Nested Observability and Re-Entry: Experimental Validation of the κ_3 Operator in the UNNS Substrate

Abstract

We report the experimental validation of the κ_3 operator within the Unbounded Nested Number Sequences (UNNS) substrate. Building on prior results establishing observability-gated selection (κ_2), this work demonstrates that observability itself exhibits persistence, collapse, and measurable re-entry under controlled gate configurations. Using calibrated observability thresholds and validated persistence metrics, we show that observable distinctions can re-emerge after collapse without modifying underlying system dynamics. This establishes κ_3 as a genuine operator acting on observability contexts rather than on states, completing the next forced layer in the UNNS operator stack.

1 Introduction

Previous UNNS chambers established that selection beyond κ_1 is contingent on observability. In particular, Chamber κ_2 demonstrated that selection is strictly gated: when parity variance is projected away, κ_2 becomes dormant, and when observability is restored, selection resumes immediately.

This result leaves an unresolved question:

> What governs the persistence, suppression, or re-entry of observability itself across layers?

The present work addresses this question experimentally. We introduce and validate the κ_3 operator, which does not select states or structures, but instead selects among observability gate configurations. Crucially, κ_3 operates without invoking any dynamical evolution law and relies solely on measured observability statistics.

2 Observability as a Measurable Stack Object

For each run, we define an observability signal $\Sigma_2^0(t)$, sampled at discrete measurement times. An observability gate Ω_2 is defined by a threshold ε_2 such that

$$\Omega_2(t) = \begin{cases} 1 & \text{if } \Sigma_2^0(t) > \varepsilon_2 \\ 0 & \text{otherwise.} \end{cases}$$

A run produces an ordered record of gate activations across layers, forming an *observability stack*. The κ_3 operator acts exclusively on these gate configurations and their empirical behavior.

No assumptions are made about the origin, stability, or dynamics of $\Sigma_2^0(t)$.

3 Persistence and Re-Entry Metrics

We define three experimentally measurable quantities.

3.1 Persistence

For a fixed gate configuration g, persistence is defined as

$$P(g) = \frac{1}{T} \sum_{k=1}^{T} \mathbf{1} \left[\Sigma_2^0(t_k) > \varepsilon_2 \right],$$

where T is the number of measurements.

3.2 Re-Entry

Re-entry counts the number of upward crossings of the observability threshold:

$$R(g) = \left| \{ k > 1 : \Omega_2(t_{k-1}) = 0 \wedge \Omega_2(t_k) = 1 \} \right|.$$

3.3 Gate Lock

A gate is classified as locked if

$$P(g) \geq p_{\text{lock}},$$

with $p_{\text{lock}} = 0.8$ fixed across all experiments.

These definitions are unchanged throughout this work.

4 Calibration of Observability Gates

Early κ_3 sweeps revealed a critical methodological issue: inappropriate Ω_2 ranges can trivially saturate persistence metrics, masking all structure.

We therefore introduce a mandatory *observability calibration pre-pass*, which empirically measures

$$\Sigma_{\min}, \quad \Sigma_{\max}, \quad \Sigma_{\text{mean}}, \quad \Sigma_{\text{std}}$$

over an initial run.

Gate sweeps are then constrained to the empirically supported range

$$\varepsilon_2 \in [\Sigma_{\min}, \Sigma_{\max}],$$

optionally padded by a fixed multiple of Σ_{std}.

This calibration step is purely instrumental and makes no interpretive claims.

5 Experimental Results

Using a 128×128 lattice, $\lambda = 0.108$, and a fine measurement stride of 2, we performed a 9×9 sweep over (Ω_1, Ω_2) configurations using a windowed contrast observable.

The results satisfy all κ_3 validation criteria:

- Observability calibration present and coherent.

- Full gate coverage (81 configurations).
- Strong persistence diversity (CV(P) > 1).
- Substantial re-entry detected ($R > 0$ for multiple gates).
- Clear lock vs. non-lock contrast.

Re-entry events were counted in the hundreds for intermediate Ω_2 bands, demonstrating that observability can re-emerge after collapse under alternative gates.

6 Key Findings

The experiments establish several nontrivial facts:

1. Observability is not monotone; collapse does not imply permanent loss.
2. Re-entry depends on gate configuration and measurement resolution, not on altered dynamics.
3. Persistence and stability are distinct properties.
4. Gate calibration is foundational; mis-calibration can erase entire operator layers.

These findings cannot be reduced to κ_2 behavior and require a distinct operator level.

7 Interpretation within the UNNS Stack

The validated operator hierarchy now reads:

- κ_0: existence
- κ_1: projection and collapse
- κ_2: selection under observability
- κ_3: persistence and re-entry of observability

κ_3 is not a stronger selector but a selector over the conditions under which selection can recur.

8 Implications and Outlook

By elevating observability to a structured, intermittent resource, κ_3 reframes collapse as context-relative rather than terminal. This resolves tensions between apparent irreversibility and later re-emergence of structure without invoking hidden dynamics.

The success of κ_3 forces the next question: whether patterns of re-entry themselves persist across layers. That question lies beyond the scope of this paper and motivates the next chamber.

Data Availability

All data supporting the findings of this study were generated using the UNNS κ_3 experimental framework and are available as structured JSON exports produced directly by the chamber implementation.

Each dataset includes:

- the full observability calibration statistics,
- the complete (Ω_1, Ω_2) gate grid,
- persistence scores and re-entry counts for all gate configurations,
- gate-lock classifications,
- and the corresponding CK3 validation results.

The datasets are fully self-describing and sufficient to reproduce all reported figures and validation outcomes without additional preprocessing. Access to the data is provided alongside the chamber implementation and supporting materials within the UNNS project repository.

Data Format

All datasets are provided in a structured JSON format conforming to the `unns.kappa3.v0.1.1` schema.

Each JSON file includes the following top-level fields:

- `config`: experimental parameters (grid size, λ, stride, lock threshold),
- `omega_calibration`: observability pre-pass statistics ($\Sigma_{\min}$, $\Sigma_{\max}$, mean, standard deviation, and sample count),
- `omega_grid`: the sampled (Ω_1, Ω_2) gate configuration space,
- `results`: persistence scores, re-entry counts, and gate-lock classifications for all gate pairs,
- `validation`: CK3.0–CK3.4 pass/fail results and overall verdict.

The schema definition explicitly specifies required fields, array length consistency, and validation semantics, and is sufficient to reconstruct all reported figures and validation outcomes programmatically. The schema version identifier is embedded in each data file to ensure forward compatibility.

Conclusion

We have experimentally validated the κ_3 operator as a genuine and necessary component of the UNNS substrate. Observability is shown to persist, collapse, and re-enter depending on gate configuration alone. This establishes nested observability as a fundamental organizing principle and completes the next forced extension of the UNNS operator stack.

Appendix A: Methods and Experimental Protocol

This appendix specifies the experimental protocol used to validate the κ_3 operator. All procedures are measurement-based and make no assumptions about system dynamics or evolution laws.

A.1 Overview of the κ_3 Measurement Pipeline

Each κ_3 run consists of four strictly ordered stages:

1. Observability calibration pre-pass
2. Gate configuration sweep
3. Persistence and re-entry evaluation
4. Validation against CK3 criteria

No stage modifies earlier measurements.

A.2 Observability Calibration Pre-Pass

The calibration pre-pass determines the empirical support of the observability signal $\Sigma_2^0(t)$ prior to any gate sweep.

Inputs

- Grid size (e.g. 64×64, 128×128)
- Control parameter λ
- Calibration depth N_{cal}
- Measurement stride s
- Observable definition (e.g. windowed contrast)

Outputs

- $\Sigma_{\text{min}}, \Sigma_{\text{max}}, \Sigma_{\text{mean}}, \Sigma_{\text{std}}$
- Recommended Ω_2 sweep interval

Pseudo-code

```
initialize system with seed
for t = 0 to N_cal:
    advance system one step
    if t mod stride == 0:
        measure Sigma2_0(t)
        store value

Sigma_min  = min(Sigma_series)
Sigma_max  = max(Sigma_series)
Sigma_mean = mean(Sigma_series)
Sigma_std  = std(Sigma_series)

recommended_Omega2_min = Sigma_min - 0.5 * Sigma_std
recommended_Omega2_max = Sigma_max + 0.5 * Sigma_std
```

This step is purely instrumental and does not activate any gate.

A.3 Gate Configuration Sweep

After calibration, a finite grid of gate configurations (Ω_1, Ω_2) is evaluated.

Inputs

- Ω_1 values (uniform grid)
- Ω_2 values (restricted to calibrated range)
- Sweep depth N_{sweep}
- Measurement stride s

Pseudo-code

```
for each Omega1 in Omega1_grid:
    for each Omega2 in Omega2_grid:
        reset system to identical initial state
        persistence_count = 0
        reentry_count = 0
        prev_active = false

        for t = 0 to N_sweep:
            advance system one step
            if t mod stride == 0:
                measure Sigma2_0(t)
                active = (Sigma2_0(t) > Omega2)

                if active:
                    persistence_count += 1
                if (not prev_active) and active:
                    reentry_count += 1

                prev_active = active

        P = persistence_count / total_measurements
        R = reentry_count

        record (Omega1, Omega2, P, R)
```

The system state is reset for each gate pair to ensure independence.

A.4 Persistence, Re-Entry, and Lock Classification

For each gate configuration g:

- Persistence:
$$P(g) = \frac{\text{active samples}}{\text{total samples}}$$

- Re-entry count:

$$R(g) = \text{number of upward threshold crossings}$$

- Gate lock:

$$\text{Lock}(g) \iff P(g) \geq p_{\text{lock}}, \quad p_{\text{lock}} = 0.8$$

No smoothing, interpolation, or temporal fitting is applied.

A.5 Validation Criteria (CK3)

Each run is validated against the following criteria:

- CK3.0: Observability calibration present
- CK3.1: ≥ 64 gate configurations evaluated
- CK3.2: Persistence diversity ($\text{CV}(P) \geq 0.3$)
- CK3.3: Re-entry detected ($\sum R \geq 1$)
- CK3.4: Lock contrast ($P_{\text{max}}/P_{\text{min}} \geq 3$)

Pseudo-code

```
CK3_0 = calibration_block_exists
CK3_1 = total_gate_pairs >= 64
CK3_2 = std(P_values) / mean(P_values) >= 0.3
CK3_3 = sum(R_values) >= 1
CK3_4 = (P_max / P_min) >= 3

if all CK3_i are true:
    verdict = VALIDATED
else:
    verdict = REJECTED
```

A.6 Reproducibility Statement

All results reported in this work are reproducible given:

- the random seed,
- grid size,
- λ,
- observable definition,
- stride,
- and gate grids.

No hidden parameters or adaptive dynamics are used. All measurements are deterministic conditioned on these inputs.

End of Appendix.