FOUNDATIONS OF MARITHMETICS

Deterministic Operator Calculus, Finite Substrate Structure, and the Authority-of-Record

Justin Grieshop
Public-Arch

Authority-of-Record Edition
(Canonical tag: `aor-20260209T040755Z`)

AoR folder: `gum/authority_archive/AOR_20260209T040755Z_0fc79a0`
Bundle sha256:
`c299b1a7a8ef77f25c3ebb326cb73f060b3c7176b6ea9eb402c97273dc3cf66c`

Abstract

This paper is the Foundations record for Marithmetics. It has two jobs, and it treats both as hard constraints:

1. **Mathematical foundation**: state the base objects, invariances, and deterministic operator calculus that define the system as a single, coherent substrate—not a collection of ad hoc observations.
2. **Cryptographic foundation**: bind every empirical claim (and every "demo result") to a single Authority-of-Record (AoR) surface that is stable, audit-grade, and independently reproducible.

Marithmetics is presented here as a finite substrate architecture with explicit cross-base semantics. "Cross-base" is not treated as a rhetorical flourish; it is treated as a conservation requirement: the same structural family must be visible (and checkable) under multiple bases when mapped through the project's explicit base semantics.

To prevent the common failure modes of numerology—hidden degrees of freedom, untracked tuning, and post-hoc narrative—this paper enforces an evidence hierarchy. Tier-A claims are those whose derivations and constraints are stated in deterministic mathematics and/or whose computed outputs are cryptographically anchored to the AoR bundle. Evidence-only claims (e.g., external overlays) may be discussed, but they are never allowed to upgrade a Tier-A statement.

This document is intended to be cited. It is also intended to be executed (in the sense that its claim surface is a map to exact artifacts and logs). Readers should treat it as a constitutional specification plus an audit index: the definitions are the constitution, and the AoR is the court record.

Keywords: Finite substrate; deterministic operator calculus; cross-base invariance; reproducible computation; cryptographic audit; authority-of-record; falsification by designed failure; integer selector; constraint pipeline.

Contents

1 Document Control and Canonical Artifacts

1.1 Canonical citation surface (the only stable AoR surface for this Foundations rewrite)

Canonical AoR tag: aor-20260209T040755Z
AoR folder: gum/authority_archive/AOR_20260209T040755Z_0fc79a0
Bundle sha256: c299b1a7a8ef77f25c3ebb326cb73f060b3c7176b6ea9eb402c97273dc3cf66c

Canonical artifacts (human-readable entry points):

- Master release zip:
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/MARI_MASTER_RELEASE_20260209T040755Z_0fc79a0.zip
- GUM Report (v32):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/report/GUM_Report_v32_2026-02-09_04-27-46Z.pdf
- GUM Report manifest (machine-verifiable):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/report/GUM_Report_v32_2026-02-09_04-27-46Z.pdf.manifest.json

Core ledgers (the "court record"):

- Claim ledger (append-only):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/claim_ledger.jsonl
- AoR summary (human orientation, not a substitute for the ledger):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/SUMMARY.md
- Runner transcript (stdout/stderr consolidation):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/runner_transcript.txt
- Run metadata (environment + parameters):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/run_metadata.json

Bundle seal + indices (machine-consumable reference tables):

- Bundle seal file:
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/bundle_sha256.txt
- Master constants table:
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/constants_master.csv
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/constants_master.json

- Demo index:
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z /gum/authority_archive/AOR_20260209T040755Z_0fc79a0 /GUM_BUNDLE_v30_20260209T040755Z /tables/demo_index.csv
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z /gum/authority_archive/AOR_20260209T040755Z_0fc79a0 /GUM_BUNDLE_v30_20260209T040755Z /tables/demo_index.json
- Falsification matrix:
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z /gum/authority_archive/AOR_20260209T040755Z_0fc79a0 /GUM_BUNDLE_v30_20260209T040755Z /tables/falsification_matrix.csv
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z /gum/authority_archive/AOR_20260209T040755Z_0fc79a0 /GUM_BUNDLE_v30_20260209T040755Z /tables/falsification_matrix.json
- Run reproducibility table:
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z /gum/authority_archive/AOR_20260209T040755Z_0fc79a0 /GUM_BUNDLE_v30_20260209T040755Z /tables/run_reproducibility.csv
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z /gum/authority_archive/AOR_20260209T040755Z_0fc79a0 /GUM_BUNDLE_v30_20260209T040755Z /tables/run_reproducibility.json

1.2 How to cite this work (required format)

A citation of Marithmetics Foundations is incomplete unless it includes:

(i) this Foundations paper (the conceptual + definitional constitution),

(ii) the AoR canonical tag (aor-20260209T040755Z), and

(iii) the bundle sha256 (`c299b1a7a8ef...cf66c`).

Recommended citation tuple (machine-stable):

- "Foundations of Marithmetics — Authority-of-Record Edition," Justin Grieshop.
- Repo tag: aor-20260209T040755Z.
- Bundle sha256: `c299b1a7a8ef77f25c3ebb326cb73f060b3c7176b6ea9eb402c97273dc3cf66c`.
- Primary evidence index: `demo_index.csv` (AoR bundle).

1.3 What this paper is (and is not)

This paper is not a manifesto, a conceptual essay, or a speculative framework. It is a Foundations record with enforceable semantics:

- **Definitions are binding**: later Marithmetics papers may extend them, but may not silently change them.
- **Symbols are scoped**: a symbol introduced here must carry the same meaning wherever it reappears in the suite, unless explicitly redefined with a stated compatibility map.
- **Evidence is cryptographically anchored**: when this paper claims a computed result, it must point to an AoR artifact (logs, tables, or ledger) under the canonical tag.

1.4 Evidence hierarchy (why this exists)

A new foundation fails in public for predictable reasons that have nothing to do with the author's confidence and everything to do with the reader's incentives. A hostile reader will assume:

- you tuned parameters after seeing the answer,
- you ran many variants and only reported the best one,
- your computational environment is irreproducible,
- your "proof" is narrative, not executable.

Marithmetics answers that stance structurally, not rhetorically: the AoR is designed so the reader does not have to trust the author's memory, prose, or intent. The AoR gives the reader a stable surface to verify claims without negotiating with the author.

2 Introduction: What "Foundations" Means Here

Foundations, in the usual mathematical sense, is where one specifies objects, axioms, and permitted operations. Foundations, in the usual computational sense, is where one specifies the environment, tests, and invariants that prevent accidental self-deception.

Marithmetics requires both notions simultaneously, because it asserts a thesis that is otherwise easy to misread:

> *Marithmetics is a finite substrate architecture whose invariants persist under controlled changes of representation (including base). Its demos are not free-form curve fits; they are constraint pipelines that either select a unique structure or fail loudly.*

This paper therefore begins by tightening the meaning of three words that will recur across the suite:

- **"Finite"**: the primary constructions are finite, explicitly enumerable, and defined without appealing to an a priori continuum.
- **"Substrate"**: a substrate is not "a model"; it is a canonical discrete arena on which invariants are defined, compared, and lifted (when needed) to analytic objects by a deterministic operator calculus.
- **"Authority"**: an authority claim is not "a strong opinion." It is a claim whose derivation and/or computation is bound to an immutable evidence surface (AoR) and whose falsification conditions are stated.

2.1 Scope: what this Foundations paper covers

This Foundations record covers:

A. The definitional substrate: the base-aware finite objects that Marithmetics treats as primary (before any physics overlays).
B. The deterministic operator calculus (DOC) relationship: which parts of the pipeline are purely mathematical, and which parts are empirical outputs produced by the AoR run.
C. The evidence architecture: the AoR's chain-of-custody, and how readers should verify that a claim is real, reproduced, and non-cherry-picked.
D. The falsification posture: how designed-failure is encoded into the workflow (and how the falsification matrix is used to separate "it works" from "it resists cheating").

This paper does not attempt to replace domain papers (Standard Model, cosmology, etc.). Instead, it provides the shared floor those papers stand on: shared definitions, shared invariances, and a shared evidence protocol.

2.2 Reader contract: the fastest honest way to audit

A reader who wants to audit the work without reading the entire suite should proceed in this order:

1. Confirm the AoR identity (tag + bundle sha256) and that the report manifest matches the report PDF.
2. Open `SUMMARY.md` for orientation, then use `claim_ledger.jsonl` as the authoritative list of claims.
3. Use `constants_master.csv` as the primary numerical table, and `demo_index.csv` to locate the exact logs that produced each entry.
4. Use `falsification_matrix.csv` to locate negative controls and verify that the system fails when the invariants are violated.

All of these artifacts are anchored in the canonical AoR folder listed in Section 0.

3 The Authority-of-Record (AoR) Model

3.1 Definition: Authority-of-Record (AoR)

An Authority-of-Record is a cryptographically anchored bundle that fixes, at minimum:

- the exact executed artifacts (code + data + configuration),
- the exact produced outputs (tables, logs, derived values), and
- the exact environment metadata sufficient to rerun the pipeline.

In Marithmetics, the AoR is surfaced by a canonical tag and a canonical archive folder. The point is not convenience; it is immutability: if a claim cannot be traced to the AoR surface, it is not a Foundations claim. It may still be a hypothesis, but it may not be treated as a result.

3.2 Why a tag (and not prose) is the primary citation

Prose can be edited; human memory can drift; PDFs can be regenerated. A tag-anchored artifact path is stable: it either exists at the cited location, or it does not.

Accordingly, Marithmetics treats the canonical tag (`aor-20260209T040755Z`) as a first-class part of the scientific record.

3.3 Claim types and tiering

We use the following claim taxonomy throughout this paper and the suite:

Tier-A (Foundations-grade)
A Tier-A statement is one of:

- A *deterministic mathematical statement*: a definition, theorem, or lemma that follows from the deterministic operator calculus baseline (DOC) and the explicitly stated finite objects; or
- A *computed statement* whose value is bound to the AoR and is reproducible via the cited logs/tables/ledger under the canonical tag.

Evidence-only (non-authority)
An evidence-only statement is one that may be useful context, but is not allowed to carry the Foundations narrative. Examples include:

- overlays from external packages,
- interpretive mappings to existing domain models when those mappings are not uniquely forced by the substrate constraints,
- exploratory correlations not sealed as hard checks in the claim ledger.

This separation is not a rhetorical safety blanket. It is the only thing that prevents the work from collapsing into ambiguity when a skeptical reader asks, "Is that a theorem, or is that an observation?"

3.4 "Hard claim" means "falsifiable at the artifact level"

In this paper, a "hard claim" is not "a strong claim." It is a claim that can be falsified by checking the AoR surface. The minimum bar is:

- the value is present in `constants_master.*` or an equivalent AoR table, and
- the producing demo log is indexed (via `demo_index.*`), and
- the claim is recorded in `claim_ledger.jsonl` with enough metadata to tie it to its producing artifact.

The presence of these indices is not optional; it is part of the definition of "hard."

3.5 Run Identity, Runtime Environment, and "Dirty Tree" Disclosure

An Authority-of-Record must disclose exactly what was run, when, where, and from what code state. For the canonical AoR run used in this paper, the run identity and environment are explicitly recorded in the AoR's `SUMMARY.md` and `run_metadata.json`.

Run identity (canonical): `20260125T043902Z_52befea`
Generated UTC: 2026-01-25 04:39:02Z
Repo head recorded by the AoR: `52befeabee33865cbd3dc2d653019e15639299b0`
Branch recorded by the AoR: `main`
Dirty flag recorded by the AoR: `true`
Python recorded by the AoR: 3.12.1
Platform recorded by the AoR: Linux-6.8.0-1030-azure-x86_64

This disclosure matters because a "clean" git state is not, by itself, what makes the AoR reproducible. What makes it reproducible is that:

1. The AoR names the repo head commit (and whether the tree was dirty) in the run metadata.
2. The AoR bundle contains per-demo code hashes and per-demo stdout/stderr hashes, enabling a verifier to confirm that the scripts and outputs match the published record.
3. The AoR bundle includes a codepack snapshot (per-demo `demo.py` scripts) sufficient to audit the execution logic used for the claims.

This is why the citation surface is the published tag `aor-20260209T040755Z`, while the runtime identity is carried inside the AoR itself (run metadata + code hashes + log hashes). The tag is how the world finds the record; the metadata is how the world audits the record.

Canonical AoR metadata (primary):

- SUMMARY.md (AoR run summary): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/SUMMARY.md
- run_metadata.json (repo head, dirty flag, Python, platform, bundle sha): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/run_metadata.json
- run_reproducibility.json (per-demo code/log hashes + paths): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/run_reproducibility.json

4 Representation Independence as a Hard Requirement

4.1 The "Base Gauge" Principle

If a theory's outputs depend on whether a number is written in base 10, base 7, or base 16, then the theory is not describing an invariant structure in mathematics—it is describing an artifact of representation.

We therefore elevate representation independence to a hard constraint:

Base-Gauge Invariance (B): a lawful computation on the substrate must be invariant under admissible changes of numeral base, provided the encode/decode contract is honored.

This is not a stylistic preference. It is a falsifiability requirement. If a result changes when you change the base, you are not seeing a property of the number—you are seeing a property of your encoding.

4.2 The Encode/Decode Contract

We define the admissible change-of-base operation in the strictest possible way: by an encode/decode round-trip identity.

Let $\text{encode\_base}_b(w)$ be a representation map from an integer $w \in \mathbb{Z}$ to a finite string over an alphabet (e.g., 0–9a–z) in base b. Let $\text{decode\_base}_b(s)$ be the inverse map from that string back to an integer.

The admissibility contract is:

$$\text{decode\_base}_b(\text{encode\_base}_b(w)) = w$$

A base change is lawful only if it respects this contract for every integer that enters the computation.

Any claim of "cross-base behavior" in Marithmetics is meaningless unless it is paired with (i) an explicit contract, and (ii) a negative control that intentionally violates the contract and triggers failure.

4.3 AoR Evidence: DEMO-64 (Selector Invariance Under Base Gauge)

DEMO-64 is the AoR's referee-ready, isolated audit of base-gauge invariance at the level of the integer selector itself—before any higher physics overlays are applied.

In **Stage 1**, the selector runs in integer mode with no encoding layer. Under the declared lane constraints, it produces a unique admissible triple:

$$(w_U, s_2, s_3) = (137, 107, 103)$$

and records derived invariants:

$$q_2 = 30, \quad q_3 = 17, \quad v_{2U} = 3, \quad \varepsilon \approx 0.18257419$$

In **Stage 2**, the demo runs the same selector with a cross-base encoding layer enabled over the tested bases:

$$b \in \{2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 16\}$$

For each base, the encode/decode contract is enforced; the round-trip failures are zero; and the selected triple and lane survivor pools match the baseline.

In **Stage 3**, the demo intentionally violates the contract ("wrong base plus one") and proves sensitivity: contract failures are triggered and the triple is not recoverable under the broken encoding.

This is the minimal "representation audit" you want before you allow any reader to accuse you of base-dependent numerology.

Evidence (DEMO-64):

- DEMO-64 stdout log (full staged audit): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/substrate__demo-64-base-gauge-invariance-integer-selector.out.txt
- DEMO-64 spec (declared bases, lanes, and the explicit contract): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/vendored_artifacts/substrate__demo-64-base-gauge-invariance-integer-selector__spec.json
- DEMO-64 deterministic record (machine-readable triple/pools and negative control summary): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/vendored_artifacts/substrate__demo-64-base-gauge-invariance-integer-selector__deterministic_record.json

4.4 AoR Evidence: DEMO-39 (Cross-Base Round-Trip of A2 Constants)

DEMO-64 establishes that the selector is invariant under base gauge. DEMO-39 goes further: it shows that the A2 constant suite itself is stable under representation changes, including explicit base-7 and base-16 encodings.

DEMO-39 reports (among other closures) the canonical gauge fractions:

$$\alpha = \frac{1}{137}, \qquad \alpha_s = \frac{2}{17}, \qquad \sin^2\theta_W = \frac{7}{30}$$

and then performs a representation audit: generate cross-base representations of multiple constants, parse them back, and show that the numeric values are preserved within a declared tolerance (with explicit maximum round-trip deltas reported).

A representative excerpt of the cross-base table (DEMO-39) is:

constant	**base10 value**	**repr (base7)**	**repr (base16)**
α	0.0072992700729927	0.002334515606303135(base7)	0.01de5d6e3f8868a000(base16)
α_s	0.11764705882352941	0.05523202611434605(base7)	0.1e1e1e1e1e1e1e0000(base16)
$\sin^2\theta_W$	0.23333333333333337	0.143014301430143014(base7)	0.3bbbbbbbbbbbbc0000(base16)
H_0	70.44939596437767193	130.310066662063631102(base7)	46.730b9d29f4dc000000(base16)

DEMO-39 also reports the Rosetta-layer integrity checks (including CRT injectivity on the "good" set and collision existence on the "bad" set), and gives maximum round-trip deltas across bases for both the phi_hat path and the A2 constants path.

The logical point is simple:

- If the same A2 constants can be expressed in multiple bases,
- and the parse round-trip returns the same numeric values to within the stated tolerance,
- then the reader is forced to concede that the constants are not artifacts of a preferred decimal encoding.

Evidence (DEMO-39):

- DEMO-39 stdout log (includes A2 closures + cross-base round-trip table): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/cosmo__demo-39-bb-a2.out.txt
- AoR constants ledger (for independent table-based lookup): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/constants_master.csv

4.5 Why This Matters Before Any Physics

This section is deliberately placed before we discuss any of the "physics-shaped" outputs. The order is intentional:

1. First establish that the computational substrate is not dependent on writing conventions.
2. Then establish that the selector and its invariants survive changes of representation.
3. Then allow physics-domain closures to be interpreted as outputs of the substrate rather than outputs of a formatting choice.

This is the minimum scaffolding needed for a hostile referee to even begin engaging with the rest of the work in good faith.

Next section (preview): we now formalize the finite substrate objects that make base-gauge invariance possible—digital-root reductions, lawful residue action, and the exact role of DRPT structure in constraining the search space.

5 Finite Substrate Objects: Residues, Digital Roots, and DRPT Dynamics

This section fixes the finite substrate layer used throughout Marithmetics: the canonical residue ring attached to a base choice, the lawful digit maps that present it, and the powering dynamics that become visible in digital-root power tables (DRPTs). This is the "floor" the rest of the system stands on. DOC adopts exactly this stratification: finite substrate → operator layer → transfer/PDE layer, with DRPTs explicitly serving as a substrate diagnostic for residue-power dynamics and unit/non-unit behavior.

5.1 Base choice and its canonical modulus

Let $b \in \{2, 3, 4, \dots\}$ be a presentation base. The base is not a physical constant; it is a coordinate choice for writing integers. The substrate object canonically attached to b is:

$$d \;:=\; b - 1, \qquad \mathbb{Z}_d := \mathbb{Z}/(d\mathbb{Z}).$$

This is the residue ring that governs digit-sum and digital-root behavior in base b. DOC explicitly uses this ring as the substrate layer ("finite substrate layer... residue rings $\mathbb{Z}_d$ with $d = b - 1$ for a chosen base b"), and then builds DRPTs as tables of powering dynamics on this ring.

Why $d = b-1$ and not $d = b$? Because all "carry-based" digit systems satisfy the fundamental congruence

$$b \equiv 1 \pmod{b-1},$$

so any base-b numeral expansion is automatically a polynomial in b evaluated at $b \equiv 1$, i.e. a digit-sum congruence modulo $b - 1$. This is not numerology; it is the algebra of positional notation.

5.2 The digit-sum congruence (lawful digit map)

Write a nonnegative integer n in base b as

$$n = \sum_{k=0}^{K} a_k b^k, \qquad a_k \in \{0, 1, \dots, b-1\}.$$

Define the base-b digit sum

$$s_b(n) := \sum_{k=0}^{K} a_k.$$

Lemma 5.1 (Digit-sum congruence). *For every $n \geq 0$,*

$$n \equiv s_b(n) \pmod{b-1}.$$

Proof. Since $b \equiv 1 \pmod{b-1}$, we have $b^k \equiv 1^k = 1 \pmod{b-1}$ for each k. Therefore

$$n = \sum_{k=0}^{K} a_k b^k \equiv \sum_{k=0}^{K} a_k = s_b(n) \pmod{b-1}.$$

□

This lemma is the precise reason that many "digit tricks" exist at all: they are residue laws in disguise.

5.3 Digital roots as residue representatives

Define the base-b digital root map $\mathrm{dr}_b : \mathbb{N} \to \{0, 1, \dots, b-1\}$ by repeated digit-sum reduction (standard), with the equivalent closed form:

$$\mathrm{dr}_b(0) = 0, \qquad \mathrm{dr}_b(n) = 1 + \big((n-1) \bmod (b-1)\big) \quad (n \geq 1).$$

So $\mathrm{dr}_b(n)$ is exactly the residue class of n in $\mathbb{Z}_{b-1}$, presented in the human-facing alphabet $\{1, 2, \dots, b-1\}$ with 0 singled out.

Lemma 5.2 (Digital root is residue)**.** *For all $n \geq 0$,*

$$\mathrm{dr}_b(n) \equiv n \pmod{b-1},$$

interpreting the "0" digital root as residue 0 and the nonzero roots $1, \ldots, b-1$ as their obvious classes.

Proof. Combine Lemma 5.1 with the fact that iterating digit sums does not change the residue mod $b-1$, and observe the closed form is precisely the canonical representative map. $\square$

Interpretation. Digital roots are not "mystical." They are a coordinate chart on $\mathbb{Z}_{b-1}$.

5.4 Units, non-units, and powering dynamics

The multiplicative behavior of residues in $\mathbb{Z}_d$ splits cleanly:

- A residue class $x \in \mathbb{Z}_d$ is a *unit* iff $\gcd(x, d) = 1$. The units form a finite group $(\mathbb{Z}_d)^\times$.
- A residue class is a *non-unit* iff $\gcd(x, d) > 1$. Non-units do not have multiplicative inverses and can exhibit "collapse" behavior under repeated powering.

Now fix $x \in \mathbb{Z}_d$ and consider the discrete dynamical system

$$x,\ x^2,\ x^3, \ldots \quad \text{in } \mathbb{Z}_d.$$

Because $\mathbb{Z}_d$ is finite, this sequence is eventually periodic (a standard finite-state argument). But DOC emphasizes the sharper qualitative distinction: units move on cycles, whereas non-units can fall into attractors (including 0 in many moduli) when viewed through powering dynamics. This distinction is one of the reasons DRPTs exist as a substrate diagnostic in the first place: DRPTs "describe the dynamics of residues under powering and distinguish units (finite cycles) from non-units (attractors)."

5.5 DRPT definition and what it actually encodes

Fix a base b and $d = b - 1$. The *Digital Root Power Table (DRPT)* in base b is the table

$$\mathrm{DRPT}_b(r, k) \ := \ \mathrm{dr}_b\big(r^k\big),$$

where $r \in \{0, 1, \ldots, b-1\}$ (a digital root representative) and $k \in \mathbb{N}$ is the exponent index.

Equivalently—and this is the definition that matters for proofs—DRPT_b is the powering table of $\mathbb{Z}_{b-1}$ expressed in the digital-root alphabet.

What DRPTs encode:

1. **Cycle structure of $(\mathbb{Z}_{b-1})^\times$.** For each unit u, the row $u^k \bmod d$ is purely periodic and its period divides the exponent of $(\mathbb{Z}_d)^\times$.
2. **Attractor structure of non-units.** For x with $\gcd(x, d) > 1$, powering can drive the sequence into idempotent/zero-like sinks depending on the CRT factorization of d.
3. **A finite transition graph.** Each row is a deterministic finite automaton on a set of size d. This gives a lawful "codebook" for powering constraints: any claim that depends on powering residues must be consistent with this automaton.

This is the correct place to say "DRPTs are the substrate," in the precise sense that they are a canonical presentation of powering dynamics on the finite residue substrate $\mathbb{Z}_{b-1}$. DOC explicitly places DRPTs in the substrate layer alongside residue rings and DFT infrastructure.

5.6 Audit capsule: base-gauge invariance is an encode/decode contract (DEMO-64)

The previous subsections were purely mathematical. We now record how the AoR operationalizes "base gauge" as a testable contract.

Define, for each base $b \geq 2$:

- $\mathrm{Enc}_b : \mathbb{N} \to \Sigma_b^*$: encode an integer as base-b digits.
- $\mathrm{Dec}_b : \Sigma_b^* \to \mathbb{N}$: decode base-$b$ digits back to an integer.

The encode/decode contract is:

$$\mathrm{Dec}_b(\mathrm{Enc}_b(n)) = n \quad \forall n \in \mathbb{N}.$$

An algorithm A that consumes "numbers" through an encoding layer is *base-gauge invariant* if its output is unchanged under replacing the presentation base b (with the contract enforced).

This is not a philosophical claim; it is explicitly audited in the AoR:

- DEMO-64 (stdout): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/substrate__demo-64-base-gauge-invariance-integer-selector.out.txt

In DEMO-64, the selector is run once in baseline integer mode, then run through cross-base encode/decode for bases 2,3,4,5,6,7,8,9,10,12,16, verifying that:

- the selected triple is invariant across bases, and
- the lane survivor pools are invariant across bases, and
- negative controls (intentionally violating the contract) break invariance (sensitivity).

These are exactly the properties required if DRPT/digit mechanics are to be treated as coordinate choices rather than hidden degrees of freedom. The AoR surfaces DEMO-64 and its vendored artifacts in the canonical URL map.

(See also the URL map entry for DEMO-64, including the corresponding stderr and vendored artifact hashes/specs: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/substrate__demo-64-base-gauge-invariance-integer-selector.out.txt https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/substrate__demo-64-base-gauge-invariance-integer-selector.err.txt)

5.7 What this substrate layer is for (and what it is not)

It is crucial to separate two roles:

- **Role A (lawful substrate constraints).** $\mathbb{Z}_{b-1}$, digital-root presentation, and DRPT dynamics define a lawful constraint system on integer powering behavior. They are admissible as proof objects because they reduce to modular arithmetic.
- **Role B (human visualization).** DRPTs also give an immediate visual compression of residue dynamics. This is epistemically useful, but it is not itself a proof. The proof is the residue law; the table is the readable representation.

Foundations uses DRPTs in Role A. A later dedicated DRPT visualization paper can fully exploit Role B (tilings, mirror symmetries, cross-base families), but those visuals will be treated there as presentations of already-lawful substrate facts, not as the source of truth.

6 Transition: From Substrate to Operators

With the substrate fixed—$\mathbb{Z}_{b-1}$, lawful digit maps, and power dynamics—we can state what "operators" mean in Marithmetics without ambiguity:

- The operator layer acts on finite state spaces built from $\mathbb{Z}_d$ and its indexed vector spaces (e.g., $\mathbb{C}^M$ with DFT basis).
- The admissibility rules (positivity, Fejér-type smoothing, projector commutation) are expressed as explicit finite conditions.
- The transfer layer then lifts operator statements to continuum analogues under explicit residual schedules.

DOC summarizes this as a three-layer architecture (substrate → operators → transfer/PDE), and it is exactly this architecture Foundations will now connect to the AoR demos and the SCFP++/GUM selection engine.

6.1 Discrete analytic space and the observer group

To speak precisely about "finite-to-continuum" without handwaving, we fix (for each finite instance) an explicit analytic state space.

Let $M \geq 2$ be an integer and define the cyclic index set

$$I_M := \{0, 1, \dots, M-1\} \cong \mathbb{Z}/M\mathbb{Z}.$$

Let $\mathcal{H}_M := \mathbb{C}^{I_M}$ with the normalized inner product

$$\langle x, y \rangle_M := \frac{1}{M} \sum_{n \in I_M} x_n \, \overline{y_n}, \qquad \|x\|_2^2 := \langle x, x \rangle_M.$$

Define the shift (translation) operator $T : \mathcal{H}_M \to \mathcal{H}_M$ by

$$(Tx)_n := x_{n-1 \pmod{M}}.$$

The finite "observer group" is the cyclic group

$$G_M := \{T^k : 0 \leq k \leq M-1\}.$$

This group action is the minimal, non-negotiable symmetry of the finite substrate: it encodes the fact that our coordinate choice on $\mathbb{Z}_M$ must not change any physical (or structural) claim. Put differently: if a claim is not stable under G_M, it is not a claim about the substrate; it is a claim about the labeling.

Definition 6.1 (Translation invariance)**.** An operator $A : \mathcal{H}_M \to \mathcal{H}_M$ is *translation-invariant* (equivalently, *circulant*) if $AT = TA$. A functional $F : \mathcal{H}_M \to \mathbb{R}$ is translation-invariant if $F(Tx) = F(x)$ for all x.

Proposition 6.2 (Fourier diagonalization of circulant operators)**.** *Let $\psi_k(n) := e^{2\pi i k n/M}$ for $k \in I_M$. Then $\{\psi_k\}_{k \in I_M}$ is an orthonormal basis of $\mathcal{H}_M$. An operator A is circulant if and only if each ψ_k is an eigenvector of A. In that case A is completely specified by its scalar multiplier $\widehat{A}(k) \in \mathbb{C}$ via*

$$A\psi_k = \widehat{A}(k) \, \psi_k.$$

This proposition is the first "anti-numerology" gate: a lawful operator is not described by a vibe or a plot, but by an explicit multiplier in the DFT basis. Everything that follows is built on this.

6.2 The Ω–Π split: universal invariants vs. fluctuations

Let $\mathbf{1} \in \mathcal{H}_M$ denote the constant vector $(1, 1, \ldots, 1)$. The span of $\mathbf{1}$ is the unique nontrivial subspace fixed by every shift T^k, and it corresponds to the $k = 0$ Fourier mode.

Definition 6.3 (Universal projector Ω)**.** Define

$$\Omega := \frac{1}{M} \mathbf{1}\,\mathbf{1}^*, \qquad \Pi := I - \Omega.$$

Then Ω is the orthogonal projector onto $\mathrm{span}\{\mathbf{1}\}$, and Π is the orthogonal projector onto its orthogonal complement.

Lemma 6.4 (Basic identities)**.** *For all $x \in \mathcal{H}_M$:*

1. $\Omega^2 = \Omega = \Omega^*$; $\Pi^2 = \Pi = \Pi^*$; $\Omega\Pi = \Pi\Omega = 0$.
2. $x = \Omega x + \Pi x$ *(orthogonal decomposition).*
3. $\|x\|_2^2 = \|\Omega x\|_2^2 + \|\Pi x\|_2^2$ *(Pythagorean split).*
4. $\Omega T = T\Omega$ *and* $\Pi T = T\Pi$ *(observer invariance).*

Interpretation:

- Ωx is the invariant "DC component" of x under the observer group.
- Πx is the fluctuation field—the part that can encode structure, dynamics, and constraints.

This Ω–Π split is not an optional convenience. It is the canonical mechanism by which the system prevents "anchoring-by-definition": anything that depends only on Ω is an invariant baseline; anything that claims structure must live in Π.

6.3 Fejér smoothing as the canonical admissible regularizer

The bridge from finite to continuum requires a controlled way to suppress high-frequency artifacts while respecting the observer group. The admissible tool for this is Fejér smoothing, because it is:

- translation-invariant (circulant),
- positive (PSD),
- normalized (preserves constants),
- explicitly diagonal in Fourier (so constraints commute cleanly).

Definition 6.5 (Discrete Fejér kernel on $\mathbb{Z}_M$)**.** Fix an integer $L \geq 1$. Define $F_L : \mathbb{Z}_M \to \mathbb{R}_{\geq 0}$ by

$$F_L(n) := \frac{1}{L^2} \sum_{j=-L+1}^{L-1} (L - |j|)\, \mathbf{1}_{\{j \equiv n \pmod{M}\}}.$$

Equivalently: F_L is the periodic "triangle" kernel of span L, normalized so its total mass is 1.

Definition 6.6 (Fejér smoothing operator $S_L^{(M)}$)**.** For $x \in \mathcal{H}_M$, define

$$(S_L^{(M)} x)_n := \sum_{m \in I_M} F_L(n - m)\, x_m.$$

This is convolution by F_L on $\mathbb{Z}_M$.

Lemma 6.7 (Admissibility properties of Fejér smoothing)**.** *For all $M \geq 2$, $L \geq 1$:*

1. $S_L^{(M)}$ *is circulant and self-adjoint.*
2. $S_L^{(M)} \succeq 0$ *(positive semidefinite).*
3. $S_L^{(M)}\mathbf{1} = \mathbf{1}$ *(preserves the invariant mode).*
4. $\|S_L^{(M)}x\|_2 \leq \|x\|_2$ *(contraction).*
5. *In the Fourier basis,* $S_L^{(M)}\psi_k = \widehat{S}_L^{(M)}(k)\psi_k$ *where*

$$\widehat{S}_L^{(M)}(k) = \begin{cases} 1, & k = 0, \\ \left(\dfrac{\sin(\pi L k/M)}{L\sin(\pi k/M)}\right)^2, & k \neq 0, \end{cases} \qquad \textit{so} \qquad 0 \leq \widehat{S}_L^{(M)}(k) \leq 1.$$

Definition 6.8 (Minor-lobe contraction factor)**.** Define

$$\rho_{M,L} := \max_{k\neq 0} \widehat{S}_L^{(M)}(k) \in [0,1).$$

Then for all x with $\Omega x = 0$, we have

$$\|S_L^{(M)}x\|_2 \leq \rho_{M,L}\,\|x\|_2.$$

This is the second "anti-numerology" gate: Fejér smoothing is not a heuristic blur; it is a certified spectral projector-weighted contraction with explicit multipliers and explicit, measurable contraction factor.

6.4 What "admissible operator" means in Marithmetics

The purpose of DOC is not to fetishize Fourier methods. The purpose is to enforce a concrete legal boundary: only operators that respect the symmetries and positivity of the finite substrate are allowed to participate in claims that survive the limit.

Definition 6.9 (Admissible operator family)**.** A family $\{A_M\}_{M\geq 2}$ with $A_M : \mathcal{H}_M \to \mathcal{H}_M$ is *admissible* if, for each M:

1. **(Observer invariance)** A_M is circulant: $A_M T = T A_M$.
2. **(Reality and stability)** $A_M = A_M^*$ and $\|A_M\|_{2\to 2} \leq C$ for a uniform constant C.
3. **(Positivity)** $A_M \succeq 0$.
4. **(Calibration)** $A_M\mathbf{1} = \mathbf{1}$, i.e., $\widehat{A}_M(0) = 1$.

When the operator acts on vector/tensor fields, these conditions apply componentwise with the appropriate inner product, and "circulant" means the operator is a Fourier multiplier in each component.

Remark 6.10 (Why these axioms).

- (1) blocks coordinate dependence.
- (2) prevents "blow-up by discretization."
- (3) blocks spurious cancellations that masquerade as structure.
- (4) forces a shared baseline mode and prevents hidden renormalizations.

These conditions are intentionally severe. They make certain common numerical stencils inadmissible unless they can be shown to reduce to a lawful multiplier plus a controlled residual.

6.5 The deterministic operator calculus boundary

We now state the foundational rule that governs every downstream bridge claim.

Principle 6.11 (DOC boundary)**.** A finite-to-continuum claim is allowed to be published as a Tier-A statement in Marithmetics only if it is expressible as an operator identity plus an explicit residual bound, with each operator factor admissible.

Concretely: if a continuum operator $\mathcal{A}$ (on the torus or a continuum configuration space) is being matched by a finite operator A_M, the claim must be of the form

$$\|A_M - \iota_M^* \mathcal{A}\, \iota_M\|_{2\to 2} \ \le\ C\, \eta_M, \qquad \eta_M \to 0 \quad (M \to \infty)$$

for an explicit deterministic residual η_M produced by an auditable schedule (the "lift pipeline"), and for an explicit injection/restriction pair ι_M that is itself part of the contract.

This makes three failure modes impossible to hide:

1. **Undeclared renormalization**: blocked by calibration and explicit ι_M.
2. **Stencil anisotropy**: revealed in operator norm residuals and commutators.
3. **Cherry-picked convergence**: blocked by schedule + ledgered runs.

6.6 AoR evidence that DOC is operational (not philosophical)

The purpose of Foundations is not to ask the reader to trust DOC as a nice-sounding principle. Foundations must show that DOC is already implemented as a working constraint in the AoR—and that it is tested against negative controls.

The canonical AoR provides controller-level demos that explicitly audit this boundary. In particular:

- **DEMO-56** (Deterministic Operator Calculus vs. finite differences).
  Stdout log:
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt
  Stderr log:
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.err.txt
- **DEMO-69** (OATB: Operator Admissibility Transfer Bridge).
  Stdout log:
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt
  Stderr log:
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.err.txt

These are cited here for one reason: they encode (in executable evidence) the claim that "DOC is the boundary" and they display what happens when one attempts to step outside the boundary (finite-difference approximations, commutator leakage, or schedule violations).

Finally, note that the AoR includes explicit tables intended to make the evidence navigable and falsification-oriented:

- demo index:
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/demo_index.csv
- falsification matrix:
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/falsification_matrix.csv
- run reproducibility table:
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/run_reproducibility.csv

In the next section we formalize what it means for "laws" to emerge from the substrate (without guessing), and we connect that formalism to the AoR's lawbook emergence evidence.

7 Lawbook Emergence: Making Selection Visible (SCFP++ → Laws)

This section is the "visibility layer." The selector is not only a mechanism that outputs integers; it is also a witness generator that makes familiar continuum laws appear as unique fixed points under finite, deterministic constraints.

The canonical evidence for the sweeps in this section is:

- `foundations__demo-53-lawbook-emergence.out.txt`
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/foundations__demo-53-lawbook-emergence.out.txt

The purpose of DEMO-53 is explicitly stated in its header: "MAKE SELECTION VISIBLE ... Stdlib-only · stdout-only · no file I/O." This is a deliberate design choice: it compresses the "lawbook emergence" story into a minimal, portable certificate that still remains deterministic and audit-grade.

7.1 SCFP++ Prime Emergence From Scratch (Lane-Gated Search)

We begin with the simplest statement: the lane-gated selector produces the canonical survivor pools and then a unique admissible triple in the primary window.

From DEMO-53 Stage 0:

- U(1) survivors: $[103, 107, 137]$
- SU(2) survivors: $[107]$
- SU(3) survivors: $[103, 137]$
- Selected triple: $(w_U, s_2, s_3) = (137, 107, 103)$

These are the same pools and the same selected triple exhibited by DEMO-64 Stage 1 (baseline integer mode) and later audited for cross-base invariance in DEMO-64 Stage 2:

- `substrate__demo-64-base-gauge-invariance-integer-selector.out.txt`
https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/substrate__demo-64-base-gauge-invariance-integer-selector.out.txt

The key point here is not the numerology of the integers; it is the structure of the selection: finite window, finite residue constraints, deterministic scoring, and uniqueness.

DEMO-53 then computes two derived quantities immediately (using only arithmetic, not fitting):

1. A derived ratio
$$h := \frac{s_3}{w_U} = \frac{103}{137} \approx 0.751824817518.$$
2. A derived modulus value
$$q_3 := 17,$$
which is consistent with the "odd part" rule described in Section 5 and explicitly printed (as a derived value) in DEMO-64.

As an example of why the demos are architected to be referee-friendly, DEMO-53 then shows a direct downstream mapping:

$$\alpha_s(M_Z) = \frac{2}{q_3} = \frac{2}{17} \approx 0.117647058824,$$

printed directly in Stage 0 of DEMO-53.

This is not yet a "full Standard Model closure"; it is the "first visible bead" of the lawbook: a rational coupling extracted from the same substrate invariants that later constrain the flagship physics demos.

7.2 Noether Visibility: Break Time-Translation, Watch Energy Fail

The next move is conceptual and deliberately familiar:

- In classical mechanics, time-translation symmetry corresponds to energy conservation (Noether's theorem).
- In this program, we treat that as a visibility experiment: we deliberately perturb time-translation invariance and observe the failure of energy conservation as a quantitative drift.

DEMO-53 Stage 1 implements exactly that: it introduces a parameter ε that breaks time-translation invariance and reports a relative energy drift. The budgets are derived from the previously computed h:
$$dt = \frac{h}{15} \approx 0.050122, \quad \text{steps} = 599.$$

The reported sweep is:

ε	relative energy drift
0.000	-1.511102×10^{-4}
0.010	-4.012386×10^{-3}
0.020	-7.868054×10^{-3}
0.050	-1.940103×10^{-2}
0.100	-3.850751×10^{-2}

DEMO-53 labels the result:

✓ PASS: energy conservation emerges only when time-translation invariance holds (eps=0)

Interpreting this correctly matters. The "point" is not that numerical schemes have drift (everyone knows they can). The point is that the demo is structured as a symmetry witness:

- When the symmetry is preserved ($\varepsilon = 0$), energy drift is minimal (here on the order of 10^{-4}, under the demo's derived budget).
- When the symmetry is broken (nonzero ε), drift increases systematically and substantially, without tuning.

This is what "lawbook emergence" means in this program: the law is not asserted; it is selected by a constraint principle, and then demonstrated by a controlled failure when the constraint is violated.

7.3 Inverse-Square Selection: Flux Fixed Point Picks $p = 2$

Stage 2 addresses a canonical continuum law-form: the inverse-square dependence that underwrites Gauss flux constancy (and, in conventional presentation, electrostatics and Newtonian gravity).

Rather than assuming an inverse-square law, DEMO-53 performs a deterministic sweep over a power p and evaluates a flux-constancy score.

The sweep is reported as:

p	score(clean)	score(5% noise)
0.00	2.818535	2.816692
0.25	2.499471	2.500606
0.50	2.174292	2.172906
0.75	1.841686	1.841238
1.00	1.500000	1.497571
1.25	1.147236	1.147526
1.50	0.781119	0.783855
1.75	0.399337	0.402925
2.00	0.000000	0.039496
2.25	0.417666	0.418745
2.50	0.852561	0.856331
2.75	1.301182	1.300953
3.00	1.757699	1.754572

And the demo certifies:

✓ PASS: Gauss flux constancy selects $p = 2$ (inverse-square) as unique law form

There are three features worth emphasizing for a skeptical reader:

1. **It is a sweep, not a solve.** The reader can see the entire landscape.
2. **The minimum is sharp.** The clean score is exactly 0 at $p = 2$ in this certificate.
3. **Noise robustness is displayed.** With 5% noise, the minimum remains near $p = 2$.

This is an example of how the program turns "laws" into fixed points under admissibility constraints rather than postulates.

7.4 Isotropic Laplacian: Continuum Rotational Symmetry Selects $w_2 = 1/6$

Stage 3 addresses the discrete-to-continuum question that sits beneath nearly every numerical PDE scheme: how does a discrete stencil recover continuum isotropy?

The demo sweeps a weight parameter w_2 that (in effect) interpolates among Laplacian stencils and reports an anisotropy metric measured at two wavenumbers ($k = 0.3$ and $k = 1.2$).

The table includes (excerpting the key lines):

w_2	aniso @ $k = 0.3$	aniso @ $k = 1.2$	note
0.000000	1.871838×10^{-3}	2.919806×10^{-2}	
0.125000	4.724979×10^{-4}	8.423430×10^{-3}	
0.142857	2.724318×10^{-4}	5.418404×10^{-3}	
0.166667	5.614453×10^{-6}	1.396797×10^{-3}	(classic 9-pt)
0.175000	8.778849×10^{-5}	1.703603×10^{-5}	(tuned high-k)

The demo's interpretive note is essential:

> *Note: you can tune w_2 to reduce anisotropy at a single lattice-scale k, but that is an explicit hidden knob. The emergent law is the continuum-isotropic fixed point.*

And then:

> ✓ PASS: small-k rotational isotropy selects $w_2 = 1/6$ uniquely (no single-scale tuning)

This is the clearest "lawbook emergence" example in the section, because it makes a clean philosophical separation:

- **Tuning** is the selection of a parameter to optimize a single scale (a hidden knob).
- **Emergence** is the selection of a parameter by continuum symmetry (small-k isotropy), which is the meaningful fixed point if one is claiming a continuum law emerges from a finite substrate.

7.5 Unitarity Selection: θ-Scheme Sweep Picks $\theta = 1/2$

Stage 4 moves to the quantum side of the bridge, but retains the same "fixed point selection" style.

The demo sets up a θ-scheme family (a standard one-parameter family of time-stepping methods), then measures norm drift. In conventional numerical analysis language, this sweep is essentially asking:

- which θ yields a unitary (norm-preserving) evolution for the test setup?

Budgets are fully derived, not tuned:

- $N = 64$
- $dx = q_3 = 17$
- $dt = h/5 \approx 0.150364963504$
- steps $= 600$

The sweep is:

θ	norm drift
0.0	$+5.975851 \times 10^{-4}$
0.1	$+4.780332 \times 10^{-4}$
0.2	$+3.584987 \times 10^{-4}$
0.3	$+2.389817 \times 10^{-4}$
0.4	$+1.194821 \times 10^{-4}$
0.5	$+4.352074 \times 10^{-14}$
0.6	-1.194647×10^{-4}
0.7	-2.389119×10^{-4}
0.8	-3.583416×10^{-4}
0.9	-4.777539×10^{-4}
1.0	-5.971487×10^{-4}

And the demo certifies:

✓ PASS: $\theta = 0.5$ is the unique unitary fixed point

In standard terminology, $\theta = 1/2$ corresponds to the Crank–Nicolson midpoint form (time-reversible and norm-preserving for the linear Schrödinger-like settings the demo is probing). The program's claim is not "we discovered Crank–Nicolson"; the claim is:

when you express the problem as "find the unitary fixed point under derived budgets," the correct method is singled out uniquely without tuning.

7.6 What "Lawbook Emergence" Means Here (And What It Does Not Mean)

This section has not proven physics in the sense of deriving Maxwell's equations from first principles in a single theorem. What it has done—concretely and auditably—is show that, within the Marithmetics framework:

1. A deterministic finite selector produces a canonical substrate triple $(137, 107, 103)$ and derived invariants such as $q_3 = 17$ and $h = 103/137$.
2. Using budgets derived from those invariants, the system performs visible parameter sweeps where continuum-meaningful fixed points appear uniquely:
   - energy conservation requires time-translation invariance (Noether visibility),
   - inverse-square is selected by flux constancy,
   - continuum isotropy selects the Laplacian weight $w_2 = 1/6$,
   - unitarity selects $\theta = 1/2$.

These are the "lawbook primitives." The later flagship demos do not treat these as independent facts; they treat them as coupled constraints—the same invariants and admissibility principles that fix the selector also fix the legal operators, the transfer rules, and the spectral budgets that propagate through the physics stack.

In other words:

- DEMO-53 is where the reader sees the laws emerge.
- DEMO-64 is where the reader sees the selector survive a hostile representational audit (cross-base invariance + negative controls).
- The flagship demos (e.g., OATB, Standard Model closure, cosmology) are where the reader sees those same constraints cash out as hard, reproducible numbers.

We now turn to the operator legality principle—admissibility—not as a philosophical slogan, but as the explicit criterion that separates lawful continuum limits from illegal ringing, negative densities, and "paradox artifacts."

8 Operator Legality: Admissibility, Transfer, and Designed Failure

The previous section explains how lawbooks emerge: a small, rigid set of finite algebraic relations that survive harsh selection. This section answers a different question:

> *When we map those finite relations into analysis—into operators that act on function spaces—how do we prevent illegal moves, hidden degrees of freedom, and "numerology by implementation"?*

Marithmetics does not treat analysis as a poetic metaphor. Analysis is a governed interface. "Operator legality" is the set of rules that makes the interface enforceable.

8.1 The DOC baseline: why this is allowed mathematics (and not a new logic)

All analytic statements in this work are written inside a conservative deterministic operator calculus (DOC): a definitional extension that packages concrete ZFC-provable facts about finite rings, Hilbert spaces, Fejér kernels, discrete operators, and their limits into an operator language.

The key point is **conservativity**:

> *If a closed analytic statement is provable in the DOC calculus, then (in the intended fragment) it is already provable in ZFC about the concrete model. The DOC adds structure and auditability; it does not add deductive power.*

This is not a slogan—it is the formal backbone that prevents "we assumed the continuum did what we wanted."

8.2 Admissibility: the minimum legality conditions for any operator claim

Marithmetics repeatedly uses three kinds of operators:

- **Projectors** (e.g., Π) encoding restriction/selection,
- **Observer/gauge actions** (e.g., Ω) encoding symmetry,
- **Smoothing / truncation operators** (e.g., S_L, Fejér-type) encoding lawful finite-to-analytic lifting.

We call an operator family "admissible" when it satisfies the minimum conditions needed to make quadratic invariants meaningful and transportable across domains.

Definition 8.1 (Admissible operator family)**.** An operator family A on a domain D is *admissible* (relative to the governing structure (Ω, Π, S_L)) if it satisfies:

1. **Positivity**: the induced quadratic form is nonnegative (PSD in the appropriate sense).
2. **Orthogonality / projector legality**: idempotence and complement structure are exact when claimed.
3. **Calibration**: the constant mode is normalized appropriately (e.g., $A\mathbf{1} = \mathbf{1}$ where required).

4. **Commutation / compatibility**: commutators with the governing structure are controlled (ideally tiny or exactly zero in the finite model, and bounded in the lift).

These are not aesthetic. They are the "no-cheating" constraints that make invariants stable under representation changes and under scale changes. In the certified operator corpus, these are tracked as axioms A1–A4 and verified by explicit audits (e.g., PSD eigenvalue auditors, idempotence checks, calibration checks, commutator norm bounds).

8.3 Admissible transfer: how finite objects become analytic objects without smuggling freedom

A central move in the system is a round-trip:

$$\text{discrete} \xrightarrow{L_h} \text{continuum} \xrightarrow{D_h} \text{discrete}$$

This is where most "numerology attacks" hide in other frameworks: if the lift is flexible, you can always tune it; if the downlift is flexible, you can always cherry-pick.

Marithmetics forbids that flexibility by requiring that the lift/downlift come with an error ledger and an explicit residual budget η_h that composes monotonically across steps.

In the admissible transfer substrate (ATLAS), this is expressed operationally as:

- a lift $L_h u_h = S_{r(h)} u_h$ whose energy deviation is bounded by η_h,
- a projector transfer bound controlled by commutator norms,
- a continuum-to-discrete downlift $D_h u$ (sampling/interpolation with optional lawful prefactors),
- and a schedule that forces $\eta_h \to 0$ under explicit scale choices.

This is exactly the kind of structure that turns "we observed convergence" into "we controlled convergence under a declared budget."

8.4 AoR empirical legality checks: DOC versus naive finite differences (and why this matters)

Even if the DOC calculus is conservative, a skeptic can still ask:

> "Fine. But did your implementation accidentally bake in the answers? Does it agree with an independent numerical method?"

This is the purpose of a class of authority demos that do **comparative legality**, not just "produce a number."

DEMO-56 is an explicit DOC-versus-finite-difference comparison. In Foundations terms, it functions as a sanity boundary: the operator calculus is audited against an independent discretization baseline so that the legal pipeline is not merely self-consistent; it is cross-consistent. The AoR evidence for this comparison is recorded as the stdout/stderr pair for DEMO-56.

Evidence (AoR):

- `https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt`
- `https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.err.txt`

8.5 The OATB bridge: admissibility must survive representation changes

The system's strongest claims are not "I can compute a constant once." They are:

- the constant is rigid under the legal pipeline,
- the constant is invariant under admissible representation changes,
- the constant is invariant under base/gauge transformations,
- and the constant persists across finite/analytic/geometric realizations (within declared tolerances).

The operator mechanism that makes this checkable is the **Operator Admissibility Transfer Bridge (OATB)**, which audits that the same invariant survives when transported through different admissible representations.

DEMO-69 is the authority record for that bridge in the AoR bundle.

Evidence (AoR):

- https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt
- https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.err.txt

8.6 Electromagnetism as an operator-stability test (not a "physics story")

A common failure mode in foundational projects is that physics appears only as a narrative overlay: definitions first, metaphors second, numbers last.

Marithmetics flips that. Physics demos are used as **operator stress tests**:

- Do the admissibility constraints survive the translation into field equations?
- Do invariants remain invariant when the domain and discretization change?
- Does the pipeline refuse illegal operator choices?

The AoR bundle includes an explicit electromagnetism demo under the controllers suite, with stdout/stderr recorded as authority evidence.

Evidence (AoR):

- https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-59-electromagnetism.out.txt
- https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-59-electromagnetism.err.txt

8.7 Designed failure: how we prevent "pretty coincidences" from surviving

A legality system is only real if it can say NO.

That is what **designed failures** are for: intentional perturbations that must break the invariants, must trip the auditors, and must refuse to pass under the same pipeline that produces the flagship results.

This is not an optional philosophical add-on. It is the mechanical difference between:

- "a pipeline that can produce outcomes," and
- "a pipeline that can certify outcomes."

In the AoR bundle, designed failures and reproducibility checks are recorded as tables that a referee can ingest directly (without trusting prose summaries). The falsification catalog and run reproducibility ledger are first-class evidence artifacts in the release bundle.

Evidence (AoR tables):

- https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/falsification_matrix.csv
- https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/falsification_matrix.json
- https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/run_reproducibility.csv
- https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/run_reproducibility.json

What matters conceptually is simple:

- If a base interpretation is wrong, invariants must drift and be flagged.
- If moduli are mis-specified (e.g., violating coprimality assumptions), injectivity must fail and produce collision witnesses.
- If a kernel violates PSD, positivity gates must fail.
- If cross-base semantics are corrupted, base-gauge invariance must fail.
- If reproducibility is broken, the ledger must not validate.

In other words: the same formalism that yields the flagship constants must, under controlled sabotage, yield a documented body count.

That is the difference between "the system looks consistent" and "the system has teeth."

9 Audit-Grade Evidence Surfaces (How "Authority" Is Implemented)

This Foundations paper makes two promises that are normally in tension:

1. **Mathematical clarity**: the reader can understand the system as a coherent theory.

2. **Audit-grade reproducibility**: the reader does not need to trust the author, the narrative, or the formatting—because every hard claim resolves to a fixed artifact.

The Authority of Record (AoR) is where this tension is resolved. The AoR is not "supporting material." It is the cryptographic substrate that makes this Foundations paper legally and scientifically meaningful.

9.1 The AoR Identity (Tag, Folder, Bundle Seal)

All public claims in this paper are bound to a single AoR release tag and archive folder:

- **AoR tag**: `aor-20260209T040755Z`
- **AoR folder**: `gum/authority_archive/AOR_20260209T040755Z_0fc79a0`
- **Bundle sha256**: `c299b1a7a8ef77f25c3ebb326cb73f060b3c7176b6ea9eb402c97273dc3cf66c`

These three elements are the reader's first integrity check. If a citation does not resolve under this tag + folder, it is not a citation for this Foundations paper.

9.2 Canonical AoR Artifacts (The Reader's Entry Points)

The AoR is intentionally published with both human-facing and machine-facing entry points:

- Master zip:
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/MARI_MASTER_RELEASE_20260209T040755Z_0fc79a0.zip
- GUM report (v32):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/report/GUM_Report_v32_2026-02-09_04-27-46Z.pdf
- Report manifest:
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/report/GUM_Report_v32_2026-02-09_04-27-46Z.pdf.manifest.json

  Interpretation:

  - The report is the narrative map: it groups results and gives the reader an orientation.
  - The manifest is the immutable binding: it is what prevents "quiet edits" to the report itself.
  - The master zip is the offline distribution surface for long-term archiving.

9.3 Core Ledgers (The Chain-of-Custody Surface)

The AoR provides a minimal ledger set that connects claims → runs → outputs:

- claim_ledger.jsonl (append-only claim stream):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/claim_ledger.jsonl
- SUMMARY.md (human index + claim grouping):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/SUMMARY.md

- runner_transcript.txt (execution transcript): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/runner_transcript.txt

- run_metadata.json (environment + run identity): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/run_metadata.json

  Operationally:

  - The claim ledger is the "legal" backbone: it is the authoritative register of what is being claimed under this AoR.
  - The summary is the "index": it lets a human find where claims live.
  - The runner transcript is the "timeline": it shows what was executed and in what order.
  - The run metadata is the "identity certificate": it records what environment produced the artifacts.

9.4 Bundle Seal and Indices (The "Scoreboard Tables")

The AoR also publishes standardized tables so results can be parsed and checked mechanically:

- bundle_sha256.txt: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/bundle_sha256.txt

- constants_master.csv / constants_master.json: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/constants_master.csv https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/constants_master.json

- demo_index.csv / demo_index.json: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/demo_index.csv https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/demo_index.json

- falsification_matrix.csv / falsification_matrix.json: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/falsification_matrix.csv https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/falsification_matrix.json

- run_reproducibility.csv / run_reproducibility.json: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/run_reproducibility.csv https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/run_reproducibility.json

Interpretation:

- constants_master.* is the machine-readable trophy case (the canonical list of derived constants / outputs).
- demo_index.* is the routing table from result → demo → artifacts.
- run_reproducibility.* is the replication contract (what the runner considers a reproducible run boundary).
- falsification_matrix.* is the negative-controls contract (what is supposed to fail, and under which perturbations).

This is the mechanism by which Marithmetics avoids the classic failure mode of "beautiful prose, hidden tuning." The project does not merely claim falsifiability; it publishes falsification as a first-class artifact.

9.5 Demo-by-Demo Evidence Surface (Stdout, Stderr, Vendored Artifacts)

Every major claim is anchored to one or more demo executions. Each demo typically has:

- a stdout log (primary evidence),
- a stderr log (secondary evidence, warnings, runtime notes),
- optionally vendored artifacts (plots, JSON results, SHA ledgers, spec records).

For example, the AoR's demo table includes (among others):

- DEMO-56 (DOC comparison demo)
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.err.txt
- DEMO-59 (Electromagnetism)
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-59-electromagnetism.out.txt
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-59-electromagnetism.err.txt

The full table is the authoritative map. If a reader cannot locate a demo artifact from the URL map, then (by definition) it is not part of this AoR.

9.6 Canonical Citation Format (What Counts as "Cited" in Marithmetics)

A citation in this paper is defined operationally as:

> *A stable URL under the AoR tag and folder, pointing to a specific ledger/table/log/artifact that is sufficient for a reader to verify the claim without additional trust.*

Practically, that means a valid Marithmetics citation includes:

1. The AoR tag (fixed),
2. A direct URL to the evidence artifact (fixed),
3. Optional: a cross-reference to the claim ledger or constants table entry.

Example (Lawbook emergence demo):

- DEMO-53 stdout (primary evidence): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/foundations__demo-53-lawbook-emergence.out.txt
- DEMO-53 stderr (secondary evidence): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/foundations__demo-53-lawbook-emergence.err.txt

From this point onward, this paper uses only URLs present in the canonical URL map as its public evidence surface.

10 Verification Workflow (How a Hostile Referee Checks This Paper)

This section gives a reader an explicit, minimal workflow for verification. The intent is not to be ceremonial. It is to remove ambiguity about what "verification" means for this project.

10.1 The 10-Minute Verification (No Code Execution)

A skeptical reader can perform a "static" verification that the AoR is internally consistent:

1. Open the GUM report (human map).
2. Open the report manifest and confirm it binds to the published PDF. (Same citation.)
3. Open `SUMMARY.md` and confirm the narrative structure matches the report and the demo table.
4. Open `constants_master.csv` and confirm it is a coherent, parseable list of results (the project's "trophy case").
5. Open `demo_index.csv` and confirm it routes results to actual demos and artifacts. (Same citation.)
6. Open `bundle_sha256.txt` and confirm the AoR publishes an explicit, stable seal for the bundle. (Same citation.)

This does not prove the science. It proves the chain-of-custody is present, and that the "authority" structure is not merely rhetorical.

10.2 The 1–2 Hour Verification (Targeted Reproduction)

A targeted reproduction means: pick a narrow claim, reproduce that claim, and confirm the output matches the AoR.

The authoritative sources for how the AoR was executed are:

- `runner_transcript.txt` (what was run, in what order).
- `run_metadata.json` (what environment produced it).
- `run_reproducibility.*` (what the project considers part of the reproducibility boundary).

A hostile referee does not need to reproduce "everything." Reproducing a single Tier-A capsule and confirming it matches the AoR already tests the central claim: that the system is deterministic under the published constraints.

10.3 The Designed-Failure Check (The Anti-Numerology Gate)

Classic numerology is resilient because it can always be retrofitted. The only reliable way to prevent this is to publish negative controls: checks that are expected to fail when the rules are violated.

This AoR publishes the negative-control surface explicitly:

- `falsification_matrix.csv / .json`

This table is interpreted as part of the definition of the system: if the falsification battery does not fail under the listed perturbations, then the system is not enforcing its own constraints and the entire interpretability claim collapses.

In other words: in Marithmetics, falsification is not a philosophical posture; it is an evidence artifact.

11 DOC/DAO Operator Legality Certificates

This section records the deterministic operator-legality certificates included in the AoR bundle. The purpose is narrow and referee-facing:

1. To show that the continuum-facing operations used anywhere in Marithmetics (smoothing, filtering, transfer, coarse-graining, control) are realized as explicit finite operators with checkable invariants (no hidden "analysis magic").
2. To show that illegal interventions are not merely discussed philosophically but are operationalized as negative controls that reliably break the invariants ("designed-fails").
3. To show "teeth": the system is sensitive to deterministic counterfactuals (alternate triples) and degrades as predicted.

The primary AoR evidence for this section is the pair of operator flagship audits:

- **DEMO-56** — "Deterministic Operator Calculus vs Classical Finite Differences"
  Evidence (AoR stdout): https://github.com/public-arch/Marithmetics/blob/release-aor-20260125T043902Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0 /GUM_BUNDLE_v30_20260209T040755Z logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt
- **DEMO-69** — "OATB Master Flagship (Operator Admissibility Transfer Bridge)"
  Evidence (AoR stdout): https://github.com/public-arch/Marithmetics/blob/release-aor-20260125T043902Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0 /GUM_BUNDLE_v30_20260209T040755Z logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt

A third, domain-instantiated reinforcement is:

- **DEMO-59** — "Electromagnetism (Electrostatics + Maxwell-class Operators)"
  Evidence (AoR stdout): https://github.com/public-arch/Marithmetics/blob/release-aor-20260125T043902Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z logs/controllers__demo-59-electromagnetism.out.txt

11.1 DOC posture in this authority record

DOC (Deterministic Operator Calculus) is used here as a conservativity discipline: it is not invoked to "win arguments" by appeal to abstract analysis. Instead, it imposes a legal grammar on which finite operators are allowed to appear in Tier-A derivations and which are only allowed as explicitly labeled negative controls.

For this paper, we use the term DAO (Deterministic Admissible Operator) as a local definition:

> *A DAO is a DOC-implementable finite operator that passes the admissibility gates enforced in the AoR logs, including (at minimum) nonnegativity (within tolerance), boundedness preservation in discontinuous tests, and sensitivity to illegal controls.*

This is not a metaphysical claim. It is a certificate class: an operator is "admissible" because the AoR harness proves it passes the gates, and "illegal" because the same harness proves it fails.

In the AoR, admissibility is not a vague preference; it is audited explicitly against two non-admissible families used as falsifiers:

- a sharp truncation / cutoff operator ("sharp"), and
- a signed operator ("signed").

These controls are not strawmen; they are included specifically because they can sometimes look superficially "good" in a single metric while violating core invariants (e.g., generating negative mass density or Gibbs overshoot). The program rejects them precisely because it is not optimizing for one metric; it is enforcing legality.

11.2 Kernel admissibility audit

At the most basic level, the admissibility grammar begins with a kernel audit: the legal family must be nonnegative (within tolerance), and the illegal controls must have negative lobes.

In DEMO-56, the kernel admissibility audit is printed explicitly:

- Fejér kernel: nonnegative (within slack)
  $k_{\min} = 1.4845867057 \times 10^{-7}$
- Sharp truncation kernel: negative lobes (non-admissible control)
  $k_{\min} = -0.1026225707$
- Signed filter kernel: negative lobes (non-admissible control)
  $k_{\min} = -0.1299921965$

Evidence (AoR stdout, DEMO-56): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt

In DEMO-69, the audit is repeated in a different setting ($N = 2048$, $r = 16$), and it additionally reports a high-frequency weight fraction beyond the admissible bandwidth:

- Fejér: min $= 5.176815 \times 10^{-10}$, HF_weight_frac($> r$)$= 0.000000$
- Sharp: min $= -3.510996 \times 10^{-3}$, HF_weight_frac($> r$)$= 0.000000$
- Signed: min $= -9.677734 \times 10^{-1}$, HF_weight_frac($> r$)$= 0.983887$

Evidence (AoR stdout, DEMO-69): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt

These audits directly support the main referee-facing defense: the legal operator is not merely "chosen"; it is constrained by a falsifiable operator legality contract, and the AoR demonstrates that controls violating the contract are caught.

11.3 DOC vs finite differences: concrete task certificates (DEMO-56)

DEMO-56 is the AoR's primary "operator legality becomes physics" bridge. It uses task batteries where illegal operators exhibit known failure modes (ringing, overshoot, negativity, HF injection) and where an admissible operator should preserve key invariants.

11.3.1 Discontinuous advection: overshoot + total variation

A classic place where illegal spectral interventions reveal themselves is linear advection of a step. The AoR prints three time snapshots. The admissible operator shows:

- zero overshoot mass ("keeps solution in $[0, 1]$"), and
- dramatically reduced total variation compared to sharp/signed controls.

Example ($t = 0.37$):

- overshoot_mass: fejer$= 0$
- overshoot_mass: sharp$= 0.0053648904$
- overshoot_mass: signed$= 0.0176566422$

Total variation at the same time:

- TV: fejer$= 1.9866737196$
- TV: sharp$= 5.2993519735$
- TV: signed$= 7.7082171866$

Evidence (AoR stdout, DEMO-56, E1): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt

This matters for foundations because it closes a common "numerology escape hatch": a pipeline that is permitted to use illegal smoothing can be tuned; a pipeline that enforces boundedness and rejects illegal ringing has a much narrower space to cheat.

11.3.2 Convergence sweep: "no tuning" monotonicity

The AoR also prints a convergence sweep under the admissible kernel (Fejér):

- $K = 30$ $L_2 = 0.0806730691$
- $K = 60$ $L_2 = 0.0571049942$
- $K = 120$ $L_2 = 0.039377692$

and then gates that the error is non-increasing across $K/2$, K, $2K$.

Evidence (AoR stdout, DEMO-56, E1-D): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt

This is a direct anti-tuning certificate: the admissible family must improve under increased budget in a deterministic sweep.

11.3.3 Counterfactual teeth: deterministic triple changes degrade

A legality regime without teeth can still hide coincidence. DEMO-56 explicitly ties budgets to invariants derived from the selected triple, and then re-runs with deterministic counterfactual triples. The AoR prints:

- Primary $(137, 107, 103)$ implies $q_3 = 17$ and a corresponding budget.
- Counterfactual triples give $q_3 = 51$, forcing a budget reduction (example shown for E1: $K = 20$) and degrading performance.

Printed counterfactual lines:

- CF $(409, 263, 239)$ $q_3 = 51$ $K = 20$ $L_2 = 0.0981630451$ degrade=True (and similarly for the other counterfactuals)

Evidence (AoR stdout, DEMO-56, E1-T): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt

This is a foundational pattern used throughout the suite: the selector does not merely pick a triple; the triple commits budgets, and the budgets constrain what the pipeline can do. Counterfactual triples are deterministic ablations, not chosen after the fact.

11.3.4 Smooth Poisson: spectral inverse vs 2nd-order FD

For a smooth manufactured Poisson problem, DEMO-56 reports:

- L_2 error: FD_2nd= $7.9380355830 \times 10^{-5}$
- spectral= $2.7752322819 \times 10^{-16}$
- ratio= $3.4961197299 \times 10^{-12}$

and gates that this beats a threshold tied to ε.

Evidence (AoR stdout, DEMO-56, E2): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt

This establishes that "admissible operator" is not a euphemism for "blurry"; in smooth regimes it can be decisively accurate.

11.3.5 Reaction–diffusion + Navier–Stokes: boundedness + HF control

For Fisher–KPP reaction–diffusion, DEMO-56 reports:

- L_2 vs internal ref: FD= 0.0312776258 Fejér= $1.8214154224 \times 10^{-6}$
- Boundedness preserved: min = 0.6107950512 max = 0.6107950512

Evidence (AoR stdout, DEMO-56, E3): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt

For 3D Navier–Stokes (Taylor–Green vortex), it reports:

- Errors vs truth: fejer= 0.5451459885 sharp= 1.1530512571 signed= 1.1530512571
- HF energy fraction: fejer= $1.9915956710 \times 10^{-22}$ sharp= $5.0365775909 \times 10^{-16}$ signed= $1.2301059546 \times 10^{-16}$

Evidence (AoR stdout, DEMO-56, E4): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt

These results are used in Foundations for one reason: they certify that the "admissible operator" is a consistent, non-ad-hoc primitive across discontinuous and smooth tasks, and that illegal interventions produce identifiable pathologies.

11.4 OATB: Operator Admissibility Transfer Bridge (DEMO-69)

DEMO-69 is an explicit bridge from operator legality to (i) transfer claims (continuum limit posture), and (ii) paradox-class sanity checks (Zeno/Grandi/Gibbs, measure partitions, collapse), all inside an auditable finite harness.

Evidence (AoR stdout, DEMO-69): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt

11.4.1 UFET near-constant witness (triangle multipliers)

DEMO-69 prints a deterministic "near-constant" witness for $K(r)$ across multiple r:

- $r = 8$: $K(r) = 0.670782$
- $r = 16$: $K(r) = 0.667820$
- $r = 32$: $K(r) = 0.666973$

It then gates:

- spread $\leq 1\%$: spread= 0.570%
- mean close to 2/3 within 2%: $|K - 2/3| = 0.001858$

This is a foundational bridge because it turns an "operator constant" (2/3) into an audited finite pattern, not an asserted analytic limit.

Evidence (AoR stdout, DEMO-69, Stage 2): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt

11.4.2 Transfer witness: legality beats "fit"

DEMO-69 includes a transfer task where distances vs truth are reported:

- Fejér$= 0.140531$
- sharp$= 0.091217$
- signed$= 0.239279$

A superficial "best fit" approach would prefer the sharp cutoff here. The program does not. It rejects sharp/signed because they violate invariants:

- sharp exhibits Gibbs overshoot: Overshoot sharp$= 0.101205$
- signed is worse: Overshoot signed$= 0.202410$
- sharp goes negative: Min(y) sharp$= -8.100846 \times 10^{-2}$
- signed goes more negative: Min(y) signed$= -1.620169 \times 10^{-1}$
- Fejér preserves nonnegativity: Min(y) Fejér$= 7.361219 \times 10^{-3}$

This is the core philosophical point made mathematically operational: legality is a constraint stronger than matching one metric.

Evidence (AoR stdout, DEMO-69, Stage 4): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt

DEMO-69 also includes deterministic counterfactual teeth at the transfer level:

- $q_{3,cf} = 51 \Rightarrow r_{cf} = 5$ dist_cf$= 0.318089$ degrade=True

Evidence (AoR stdout, DEMO-69, CF1): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt

11.4.3 Paradox pack: Zeno, Grandi, Gibbs, measure partitions, collapse

DEMO-69 contains a "paradox pack" because paradoxes are where illegal operators and illegal limits hide. The harness treats these as finite checks with explicit outputs.

Zeno partial sum:

- sum$= 0.999999999069$ with err$= 9.313 \times 10^{-10}$

Grandi series Cesàro:

- cesaro$= 0.500000$ with err$= 0.000 \times 10^{+00}$

Gibbs overshoot suppression is audited directly:

- Gibbs overshoot: Dirichlet$= 0.091747$ Fejér$= -0.011686$
- Undershoot(min): Dirichlet$= -0.086785$ Fejér$= 0.001998$

Measure partition sanity is audited via "mass conservation" and a designed-fail signed partition:

- Hilbert-hotel shifts preserve total mass mass$= 0.250000$

- signed partitions generate illegal negative mass signed_mass= -0.234848

Finally, the harness asserts that a "sharp collapse" injects vastly more HF than the admissible bridge:

- Gate C2: sharp collapse injects vastly more HF than OATB ratio= $9.28 \times 10^{+26}$

Evidence (AoR stdout, DEMO-69, Stage 5): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt

In this Foundations paper, these blocks serve a strict role: they show that the program does not "hand-wave" paradox resolution. It demonstrates that admissibility (nonnegative legal operators) produces stable finite behavior, while illegal interventions produce exactly the known paradox-adjacent pathologies.

11.4.4 Ω reuse across PDEs: portability certificate

DEMO-69 further certifies that the same Ω operator, when treated as an admissible primitive, improves tracking and suppresses HF error across multiple PDE settings:

- 3D heat: tracking improves, factor= 1.66, HF error suppressed, ratio= 0.809
- 4D heat: tracking improves, factor= 1.29, HF error suppressed, ratio= 0.715
- 4D vector field: incompressibility improved, div_ratio= 2.739×10^{-19}, KE damped, ke_ratio= 2.334×10^{-4}, HF KE damped, ratio= 3.005×10^{-36}

Evidence (AoR stdout, DEMO-69, Stage 6): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt

11.5 Closure statement (Hard)

Claim 11.1 (Operator legality certificates, Hard)**.** *The AoR run certifies that the operator primitives used by the Marithmetics pipeline satisfy the declared admissibility grammar and exhibit designed-fail sensitivity. In particular:*

1. *Admissible kernels (Fejér family) pass nonnegativity audits (within tolerance) in multiple independent harnesses.*
2. *Non-admissible controls (sharp, signed) fail the audits by exhibiting negative lobes and/or HF leakage, and they demonstrably produce ringing, overshoot, and negativity in discontinuous tests.*
3. *The admissible primitives preserve boundedness where required and suppress HF injection in PDE tasks; illegal controls violate these invariants.*
4. *Deterministic counterfactual triples degrade performance under budget changes, providing non-genericity teeth.*
5. *Paradox-class sanity checks (Zeno/Grandi/Gibbs/measure partitions/collapse) behave stably under admissibility and fail under illegal interventions.*

Primary evidence (AoR):

- DEMO-56 stdout: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-56-deterministic-operator-calculus-vs-fd.out.txt
- DEMO-69 stdout: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/controllers__demo-69-oatb-operator-admissibility-transfer-bridge.out.txt
- DEMO-59 stdout (reinforcement): https://github.com/public-arch/Marithmetics/blob/release-aor-20260125T043902Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/controllers__demo-59-electromagnetism.out.txt

11.6 Falsifiability for this section

This section is falsified under reproduction and verify if any of the following occur:

1. DEMO-56 or DEMO-69 no longer reports an admissible kernel as nonnegative (within tolerance), or no longer reports negative lobes for the declared illegal controls.
2. The admissible operator begins to exhibit overshoot/negativity in the discontinuous task audits where boundedness is a certified gate.
3. The illegal controls cease to exhibit the expected pathologies (ringing/overshoot/negativity/HF injection), i.e., designed-fails no longer fail.
4. Deterministic counterfactual teeth no longer degrade (the "not generic" certificate is lost).
5. The paradox pack gates (Zeno/Grandi/Gibbs/measure/collapse) no longer reproduce the printed values and PASS/FAIL structure.
6. The AoR bundle fails integrity verification against its declared ledgers and seal.

39:

- DEMO-39 AoR log (primary evidence): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260125T043902Z/logs/cosmo__demo-39-bb-a2.out.txt

11.7 Gauge lawbook derivation: uniqueness under declared contracts

The capsule begins by constructing the gauge lawbook as a unique object under the declared admissibility/rigidity contracts. The AoR log records:

- Candidates found: 1
- Best modulus triple: $q = (17, 13, 17)$ with residue sets
- U(1): residues $[1, 5]$ mod 17
- SU(2): residues [3] mod 13
- SU(3): residues [1] mod 17
- Unique triple (selected):
$$(w_U, s_2, s_3) = (137,\ 107,\ 103)$$
- Closure statement (AoR): "✓ CLOSED: unique canonical lawbook under declared contracts."

This matters because it locks the "starting point" of the entire architecture to a single finite object rather than a family of tunable seeds.

11.8 Gauge selector: survivor pools, invariants, and Θ

The capsule then makes the selection mechanically visible by listing the survivor pools in a specified span, with declared density thresholds τ.

- Search span: $97 \le w \le 180$
- Density thresholds: $\tau_{U1} = 0.31,\ \tau_{SU2} = 0.30,\ \tau_{SU3} = 0.30$
- Lane outputs (AoR):
  - U(1): survivors $[103, 107, 137]$
  - SU(2): survivors $[107]$
  - SU(3): survivors $[103, 137]$
- Unique triple: $(137, 107, 103)$

From that triple, DEMO-39 records the canonical invariants:

- $q_2 = 30$
- $v_2 = 3$
- $\varphi(q_2) = 8$
- $\Theta = \frac{4}{15}$
- Closure statement (AoR): "✓ CLOSED: canonical $(137, 107, 103)$ and invariants $(q_2 = 30, v_2 = 3, \Theta = 4/15)$."

11.9 Φ-mapping: the structural gauge rationals

Using the Φ-mapping (as defined earlier in this paper), DEMO-39 records the core structural fractions:

$$\alpha = \frac{1}{137}, \qquad \alpha_s = \frac{2}{17}, \qquad \sin^2\theta_W = \frac{7}{30}.$$

Numerically (AoR):

- $\alpha = 1/137 \approx 0.0072992700729927$
- $\alpha_s = 2/17 \approx 0.1176470588235294$
- $\sin^2\theta_W = 7/30 \approx 0.2333333333333333$

AoR closure statement: "✓ CLOSED: $\alpha = 1/137$, $\alpha_s = 2/17$, $\sin^2 = 7/30$."

11.10 Yukawa closure: Palette-B as a unique best tuple

The capsule then records a Yukawa-sector closure in terms of a discrete palette of rational exponents ("Palette-B"), selected by gates and witness scans.

The AoR log records:

- Palette-B (best tuple):
$$\left\{0,\ \tfrac{4}{3},\ \tfrac{7}{4},\ \tfrac{8}{3},\ 4,\ \tfrac{11}{3},\ \tfrac{13}{8},\ \tfrac{21}{8},\ \tfrac{9}{2}\right\}.$$
- Pipeline (AoR summary):
  - D1 total candidates: 486

- D1 survivors: 81
- D1 best = Palette-B tuple above
- $\Delta_{\mathrm{iso}}(\text{best}) = 0.0416666666666667 = 1/24$
- "✓ CLOSED: D1 selects Palette-B as unique best tuple."

A second sweep ("offset grid") is recorded as a robustness witness:

- offset grid size: 2401
- Monte-Carlo draws: 2000 (seed 123)
- canonical rank: 1
- "hits summary: {'zero': 2401, 'one': 0, 'ge2': 0}" (AoR)

The key point for Foundations is not an appeal to taste: the AoR explicitly records that the declared gates isolate Palette-B as a unique best object under the capsule's contracts.

11.11 Cosmology closure: BB-36 Ω-templates, near-flatness, and rank-1 parameters

DEMO-39 records a BB-36 cosmology closure with explicit sector values and a near-flatness check:

- $\Omega_b = 0.044981893088497$
- $\Omega_c = 0.2429783126554844$
- $\Omega_\Lambda = 0.7118999678358255$
- $\Omega_r = 8.42249728384458 \times 10^{-5}$
- $\Omega_{\mathrm{tot}} = 0.9999443985526454$

AoR closure statement: "✓ CLOSED: BB-36 Ω templates + near-flatness ($\varepsilon \le 10^{-3}$)."
The capsule also records H_0 closure:

- $d_{\mathrm{contract}} = 7$
- $H_0 = 70.4493959643776719$ with exponent signature $(-6, 1, 2, 7)$, rank$=1$
- AoR: "✓ CLOSED: BB-36 H_0 is rank-1."

And the primordial trio:

- $A_s = 2.099094113200873 \times 10^{-9}$ (rank-1)
- $n_s = 0.9647460711549101$ (rank-1)
- $\tau = 0.0539794849701662$ (rank-1)
- AoR: "✓ CLOSED: primordial trio is rank-1."

11.12 Neutrino capsule: hierarchy contract pass

DEMO-39 records neutrino closure templates:

- $\Delta m^2_{21} = 7.499740433804768 \times 10^{-5}$
- $\Delta m^2_{31} = 0.002499999351254$
- $\Sigma m_\nu = 0.059994871726098$
- $\Delta m^2_{31}/\Delta m^2_{21} = 33.3344783505486149$
- AoR: "✓ CLOSED: neutrino templates + hierarchy contract pass."

11.13 Amplitude windows, ℓ_1, and an explicit negative control

DEMO-39 records additional amplitude-window checks and one explicitly failing negative control:

- $\eta_B = 6.115445477325657 \times 10^{-10}$ (in_win=True)
- $Y_{\mathrm{He}} = 0.247941613961382$ (in_win=True)
- $\delta_{\mathrm{CMB}} = 1.000475284975453 \times 10^{-5}$ (in_win=True)
- Negative control: "NC(deltaCMB using FCMB_d-1): in_win=False"
- $\ell_1 = 219.9490873240763733$ with exponent signature $(-7, 4, 6, -2)$ and $C = 1/e$
- AoR: "✓ CLOSED: amplitude windows pass; NC breaks deltaC window."

This is important as a pattern: the AoR does not only record successes; it records a designed failure where the system is supposed to fail.

11.14 Cross-base invariance: the same constants survive representation changes

DEMO-39 includes a Rosetta-layer cross-base audit and records:

- rosetta passes:
  - crt_collision_exists_on_bad: true
  - crt_injective_on_good: true
  - designed_fail_digit_injection_trips: true
  - phi_hat_float_decimal_agree: true
  - phi_hat_roundtrip_parse_ok: true
- roundtrip max $|\Delta|$ for $\hat{\varphi}$: 7.77×10^{-16} (b7), 2.22×10^{-16} (b10), 0 (b16)
- A2 constants roundtrip max $|\Delta|$: 4.72×10^{-16} (b7), 5.55×10^{-17} (b10), 2.00×10^{-22} (b16)
- "A2 constants roundtrip (all): PASS"

It then lists explicit multi-base representations for several constants (example entries from AoR):

- α base-7: $0.002334515606303135_{(7)}$; base-16: $0.01de5d6e3f8868a000_{(16)}$
- α_s base-7: $0.055232026114346405_{(7)}$; base-16: $0.1e1e1e1e1e1e1e0000_{(16)}$
- $\sin^2\theta_W$ base-7: $0.143014301430143014_{(7)}$; base-16: $0.3bbbbbbbbbbbc0000_{(16)}$

AoR closure statement: "✓ CLOSED: base representations (b7/b10/b16) roundtrip to the same numeric values (tol$= 10^{-15}$)."

11.15 A2 scoreboard: explicit PASS list

DEMO-39 ends with an explicit scoreboard of capsule claims, all recorded as PASS in the AoR log, including:

- gauge_lawbook_unique
- gauge_selector_unique
- gauge_invariants_match
- phi_mapping_expected_fractions
- yukawa_D1_best_is_palette_B
- cosmology_templates_match_BB36
- cosmology_flatness_eps_1e-3
- H0_BB36_rank1
- primordial_As_rank1, primordial_ns_rank1, primordial_tau_rank1
- neutrinos_closed
- amplitudes_etaB_in_window, amplitudes_deltaC_in_window
- ell1_value_finite
- rosetta_all_pass
- a2_constants_roundtrip_parse_ok
- a2_numeric_invariant_under_repr

This capsule is the Foundation's compact ledger: it is the place a skeptical reader can start to see the system as a single object (selector → invariants → mappings → closures → invariance checks), before following domain-specific papers into deeper derivations and broader applications.

12 The BB-36 Master Flagship (DEMO-36): Bridges, Controls, "Teeth," and an Optional External Overlay

The A2 capsule in §14 is the minimal audit: it lists the closure facts and the base-invariance checks in one place. The BB-36 Master Flagship (DEMO-36) is the maximal demonstration of the same closure family: it is where the reader can see the selector, the bridges, the counterfactual controls ("teeth"), and the optional external overlay executed end-to-end.

Primary AoR evidence for this section:

- DEMO-36 stdout: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/cosmo__demo-36-big-bang-master-flagship.out.txt
- DEMO-36 stderr: https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/cosmo__demo-36-big-bang-master-flagship.err.txt

Vendored artifacts (AoR-pinned):

- CAMB planck-vs-GUM overlay (image): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/vendored_artifacts/cosmo__demo-36-big-bang-master-flagship__camb_planck_vs_gum_overlay.png
- BB-36 master plot (image): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/vendored_artifacts/cosmo__demo-36-big-bang-master-flagship__bb36_master_plot.png
- CAMB overlay plot (image): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/vendored_artifacts/cosmo__demo-36-big-bang-master-flagship__camb_overlay.png
- BB-36 results (JSON): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/vendored_artifacts/cosmo__demo-36-big-bang-master-flagship__bb36_master_results.json
- CAMB overlay note (text): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/vendored_artifacts/cosmo__demo-36-big-bang-master-flagship__camb_overlay_note.txt
- CAMB Planck parameters used (JSON): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/vendored_artifacts/cosmo__demo-36-big-bang-master-flagship__camb_planck_params.json

12.1 What "BB-36" denotes in this Foundations paper

In this paper, BB-36 names a particular closure contract:

1. **Selector (finite)**: a unique gauge-lawbook triple is selected under declared admissibility/rigidity tests.
2. **Structural outputs (finite)**: dimensionless parameters (e.g., H_0 proxy, ℓ_1, amplitude proxies) are produced as deterministic functions of the selector and the declared transfer operators.
3. **Bridges (lawful)**: each "physics-looking" output is backed by at least one operator-level bridge whose legality is testable (admissible vs. illegal kernels).
4. **Teeth (falsification controls)**: deterministic counterfactual seeds are required to fail gates by a margin tied to ε (the closure's internal tolerance scale).

DEMO-36 is therefore not "cosmology by curve-fit." It is a closure capsule whose internal logic includes explicit controls designed to break it.

12.2 Stage 1 — Selection and deterministic counterfactuals

DEMO-36 starts by printing the raw survivor pools and then enforcing coherence:

- Lane survivor pools:
  U(1): $[103, 107, 137]$, SU(2): $[107]$, SU(3): $[103]$
- After the U(1) coherence condition $v_2(w_U - 1) = 3$, the U(1) pool collapses to $[137]$.
- The primary triple is confirmed as:

$$(w_U, s_2, s_3) = (137, 107, 103).$$

It then produces deterministic counterfactual triples (not randomly chosen), and asserts that a minimum number are captured (DEMO-36 records four):

$$(409, 263, 239), \quad (409, 263, 307), \quad (409, 367, 239), \quad (409, 367, 307).$$

These counterfactuals are used later as "teeth": they must systematically fail the budgeted gates.

12.3 Stage 2 — Structural closure outputs and the role of "monomials"

DEMO-36 lists a set of "structural cosmo outputs (DEMO-36 monomials)" including:

- $H_0 \approx 70.449395964378$
- $\Omega_b \approx 0.044981893088$
- $\Omega_c \approx 0.126211331093$
- $\Omega_\Lambda \approx 0.828806775819$
- $\Omega_b h^2 \approx 0.022325041590$
- $\Omega_c h^2 \approx 0.062640165238$
- $A_s \approx 2.099094113201 \times 10^{-9}$
- $n_s \approx 0.964746071155$
- $\tau \approx 0.053979484970$
- $\ell_1 \approx 219.949087324076$
- $\delta_{\mathrm{CMB}} \approx 1.000475284975 \times 10^{-5}$

It then checks each against broad plausibility gates (S-gates) and records PASS.

Important: in this Foundations paper, the DEMO-36 "monomial" lines are treated as internal structural outputs used to drive the bridge tests and the control logic. The canonical, publish-grade cosmology template partition for BB-36 is cited from the A2 archive capsule (§14, DEMO-39), which records $(\Omega_b, \Omega_c, \Omega_\Lambda, \Omega_r, \Omega_{\mathrm{tot}})$ as the closure template with a near-flatness check.

12.4 Stage 3 — The tilt bridge: admissible operator vs illegal controls

DEMO-36 then performs a tilt bridge test over a specified band and prints:

- A structural tilt target: $n_s - 1$
- An "admissible" kernel tilt estimate
- Two explicit illegal controls ("sharp" and "signed"), each of which exhibits a failure mode the admissible kernel is designed to avoid (negative lobes, injected high-frequency energy, or curvature distortions).

The key Foundations point is methodological: the "tilt" is not asserted because it is close to a desired number; it is asserted because the operator legality condition is checkable and enforced (e.g., Fejér-type nonnegativity), and the illegal controls do what illegal controls should do: they fail.

12.5 Stage 4 — The δ_{CMB} amplitude bridge: proxy + controls + teeth

DEMO-36 repeats the same structure for an amplitude proxy:

1. It prints $\delta_{\mathrm{CMB,struct}}$.
2. It evaluates the proxy under:
   - a truth tier budget K_{truth},
   - an admissible kernel at the demo budget K,
   - illegal controls (sharp / signed).
3. It asserts that the admissible proxy is within ε while illegal controls worsen error and/or inject high-frequency energy beyond a declared floor.

It then deploys counterfactual teeth at the same budget K: each counterfactual triple must miss by score margin, and DEMO-36 records the required strong pass rate.

12.6 Stage 5 — Optional CAMB TT overlay (informational, not Tier-A)

DEMO-36 can optionally run an external CAMB stage and compares a first-peak location against ℓ_1 derived internally:

- $\ell_{1,\mathrm{struct}} \approx 219.949$
- $\ell_{1,\mathrm{CAMB}} \approx 232.000$
- $\Delta\ell_1 \approx 12.051$

In this Foundations paper the CAMB stage is treated as **evidence-only and informational**. It is not part of the Tier-A substrate closure, because CAMB is an external solver with its own modeling assumptions. The reason it belongs in AoR at all is audit discipline: if an external overlay is shown, it must be pinned, reproducible, and clearly labeled as an overlay.

12.7 Stage 6 — Artifact emission and determinism seal

DEMO-36 records:

- emission of a JSON results file (vendored into AoR),
- optional plotting artifacts (vendored into AoR),
- and a determinism hash.

This is the functional end of the BB-36 Master Flagship: the demo is both a scientific demonstration and a custody-grade record.

13 The AoR Suite Map: Where Each Domain Is Demonstrated (and Where It Is Not)

Foundations is not meant to replace domain papers; it is meant to guarantee that domain papers sit on a single, falsifiable, reproducible substrate. The AoR provides an explicit map from domain to executable evidence.

The canonical indices for navigation are:

- demo index (CSV): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/demo_index.csv
- demo index (JSON): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/demo_index.json
- master constants table (CSV): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/constants_master.csv
- master constants table (JSON): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/tables/constants_master.json

What follows is a human-readable map of the AoR demo surface used by this Foundations paper.

13.1 Substrate and base invariance

- Universe-from-zero substrate demo (DEMO-40): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/substrate__demo-40-universe-from-zero.out.txt https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/substrate__demo-40-universe-from-zero.err.txt
- Base-gauge invariance integer selector (DEMO-64): https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/substrate__demo-64-base-gauge-invariance-integer-selector.out.txt https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/substrate__demo-64-base-gauge-invariance-integer-selector.err.txt

These are the demos that justify saying "cross-base behavior is audited," rather than merely claimed.

13.2 Foundations demos: lawbooks, action, and the prediction ledger

- Lawbook emergence (DEMO-53):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/foundations__demo-53-lawbook-emergence.out.txt
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/foundations__demo-53-lawbook-emergence.err.txt

- One-Action master flagship (DEMO-71):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/foundations__demo-71-one-action-master-flagship.out.txt
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/foundations__demo-71-one-action-master-flagship.err.txt

- Prediction-ledger master flagship (DEMO-75):
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/foundations__demo-75-prediction-ledger-master-flagship.out.txt
  https://github.com/public-arch/Marithmetics/blob/aor-20260209T040755Z/gum/authority_archive/AOR_20260209T040755Z_0fc79a0/GUM_BUNDLE_v30_20260209T040755Z/logs/foundations__demo-75-prediction-ledger-master-flagship.err.txt

These are the demos that turn "methodology" into "constitutional behavior": emergence, action, and forward-commitment to falsifiable predictions.

13.3 Cosmology demos: flagship vs archive capsule

- BB-36 Master Flagship (DEMO-36): see §15.
- BB-36 A2 archive capsule (DEMO-39): see §14.