21] \cap [0.25, 0.5]$,
i.e., $\mathbf{r}_{\text{new}} = \emptyset$.

- *Conclusion:* the original equation has no solutions.

31. Solving Systems of Equations: Second Example

- *Example:* $x - x^2 = 0$, $x \in [0, 1]$.
- *Parsing:* $r_1 = x^2$, $0\ (= r_2)\ = x - r_1$.
- *Rules:* (1) $r = x^2$; (2) $x = \sqrt{r}$; (3) $r = x$; (4) $x = r$.
- *We start with:* $\mathbf{x} = [0, 1]$, $\mathbf{r} = (-\infty, \infty)$.
- *Problem:* after Rule 1, we're stuck with $\mathbf{x} = \mathbf{r} = [0, 1]$.
- *Solution:* bisect $\mathbf{x} = [0, 1]$ into $[0, 0.5]$ and $[0.5, 1]$.
- *For 1st subinterval:*
  - Rule 1 leads to $\mathbf{r}_{\text{new}} = [0, 0.5]^2 \cap [0, 0.5] = [0, 0.25]$;
  - Rule 4 leads to $\mathbf{x}_{\text{new}} = [0, 0.25]$;
  - Rule 1 leads to $\mathbf{r}_{\text{new}} = [0, 0.25]^2 = [0, 0.0625]$;
  - Rule 4 leads to $\mathbf{x}_{\text{new}} = [0, 0.0625]$; etc.
  - we converge to $x = 0$.
- *For 2nd subinterval:* we converge to $x = 1$.

32. Optimization: Extending Known Algorithms to Situations with Interval Uncertainty

- *Problem:* find $y_1, \ldots, y_m$ for which

$$g(y_1, \ldots, y_m, a_1, \ldots, a_m) \to \max .$$

- *We know:* a_i with interval uncertainty: $a_i \in [\underline{a}_i, \overline{a}_i]$;
- *We want:* to find the corresponding ranges of y_j.
- *First case:* for exactly known a_i, we have an algorithm $y_j = f_j(a_1, \ldots, a_n)$ for solving the optimization problem.
- *Example:* quadratic objective function g.
- *Solution:* apply interval computations techniques to find the range $f_j([\underline{a}_1, \overline{a}_1], \ldots, [\underline{a}_n, \overline{a}_n])$.
- *Better solution:* for specific f, we often already know which ideas work best.

33. Optimization When No Algorithm Is Known

- *Idea:* divide the original box $\mathbf{x}$ into subboxes $\mathbf{b}$.
- If $\max\limits_{x\in\mathbf{b}} g(x) < g(x')$ for a known x', dismiss $\mathbf{b}$.
- *Example:* $g(x) = x \cdot (1 - x)$, $\mathbf{x} = [0, 1]$.
- Divide into 10 (?) subboxes $\mathbf{b} = [0, 0.1], [0.1, 0.2], \ldots$
- Find $g(\widetilde{b})$ for each $\mathbf{b}$; the largest is $0.45 \cdot 0.55 = 0.2475$.
- Compute $G(\mathbf{b}) = g(\widetilde{b}) + (1 - 2 \cdot \mathbf{b}) \cdot [-\Delta, \Delta]$.
- Dismiss subboxes for which $\overline{Y} < 0.2475$.
- *Example:* for $[0.2, 0.3]$, we have

$$0.25 \cdot (1 - 0.25) + (1 - 2 \cdot [0.2, 0.3]) \cdot [-0.05, 0.05].$$

- Here $\overline{Y} = 0.2175 < 0.2475$, so we dismiss $[0.2, 0.3]$.
- *Result:* keep only boxes $\subseteq [0.3, 0.7]$.
- *Further subdivision:* get us closer and closer to $x = 0.5$.