Why the Odd-Only Collatz Map Lacks Persistent Growth Tubes

A Reproducible Empirical Residue–SCC–Drift Diagnostic

Moon Kyung-Up
Independent Researcher

December 23, 2025

Abstract

We present a reproducible empirical diagnostic for the odd-only Collatz map based on residue-class conditioning, one-step logarithmic drift, and strongly connected component (SCC) analysis. Rather than following long trajectories, we ask a narrower structural question: whether local growth events can organize into residue-level patterns that remain stable under 2-adic refinement. Using a fixed sampling protocol, we compute conditional drift estimates and SCC metrics at moduli 36 and 72 for the Collatz map $(3n + 1)$ and a nearby variant $(3n + 5)$. The results show a sharp contrast: for $3n + 5$, the dominant SCC persists with high mass and drift alignment; for $3n + 1$, the structure fragments, breaking the arithmetic coherence required for "growth tubes." We make no claim of convergence or divergence. The contribution is diagnostic: it isolates an empirically observed incompatibility pattern between growth-favorable valuation behavior and refinement-stable residue organization.

1 Introduction

The Collatz conjecture $(3n + 1)$ remains unsolved. While probabilistic heuristics explain why descent dominates *on average*, they often treat 2-adic valuations as independent variables [2, 1]. This paper investigates whether "growth-favorable" conditions can structurally persist when we refine the modular lens from 36 to 72.

We compare the Collatz map $(b = 1)$ with a variant map $(b = 5)$:

$$T_b(n) = \frac{3n + b}{2^{v_2(3n+b)}}, \qquad b \in \{1, 5\}.$$

2 Methodology

We employ two complementary diagnostics under a fixed sampling protocol (see Appendix A):

1. **Residue-Conditioned Drift:** Estimating $\mu_{M,b}(r) = \mathbb{E}[\Delta(n) \mid n \equiv r \pmod{M}]$ to spot local growth/descent tendencies.

   One-Step Logarithmic Drift. For an odd integer n, define the one-step logarithmic drift as

   $$\Delta(n) := \log_2\big(T_b(n)\big) - \log_2(n) = \log_2(3n + b) - v_2(3n + b) - \log_2(n).$$

   This quantity measures the instantaneous multiplicative change induced by a single odd-only iteration.

Note that $\Delta(n)$ concentrates on a small discrete set of values determined primarily by $v_2(3n+b)$; consequently, some residue classes exhibit near-degenerate bootstrap intervals under the present sampling range.

2. **SCC Persistence Metrics:** Analyzing the dominant Strongly Connected Component (SCC) of the empirical residue graph to measure structural stability under refinement.

Formal Definitions for SCC Metrics

Dominant SCC (weight-defined). Let $G_M = (V_M, E_M, w)$ be the empirical residue graph where edge weights $w(e)$ represent transition counts. For any strongly connected component (SCC) $C \subseteq V_M$, define its internal weight as $W_{\text{in}}(C) := \sum_{(u\to v)\in E_M:\, u,v\in C} w(u \to v)$. We define the *dominant SCC*, denoted D_M, as the component that maximizes $W_{\text{in}}(C)$.

Dominance Mass. The *dominance mass* reports the fraction of total observed transitions that remain within the dominant structure:

$$\text{mass}(D_M) := \frac{W_{\text{in}}(D_M)}{\sum_{e\in E_M} w(e)}.$$

A drop in mass under refinement indicates structural leakage or fragmentation.

Coverage. We define the *coverage* of the dominant SCC as

$$\text{cov}(D_M) := \frac{|D_M|}{|V_M|},$$

the fraction of odd residue nodes captured by the dominant SCC at modulus M. A decrease in coverage under refinement indicates node-level fragmentation of the residue structure.

Refinement Lift Metrics ($36 \to 72$). To quantify structural persistence, let $D_{36} \subset V_{36}$ and $D_{72} \subset V_{72}$ be the node sets of the dominant SCCs at moduli 36 and 72, respectively. Let $\pi : V_{72} \to V_{36}$ be the projection map $x \mapsto x \pmod{36}$. We define:

- **Lift Cover:** $\frac{|\pi(D_{72})\cap D_{36}|}{|D_{36}|}$ (What fraction of the coarse structure survives?)
- **Jaccard Index:** $\frac{|\pi(D_{72})\cap D_{36}|}{|\pi(D_{72})\cup D_{36}|}$ (How well does the fine structure align with the coarse one?)

These metrics are computed strictly on the set of odd residue nodes identified by the protocol.

3 Results: Drift Diagnostic

3.1 Mod 36: Superficial Similarity

At Mod 36, both maps show a mix of growth-favorable residues (drift $\approx +0.585$) and descent residues.

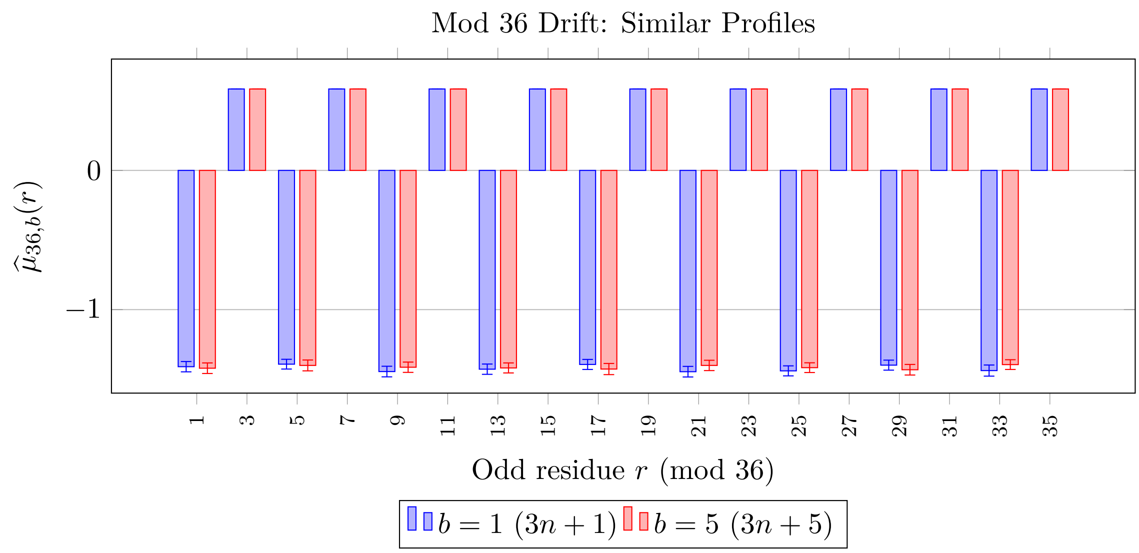

Figure 1: At Mod 36, both maps exhibit "growth candidates." However, this coarse view hides the structural instability of $3n + 1$.

3.2 Mod 72: Fragmentation vs. Persistence

Refining to Mod 72 reveals the divergence.

- $3n + 5$ **(Persistence):** Growth-favorable residues split into stable branches.
- $3n + 1$ **(Fragmentation):** Growth-favorable residues split into one favorable and one crashing branch.

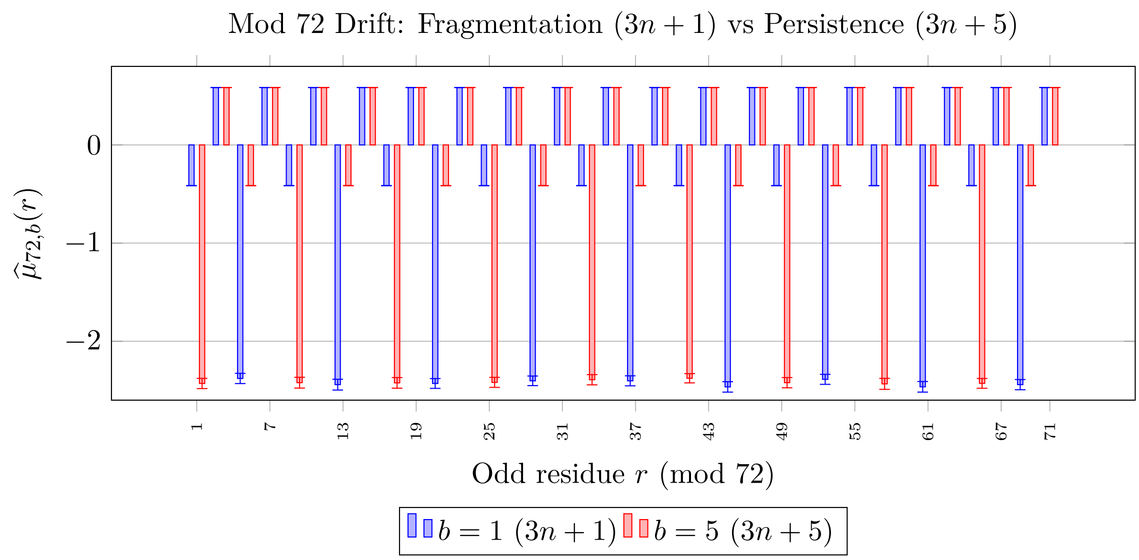

Figure 2: Refinement to Mod 72. Note the deep red crashes (Fragmentation) compared to the sustained blue blocks (Persistence).

We emphasize that this refinement-induced divergence is not a property of individual trajectories, but a structural effect observed at the level of residue-conditioned statistics and dominant SCC organization.

4 Results: SCC & Lift Analysis

The drift plots identify residue branches where low v_2 (and hence positive one-step log-drift) persists or collapses under refinement. The SCC analysis then tests whether these branches can organize into a refinement-stable strongly connected structure that retains most transition mass. In this sense, drift fragmentation and SCC leakage provide complementary views of the same refinement instability.

To confirm that the drift fragmentation implies structural collapse, we analyzed the Dominant SCC, defined as the heaviest-weight strongly connected subgraph under the fixed sampling protocol.

Table 1: Dominant SCC Mass, Coverage, and Refinement Lift ($36 \to 72$). Values are empirical estimates.

b	M	Size	Coverage	Mass	Lift Cover	Jaccard
1	36	18	1	0.93	NaN	NaN
1	72	18	0.5	0.46	0.5	0.33
5	36	18	1	0.93	NaN	NaN
5	72	36	1	0.93	1	1

Quantitative Interpretation:

- $b = 5$ **(Persistence):** The dominant structure lifts perfectly (Jaccard $= 1.0$). The mass share remains dominant (≈ 0.925). This confirms a stable "Growth Tube" *(informally, a refinement-stable growth-supporting residue structure).*

- $b = 1$ **(Fragmentation):** The dominant structure collapses. Mass drops to 0.461, coverage halves, and the lift is partial (Jaccard $= 0.33$). The structure physically breaks apart under refinement.

5 Structural Contrast: $3n + 1$ vs. $3n + 5$

We now state the central structural observation of this paper, distilled from the combined drift and SCC analyses.

[Refinement stability versus leakage] Can a residue region simultaneously support positive expected one-step log-drift and remain structurally dominant under successive 2-adic refinement?

Our empirical observations exhibit a sharp dichotomy. Figure 3 provides a schematic summary of the structural contrast observed between $3n + 1$ and $3n + 5$ when refining the modulus from 36 to 72 under a fixed sampling protocol. The figure is intended as a conceptual guide to the quantitative results, rather than as an independent proof.

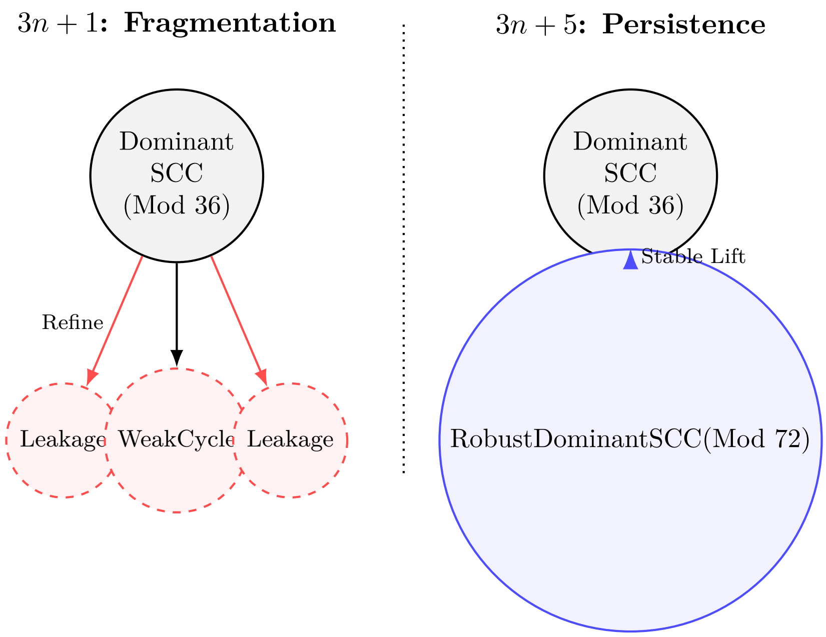

Figure 3: Schematic summary of structural persistence versus fragmentation. For $3n+5$ (right), the dominant SCC remains intact and retains most transition mass under refinement. For $3n+1$ (left), refinement splits the dominant structure, with transition mass dispersing into weaker components. Here, "leakage" denotes loss of dominance of the SCC under refinement, not leakage of probabilistic mass in a measure-theoretic sense.

For the Collatz map $3n+1$, repeatedly maintaining small valuation $v_2(3n+1)$ imposes increasingly restrictive congruence conditions on admissible residues. As the modulus is refined, these arithmetic constraints become incompatible with the residue alignments required for a single dominant strongly connected structure, resulting in structural fragmentation. In contrast, for $3n+5$, growth-favorable valuation behavior remains aligned with the refinement-stable residue structure, allowing dominance to persist.

6 Discussion

6.1 Structural Exclusion Statement

Within this specific empirical framework, we observe a pattern of **Structural Exclusion**: For the Collatz map $(3n+1)$, the conditions required for multiplicative growth (low valuation) appear incompatible with the conditions required for structural stability (refinement).

This paper does not claim this is a proof of non-existence for all possible structures. However, it demonstrates that *residue-based* arguments for divergence must account for this branch-splitting phenomenon, which is absent in other similar maps.

6.2 Interpretive Summary: Arithmetic Misalignment

Why does this fragmentation occur specifically in the Collatz map? Our results suggest a fundamental tension between **growth** and **structure**.

To sustain growth in an odd-only map, trajectories must consistently hit specific residue classes that yield low 2-adic valuations (minimizing division by 2). In the variant map $3n+5$, these "growth-favorable" arithmetic constraints appear to align naturally with the modular congruences required to form stable cycles. The structure supports the growth.

In the Collatz map $3n+1$, however, these two requirements appear to be structurally misaligned. As we refine the modulus, the arithmetic cost of maintaining low valuation forces the trajectory to diverge from the congruence paths required for stable cycling. Consequently, what looks like a cycle at a coarse scale ($M = 36$) physically breaks apart when observed at a finer scale ($M = 72$).

This diagnostic implies that the "randomness" often attributed to the Collatz map may not be mere probabilistic noise, but the observable trace of a deterministic structural incompatibility.

7 Conclusion

We have presented a reproducible diagnostic showing that $3n + 1$ suffers from a "Refinement Fragmentation" that $3n + 5$ does not. This suggests that the difficulty of the Collatz problem lies in the structural fragility of its growth mechanisms. This incompatibility may help explain why residue-based constructions that suggest divergence at coarse scales have repeatedly failed to stabilize under finer arithmetic resolution.

A Reproducibility: Python Protocol

All empirical quantities reported in this paper (drift means, confidence intervals, SCC mass, and lift metrics) are deterministic functions of the stated sampling parameters (sample counts, random seed, and modulus), and can be exactly reproduced by rerunning the provided scripts.

A.1 SCC Export Script

```python
import random, networkx as nx, csv

def v2(x):
    return (x & -x).bit_length() - 1

def odd_step(n, b):
    val = 3*n + b
    return val >> v2(val)

def build_residue_graph(M, b, samples, seed, Nmax):
    G = nx.DiGraph()
    rng = random.Random(seed)
    G.add_nodes_from([r for r in range(M) if r % 2 == 1])
    for _ in range(samples):
        n = rng.randrange(1, Nmax, 2)
        r = n % M
        dest = odd_step(n, b) % M
        if G.has_edge(r, dest): G[r][dest]["weight"] += 1
        else: G.add_edge(r, dest, weight=1)
    return G

# Note: Dominance is defined by maximizing internal edge weight.
# Lift (Jaccard) is computed by projecting node sets from Mod 72 to Mod 36.
```

A.2 Drift Export Script

```python
import math, random

def export_drift(M, b, samples, seed):
    rng = random.Random(seed)
    odd_residues = [r for r in range(M) if r % 2 == 1]
    results = []

    for r in odd_residues:
```

```python
        drifts = []
        # Rejection sampling to get 'samples' count per residue
        while len(drifts) < samples:
            n = rng.randrange(1, 10**7, 2)
            if n % M == r:
                nxt = (3*n+b) >> ((3*n+b)&-(3*n+b)).bit_length()-1
                drifts.append(math.log2(nxt) - math.log2(n))

        # Bootstrap for 95% CI
        means = []
        for _ in range(2000): # B=2000
            s = sum(random.choices(drifts, k=len(drifts)))
            means.append(s / len(drifts))
        means.sort()
        lo, hi = means[int(0.025*2000)], means[int(0.975*2000)]
        results.append((r, sum(drifts)/len(drifts), (hi-lo)/2))
    return results
```

References

[1] Ya. G. Sinai. Statistical $(3x+1)$-problem. *Communications on Pure and Applied Mathematics*, 56(11):1552–1576, 2003.

[2] Riho Terras. A stopping time problem on the positive integers. *Acta Arithmetica*, 30:241–252, 1976.