Exercise 4: Stochastic Arithmetic
Discrete Stochastic Arithmetic (DSA) and CADNA Analysis

Floating-point Arithmetic and Error Analysis (AFAE)

Master of Computer Science 2nd year - CCA
Year 2025/2026

Contents

1 Problem Statement

1.1 Linear System

Consider the linear system $AX = B$ with:

$$A = \begin{pmatrix} a & b \\ b & c \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ p \end{pmatrix} \tag{1}$$

Given constraints:

$$a = b + 1 \quad \text{and} \quad c = b - 1 \tag{2}$$

1.2 Equivalent System After Gaussian Elimination

The equivalent system after the first step of Gaussian elimination is $A'X = B$ with:

$$A' = \begin{pmatrix} 1 & b/a \\ 0 & c - b^2/a \end{pmatrix} \tag{3}$$

2 Question 1: Exact Solution

Problem

Show that the exact solution is:

$$X_{\text{sol}} = \begin{pmatrix} pb \\ -pa \end{pmatrix} \tag{4}$$

Solution

Step 1: Write the system equations

Starting with the system:

$$\begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ p \end{pmatrix} \tag{5}$$

This gives us:

$$ax_1 + bx_2 = 0 \quad ...(1) \tag{6}$$
$$bx_1 + cx_2 = p \quad ...(2) \tag{7}$$

Step 2: Solve for x_1 from equation (1)

From equation (1):

$$x_1 = -\frac{b}{a} x_2 \tag{8}$$

Step 3: Substitute into equation (2)

Substituting into equation (2):

$$b\left(-\frac{b}{a}x_2\right) + cx_2 = p \tag{9}$$

$$-\frac{b^2}{a}x_2 + cx_2 = p \tag{10}$$

$$x_2\left(c - \frac{b^2}{a}\right) = p \tag{11}$$

Step 4: Simplify using the constraints

Now, using $a = b + 1$ and $c = b - 1$:

$$c - \frac{b^2}{a} = (b-1) - \frac{b^2}{b+1} \tag{12}$$

$$= \frac{(b-1)(b+1) - b^2}{b+1} \tag{13}$$

$$= \frac{b^2 - 1 - b^2}{b+1} \tag{14}$$

$$= \frac{-1}{b+1} \tag{15}$$

$$= \frac{-1}{a} \tag{16}$$

Step 5: Solve for x_2

Therefore:

$$x_2 \cdot \frac{-1}{a} = p \implies x_2 = -pa \tag{17}$$

Step 6: Solve for x_1

And:

$$x_1 = -\frac{b}{a}x_2 = -\frac{b}{a}(-pa) = pb \tag{18}$$

Answer

The exact solution is:

$$\boxed{X_{\text{sol}} = \begin{pmatrix} pb \\ -pa \end{pmatrix}} \tag{19}$$

3 Question 2: Analysis with $b = 303$, $p = 3$ in binary32

3.1 Results Comparison

3.1.1 Classical Floating-Point (binary32, round to nearest)

$$X(0) = 9.07950562 \times 10^2 \quad \text{vs} \quad X_{\text{sol}}(0) = 9.09 \times 10^2 \tag{20}$$
$$X(1) = -9.10947083 \times 10^2 \quad \text{vs} \quad X_{\text{sol}}(1) = -9.12 \times 10^2 \tag{21}$$

3.1.2 DSA (CADNA)

$$X(0) = 0.9 \times 10^3 \quad \text{vs} \quad X_{\text{sol}}(0) = 0.9089999 \times 10^3 \tag{22}$$
$$X(1) = -0.9 \times 10^3 \quad \text{vs} \quad X_{\text{sol}}(1) = -0.9120000 \times 10^3 \tag{23}$$

Warning: 1 numerical instability - LOSS OF ACCURACY DUE TO CANCELLATION

3.2 Analysis

3.2.1 Comparison of Computed Results

- Classical arithmetic gives approximately 2-3 significant digits of accuracy
- Exact values: $X_{\text{sol}}(0) = 909$, $X_{\text{sol}}(1) = -912$
- Classical errors: approximately 1-2 in the last displayed digit
- DSA shows only 1 significant digit (0.9×10^3)

3.2.2 Consistency Assessment

The results are **consistent**. The key insight is:

Key Observation

DSA correctly identifies that only **1 significant digit** is reliable, while classical floating-point arithmetic displays all 9 digits, creating a **false sense of precision**.
The "true" accuracy is what DSA reveals (1 correct digit), not what classical arithmetic displays (9 digits).

Detailed Analysis:

1. **Classical arithmetic**: Displays 9.07950562×10^2, suggesting 8 significant digits
2. **Reality**: Only the first digit (9) is reliable
3. **DSA result**: More honest - displays only 0.9×10^3 (1 significant digit)
4. **Conclusion**: DSA provides the statistically significant digits based on CESTAC method analysis

The DSA result reflects the *actual* accuracy, not the *displayed* accuracy.

4 Question 3: Type of Instability Detected

4.1 Identification

Type of instability: Catastrophic cancellation

4.2 Definition

Definition 1 (Catastrophic Cancellation)**.** Catastrophic cancellation occurs when subtracting two nearly equal numbers. The significant digits cancel out, leaving only round-off errors in the result.

4.3 Location in the Computation

Responsible computation: In Gaussian elimination, we compute:

$$c - \frac{b^2}{a} = (b-1) - \frac{b^2}{b+1} \tag{24}$$

4.4 Numerical Analysis with $b = 303$

4.4.1 Computing the values

$$a = b + 1 = 304 \tag{25}$$

$$c = b - 1 = 302 \tag{26}$$

$$\frac{b^2}{a} = \frac{303^2}{304} = \frac{91809}{304} \approx 302.003289473684... \tag{27}$$

4.4.2 The cancellation

So we're computing:

$$c - \frac{b^2}{a} = 302 - 302.003289... = -0.003289... \tag{28}$$

Massive Cancellation

This is a **massive cancellation**: we're subtracting two numbers that agree in their first 3 significant digits (302.xxx - 302.xxx).
Digits lost:

$$\log_{10}\left(\frac{302}{0.003289}\right) \approx \log_{10}(91800) \approx 5 \text{ decimal digits} \tag{29}$$

4.5 Why This Causes Instability

1. Initial numbers have ∼7-8 significant digits (binary32: $-\log_{10}(2^{-24}) \approx 7.2$ digits)
2. Cancellation eliminates ∼5 digits
3. Only ∼2-3 significant digits remain in the result
4. This propagates through back-substitution, degrading the final solution
5. Final solution has only 1-2 reliable digits

4.6 Mathematical Explanation

For $x \approx y$, when computing $x - y$:

$$\text{Relative error} \approx \frac{|x|}{|x-y|} \times u \tag{30}$$

where u is the unit roundoff. The factor $\frac{|x|}{|x-y|}$ can be enormous when $x \approx y$.

5 Question 4: Validity of DSA Estimation

5.1 Main Question

Can catastrophic cancellation invalidate DSA estimation?

5.2 Answer

Answer

No, cancellation alone does **not** invalidate DSA estimation.
DSA is specifically designed to detect this kind of instability! The CESTAC method with random rounding will produce different results when cancellation occurs, allowing DSA to correctly estimate the reduced accuracy.

5.3 How DSA Detects Cancellation

1. DSA runs the computation N times (typically $N = 3$) with random rounding modes
2. When cancellation occurs, the rounding errors are amplified differently in each run
3. The variance σ^2 between runs increases
4. The estimated accuracy C_R decreases accordingly:

$$C_R \approx \log_{10}\left(\frac{\sqrt{N}\,|\bar{R}|}{\sigma\tau_\beta}\right) \tag{31}$$

5. Large σ (from cancellation) $\Rightarrow$ small C_R (few significant digits)

5.4 Types of Instabilities That MAY Invalidate DSA

5.4.1 1. Branching Instabilities

When control flow depends on unstable computations:

```
if (x > 0) {
    // branch A
} else {
    // branch B
}
```

If x is affected by round-off errors near zero, different random runs might take different branches, making statistical analysis invalid.

5.4.2 2. Underflow/Overflow

When some random runs produce overflow or underflow while others don't, the distribution is no longer quasi-Gaussian. The assumptions of CESTAC break down.

5.4.3 3. Non-Smooth Functions

Near discontinuities or in chaotic systems:

- Discontinuous functions
- Chaotic dynamical systems
- Functions with sharp gradients

Small perturbations can lead to drastically different results that don't follow the assumed statistical model.

5.4.4 4. Deterministic Algorithms Assuming Exact Arithmetic

Some algorithms (e.g., exact linear algebra algorithms over rationals) may fail in unexpected ways when arithmetic is inexact. Their behavior becomes unpredictable.

5.5 Summary

Type of Instability	**DSA Valid?**	**Reason**
Catastrophic cancellation	Yes	Detected by variance
Accumulation of errors	Yes	Detected by variance
Branching on unstable values	No	Different code paths
Underflow/Overflow	No	Non-Gaussian distribution
Discontinuous functions	No	Non-smooth behavior
Chaotic systems	No	Exponential divergence

Table 1: Validity of DSA for different types of instabilities

6 Question 5: Analysis with binary64

6.1 Results Comparison

6.1.1 Classical Floating-Point (binary64)

$$X(0) = 9.090000000078280 \times 10^2 \quad \text{vs} \quad X_{\text{sol}}(0) = 9.09 \times 10^2 \tag{32}$$
$$X(1) = -9.120000000078539 \times 10^2 \quad \text{vs} \quad X_{\text{sol}}(1) = -9.12 \times 10^2 \tag{33}$$

6.1.2 DSA (CADNA)

$$X(0) = 0.9089999999 \times 10^3 \quad \text{vs} \quad X_{\text{sol}}(0) = 0.908999999999999 \times 10^3 \tag{34}$$
$$X(1) = -0.9119999999 \times 10^3 \quad \text{vs} \quad X_{\text{sol}}(1) = -0.912000000000000 \times 10^3 \tag{35}$$

Warning: 1 numerical instability - LOSS OF ACCURACY DUE TO CANCELLATION

6.2 Analysis

6.2.1 Comparison

- **Classical arithmetic**: Error $\sim 10^{-13}$ (relative), about 10 correct significant digits
- **DSA**: Shows 10 significant digits (0.9089999999×10^3)
- **Much better than binary32!**

6.2.2 Error Analysis

For $X(0) = 909$:

$$\text{Classical error} = |909 - 909.0000000078280| \approx 7.8 \times 10^{-9} \tag{36}$$
$$\text{Relative error} = \frac{7.8 \times 10^{-9}}{909} \approx 8.6 \times 10^{-12} \tag{37}$$
$$\text{Significant digits} \approx -\log_{10}(8.6 \times 10^{-12}) \approx 11.1 \tag{38}$$

6.2.3 Consistency Assessment

Consistency

Yes, very consistent.
Both classical and DSA show approximately 10 correct decimal digits. The DSA result correctly reflects the actual precision available.
The cancellation still occurs but has **much less impact** due to the higher starting precision of binary64.

6.3 Why binary64 Performs Better

1. **Higher starting precision**:
   - binary32: ~ 7.2 decimal digits
   - binary64: ~ 15.9 decimal digits
2. **Same loss of digits**: ~ 5 digits lost due to cancellation
3. **More digits remaining**:
   - binary32: $7.2 - 5 \approx 2$ digits
   - binary64: $15.9 - 5 \approx 11$ digits

7 Question 6: Consistency Between binary32 and binary64

7.1 Main Question

Is the accuracy consistent between binary32 and binary64?

7.2 Answer

Key Theorem

Yes, the accuracy is consistent with the fundamental theorem on numerical accuracy:

Theorem 2 (Independence of Accuracy Loss)**.** The **loss of accuracy** during a numerical computation is **independent** of the precision used for the floating-point representation.

7.3 Quantitative Analysis

7.3.1 Loss of Accuracy in Cancellation

The number of digits lost is:

$$\text{Digits lost} = \log_{10}\left(\frac{302}{0.003289}\right) \approx \log_{10}(91800) \approx 5 \text{ decimal digits} \tag{39}$$

7.3.2 Starting Precision

- **binary32**: $-\log_{10}(2^{-24}) \approx 7.2$ decimal digits
- **binary64**: $-\log_{10}(2^{-53}) \approx 15.9$ decimal digits

7.3.3 Remaining Precision

- **binary32**: $7.2 - 5 \approx 2$ digits $\rightarrow$ DSA shows 1 digit ✓
- **binary64**: $15.9 - 5 \approx 11$ digits $\rightarrow$ DSA shows 10 digits ✓

7.4 Verification

The loss of ∼5 digits is **independent** of the precision format, confirming the theorem!

Format	Starting Digits	Lost Digits	Remaining Digits
binary32	7.2	5	$2.2 \approx$ 1-2
binary64	15.9	5	$10.9 \approx$ 10-11
Loss	—	**Same!**	—

Table 2: Comparison of accuracy loss across precision formats

7.5 General Formula

For any cancellation $x - y$ where $x \approx y$:

$$\boxed{\text{Digits lost} \approx \log_{10}\left(\frac{|x|}{|x - y|}\right)} \tag{40}$$

This formula is **independent** of the floating-point format!

8 Question 7: Change to $b = 3143756$

8.1 Results with $b = 3143756$ in binary64

8.1.1 DSA (CADNA)

$$X(0) = 0.94 \times 10^7 \quad \text{vs} \quad X_{\text{sol}}(0) = 0.943126799999999 \times 10^7 \tag{41}$$
$$X(1) = -0.94 \times 10^7 \quad \text{vs} \quad X_{\text{sol}}(1) = -0.943127100000000 \times 10^7 \tag{42}$$

Severe Loss of Accuracy

Only **2 significant digits**!
This is a dramatic loss compared to the 10 digits with $b = 303$.

8.2 Explanation

8.2.1 Computing the cancellation

With $b = 3143756$:

$$a = b + 1 = 3143757 \tag{43}$$
$$c = b - 1 = 3143755 \tag{44}$$
$$b^2 = 9882996059536 \tag{45}$$
$$\frac{b^2}{a} = \frac{9882996059536}{3143757} \approx 3143755.000000318... \tag{46}$$

8.2.2 The massive cancellation

$$c - \frac{b^2}{a} = 3143755 - 3143755.000000318... \tag{47}$$
$$= -0.000000318... \tag{48}$$
$$\approx -3.18 \times 10^{-7} \tag{49}$$

8.2.3 Loss of accuracy

$$\text{Digits lost} = \log_{10}\left(\frac{3143755}{0.000000318}\right) \approx \log_{10}(9.9 \times 10^{12}) \approx 13 \text{ decimal digits!} \tag{50}$$

8.3 Analysis for binary64

$$\text{Starting precision} \approx 16 \text{ decimal digits} \tag{51}$$
$$\text{Loss} \approx 13 \text{ decimal digits} \tag{52}$$
$$\text{Remaining} \approx 16 - 13 = 3 \text{ digits} \tag{53}$$

DSA shows 2 digits ✓ (conservative estimate)

8.4 Prediction for binary32

Prediction for binary32

Starting precision: $\sim$7.2 decimal digits
Loss: 13 decimal digits
Remaining: $7.2 - 13 < 0$ digits
Prediction: With binary32, the result would have **0 significant digits** (completely meaningless)!
The DSA would likely not display any significant digits at all, or show results like:

- $X(0) = @.@ \times 10^7$ (indicating no reliable digits)
- Or display a warning that the computation is completely unstable

The result would be **complete garbage** - random noise with no correlation to the true answer.

8.5 Summary Table

Parameter	$b = 303$	$b = 3143756$	**Ratio**
Cancellation value	-3.29×10^{-3}	-3.18×10^{-7}	10^4
Digits lost	5	13	—
binary32 remaining	2	< 0	—
binary64 remaining	11	3	—

Table 3: Comparison of cancellation severity

9 Key Formulas and Concepts

9.1 CESTAC/DSA Accuracy Estimation

$$C_R \approx \log_{10}\left(\frac{\sqrt{N}\,|\bar{R}|}{\sigma\tau_\beta}\right) \tag{54}$$

where:

- $\bar{R} = \frac{1}{N}\sum_{i=1}^{N} R_i$ is the mean of N samples
- $\sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}(R_i - \bar{R})^2$ is the variance
- τ_β is the Student's t-distribution value
- N is the number of samples (typically 3)
- β is the confidence level (typically 95%)

9.2 Loss of Accuracy Theorem

Theorem 3 (Cancellation Loss)**.** For cancellation $x - y$ where $x \approx y$:

$$\text{Digits lost} \approx \log_{10}\left(\frac{|x|}{|x-y|}\right) \tag{55}$$

This loss is **independent** of the precision format used!

9.3 Significant Digits in Floating-Point Formats

$$\text{binary32 (float):} \quad -\log_{10}(2^{-24}) \approx 7.2 \text{ decimal digits} \tag{56}$$

$$\text{binary64 (double):} \quad -\log_{10}(2^{-53}) \approx 15.9 \text{ decimal digits} \tag{57}$$

10 Summary and Key Takeaways

10.1 Main Lessons

1. **Catastrophic cancellation** is a major source of numerical instability
2. **DSA/CADNA** effectively detects and quantifies accuracy loss
3. **Loss of accuracy is independent of precision format** - this is a fundamental theorem
4. **Higher precision helps** but doesn't eliminate cancellation - it just gives more "buffer"
5. **Classical floating-point displays all digits** regardless of reliability - DSA only shows reliable digits
6. **Problem scaling matters**: larger b leads to worse cancellation in this problem

10.2 Practical Implications

- Always be suspicious of results involving subtraction of similar values
- Use DSA/CADNA or similar tools to assess actual accuracy
- Don't trust all displayed digits in floating-point output
- Consider problem reformulation to avoid cancellation
- Higher precision is not a cure-all - algorithmic changes may be needed

10.3 This Exercise Demonstrates

1. How Gaussian elimination can be unstable
2. The power of DSA for detecting instabilities
3. The fundamental theorem of accuracy loss independence
4. The difference between displayed precision and actual accuracy
5. The importance of understanding numerical stability