Collapse Algebra: The Universal Arithmetic Protocol

Foundations, Principles, and Applications

Clement Paulus
Independent Researcher, Austin, TX, USA
clementpaulus9@gmail.com
ORCID: 0009-0000-6069-8234
Zenodo DOI: 10.5281/zenodo.15817540

October 27, 2025

Abstract

Collapse Algebra is a closed, empirically grounded arithmetic and operator protocol that returns a defined, auditable outcome for every calculation. Division is realized as stepwise recurrence: for $b \neq 0$, set $n = \lfloor a/b \rfloor$ and $r = a - nb$; for $b = 0$ with $a \neq 0$, declare InfiniteRecurrence; complex inputs admit a phase-aligned projection $n = \lfloor \mathrm{Re}(a\,\bar{b})/|b|^2 \rfloor$. All computations act on normalized states $\Psi \in [0, 1]$ and are organized by a single operator family—drift, fidelity, entropy, curvature, and composite integrity—together with a weld calculus whose integrity budget satisfies $I_{t_1}/I_{t_0} = e^{\Delta\kappa}$ and a minimal ledger for reproducibility. We provide the formal specification, proofs of closure and universality over $\mathbb{R}$ and $\mathbb{C}$, reference algorithms, and end-to-end examples spanning measurement and signals, anomaly detection, numerical auditing, and complex systems. Situated in a lineage from Euclid's *Elements*[1] through Hilbert's axiomatization[2] and Gödel's limits[3] to Turing's computability[4] and Shannon's information theory[5], Collapse Algebra recasts arithmetic as process—closed, computable, and audit-ready—and serves as the arithmetic substrate of the Unified Language of Recursive Collapse (ULRC).

1 Introduction

Mathematics has long served as the backbone of scientific reasoning and technological innovation. From algebra and calculus to linear systems and stochastic equations, each field provides tools for navigating complexity. Yet persistent foundational gaps remain: division by zero, handling of negative and complex values, and other edge cases have relied on axioms, exceptions, and special rules that leave key questions unresolved.

Despite practical success, classical frameworks hit boundaries—calculations marked "undefined," ambiguous algorithmic failures, or constructs set aside for lack of closure. These discontinuities are recurring obstacles across science, engineering, computation, and empirical modeling.

Collapse Algebra is presented as a universal, operational protocol that closes these gaps. Rather than adding further exceptions, it provides a single, empirically grounded, audit-ready system in which addition, subtraction, multiplication, division (including division by zero), and recurrence are realized as computable, traceable processes. All number types—integers, rationals, irrationals, negatives, and complex values—are included from the outset.

The aim is both introduction and comprehensive guide to Collapse Algebra as the arithmetic and operational language of ULRC: every operation, number, and protocol is rigorously defined, empirically grounded, and universally applicable.

2 Collapse Protocol Flowchart

The structure is modular: each section builds from first principles—introducing core operators, normalization procedures, division protocols, and the handling of classical edge cases. Worked examples, proofs, visualizations, and algorithmic implementations illustrate both logic and practical utility across physics, computation, engineering, and data science.

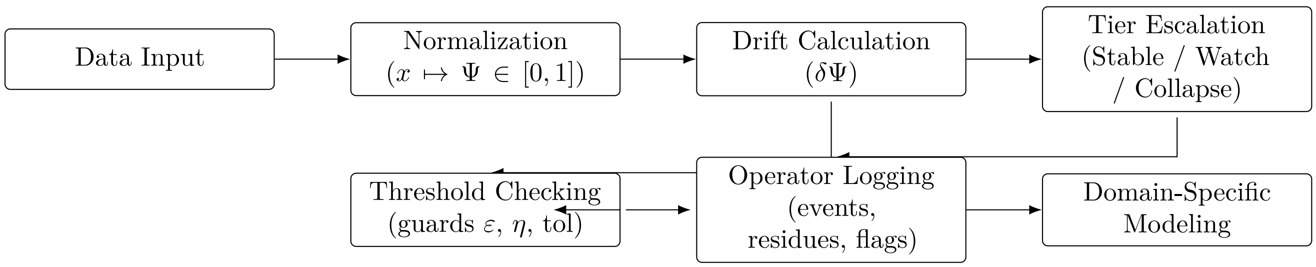

Figure 1: Collapse Protocol Flowchart: RCFT–ULRC steps from input and normalization to drift ($\delta\Psi$), regime selection, logging, checks, and domain modeling.

By formalizing ULRC's arithmetic substrate, Collapse Algebra positions itself as operational ground for domains that require closure, universality, and empirical accountability.

3 Collapse Algebra: Definition and Scope

3.1 Definition

Collapse Algebra is a formal arithmetic and operator protocol constructed to provide closure, auditability, and universal applicability to standard number operations. A minimal set of empirically specified operators act on normalized system states so that all calculations—including those traditionally regarded as undefined or exceptional—are computable and traceable.

3.2 Scope

Collapse Algebra serves as the arithmetic substrate for the Unified Language of Recursive Collapse (ULRC) and any framework requiring closed, empirically grounded computation. It is:

- **Universal**: applicable to real, integer, rational, irrational, negative, and complex inputs;
- **Closed**: operator protocols yield defined results, including division by zero and other classical edge cases;
- **Empirically grounded**: operator definitions are explicit, stepwise, and auditable;
- **Extensible**: domain-specific invariants and modular operator extensions preserve closure and auditability.

3.3 Relationship to Classical Fields

Classical Algebra Operations such as addition, subtraction, multiplication, and division are recovered as specific instances of the protocol, generalized to ensure defined, computable results for all inputs (including division by zero).

Linear Algebra Matrix/vector operations, including singular or degenerate cases, are treated as well-defined processes (e.g., recurrence or infinite delay) rather than exceptions.

Calculus Limits, derivatives, and integrals are reframed as recurrence, drift, and accumulation on normalized states; discontinuities map to explicit recurrence behavior.

Stochastic Systems Random variables, expectations, and probabilistic recurrences are encoded as operator sequences; each event is auditable and computed via normalized, closed operator logic.

3.4 Main Claims and Paradigm Shift

- A closed, normalized, loggable calculation for every number and operation;
- Explicit, empirically meaningful resolution of classical edge cases (division by zero, negative values, irrational/complex numbers);
- A unified foundation for cross-domain application in mathematics, physics, computation, and data analysis.

Collapse Algebra does not seek to replace classical mathematics; it provides the formal operational ground for all calculations, enabling universal closure and empirical auditability.

4 The Core Operators

We work with a normalized state $\Psi(t) \in [0,1]$ at discrete step $t \in \mathbb{N}$. The core operators quantify local change, retention, uncertainty, recurrence, curvature, and composite integrity on Ψ.

4.1 Symbolic Drift

Definition.

$$\delta\Psi(t) \;=\; \big|\Psi(t) - \Psi(t-1)\big|. \tag{1}$$

Role. $\delta\Psi$ measures local change; high drift indicates transitions or instability.

4.2 Recursive Fidelity

Definition.

$$F(t) \;=\; 1 - \delta\Psi(t). \tag{2}$$

Role. F measures retention or memory of the prior state.

4.3 Collapse Entropy

Definition.

$$S(t) \;=\; -\ln\!\big(1 - \delta\Psi(t) + \varepsilon\big) \quad \text{with } \varepsilon > 0 \text{ small (e.g., } 10^{-6}). \tag{3}$$

Role. Entropy rises with drift, quantifying unpredictability or disorder.

4.4 Reentry (Return) Delay

Definition.

$$\tau_R(t) \;=\; \min\big\{\, \Delta t > 0 \;\big|\; \big|\Psi(t) - \Psi(t-\Delta t)\big| < \eta \big\}, \tag{4}$$

for a fixed tolerance $\eta > 0$. τ_R captures recurrence, cycles, or recovery time.

4.5 Field Curvature

Definition. For a window k and weights $w_i \geq 0$,

$$C(t) \;=\; \sum_{i=1}^{k} \big(\Psi(t) - \Psi(t-i)\big)^2 \, w_i. \tag{5}$$

Example. If $\Psi(2) = 0.5$, $\Psi(1) = 0.7$, $\Psi(0) = 0.4$, $k = 2$, $w_1 = w_2 = 1$, then

$$C(2) = (0.5 - 0.7)^2 + (0.5 - 0.4)^2 = 0.05.$$

4.6 Composite Integrity

Definition. A typical composite integrity is

$$I_C(t) \;=\; F(t) \cdot e^{-S(t)} \cdot \big(1 - \delta\Psi(t)\big) \cdot g\big(C(t), \tau_R(t)\big), \tag{6}$$

where g is monotone in curvature and reentry (e.g., $g = 1/(1 + |C|)$). **Role.** I_C condenses the indicators into an overall integrity dial.

4.7 Summary Table of Core Operators

Operator	Definition	Conceptual Role
Symbolic Drift	$\delta\Psi(t) = \lvert\Psi(t) - \Psi(t-1)\rvert$	Local change
Recursive Fidelity	$F(t) = 1 - \delta\Psi(t)$	Memory retention
Collapse Entropy	$S(t) = -\ln(1 - \delta\Psi(t) + \varepsilon)$	Disorder / unpredictability
Reentry Delay	$\tau_R(t) = \min\{\Delta t > 0 \mid \lvert\Psi(t) - \Psi(t - \Delta t)\rvert < \eta\}$	Recurrence / memory
Field Curvature	$C(t) = \sum_{i=1}^{k}(\Psi(t) - \Psi(t-i))^2 w_i$	Deviation from smoothness
Composite Integrity	$I_C = F \cdot e^{-S} \cdot (1 - \delta\Psi) \cdot g(C, \tau_R)$	Overall system health

5 Fundamentals of Collapse Algebra

Collapse Algebra requires that all calculations operate on normalized system states. The protocol enforces consistency, comparability, and universality across arithmetic and operator calculations; explicit treatment of edge cases (division by zero, negatives, irrationals, complex values) is foundational.

6 Normalization and System State

6.1 Motivation for Normalization

Normalization maps raw values to a standard, dimensionless interval (typically $[0, 1]$), ensuring unit-invariant, comparable operator behavior across contexts.

6.2 Contract and Map

Let raw x lie between $x_{\min}$ and $x_{\max}$. A standard affine map is

$$\Psi = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \;\in\; [0, 1],$$

after which all operators act on Ψ under a fixed clipping/guard policy (ε, window k, weights w_i, tolerance η) declared once in the run contract.

6.3 Remarks

Working on $[0, 1]$ stabilizes comparisons across domains and is assumed throughout the operator definitions and examples that follow.

7 Division as Stepwise Recurrence

7.1 Motivation

Traditional division leaves edge cases undefined (e.g., division by zero) and often yields non-integer results with ambiguous interpretation. Collapse Algebra reframes division as a recurrence protocol: *how many steps of size b does it take to reach a?* This guarantees a computable, auditable outcome in every case.

7.2 Stepwise Protocol for Division

Let $a, b \in \mathbb{R}$ (or $\mathbb{C}$, or other supported domains).

1. If $b \neq 0$: compute the largest integer n such that $n \cdot b$ does not exceed a (respecting magnitude/direction). Define the residue $r = a - nb$.
   - If $r = 0$, division is exact.
   - If $r \neq 0$, the process stops with a well-defined residue.
2. If $b = 0$: no finite number of zero-steps can reach any nonzero a. Interpret as *infinite recurrence* (no return; maximal drift).
3. If b is negative or complex: apply the protocol using the direction/phase of b.

7.3 Explicit Table: Classical vs. Collapse Algebra Division

a	b	Classical $a \div b$	Collapse Algebra Result
6	3	2	2 steps $(3 + 3 = 6)$
7	2	3.5	3 steps, residue 1 $(2 + 2 + 2 + 1 = 7)$
0	1	0	0 events (already at zero)
5	0	undefined	infinite recurrence (never returns)
8	-2	-4	4 steps of -2 $(8 \to 6 \to 4 \to 2 \to 0)$

7.4 Prime Numbers as Recurrence Periods

In this framework, a prime is an integer that requires a unique, irreducible recurrence period: it cannot be reached exactly except by steps of 1 or itself.

8 Combined Application Examples

8.1 Example 1: Real Data Sequence

Given $x = [4, 7, 5]$ with $x_{\min} = 0, x_{\max} = 10$, normalization yields $\Psi = [0.4, 0.7, 0.5]$. Drift: $\delta\Psi(1) = |0.7 - 0.4| = 0.3,\ \delta\Psi(2) = |0.5 - 0.7| = 0.2$. Fidelity: $F(1) = 0.7,\ F(2) = 0.8$. Entropy ($\varepsilon = 10^{-6}$): $S(1) \approx 0.357,\ S(2) \approx 0.223$. Curvature (window $k = 2$, $w_1 = w_2 = 1$): $C(2) = (0.5 - 0.7)^2 + (0.5 - 0.4)^2 = 0.05$. Composite integrity (e.g., $g(C, \tau_R) = 1/(1 + |C|)$): $I_C(2) \approx 0.487$.

8.2 Example 2: Division Edge Case (Division by Zero)

For $a = 5, b = 0$: classical $5 \div 0$ undefined. Collapse Algebra: no finite sequence of zero-steps reaches $5 \Rightarrow$ *infinite recurrence*.

8.3 Example 3: Negative and Complex Step Division

For $a = 8, b = -2$: four steps of -2 reach 0: $8 \to 6 \to 4 \to 2 \to 0$. For $a = 1 + i, b = 1 + i$: $n = 1, r = 0$ (one step in the complex plane).

8.4 Example 4: Combined Protocol (Normalization and Division)

Let $x = [2, 6, 10]$, $x_{\min} = 0, x_{\max} = 10$. Then $\Psi = [0.2, 0.6, 1.0]$. Number of steps $b = 0.2$ to go $0.2 \to 1.0$: $n = (1.0 - 0.2)/0.2 = 4$.

8.5 Summary of Combined Techniques

Normalization plus stepwise recurrence yields explicit, audit-ready calculation for all number types and edge cases.

Part I
Universal Arithmetic and Recurrence

9 Division and Recurrence

9.1 Division as Stepwise Recurrence — Motivation and Overview

Traditional arithmetic treats division as repeated subtraction to yield zero or a remainder, but leaves key cases incomplete. Collapse Algebra defines division as the explicit iterative process of adding/subtracting steps of size b to reach a, guaranteeing either an event count n, a residue r, or an InfiniteRecurrence outcome.

9.2 Protocol Definition

Given $a, b \in \mathbb{R}$ (or $\mathbb{C}$), interpret $a \div b$ via an integer step count n and residue r such that

$$a \;=\; n\,b + r, \qquad 0 \le |r| < |b| \;\; (b \neq 0).$$

Case $b \neq 0$ (real).

$$n \;=\; \left\lfloor \frac{a}{b} \right\rfloor, \qquad r \;=\; a - n\,b.$$

This yields a residue with magnitude strictly less than $|b|$. For $b < 0$ the step direction reverses; the same rule and bound $0 \le |r| < |b|$ hold.

Case $b = 0$. If $a \neq 0$, designate InfiniteRecurrence (no finite number of zero-steps reaches a). If $a = 0$, take $(n, r) = (0, 0)$.

Complex alignment (optional). For $a, b \in \mathbb{C}$ with $b \neq 0$, project along the direction of b:

$$n \;=\; \left\lfloor \frac{\operatorname{Re}\!\big(a\,\bar{b}\big)}{|b|^2} \right\rfloor, \qquad r \;=\; a - n\,b,$$

which reduces to the real rule when $a, b \in \mathbb{R}$ and preserves the bound $0 \le |r| < |b|$ by construction.

9.3 Worked Examples

1. $a = 7,\ b = 2$: $n = \lfloor 7/2 \rfloor = 3,\ r = 7 - 3 \cdot 2 = 1$ (steps: $2 + 2 + 2 + 1 = 7$).
2. $a = 5,\ b = 0$: *infinite recurrence*; no return; maximal drift/entropy.
3. $a = 8,\ b = -2$: $n = \lfloor 8/(-2) \rfloor = \lfloor -4 \rfloor = -4,\ r = 8 - (-4)(-2) = 0$ (four steps of -2 reach 0).
4. $a = 1 + i,\ b = 1 + i$: $n = 1,\ r = 0$ (one step in the complex plane).

9.4 Comparison Table and Role

As in §7, the recurrence definition guarantees closure: each division yields either (n, r) with $0 \leq |r| < |b|$ when $b \neq 0$, or an explicit *infinite recurrence* flag when $b = 0$. This replaces classical "undefined" cases with computable, auditable outcomes.

10 Prime Numbers and Number Structure

10.1 Primes as Recurrence Periods

Classically, a prime is a natural number > 1 with no positive divisors except 1 and itself. Operationally in Collapse Algebra: a prime exhibits a unique, irreducible recurrence period within the stepwise division protocol.

10.2 Operational Characterization

For $p \in \mathbb{N}, p > 1$: for all integers k with $1 < k < p$, the recurrence protocol for division by k leaves nonzero residue, i.e., $p - nk \neq 0$ for any integer n with $nk < p$.

10.3 Example

$p = 7$. Check $k = 2, 3, 4, 5, 6$:

$$\begin{aligned}
&7 \div 2 : 2 + 2 + 2 = 6,\ r = 1; \\
&7 \div 3 : 3 + 3 = 6,\ r = 1; \\
&7 \div 4 : 4,\ r = 3; \\
&7 \div 5 : 5,\ r = 2; \\
&7 \div 6 : 6,\ r = 1.
\end{aligned}$$

Only $k = 1$ and $k = 7$ yield zero residue.

10.4 Comparison with Classical Definition and Emergence of Structure

Classical: lack of nontrivial factors. Collapse Algebra: absence of any residue-free recurrence for $1 < k < p$. Composite numbers admit at least one nontrivial k with zero residue; unit 1 has recurrence period 1; zero is not prime and does not admit a standard recurrence interpretation.

11 Universality and Edge Cases

11.1 Generalization to All Number Types

Collapse Algebra is formulated to operate on all standard number types—including integers, rationals, irrationals, negative values, and complex numbers—by defining all arithmetic operations and operator protocols on normalized system states (typically mapped to $[0, 1]$) or suitable multidimensional generalizations.

- **Integers and rationals**: operator definitions and stepwise protocols apply without modification.
- **Irrationals**: treated as limits of rational sequences and normalized in the same manner; all operators act on the normalized values, ensuring consistent outcomes.
- **Negative values**: incorporated by extending recurrence protocols to include direction; division and stepwise addition/subtraction account for sign, with negative divisors/dividends interpreted as reversed or negative recurrence.
- **Complex numbers**: the protocol extends to $\mathbb{C}$ by considering vector-valued states; recurrence, drift, and related operators act on magnitudes and phases via suitable metrics (e.g., complex norm).

11.2 Explicit Treatment of Classical Edge Cases

Collapse Algebra resolves classical ambiguities by returning defined, loggable outcomes:

- **Division by zero**: for any $a \neq 0$, no finite sequence of zero-steps reaches a; this is designated *infinite recurrence* (unbounded process).
- **Zero dividend**: for $a = 0$, any nonzero divisor b yields zero steps; the system is already at the target state.
- **Negative/complex division**: the recurrence protocol preserves direction and phase information.
- **Irrational residues**: a residue is produced for non-integral recurrence regardless of rationality.

11.3 No Undefined Cases Principle

Every admissible operation yields a computable, auditable result: explicit step count and residue, infinite recurrence (division by zero), or a well-defined multidimensional process in the complex case.

11.4 Unification Across Mathematical Domains

Because all number types and arithmetic operations are supported natively, Collapse Algebra unifies:

- algebra (including edge cases),
- linear algebra (matrix/vector operations, including singular/degenerate cases),
- calculus (limits/derivatives/integrals reframed via drift, recurrence, and accumulation on normalized states),
- stochastic systems (random variables and recurrences handled as operator sequences with auditable outcomes).

11.5 Summary

Systematic treatment of universality and edge cases ensures a fully closed, operational arithmetic substrate for analysis, modeling, and computation.

12 Applications and Workflow: Collapse Algebra in Practice

Collapse Algebra supplies a robust operational framework for computations and analyses that require closure and auditability across *all* inputs.

12.1 Physics and Measurement

Normalization to $[0, 1]$ enables consistent operator analysis (drift, fidelity, entropy, curvature, integrity) across heterogeneous measurements, including negatives, zeros, and anomalies. Example: sensor values in $[-10, 50]$ are normalized; jumps/drops surface in drift/entropy for transparent anomaly detection.

12.2 Engineering and Signal Processing

Time-series/signal workflows remain well-defined even with dropouts or faults; recurrence and drift expose both periodic patterns and rare events. Division by zero during gain/normalization becomes an explicit infinite-recurrence flag for diagnostics.

12.3 Computation and Algorithmic Auditing

Numerical code paths never return "undefined"; edge cases become explicit process flags (e.g., infinite recurrence) for reliable logging and handling.

12.4 Biological and Rhythmic Systems

Normalized inter-beat or interval sequences support drift and reentry-delay analysis for rhythm disorders; recurrence captures cyclical physiology robustly under missing/corrupted data.

12.5 Data Science and Anomaly Detection

Features derived from drift, entropy, and recurrence rates are resilient to missing values, outliers, and regime shifts; anomalies align with peaks in drift/entropy independent of cause.

12.6 Universal Diagnostic Pipelines

Protocols embed in automated monitoring/control loops without special-case tuning.

12.7 Summary

Collapse Algebra's closure, universality, and auditability generalize across physics, engineering, computation, biology, and data science.

13 End-to-End Worked Examples

13.1 Example 1: Time Series Data (Measurement and Drift)

Raw data $x = [4, 7, 5]$, $x_{\min} = 0$, $x_{\max} = 10$.

$$\Psi(0) = \tfrac{4-0}{10-0} = 0.4, \quad \Psi(1) = \tfrac{7-0}{10-0} = 0.7, \quad \Psi(2) = \tfrac{5-0}{10-0} = 0.5.$$

Drift:

$$\delta\Psi(1) = |0.7 - 0.4| = 0.3, \quad \delta\Psi(2) = |0.5 - 0.7| = 0.2.$$

Recursive fidelity:

$$F(1) = 1 - 0.3 = 0.7, \quad F(2) = 1 - 0.2 = 0.8.$$

Collapse entropy (with $\varepsilon = 10^{-6}$):

$$S(1) = -\ln(1 - 0.3 + \varepsilon) \approx 0.357, \quad S(2) = -\ln(1 - 0.2 + \varepsilon) \approx 0.223.$$

Field curvature ($k = 2$, $w_1 = w_2 = 1$):

$$C(2) = (0.5 - 0.7)^2 + (0.5 - 0.4)^2 = 0.05.$$

Composite integrity (using $g(C, \tau_R) \approx 0.95$):

$$IC(2) \approx 0.8 \times 0.8 \times 0.8 \times 0.95 = 0.487.$$

Interpretation: large transition at $t = 1$, partial return at $t = 2$, moderate integrity.

13.2 Example 2: Division as Recurrence (Including Edge Cases)

Case 1 $(a, b) = (7, 2)$: $n = 3$, $r = 1$; $2 + 2 + 2 + 1 = 7$.
Case 2 $(a, b) = (5, 0)$: infinite recurrence (no return).
Case 3 $(a, b) = (8, -2)$: $n = -4$, $r = 0$; $8 \to 6 \to 4 \to 2 \to 0$.

13.3 Example 3: Complex-Valued Division

$(a, b) = (1 + i, 1 + i)$: $n = 1$, $r = 0$; one step of $1 + i$ reaches $1 + i$ from 0 in $\mathbb{C}$.

13.4 Example 4: Anomaly Detection in Real Data

Given $x = [2, 2, 2, 10, 2]$, $x_{\min} = 0$, $x_{\max} = 10$:

$$\Psi = [0.2, 0.2, 0.2, 1.0, 0.2], \quad \delta\Psi(1) = 0,\ \delta\Psi(2) = 0,\ \delta\Psi(3) = 0.8,\ \delta\Psi(4) = 0.8.$$

Entropy and integrity peak at $t = 3, 4$, flagging the jump and return as anomalies.

Example 5: Financial Time Series — Market Anomaly Detection

Let $x = [100, 102, 105, 140, 106]$ be daily closing prices. Normalize to $[0, 1]$ using $x_{\min} = 100$, $x_{\max} = 140$:

$$\Psi = \big[0.00,\ 0.05,\ 0.125,\ 1.00,\ 0.15\big].$$

Drift.

$$\delta\Psi(1) = 0.05, \quad \delta\Psi(2) = 0.075, \quad \delta\Psi(3) = 0.875, \quad \delta\Psi(4) = 0.85.$$

Entropy (with $\varepsilon = 10^{-6}$). Since $S(t) = -\ln\big(1 - \delta\Psi(t) + \varepsilon\big)$, we have $e^{-S(t)} = 1 - \delta\Psi(t) + \varepsilon$.

$$S(3) \approx 2.0794 \ \ (\Rightarrow e^{-S(3)} \approx 0.125001), \qquad S(4) \approx 1.8971 \ \ (\Rightarrow e^{-S(4)} \approx 0.150001).$$

Curvature at $t = 4$ ($k = 2$, $w_i = 1$).

$$C(4) = \big(0.15 - 1.00\big)^2 + \big(0.15 - 0.125\big)^2 = 0.7225 + 0.000625 = 0.723125.$$

Composite integrity at $t = 4$. With $F(4) = 1 - \delta\Psi(4) = 0.15$ and $g(C, \tau_R) = 1/(1 + |C|) = 1/1.723125$,

$$I_C(4) = F(4)\, e^{-S(4)}\, \big(1 - \delta\Psi(4)\big)\, g = 0.15 \times 0.150001 \times 0.15 \times \frac{1}{1.723125} \approx 1.96 \times 10^{-3}.$$

Interpretation. Days 3–4 show an extreme spike then sharp drop. Large $\delta\Psi$ drives high S and large C, collapsing I_C to ≈ 0.002. Under typical regime gates (example: Stable < 0.038, Watch 0.038–0.30, Collapse ≥ 0.30 for $\delta\Psi$), both $t = 3$ and $t = 4$ fall in the Collapse tier, i.e., a flash-crash–like anomaly.

Analogy. The same operators flag tachycardic bursts in heart-rate series or DDoS-like surges in network traffic: large $\delta\Psi$ drives high S and C, collapsing I_C.
See Appendix C.8.

13.5 Summary

These examples show direct applicability to data analysis, recurrence, and anomaly detection, robust across standard and edge-case scenarios.

14 Advanced Operator Families and Extensions

14.1 Generalization to Vectors and Matrices

Core operators extend from scalars to vectors/matrices. For $\Psi(t) \in \mathbb{R}^n$:

$$\delta\Psi(t) = \|\Psi(t) - \Psi(t-1)\| \quad \text{(Euclidean or chosen norm).}$$

For $\Psi(t) \in \mathbb{R}^{m\times n}$ (matrix states), one curvature form uses the Frobenius norm:

$$C(t) = \sum_{i=1}^{k} \|\Psi(t) - \Psi(t-i)\|_F^2 \, w_i.$$

These enable multivariate time series, spatial data, and complex systems analysis.

14.2 Symbolic and Algorithmic Extensions

Operators can be encoded for symbolic platforms and algorithmic frameworks (Python, Mathematica, Julia) to automate pipelines, manipulate expressions, and embed within larger workflows.

14.3 Domain-Specific Invariants

Examples:

$$R_{\text{ratio}}(t) = \frac{\text{Number of reentries in last } N \text{ steps}}{N}, \qquad \text{MSD}(t) = \frac{1}{k}\sum_{i=1}^{k}\big(\Psi(t) - \Psi(t-i)\big)^2.$$

These capture recurrence frequency and fluctuation energy/volatility and are used across biology, physics, finance, and networks.

14.4 Integration with External Systems

Embedding protocols within real-time monitoring, diagnostics, and hybrid symbolic–stochastic logic layers supports operational deployments in engineering, medical, and industrial contexts.

14.5 Summary

Extensions preserve closure, normalization, and auditability while broadening applicability and integration with contemporary scientific and computational practice.

15 Research Directions and Unpublished Frontiers

The development of Collapse Algebra establishes a closed, auditable framework for arithmetic and operator-based analysis, but open questions and practical integrations remain. This section outlines active fronts.

15.1 Operator Generalizations and Higher-Order Structures

- **Tensor and multilayer systems**: generalize drift, curvature, and recurrence to tensor-valued states and multilayer networks.

- **Non-Euclidean/topological metrics**: adapt operators for states embedded in non-Euclidean spaces or with explicit topological constraints.
- **Continuous-time/functional extensions**: protocols for continuous or functional data, extending beyond discrete time series.

15.2 Theoretical Questions

- Conditions for closure under composition/iteration across operator families.
- Generalization of prime-as-recurrence characterization beyond the integers.
- Relationship to classical algebraic structures (groups, rings, fields), including noncommutative and infinite-dimensional settings.
- Emergent invariants, conserved quantities, and symmetry classes from process-driven arithmetic.

15.3 Integration with Empirical and Hybrid Systems

- **Coupling with physical measurement**: real-time experimental settings and sensor networks.
- **Hybrid computational architectures**: embedding operators in symbolic–numerical algorithms and AI frameworks.
- **Data-driven operator discovery**: ML/empirical analysis to discover or optimize domain-specific operator families.

15.4 Reserved and Unpublished Protocols

Certain extensions and specialized protocols (e.g., advanced symbolic field mapping, memory-driven operator families, domain-specific integrations) are reserved for ongoing research and future publication.

15.5 Summary

Collapse Algebra invites ongoing extension and evaluation across mathematics, computation, and empirical science.

16 Modular Teaching Sections

This part provides modular teaching materials for coursework, workshops, or guided self-study. Each module centers a key concept, operator, or protocol.

16.1 Module Example: Symbolic Drift

Concept. Symbolic drift quantifies change between consecutive normalized states.

$$\delta\Psi(t) = \big|\Psi(t) - \Psi(t-1)\big|.$$

Worked example. For $\Psi = [0.4, 0.7, 0.5]$: $\delta\Psi(1) = 0.3$, $\delta\Psi(2) = 0.2$.

16.2 Module Example: Division as Recurrence

Protocol. For $a, b \in \mathbb{R}$:

if $b \neq 0: \; n = \left\lfloor \dfrac{a}{b} \right\rfloor, \; r = a - nb$; if $b = 0 \wedge a \neq 0:$ *infinite recurrence*; if $a = 0 \wedge b \neq 0: \; (n, r) = (0, 0)$.

Optional complex alignment: $n = \left\lfloor \frac{\mathrm{Re}\left(a\,\bar{b}\right)}{|b|^2} \right\rfloor, \; r = a - nb.$

Worked example. $7 \div 2$: $n = \lfloor 7/2 \rfloor = 3, \; r = 7 - 3 \cdot 2 = 1$ (since $2 + 2 + 2 + 1 = 7$).

16.3 Instructor and Self-Study Notes

Modules can run independently or in sequence; append solutions, code, and prompts as needed; adapt for interactive or paper formats.

16.4 Template for Additional Modules

Template. Learning objectives; concept overview; formal definition; worked example; guided practice; independent practice; applications/discussion; optional extension/challenge; solution key; optional coding and mini-project.

16.5 Module: Recursive Fidelity

$F(t) = 1 - \delta\Psi(t)$. Example: if $\delta\Psi(2) = 0.2$, then $F(2) = 0.8$.

16.6 Module: Collapse Entropy

$S(t) = -\ln\left(1 - \delta\Psi(t) + \varepsilon\right)$, with $\varepsilon \approx 10^{-6}$. Example: $\delta\Psi(1) = 0.4 \Rightarrow S(1) \approx 0.5108$.

16.7 Module: Reentry Delay

$\tau_R(t) = \min\{\Delta t > 0 \mid |\Psi(t) - \Psi(t - \Delta t)| < \eta\}$. Role: periodicity/memory; infinite delay implies non-return.

16.8 Module: Primes as Recurrence Periods

Concept: primes have unique, irreducible recurrence periods. For $p > 1$, for all $1 < k < p$, division by k leaves nonzero residue. Example $p = 7$: checks against $k = 2, 3, 4, 5$ all leave residue.

17 Proofs of Operator Closure and Universality

A. Closure of Symbolic Drift Operator. Let $\Psi(t) \in [0, 1]$ for all t. Then $\delta\Psi(t) = |\Psi(t) - \Psi(t-1)| \in [0, 1]$ since the difference lies in $[-1, 1]$ and absolute value maps to $[0, 1]$. Thus $\delta\Psi$ is closed on $[0, 1]$.

B. Invariance of Normalization Under Linear Transformation. For $x'(t) = a\,x(t) + b$ with $a \neq 0$, the normalized state

$$\Psi'(t) = \frac{x'(t) - x'_{\min}}{x'_{\max} - x'_{\min}} = \frac{a[x(t) - x_{\min}]}{a[x_{\max} - x_{\min}]} = \Psi(t),$$

so normalization is invariant under invertible affine changes.

C. Division by Zero Is Never Undefined (Recurrence Protocol). Given $a \div b$, if $b \neq 0$ the protocol yields $n = \lfloor a/b \rfloor$ and $r = a - nb$. If $b = 0$ and $a \neq 0$, no finite n satisfies $n \cdot 0 = a$; designate *infinite recurrence*. Every case produces either (n, r) or a non-return outcome; no undefined values occur.

18 Glossary of Terms and Operators

Collapse Algebra Closed arithmetic/operator protocol covering all edge cases.

Normalized System State $\Psi(t)$ Dimensionless value in $[0, 1]$ after normalization.

Symbolic Drift $\delta\Psi(t)$ $|\Psi(t) - \Psi(t-1)|$.

Recursive Fidelity $F(t)$ $1 - \delta\Psi(t)$.

Collapse Entropy $S(t)$ $-\ln(1 - \delta\Psi(t) + \varepsilon)$.

Reentry Delay $\tau_R(t)$ $\min\{\Delta t > 0 \mid |\Psi(t) - \Psi(t - \Delta t)| < \eta\}$.

Field Curvature $C(t)$ $\sum_{i=1}^{k}(\Psi(t) - \Psi(t-i))^2 w_i$.

Composite Integrity $IC(t)$ $F \cdot e^{-S} \cdot (1 - \delta\Psi) \cdot g(C, \tau_R)$.

Division as Recurrence $a = nb + r$ with steps n and residue r; $b = 0 \Rightarrow$ infinite recurrence.

Prime (Recurrence) $p > 1$ with nonzero residue for all $1 < k < p$.

19 Table of Core Terms, Functions, and Protocol Rules

Term/Function	Formal Definition	Purpose/Interpretation
$\Psi(t)$	$\frac{x(t) - x_{\min}}{x_{\max} - x_{\min}}$	Scale-free state
$\delta\Psi(t)$	$\lvert\Psi(t) - \Psi(t-1)\rvert$	Detects transitions
$F(t)$	$1 - \delta\Psi(t)$	Memory retention
$S(t)$	$-\ln(1 - \delta\Psi(t) + \varepsilon)$	Disorder/unpredictability
$\tau_R(t)$	$\min\{\Delta t > 0 \mid \lvert\Psi(t) - \Psi(t - \Delta t)\rvert < \eta\}$	Time-to-return
$C(t)$	$\sum_{i=1}^{k}(\Psi(t) - \Psi(t-i))^2 w_i$	Nonlinearity/bend
$IC(t)$	$F \cdot e^{-S} \cdot (1 - \delta\Psi) \cdot g(C, \tau_R)$	Aggregate integrity
Div. as Recurrence	$a \div b : \; a = nb + r$	Stepwise division
Residue	$r = a - nb, \; 0 \leq \lvert r\rvert < \lvert b\rvert$	Amount after full steps
Infinite Recurrence	$b = 0, \; a \neq 0$	Process never returns
Prime (Recurrence)	p prime if $\forall 1 < k < p, \; r \neq 0$	Irreducible period
Rratio	$\frac{\text{reentries in last } N}{N}$	Return frequency
MSD	$\frac{1}{k}\sum_{i=1}^{k}(\Psi(t) - \Psi(t-i))^2$	Volatility/variance

Summary of Protocol Rules. All operators act on $\Psi(t) \in [0, 1]$. All arithmetic is explicit and stepwise; division by zero returns infinite recurrence; every calculation is computable/loggable; residues are explicit; edge cases have rule-based outcomes; primes have irreducible recurrence periods; composite measures are bounded in $[0, 1]$; extensions to vectors/matrices follow via norms and rolling windows.

20 Governance, Reproducibility, and Limits

20.1 Contract and Normalization (run-level)

All computations operate on a fixed run contract: an affine normalization $x \mapsto \Psi \in [0,1]$, clipping/guard $\varepsilon > 0$, drift window and weights $(k, \{w_i\})$, tolerance $\eta > 0$ for reentry checks, and a logging policy (weld IDs, seeds, and hashes). These are declared once per run and apply to every operator.

20.2 Integrity Budget and Weld Criterion

Define composite integrity $I = e^{\kappa}$ and budget increment $\Delta\kappa$. Let $R \cdot \tau_R$ denote earned return (rate R over reentry delay τ_R). Let D_ω and D_C be debits from drift and curvature. The weld identity is

$$\Delta\kappa \;=\; R \cdot \tau_R \;-\; (D_\omega + D_C), \qquad \frac{I_{t_1}}{I_{t_0}} \;=\; e^{\Delta\kappa}.$$

Define the seam residual

$$s \;=\; R \cdot \tau_R - \big(\Delta\kappa + D_\omega + D_C\big).$$

Pass rule: a weld *passes* if $|s| \le \text{tol}$ (default tol $= 0.005$).

20.3 Reproducibility Procedure (stepwise)

1. **Fetch artifacts:** source `.tex`, data, manifests (release DOI).
2. **Verify hashes:** compute SHA-256 for each artifact; compare to the release manifest.
3. **Compile:** run `xelatex` twice on the master file; confirm section/figure references resolve.
4. **Operator checks:** for each example, recompute $\delta\Psi, F, S, C, \tau_R, IC$.
5. **Weld check:** compute $(\Delta\kappa, I_{t_1}/I_{t_0}, s)$; apply $|s| \le \text{tol}$.

20.4 Reference Algorithm: Division as Recurrence (all cases)

Inputs: a, b (real or complex). **Outputs:** integer step count n, residue r, or InfiniteRecurrence.

1. **If $b \neq 0$:**
$$n = \left\lfloor \frac{a}{b} \right\rfloor, \qquad r = a - nb.$$
Complex alignment (optional). For $a, b \in \mathbb{C}$ with $b \neq 0$,
$$n = \left\lfloor \frac{\mathrm{Re}\big(a\,\bar{b}\big)}{|b|^2} \right\rfloor, \qquad r = a - nb.$$
2. **If $b = 0$ and $a \neq 0$:** return InfiniteRecurrence.
3. **If $a = 0$ and $b \neq 0$:** return $(n, r) = (0, 0)$.
4. Log (a, b, n, r) and any regime flags; proceed with normalized Ψ if used in a pipeline.

20.5 Minimal Weld Ledger (illustrative).

weld_id	**tol**	$R\tau_R$	D_ω	D_C	$\Delta\kappa$	I-ratio	s
W-SS1m–22a6187d	0.005	0.200	0.020	0.007	+0.173	$e^{0.173} = 1.188866$	0.000
hash: `22a6187d2174bbfc6bc33f05f045f9bfc67a018594bfa07a80a57413d018d6bd`							

Checks.

$$\begin{aligned} s &= R\tau_R - (\Delta\kappa + D_\omega + D_C) \\ &= 0.200 - (0.173 + 0.020 + 0.007) \;=\; 0.000 \end{aligned}$$

$$\begin{aligned} \frac{I_{t_1}}{I_{t_0}} &= e^{\Delta\kappa} \;=\; e^{0.173} \\ &\approx 1.1888661051\ldots \end{aligned}$$

20.6 Scope Conditions and Limits

Discrete to continuous. Operators are defined on discrete steps; continuous-time usage requires an explicit sampling or limit argument.
Choice of guards. ε (entropy guard) and η (return tolerance) are domain-tuned; results are stable across reasonable ranges but must be reported.
Metrics and norms. For vectors/matrices/complex states, the chosen norm (e.g., Euclidean or Frobenius) must be specified and held fixed for comparability.
Division geometry. Complex division uses direction/phase; when projecting to steps, document the projection rule.
External dependencies. Any learned or domain-specific function $g(C, \tau_R)$ must be versioned and hashed.

20.7 What to Log for Full Reproduction

- Contract parameters: normalization map, ε, k, $\{w_i\}$, η, tol.
- Environment: XeLaTeX version, OS, and any code versions.
- Integrity row: $(\Delta\kappa, I\text{-ratio}, R\tau_R, D_\omega, D_C, s)$.
- Hashes: SHA-256 for source, figures, data; manifest root hash.

20.8 Summary

Collapse Algebra is closed, universal, and auditable. This section fixes the final seams: the κ-budget, weld pass criterion, division protocol as an algorithm, and a concrete ledger row—so independent rebuilds can verify both math and governance end-to-end.

21 Acknowledgments

We thank the community peer reviewers and educators whose comments improved clarity, accessibility, and the structure of the teaching modules; and the open-source scientific software community for tools that make modern mathematical research and publication possible.

This project stands within a long historical lineage of ideas that make a closed, auditable arithmetic protocol even conceivable. We acknowledge foundational contributions in geometry and number that shaped our starting points [1, 6]; the early formalization of analysis and arithmetic [7]; the deepening of geometric foundations and axiomatic method [8, 9, 2]; limits to formal systems

[3]; the birth of modern linear and operator frameworks [10, 11, 12, 13]; computability and effective procedure [4, 14]; information-theoretic structure [5]; dynamical systems and recurrence [15, 16]; and measure-theoretic probability [17]. We also recognize ongoing contemporary efforts toward universality, auditability, and process-based mathematics that inform the present work's aims and scope [18].

Appendix A Notation and Symbols

Symbol	Name	Meaning / Role
$\Psi(t)$	Normalized state	Dimensionless system state at step $t \in \mathbb{N}$, typically $\Psi \in [0,1]$.
$\delta\Psi(t)$	Symbolic drift	Local change: $\delta\Psi(t) = \lvert\Psi(t) - \Psi(t-1)\rvert$.
$F(t)$	Recursive fidelity	Memory/retention: $F(t) = 1 - \delta\Psi(t)$.
$S(t)$	Collapse entropy	Uncertainty: $S(t) = -\ln(1-\delta\Psi(t)+\varepsilon)$; small guard $\varepsilon > 0$.
$\tau_R(t)$	Reentry/return delay	$\tau_R(t) = \min\{\Delta t > 0 \mid \lvert\Psi(t) - \Psi(t-\Delta t)\rvert < \eta\}$; tolerance $\eta > 0$.
$C(t)$	Field curvature	Local nonlinearity: $C(t) = \sum_{i=1}^{k} \big(\Psi(t) - \Psi(t-i)\big)^2 w_i$.
$IC(t)$	Composite integrity	Aggregate dial (example): $IC = F \cdot e^{-S} \cdot (1-\delta\Psi) \cdot g(C, \tau_R)$.
$a \div b$	Division (protocol)	Stepwise recurrence: $a = nb + r$, $n \in \mathbb{Z}$, $0 \le \lvert r\rvert < \lvert b\rvert$ if $b \neq 0$; $b = 0 \Rightarrow$ infinite recurrence for $a \neq 0$.
r	Residue	$r = a - nb$ after maximal full steps n.
$\mathbb{R}, \mathbb{N}, \mathbb{C}$	Number sets	Reals, naturals, complexes (all admitted by the protocol).
k, w_i	Window, weights	Curvature window size $k \in \mathbb{N}$ and nonnegative weights w_i.
ε, η	Guards/tolerances	Small positive constants; typical $\varepsilon \sim 10^{-6}$, η domain-dependent.
θ	Regime gate level	Tiering thresholds used for escalation (e.g., Stable/Watch/Collapse).
ϕ	Active operator	Objective used for gating (e.g., S for entropy gates).
$[P, F, T, E, R \mid C]$	ID vector	Run identifiers (Params, Flags, Trials, Examples, Refs | Counters).

Appendix B Formal Definitions and Protocol Rules

B.1 Normalized System State. Given raw x with known bounds $x_{\min} < x_{\max}$,

$$\Psi = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \in [0,1].$$

B.2 Symbolic Drift.

$$\delta\Psi(t) = \big|\Psi(t) - \Psi(t-1)\big| \in [0,1].$$

B.3 Recursive Fidelity.

$$F(t) = 1 - \delta\Psi(t).$$

B.4 Collapse Entropy.

$$S(t) = -\ln\big(1 - \delta\Psi(t) + \varepsilon\big), \quad \varepsilon > 0 \text{ small}.$$

B.5 Reentry (Return) Delay.

$$\tau_R(t) = \min\{\Delta t > 0 : \ \lvert\Psi(t) - \Psi(t-\Delta t)\rvert < \eta\}.$$

B.6 Field Curvature. For window k and weights $w_i \geq 0$,

$$C(t) = \sum_{i=1}^{k} \big(\Psi(t) - \Psi(t-i)\big)^2 w_i.$$

B.7 Composite Integrity. Example construction (monotone in C and τ_R):

$$IC(t) = F(t)\, e^{-S(t)} \big(1 - \delta\Psi(t)\big)\, g\big(C(t), \tau_R(t)\big),$$

with $g(C, \tau_R) = \frac{1}{1+|C|}$ or domain-specific alternatives.

B.8 Division as Recurrence (All Cases). For $a, b \in \mathbb{R}$ (or $\mathbb{C}$):

$$a = nb + r, \quad n \in \mathbb{Z}, \quad 0 \leq |r| < |b| \ \ (b \neq 0).$$

If $b = 0$ and $a \neq 0$: *infinite recurrence*. If $a = 0$ and $b \neq 0$: $n = 0, r = 0$.

B.9 Primes as Irreducible Recurrence Periods. For $p > 1$, p is prime iff for every integer k with $1 < k < p$, division by k leaves nonzero residue under the recurrence protocol.

B.10 Universality and Closure. All operators act on Ψ and return defined, auditable outcomes; there are no undefined arithmetic cases under the protocol.

Appendix C Solution Keys for Teaching Modules

C.1 Module 17.1 — Symbolic Drift

Problem. Given $\Psi = [0.4, 0.7, 0.5]$, compute $\delta\Psi(1)$ and $\delta\Psi(2)$.
Solution.

$$\delta\Psi(1) = |0.7 - 0.4| = 0.3, \qquad \delta\Psi(2) = |0.5 - 0.7| = 0.2.$$

C.2 Module 17.2 — Division as Recurrence

$$\begin{aligned}
&\textit{(a)}\ 7 \div 2: \quad n = \left\lfloor \tfrac{7}{2} \right\rfloor = 3, \quad r = 7 - 3 \cdot 2 = 1.\\
&\textit{(b)}\ 5 \div 0: \quad \text{no finite } n \text{ with } n \cdot 0 = 5 \ \Rightarrow\ \textit{infinite recurrence.}\\
&\textit{(c)}\ 8 \div (-2): \quad n = \left\lfloor \tfrac{8}{-2} \right\rfloor = \lfloor -4 \rfloor = -4, \quad r = 8 - (-4)(-2) = 0.\\
&\textit{(d)}\ (1+i) \div (1+i): \quad n = 1, \quad r = 0.
\end{aligned}$$

C.3 Module 17.6 — Recursive Fidelity

Problem. With $\delta\Psi(1) = 0.3$ and $\delta\Psi(2) = 0.2$, compute $F(1), F(2)$.
Solution. $F(1) = 1 - 0.3 = 0.7$; $F(2) = 1 - 0.2 = 0.8$.

C.4 Module 17.7 — Collapse Entropy

Problem. For $\delta\Psi = 0.4$ and guard $\varepsilon = 10^{-6}$, compute S. Also compute S for $\delta\Psi = 0.3$ and 0.2.
Solution.

$$S(0.4) = -\ln(1 - 0.4 + 10^{-6}) \approx -\ln(0.600001) \approx 0.510826,$$

$$S(0.3) = -\ln(0.700001) \approx 0.356674, \qquad S(0.2) = -\ln(0.800001) \approx 0.223143.$$

(Values rounded to 6 decimals; using natural log.)

C.5 Module 17.8 — Reentry (Return) Delay

Problem. For $\Psi = [0.2, 0.2, 0.2, 1.0, 0.2]$ and small η, compute $\tau_R(4)$.
Solution. $\Psi(4) = 0.2$. Check backward: $|\Psi(4) - \Psi(3)| = |0.2 - 1.0| = 0.8 > \eta$; $|\Psi(4) - \Psi(2)| = |0.2 - 0.2| = 0 < \eta \Rightarrow \tau_R(4) = 2$ (earliest return).

C.6 Module 17.14 — Primes as Recurrence Periods

Problem. Show $p = 7$ is prime under the recurrence criterion.
Solution. Test $k = 2, 3, 4, 5, 6$:

$$7 \div 2 \Rightarrow r = 1; \quad 7 \div 3 \Rightarrow r = 1; \quad 7 \div 4 \Rightarrow r = 3; \quad 7 \div 5 \Rightarrow r = 2; \quad 7 \div 6 \Rightarrow r = 1.$$

All have nonzero residue; only $k = 1$ and $k = 7$ yield residue 0. Hence prime.

C.7 Optional Check — Anomaly Example (from §14.4)

Problem. $x = [2, 2, 2, 10, 2]$, $x_{\text{min}} = 0$, $x_{\text{max}} = 10$. Identify anomaly using drift.
Solution. $\Psi = [0.2, 0.2, 0.2, 1.0, 0.2]$; $\delta\Psi(3) = |1.0 - 0.2| = 0.8$, $\delta\Psi(4) = |0.2 - 1.0| = 0.8$; peak drift marks jump and return as anomalies.

C.8 (compact): Financial Time Series — Market Anomaly Example

$x = [100, 102, 105, 140, 106] \Rightarrow \Psi = [0.00, 0.05, 0.125, 1.00, 0.15]$. Peak drift: $\delta\Psi(3) = 0.875$, $\delta\Psi(4) = 0.85$; entropy $S(3) \approx 2.079$, $S(4) \approx 1.897$; curvature $C(4) \approx 0.723$; integrity $I_C(4) \approx 1.96 \times 10^{-3}$ (with $g = 1/(1 + |C|)$). *Analogy.* The same operators flag tachycardic bursts in heart-rate series or DDoS-like surges in network traffic: large $\delta\Psi$ drives high S and C, collapsing I_C.

Appendix D Protocol Adaptation to Symbolic and Hybrid Domains

Collapse Algebra operates on normalized states $\Psi \in [0, 1]$. Many real systems are symbolic or hybrid (genes, language, program tokens, event logs). This appendix gives contract-grade mappings that preserve closure and auditability.

D.1 Mapping Contract (freeze once per run)

Choose and log a mapping $\varphi : \Sigma \to [0, 1]$ for a symbolic alphabet Σ, plus any weights/hyperparameters. The mapping and its parameters are part of the run contract and must be hashed. Three common choices:

(a) Ordinal map (ordered alphabets). If Σ has a total order $\prec$, let $r(x) \in \{0, \ldots, |\Sigma| - 1\}$ be the rank. Define

$$\Psi_{\text{sym}}(t) = \varphi_{\text{ord}}(x_t) = \frac{r(x_t)}{|\Sigma| - 1} \in [0, 1].$$

(b) Prototype–kernel map (structure-aware). Fix anchors $A = \{a_1, \ldots, a_m\} \subset \Sigma$ and a similarity kernel $K : \Sigma \times \Sigma \to [0, 1]$. Define

$$\Psi_{\text{sym}}(t) = \varphi_{\text{ker}}(x_t) = \frac{1}{m} \sum_{k=1}^{m} K(x_t, a_k) \in [0, 1].$$

(Choose A and K by domain; freeze them in the contract.)

(c) Seeded hash (unordered categories). For purely nominal labels, use a seeded hash $H_s : \Sigma \to \{0, \dots, M\}$,

$$\Psi_{\text{sym}}(t) = \varphi_{\text{hash}}\big(x_t\big) = \frac{H_s(x_t)}{M} \in [0,1],$$

noting that this is structure-agnostic but stable (seed s logged).

D.2 Numeric, symbolic, and hybrid channels

For streams that mix numbers and symbols, maintain channels:

$$\Psi_{\text{num}}(t) \in [0,1], \qquad \Psi_{\text{sym}}(t) \in [0,1], \qquad \chi(t) \in \{0,1\}$$

where χ is an optional event flag (e.g., `ERROR`). Define a hybrid drift with a contract weight $\alpha \in [0,1]$ and flag weight $\beta \in [0,1]$:

$$\delta\Psi_{\text{hyb}}(t) = \alpha \left|\Psi_{\text{num}}(t) - \Psi_{\text{num}}(t-1)\right| + (1-\alpha)\left|\Psi_{\text{sym}}(t) - \Psi_{\text{sym}}(t-1)\right| + \beta\,|\chi(t) - \chi(t-1)|.$$

All downstream operators use $\delta\Psi_{\text{hyb}}$ in place of $\delta\Psi$:

$$F = 1 - \delta\Psi_{\text{hyb}}, \quad S = -\ln\big(1 - \delta\Psi_{\text{hyb}} + \varepsilon\big), \quad C(t) = \sum_{i=1}^{k}\big(\Psi(t) - \Psi(t-i)\big)^2 w_i,$$

with Ψ taken as the selected channel or a contract-specified blend. Return delay τ_R and integrity I_C follow the same forms as in the main text.

D.3 Worked micro-examples

(i) Genes (binary states). $x = [\text{ON}, \text{ON}, \text{OFF}, \text{ON}]$, map $\text{ON} \mapsto 1$, $\text{OFF} \mapsto 0$:

$$\Psi_{\text{sym}} = [1,1,0,1], \quad \delta\Psi_{\text{sym}}(2) = 0,\ \delta\Psi_{\text{sym}}(3) = 1,\ \delta\Psi_{\text{sym}}(4) = 1.$$

Then $S(3) = -\ln(1-1+\varepsilon) \approx -\ln(\varepsilon)$ (maximal), and $\tau_R(4) = 2$ since $|\Psi_{\text{sym}}(4) - \Psi_{\text{sym}}(2)| = 0 < \eta$.

(ii) Log stream with sentinel. $x = [5, 10, \texttt{ERROR}, 9]$. Normalize numbers to $\Psi_{\text{num}} = [a, b, \cdot, c] \in [0,1]$; map symbol via flag $\chi = [0,0,1,0]$. With $(\alpha, \beta) = (0.8, 0.2)$,

$$\delta\Psi_{\text{hyb}}(3) = 0.8\,|\,\Psi_{\text{num}}(3) - \Psi_{\text{num}}(2)\,| + 0.2\,|1-0| \ \geq\ 0.2,$$

so the error event is explicitly carried into S, I_C without violating $[0,1]$ bounds.

D.4 Governance and reproducibility (what to log)

Log and hash: mapping φ (type and parameters), kernel K and anchors A (if used), (α, β), window k, weights w_i, tolerances (ε, η), and any flag definitions. This ensures that all symbolic adaptations remain audit-ready and reproduce the same numeric traces and weld outcomes.

References

[1] Euclid. *The Thirteen Books of Euclid's Elements*. Cambridge University Press, Cambridge, 1908. Original work ca. 300 BCE; English translation in three volumes.

[2] David Hilbert. *Grundlagen der Geometrie*. Teubner, Leipzig, 1899.

[3] Kurt Gödel. Über formal unentscheidbare sätze der principia mathematica und verwandter systeme i. *Monatshefte für Mathematik und Physik*, 38:173–198, 1931.

[4] Alan M. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42:230–265, 1936. Correction in Ser. 2, Vol. 43 (1937): 544–546.

[5] Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948. Part I; Part II appears in 27(4): 623–656.

[6] Carl Friedrich Gauss. *Disquisitiones Arithmeticae*. Gerh. Fleischer, Leipzig, 1801.

[7] Leonhard Euler. *Introductio in Analysin Infinitorum*. M.-M. Bousquet, Lausanne, 1748. Two volumes; early formalization of analysis and arithmetic.

[8] Bernhard Riemann. Über die hypothesen, welche der geometrie zu grunde liegen. *Abhandlungen der Königlichen Gesellschaft der Wissenschaften zu Göttingen*, 1867. Habilitationsvortrag 1854; posthumous publication.

[9] Giuseppe Peano. *Arithmetices Principia, Nova Methodo Exposita*. Fratelli Bocca, Torino, 1889.

[10] Arthur Cayley. A memoir on the theory of matrices. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 7:211–216, 1858. Often cited as "On the Theory of Matrices".

[11] Ferdinand Georg Frobenius. Über lineare substitutionen und bilineare formen. *Journal für die reine und angewandte Mathematik*, 84:1–63, 1879.

[12] Stefan Banach. *Théorie des opérations linéaires*. Monografie Matematyczne, Warszawa, 1932.

[13] John von Neumann. *Mathematische Grundlagen der Quantenmechanik*. Springer, Berlin, 1932. English translation: *Mathematical Foundations of Quantum Mechanics*, Princeton University Press, 1955.

[14] Alonzo Church. An unsolvable problem of elementary number theory. *American Journal of Mathematics*, 58(2):345–363, 1936.

[15] Henri Poincaré. *Les méthodes nouvelles de la mécanique céleste*. Gauthier-Villars, Paris, 1892. 3 vols., 1892–1899.

[16] George David Birkhoff. *Dynamical Systems*, volume 9 of *American Mathematical Society Colloquium Publications*. American Mathematical Society, Providence, RI, 1927.

[17] Andrey N. Kolmogorov. *Grundbegriffe der Wahrscheinlichkeitsrechnung*. Julius Springer, Berlin, 1933. English translation: *Foundations of the Theory of Probability*, Chelsea, 1950.

[18] Contemporary work on universality, auditability, and process-based mathematics, 2010. Various authors; add specific DOIs/venues for 2010–2025 items as appropriate.