Interval Arithmetic, Verification Methods, and Multiple Precision

Contents

1 Interval Arithmetic

1.1 Basic Interval Operations

Definition 1.1 (Interval notation)**.**

$$\mathbf{x} = [\underline{x}; \overline{x}] = \{x \in \mathbb{R} : \underline{x} \leq x \leq \overline{x}\} \tag{1}$$

Definition 1.2 (Midpoint and width)**.**

$$\mathrm{mid}(\mathbf{x}) = \frac{\overline{x} + \underline{x}}{2} \tag{2}$$

$$w(\mathbf{x}) = \overline{x} - \underline{x} \tag{3}$$

1.2 Operations with Directed Rounding

1.2.1 Addition

$$\mathbf{x} + \mathbf{y} = [\nabla(\underline{x} + \underline{y}), \Delta(\overline{x} + \overline{y})] \tag{4}$$

1.2.2 Subtraction

$$\mathbf{x} - \mathbf{y} = [\nabla(\underline{x} - \overline{y}), \Delta(\overline{x} - \underline{y})] \tag{5}$$

1.2.3 Multiplication

$$\mathbf{x} \times \mathbf{y} = [\min\{\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{xy}\}, \max\{\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{xy}\}] \tag{6}$$

1.2.4 Division

If $0 \notin [\underline{y}, \overline{y}]$:

$$\mathbf{x}/\mathbf{y} = \mathbf{x} \times \frac{1}{\mathbf{y}} \tag{7}$$

where

$$\frac{1}{\mathbf{x}} = [1/\overline{x}; 1/\underline{x}] \tag{8}$$

2 Interval Newton Method

Definition 2.1 (Interval Newton operator)**.**

$$N(\tilde{x}, \mathbf{X}) := \tilde{x} - \frac{f(\tilde{x})}{f'(\mathbf{X})} \tag{9}$$

Theorem 2.2 (Convergence conditions)**.** Let $\mathbf{X}$ be an interval and $\tilde{x} \in \mathbf{X}$. Assume $0 \notin f'(\mathbf{X})$.

- If $N(\tilde{x}, \mathbf{X}) \subset \mathbf{X}$, then $\mathbf{X}$ contains a **unique root** of f
- If $N(\tilde{x}, \mathbf{X}) \cap \mathbf{X} = \emptyset$, then $\mathbf{X}$ contains **no roots** of f

3 Fixed Point Theorem (Brouwer)

3.1 For Nonlinear Systems

For nonlinear systems:

$$f(\mathbf{x}) = 0 \Leftrightarrow g(\mathbf{x}) = \mathbf{x} \tag{10}$$

where

$$g(\mathbf{x}) := \mathbf{x} - Rf(\mathbf{x}) \quad \text{with } \det(R) \neq 0 \tag{11}$$

Theorem 3.1 (Key result)**.**

$$\mathbf{X} \in \mathbb{IR}^n, \quad g(\mathbf{X}) \subseteq \mathbf{X} \quad \Rightarrow \quad \exists \hat{\mathbf{x}} \in \mathbf{X}, \quad g(\hat{\mathbf{x}}) = \hat{\mathbf{x}} \tag{12}$$

3.2 Mean Value Form

$$-Rf(\tilde{\mathbf{x}}) + (I - RM)\mathbf{Y} \subseteq \mathbf{Y} \quad \Rightarrow \quad g(\mathbf{X}) \subseteq \mathbf{X} \tag{13}$$

where M is an interval matrix containing all Jacobians in $\mathbf{X}$.

4 Matrix Nonsingularity Test

Theorem 4.1.

$$|I - RA| < 1 \quad \Rightarrow \quad A \text{ is nonsingular} \tag{14}$$

Application: Choose $R \approx A^{-1}$ and compute $|I - RA|$ with interval arithmetic.

5 Verification of Nonlinear Systems

Theorem 5.1 (Verification theorem for nonlinear systems)**.** Let $f : \mathbb{R}^n \to \mathbb{R}^n$ with $f \in C^1$, $\tilde{\mathbf{x}} \in \mathbb{R}^n$, $\mathbf{X} \in \mathbb{IR}^n$ with $0 \in \mathbf{X}$.

Let $M \in \mathbb{IR}^{n\times n}$ such that:

$$\{\nabla f_i(\zeta) : \zeta \in \tilde{\mathbf{x}} + \mathbf{X}\} \subseteq M_{i,:} \tag{15}$$

If:

$$-Rf(\tilde{\mathbf{x}}) + (I - RM)\mathbf{X} \subseteq \operatorname{int}(\mathbf{X}) \tag{16}$$

Then:

- There exists a **unique** $\hat{\mathbf{x}} \in \tilde{\mathbf{x}} + \mathbf{X}$ with $f(\hat{\mathbf{x}}) = 0$
- The Jacobian $J_f(\hat{\mathbf{x}})$ is nonsingular

6 Multiple Roots Verification

For double roots, solve the system:

$$G(\mathbf{x}, e) = \begin{pmatrix} f(x) - e \\ f'(x) \end{pmatrix} = 0 \tag{17}$$

Jacobian:

$$J_G(x, e) = \begin{pmatrix} f'(x) & -1 \\ f''(x) & 0 \end{pmatrix} \tag{18}$$

This system is **well-conditioned** for double roots (avoids the ill-conditioning).

7 Multiple Precision Arithmetic

7.1 Double-Double Numbers

Definition 7.1 (Double-double)**.** A **double-double** is a pair (a_h, a_l) satisfying:

$$a = a_h + a_l \quad \text{and} \quad |a_l| \leq u|a_h| \tag{19}$$

where u is the unit roundoff of the base precision.

Theorem 7.2 (Error bound for double-double operations)**.**

$$\mathrm{fl}(a \odot b) = (1 + \delta)(a \odot b), \quad |\delta| \leq 4 \cdot 2^{-106} \tag{20}$$

where $\odot \in \{+, \times\}$

Precision: Double-double in IEEE 754 double precision gives approximately **106 bits** of precision (double the 53 bits of standard double precision).

7.2 Representation Formats

7.2.1 Multiprecision number

$$s \cdot m \cdot \beta^e \tag{21}$$

where:

- s: sign
- m: mantissa (arbitrary length)
- β: base (typically 2)
- e: exponent

7.2.2 With integers

$$m = \sum_{i=0}^{n} m_i B^i \tag{22}$$

where m_i are machine integers and B is the word size.

7.2.3 With expansions

$$x = \sum_{i=0}^{n} f_i \tag{23}$$

where f_i are floating-point numbers with non-overlapping mantissas.

8 Kulisch Accumulator

Purpose: Compute exact sum/dot product without rounding errors.

8.1 Register Length

For exact dot product (double precision):

$$L = k + 2e_{\max} + 2|e_{\min}| + 2n = 4288 \text{ bits} \tag{24}$$

where:

- $n = 53$ bits (mantissa precision)
- $e_{\min} = -1022$ (minimum exponent)
- $e_{\max} = 1023$ (maximum exponent)
- $k = 92$ bits (for products: $2n - 2 = 104$ rounded up)

Key property: All intermediate results fit exactly in the accumulator, so only one rounding occurs at the end.

9 Error Analysis

9.1 Forward vs Backward Error

$$\text{forward error} \approx \text{condition number} \times \text{backward error} \tag{25}$$

9.1.1 Number of Correct Digits

$$\text{Number of correct digits} = -\log_{10}(\text{forward error}) \tag{26}$$

Rule of thumb:

$$\text{Number of correct digits} \approx -\log_{10}(u) - \log_{10}(\kappa) \tag{27}$$

where u is the unit roundoff and κ is the condition number.

9.2 Multiple Roots

Key insight: Forward error is $O(u^{1/m})$

- Single root ($m = 1$): error $\sim u$ ✓ well-conditioned
- Double root ($m = 2$): error $\sim \sqrt{u}$ × ill-conditioned
- Triple root ($m = 3$): error $\sim u^{1/3}$ ×× very ill-conditioned

Multiple roots are always ill-conditioned!

10 Key Algorithms

10.1 TwoSum (Error-Free Transformation)

Algorithm 1 TwoSum

function TwoSum(a, b)
  $s \leftarrow \mathrm{fl}(a + b)$
  $z \leftarrow \mathrm{fl}(s - a)$
  $t \leftarrow \mathrm{fl}(b - z)$
  return (s, t)
end function

Property 10.1. $s + t = a + b$ (exact), where s is the rounded sum and t is the rounding error.

10.2 FastTwoSum (when $|a| \geq |b|$)

Algorithm 2 FastTwoSum

function FastTwoSum(a, b)
  $s \leftarrow \mathrm{fl}(a + b)$
  $z \leftarrow \mathrm{fl}(s - a)$
  $t \leftarrow \mathrm{fl}(b - z)$
  return (s, t)
end function

Property 10.2. Same as TwoSum but requires $|a| \geq |b|$. One fewer operation than TwoSum.

Critical: Uses **Sterbenz's lemma**: if $a/2 \leq b \leq 2a$, then $a - b$ is exact.

10.3 TwoProduct (Error-Free Transformation)

Algorithm 3 TwoProduct

function TwoProduct(a, b)
  $p \leftarrow \mathrm{fl}(a \times b)$
  $e \leftarrow \mathrm{FMA}(a, b, -p)$ ▷ or use Dekker's algorithm
  return (p, e)
end function

Property 10.3. $p + e = a \times b$ (exact), where p is the rounded product and e is the error.

11 Condition Numbers (Quick Reference)

11.1 For Different Problems

11.1.1 Summation

$$\mathrm{cond}\left(\sum p_i\right) = \frac{\sum |p_i|}{|\sum p_i|} \tag{28}$$

11.1.2 Matrix-vector product

$$\text{cond}(Ax) \le |A||x|/|Ax| \tag{29}$$

11.1.3 Linear systems ($Ax = b$)

$$\kappa(A) = |A||A^{-1}| \tag{30}$$

11.1.4 Polynomial evaluation at x

$$\text{cond}(p, x) = \frac{\tilde{p}(|x|)}{|p(x)|} \tag{31}$$

where $\tilde{p}$ has absolute value coefficients.

11.1.5 Polynomial root (simple root α)

$$K(p, \alpha) = \frac{\tilde{p}(|\alpha|)}{|\alpha||p'(\alpha)|} \tag{32}$$

12 Common Error Bounds

12.1 Summation (Recursive)

$$\left|\frac{\tilde{s} - s}{s}\right| \le nu \cdot \text{cond}\left(\sum p_i\right) \tag{33}$$

where n is the number of terms and u is the unit roundoff.

12.2 Matrix Multiplication

12.2.1 Inner product $x^T y$

$$\text{fl}(x^T y) = \sum_{i=1}^{n} x_i y_i (1 + \theta_i), \quad |\theta_i| \le nu \tag{34}$$

12.2.2 Matrix-matrix AB

$$\text{fl}(AB)_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} (1 + \theta_{ijk}), \quad |\theta_{ijk}| \le nu \tag{35}$$

12.3 Linear Systems

12.3.1 LU factorization

$$\text{fl}(LU) = A + E, \quad |E| \le nu|A| \tag{36}$$

12.3.2 Forward error after solving

$$\frac{|\tilde{x} - x|}{|x|} \lesssim \kappa(A) \cdot u \tag{37}$$

12.3.3 Iterative refinement (one iteration)

$$\frac{|\tilde{x} - x|}{|x|} \lesssim \kappa(A)^2 \cdot u \tag{38}$$

13 Key Symbols and Notation

Symbol	Meaning
∇	Rounding toward $-\infty$ (round down)
Δ	Rounding toward $+\infty$ (round up)
$\mathbb{IR}$	Set of intervals
u	Unit roundoff (machine epsilon)
	fp64: $u = 2^{-53} \approx 1.11 \times 10^{-16}$
	fp32: $u = 2^{-24} \approx 5.96 \times 10^{-8}$
	fp16: $u = 2^{-11} \approx 4.88 \times 10^{-4}$
$\text{int}(\mathbf{X})$	Interior of interval $\mathbf{X}$
$\kappa(A)$	Condition number of matrix A
$\text{fl}(x)$	Floating-point representation of x
$\tilde{x}$	Computed (approximate) value
$\hat{x}$	Exact value

14 Quick Tips for the Exam

1. **Always check**: Is the problem well-conditioned? ($\kappa \ll 1/u$)
2. **Multiple roots**: Reformulate using the $G(x, e)$ system to avoid ill-conditioning
3. **Interval Newton**: Check $0 \notin f'(\mathbf{X})$ before claiming uniqueness
4. **Directed rounding**:
   - Round down for lower bounds: ∇
   - Round up for upper bounds: Δ
5. **Double-double**: Gives $\sim 2\times$ precision (~ 106 bits vs 53 bits)
6. **Iterative refinement**: Needs high-precision residual (u_r) to converge
7. **Error accumulation**:
   - Worst case: $O(nu)$
   - Probabilistic: $O(\sqrt{n}u)$ with random rounding