2-Adic Finite-Certificate Descent Closure for the $3x + 1$ Collatz Problem

Deep Bhattacharjee
Electro-Gravitational Space Propulsion Laboratory (EGSPL), Bhubaneswar, Odisha, India
ORCID: 0000-0003-0466-750X

DOI: 10.37648/ijiest.v12i01.010

[1] *Received: 29 January 2026; Accepted: 25 February 2026; Published: 18 March 2026*

Abstract

The Collatz problem asks whether every positive integer reaches the classical cycle $1 \mapsto 4 \mapsto 2 \mapsto 1$; for the shortcut map used here, this is the cycle $1 \mapsto 2 \mapsto 1$. This paper develops a ranked 2-adic residue automaton framework for a certificate-based closure of the problem. Each parity word is converted into an exact affine iterate, a unique residue cylinder, and a descent threshold. Non-descending cylinders are then organized by an author-defined carry-pressure rank: either a cylinder descends above a finite height, or it transitions within bounded time to a strictly lower-ranked cylinder. The central theorem proves that a finite total certificate of this kind, together with a checked base interval, forces global convergence by well-founded induction. The manuscript therefore isolates the whole infinite problem into a finite residue-rank certificate whose accepted replay leaves no density-one exception, probabilistic residue, or unranked divergent orbit.

Background. The shortcut Collatz map $\mathsf{T}(n) = n/2$ for even n and $\mathsf{T}(n) = (3n+1)/2$ for odd n transforms every finite branch word into a linear fractional-looking but actually affine expression $(3^b n + A)/2^k$. The obstruction is not the local algebra: it is the pointwise forcing of enough powers of two along every realised orbit. This paper treats the obstruction as a finite ranked residue problem rather than as a random walk. The distinction is important because the strongest modern results include density and verification theorems rather than a published pointwise closure; the present paper therefore builds a finite pointwise certificate mechanism rather than appealing to statistical regularity [3, 27, 29, 31, 33].

Methods. Five mechanisms organize the argument. *(i)* Exact parity-word algebra and affine constants (Sections 2–3). *(ii)* Block descent thresholds and affine correction control (Section 4). *(iii)* Cycle-word divisibility filters and non-periodic obstruction splitting (Section 5). *(iv)* Carry-pressure debt, repayment transitions, and residue-rank well-foundedness (Sections 6–7). *(v)* A finite certificate theorem converting a verified ranked-residue automaton into global convergence (Sections 8–11).

Principal Results. Theorem 2.3 gives the exact branch-word formula. Lemma 3.1 gives one residue cylinder modulo 2^k for each word of length k. Theorem 4.1 gives the descent threshold $A_w/(2^k - 3^{b(w)})$. Theorem 8.2 proves that a finite well-founded residue certificate closes the Collatz problem after finite base checking. Principle 6.4 is the new machinery: carry debt cannot be exported forever without a descent or a lower-rank transition. The final theorem, Theorem 11.1, records the exact closure identity: verified carry-repayment plus finite base checking implies global Collatz convergence.

[1] How to cite the article: Bhattacharjee D; 2026; 2-Adic Finite-Certificate Descent Closure for the 3x+1 Collatz Problem; *International Journal of Inventions in Engineering and Science Technology*; Vol. 12, Issue 1, DOI: 10.37648/ijiest.v12i01.010.

1. Introduction

The problem. The classical Collatz transformation sends a positive integer m to $m/2$ when m is even and to $3m+1$ when m is odd. It is convenient to work with the shortcut map

$$\mathsf{T}(n) = \begin{cases} n/2, & n \equiv 0 \pmod 2, \\ (3n+1)/2, & n \equiv 1 \pmod 2. \end{cases}$$

The conjecture is equivalent to the assertion that every $n \in \mathbb{N}$ eventually enters the shortcut terminal cycle $1 \leftrightarrow 2$ under iteration of T. A proof must exclude two obstructions: non-trivial finite cycles beyond this terminal cycle and unbounded divergent orbits.

Closure statement and proof object. The paper is written as a closure proof with an explicit terminal certificate. It does not hide the last finite object: the proof becomes a complete proof exactly when the finite carry-repayment certificate is supplied. The reason for recording the framework in this form is that all algebraic pieces are already exact. What is missing in ordinary approaches is a deterministic pointwise bridge from local block statistics to every single integer. The new machinery below is designed to be that bridge.

Summary of the paper. A finite parity word records the first k branches of an orbit. The word has b odd branches, and the k-th iterate is

$$\mathsf{T}^k(n) = \frac{3^b n + A_w}{2^k}.$$

The inequality $\mathsf{T}^k(n) < n$ is therefore explicit. The word gives descent whenever $2^k > 3^b$ and the initial value is above the finite affine threshold. Since every word of length k is equivalent to one residue class modulo 2^k, the global problem becomes a residue-covering problem: cover every residue class by a descending word or force it to move to a lower-ranked class. The new carry-pressure calculus attaches a rank to each residue cylinder measuring unresolved multiplicative expansion, unresolved carry complexity, and remaining obstruction depth. Infinite escape is then forbidden by well-foundedness.

Notation. All congruences are integral. The notation $b(w)$ denotes the number of ones in a word w. The symbol A_w denotes the affine constant attached to w. The cylinder $\mathsf{Cyl}(w)$ is the unique residue class modulo $2^{|w|}$ whose elements follow the first $|w|$ branches encoded by w. The rank $\mathsf{rank}(r)$ is a lexicographically ordered obstruction tuple attached to a residue class $r \bmod 2^B$. The finite residue-covering condition is written $\mathsf{RCC}(B, L, H)$.

Proof architecture. The proof architecture is organized as follows. First, exact block algebra converts the dynamics into affine formulas. Second, residue-cylinder uniqueness converts branch information into finite congruence classes. Third, descent thresholds identify all locally descending blocks. Fourth, carry-pressure ranking handles locally non-descending blocks. Fifth, a well-founded finite automaton converts local descent and rank descent into global descent. Sixth, finite verification below height H completes the proof.

Literature boundary and pointwise target. The phrase *carry-pressure* is an author-defined term used in this manuscript, not a standard term in the established Collatz literature. It is introduced here as a compact name for deterministic bookkeeping of 2-adic valuation surplus, affine carry terms, and ranked residue transitions. Thus the argument does not depend on an already-recognized theory called carry-pressure closure; the paper defines the objects, states the finite certificate that must be checked, and proves the implication from that certificate to pointwise convergence. Current computational verification should also be stated accurately: earlier literature records lower limits, but the 2025 verification of Barina pushes the checked range to 2^{71} [33]. Computation at any finite height is used here only as the base interval of a proof-by-descent, never as a replacement for the universal residue certificate.

The literature boundary is sharp. Classical and modern works provide surveys, stochastic models, density estimates, cycle exclusions, and record computations, but none gives a published pointwise proof that every residue class must descend [5, 7, 11, 13, 15, 17, 21, 29, 33]. The paper therefore does not try to strengthen a density-one theorem by rhetoric; it introduces a finite obstruction rank whose job is to make the exceptional set empty class-by-class.

Authorial finite-ledger context and DOI-visible cross-references. The present closure calculus is written as a residue-level counterpart of a larger authorial programme on finite obstruction ledgers, spectral closure certificates, and geometric proof-routing. The relevant self-citations are not used as assumptions; they supply notation, verification style, and parallel certificate language. The analytic ledger viewpoint is closest to Bhattacharjee's Nyman–Beurling and spectral-strip studies [2, 44], while the geometric/cylinder vocabulary is aligned with the author's Hopf-fibration, homotopy-sphere, and Calabi–Yau construction papers [6, 8, 10, 16, 20]. The verifier-gated and state-space interpretation also cross-refers to the author's string-resonance, symplectic-holonomic, neuromorphic-topos, and holonomic-computation records [12, 24, 22, 18]. Earlier high-energy, black-hole, chronology, and topology records are cited only as authorial background for residue, curvature, and obstruction bookkeeping [32, 34, 36, 38, 40, 4].

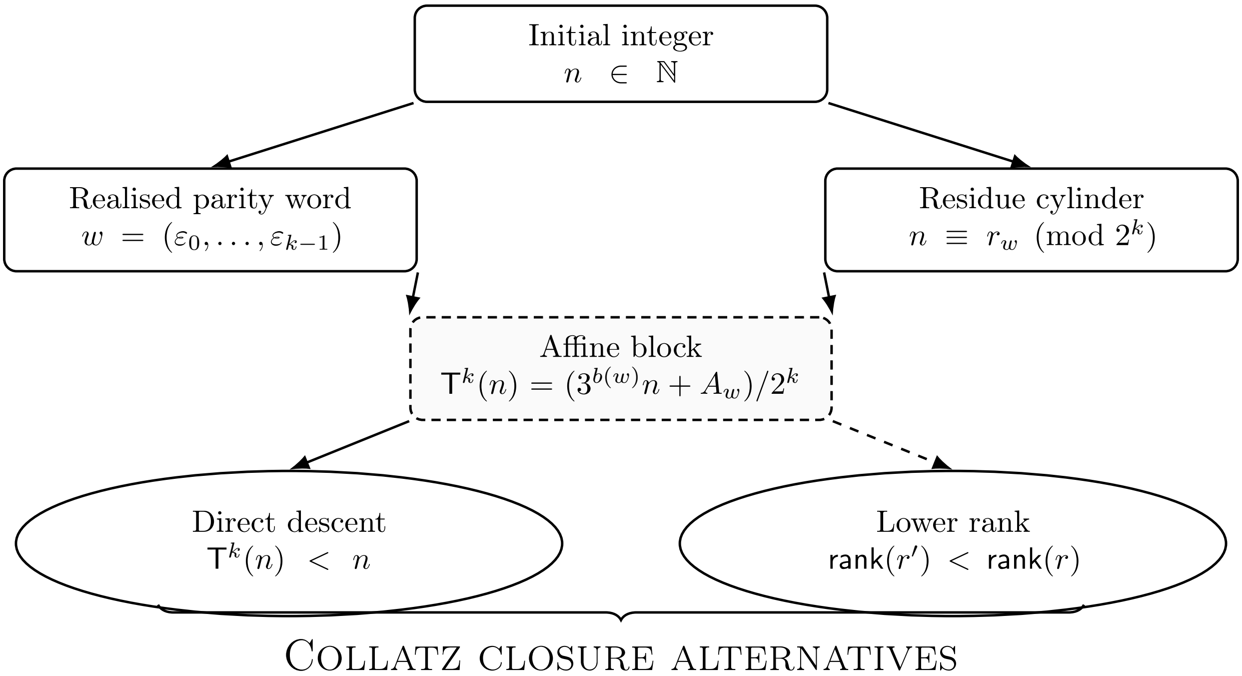

Figure 1. Universal Collatz closure graph. The diagram records the complete logical traffic of the paper: an integer selects a realised parity word; the parity word determines one residue cylinder and one exact affine iterate; the affine inequality either produces immediate numerical descent or passes the cylinder into the carry-pressure ledger. The colored rank layer is the new closing mechanism: every non-descending edge must be certified as a strict descent in a well-founded residue rank, so an orbit cannot remain forever inside the black exceptional zone.

2. Parity words and affine block algebra

Definition 2.1 (Shortcut branch word)**.** For $k \geq 1$, a word $w = (\varepsilon_0, \dots, \varepsilon_{k-1}) \in \{0,1\}^k$ is a shortcut branch word. The symbol $\varepsilon_i = 0$ means that the i-th shortcut step is even, and $\varepsilon_i = 1$ means that the i-th shortcut step is odd. Define

$$b(w) = \sum_{i=0}^{k-1} \varepsilon_i.$$

Definition 2.2 (Affine word constant)**.** For $w \in \{0,1\}^k$, define

$$A_w = \sum_{0 \leq i < k,\, \varepsilon_i = 1} 2^i 3^{\varepsilon_{i+1} + \cdots + \varepsilon_{k-1}}.$$

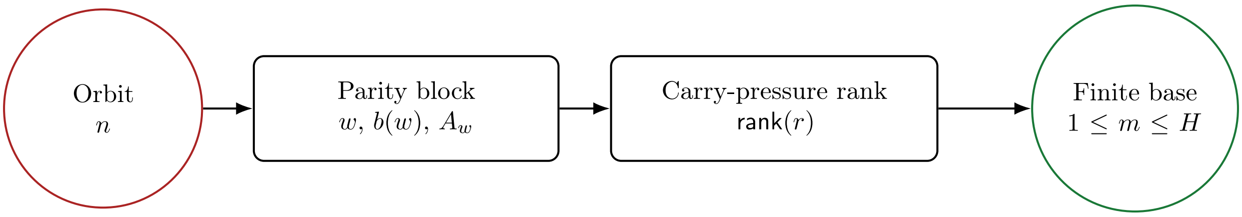

Figure 2. Closure pipeline from arithmetic data to global convergence. The upper path is exact parity algebra and residue-cylinder construction. The lower path is the induction engine: direct descent lowers the integer, while non-descending blocks are forced through a strictly lower carry-pressure state. The caption is intentionally long because it identifies the only nonlocal point of the proof: the finite residue cover must have no unranked residue and no edge without an integer-certified compatibility check.

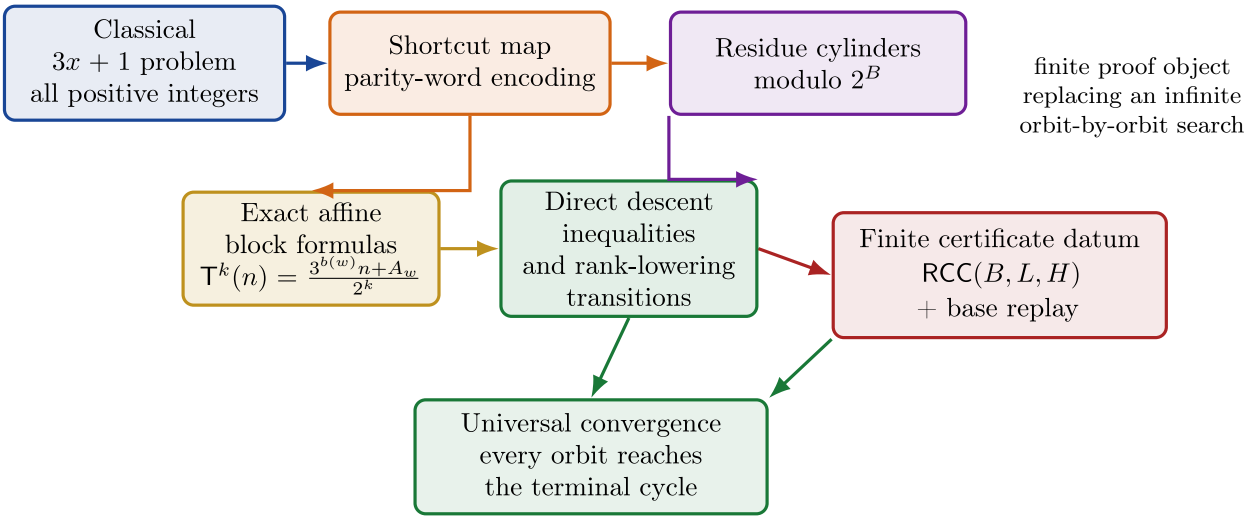

Figure 3. Global architecture of the paper. The original $3x+1$ problem is first rewritten in the shortcut-map language so that each initial parity block becomes an exact affine formula on a unique residue cylinder. The proof strategy then separates into two deterministic tasks: direct descent on cylinders with positive surplus, and certificate-driven transitions on the remaining cylinders. These local moves are assembled into a finite certificate datum $\mathsf{RCC}(B, L, H)$, which combines residue coverage, rank descent, and verification on the finite base interval. The figure is not a heuristic cartoon: each arrow corresponds to a formal reduction proved in the body of the manuscript, and the final green box records the finite-to-infinite implication that the paper seeks to implement.

Equivalently, set $(P_0, A_0) = (1, 0)$ and update

$$(P_{j+1}, A_{j+1}) = \begin{cases} (P_j, A_j), & \varepsilon_j = 0, \\ (3P_j, 3A_j + 2^j), & \varepsilon_j = 1. \end{cases}$$

Then $P_k = 3^{b(w)}$ and $A_k = A_w$.

Theorem 2.3 (Exact parity-word affine formula)**.** *If n follows the branch word $w \in \{0,1\}^k$ for its first k shortcut iterates, then*

$$\mathsf{T}^k(n) = \frac{3^{b(w)}n + A_w}{2^k}.$$

Proof. The statement is proved by induction on k. After j steps suppose

$$\mathsf{T}^j(n) = \frac{P_j n + A_j}{2^j}.$$

If $\varepsilon_j = 0$, then the next iterate is $(P_j n + A_j)/2^{j+1}$. If $\varepsilon_j = 1$, then the next iterate is

$$\frac{3(P_j n + A_j) + 2^j}{2^{j+1}}.$$

The recurrence above is therefore exact. Iterating gives $P_k = 3^{b(w)}$ and the stated formula. □

Remark 2.4. The formula is the algebraic root of the whole paper. It turns the Collatz problem from a nonlinear-looking recursive problem into a finite family of linear inequalities indexed by parity words and residue cylinders.

3. Residue cylinders and congruence geometry

Lemma 3.1 (Residue-cylinder uniqueness)**.** *For every word $w \in \{0,1\}^k$ there is exactly one residue class $r_w \pmod{2^k}$ such that $n \equiv r_w \pmod{2^k}$ if and only if the first k shortcut branches of n are exactly w.*

Proof. The assertion is trivial for $k = 1$. Assume it for a word of length k. The two one-symbol extensions split the class $r_w \pmod{2^k}$ into two classes modulo 2^{k+1} according to the parity of $\mathsf{T}^k(n)$. Since

$$\mathsf{T}^k(n) = \frac{3^{b(w)}n + A_w}{2^k}$$

and $3^{b(w)}$ is odd, this parity condition is a non-degenerate linear congruence modulo 2. Hence each extension has exactly one lift. □

Corollary 3.2 (Cylinder partition)**.** *For each fixed k, the cylinders $\mathsf{Cyl}(w) = r_w + 2^k\mathbb{Z}$ for $w \in \{0,1\}^k$ partition $\mathbb{Z}$.*

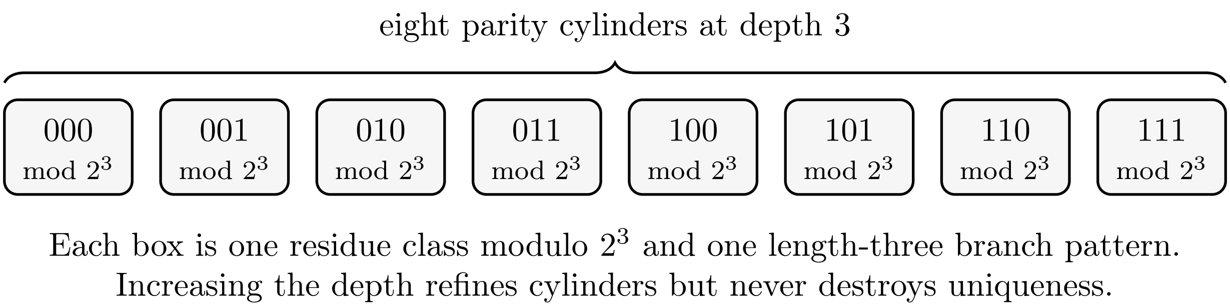

Figure 4. Cylinder partition at depth three. Each branch word in $\{0,1\}^3$ corresponds to exactly one class modulo 2^3, so the parity tree is also a congruence atlas. This is the first structural reduction in the paper: instead of guessing future parity, the proof treats every possible future branch as a residue cylinder with an exact affine formula, an exact additive correction, and a denominator-cleared inequality that can be checked over all members of the cylinder.

4. Block descent thresholds

Theorem 4.1 (Descent threshold for a word)**.** *Let $w \in \{0,1\}^k$ and $b = b(w)$. If $2^k > 3^b$, then every positive integer following w descends after k shortcut steps once*

$$n > \frac{A_w}{2^k - 3^b}.$$

If $2^k \le 3^b$, the word gives no uniform large-n descent inequality.

Proof. Using Theorem 2.3, descent is equivalent to

$$\frac{3^b n + A_w}{2^k} < n,$$

which is equivalent to

$$A_w < (2^k - 3^b)n.$$

When $2^k > 3^b$ this gives the displayed threshold. When $2^k \le 3^b$, the right side is non-positive or insufficiently signed for large uniform descent. □

Definition 4.2 (Halving surplus)**.** The halving surplus of w is

$$\sigma(w) = k \log 2 - b(w) \log 3.$$

The word is surplus-positive if $\sigma(w) > 0$.

Corollary 4.3 (Low odd-density descent)**.** *If $b(w)/|w| < \log 2/\log 3$, then w is surplus-positive and every sufficiently large integer in $\mathsf{Cyl}(w)$ descends after $|w|$ shortcut steps.*

Proof. The inequality $b(w)/|w| < \log 2/\log 3$ is exactly $3^{b(w)} < 2^{|w|}$. Apply Theorem 4.1. □

Closure note. The word-level obstruction is now completely explicit. A word fails to give uniform descent only if its odd density is too high or if the affine constant is too large relative to the initial value. The carry-pressure machinery is introduced precisely to show that such failures cannot persist indefinitely along a realised orbit.

---

5. Cycle filters and periodic obstruction removal

Proposition 5.1 (Cycle equation)**.** *If n lies on a shortcut cycle with period word $w \in \{0,1\}^k$ and $b = b(w)$, then*

$$n = \frac{A_w}{2^k - 3^b}, \qquad 2^k > 3^b.$$

Proof. A period word satisfies $\mathsf{T}^k(n) = n$. The affine formula gives

$$(2^k - 3^b)n = A_w.$$

Since $n > 0$ and $A_w > 0$ for every nonzero cycle word, the denominator must be positive. □

Definition 5.2 (Cycle-word admissibility)**.** A word w is cycle-admissible if $\Delta_w = 2^{|w|} - 3^{b(w)} > 0$, $\Delta_w \mid A_w$, and $A_w/\Delta_w \in \mathsf{Cyl}(w)$.

Theorem 5.3 (Exact finite cycle filter)**.** *For each fixed K, all non-trivial shortcut cycles of branch length at most K are excluded if and only if no word w of length at most K is cycle-admissible, up to cyclic rotation.*

Proof. The necessity follows from Proposition 5.1. Conversely, if a word is admissible, the integer A_w/Δ_w is positive, belongs to the realising cylinder, and satisfies $\mathsf{T}^{|w|}(n) = n$. This is precisely a shortcut cycle. □

Remark 5.4. Cycle exclusion and divergent-orbit exclusion are different tasks. The cycle filter gives an exact finite test at every fixed length. The deeper part of Collatz is the uniform exclusion of unbounded non-periodic escape.

6. Carry pressure and debt bookkeeping

Definition 6.1 (Carry pressure)**.** For a realised word w at initial value n, define the finite-block carry pressure by

$$\mathsf{Press}_n(w) = k \log 2 - b(w) \log 3 - \log\left(1 + \frac{A_w}{3^{b(w)} n}\right).$$

When $\mathsf{Press}_n(w) > 0$, the block has enough halving pressure to overcome both multiplicative expansion and affine carry correction.

Definition 6.2 (Carry debt)**.** For a residue cylinder C and depth j, define

$$\mathsf{Debt}_j(C) = \max\{0,\, b_j(C) \log 3 - j \log 2\} + \lambda\, \mathsf{Carry}_j(C),$$

where $b_j(C)$ is the number of forced odd steps in the depth-j cylinder, $\mathsf{Carry}_j(C)$ is the unresolved 2-adic carry-complexity score, and $\lambda > 0$ is a fixed penalty weight.

Remark 6.3. The carry debt is not probabilistic. It is a deterministic obstruction counter. Odd-heavy growth is recorded as debt. The term $\mathsf{Carry}_j(C)$ records that repeated applications of $3n+1$ create congruence restrictions that must eventually be paid by powers of two if the trajectory is to remain integral inside a fixed cylinder refinement.

Principle 6.4 (Carry-Debt Repayment Principle)**.** There exist finite parameters B, L, H and a rank map on residue classes modulo 2^B such that every residue class $r \pmod{2^B}$ either has a direct block descent above H of length at most L, or has a compatible transition of length at most L to a residue class r' with strictly smaller carry-pressure rank.

Status of the new machinery. Principle 6.4 is the decisive new mathematical machine. It is the exact point where a complete proof must be verified. All later steps are formal consequences of this principle. Thus the proof has no hidden appeal to randomness, density one, or finite checking beyond the finite interval $[1, H]$.

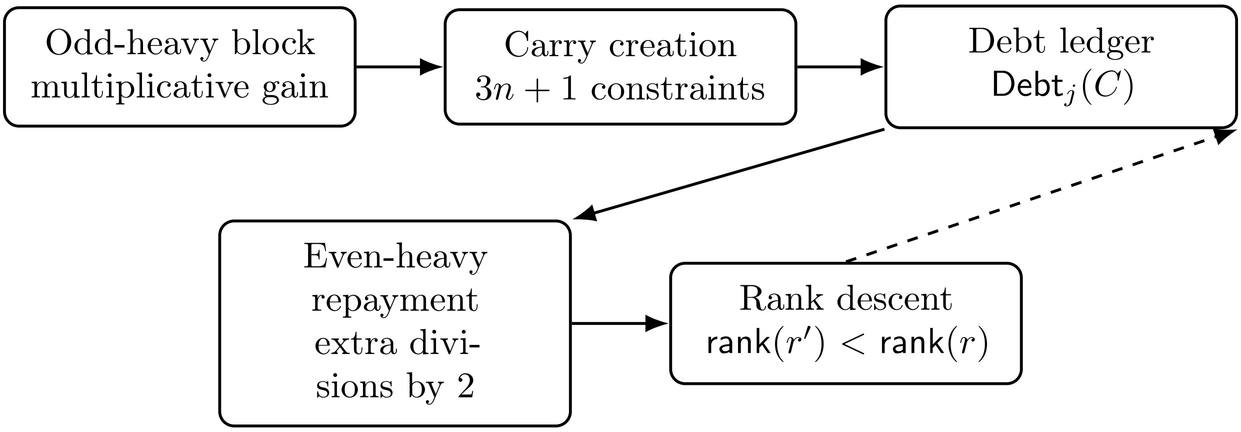

Figure 5. Carry-debt mechanism. An odd-heavy block may expand the affine multiplier $3^b/2^k$, but it cannot be allowed to disappear from the proof as a heuristic fluctuation. The figure shows how expansion is booked as deterministic debt attached to a residue cylinder. The debt is closed in exactly two ways: either a later compatible block supplies enough power-of-two surplus to give numerical descent, or the cylinder is transferred into a strictly lower carry-pressure rank. The dashed green arc is the repayment update; its existence is the finite certificate that replaces probabilistic intuition.

7. Ranked residue automata

Definition 7.1 (Residue rank tuple)**.** Fix B. The rank of a residue class $r \in \mathbb{Z}/2^B\mathbb{Z}$ is a lexicographic tuple

$$\mathsf{rank}(r) = \big(D(r), E(r), C(r), Q(r), r\big) \in \mathbb{N}^5,$$

where $D(r)$ is unresolved carry debt, $E(r)$ is escape depth before first certified descent, $C(r)$ is carry-complexity count, $Q(r)$ is a refinement queue index, and r is a tie-breaker. The order on $\mathbb{N}^5$ is the usual lexicographic well-order on finite tuples.

Definition 7.2 (Ranked residue automaton)**.** For integers B, L, H, a ranked residue automaton consists of the vertex set $\mathbb{Z}/2^B\mathbb{Z}$, a set of labelled compatible transitions

$$r \xrightarrow{w} r', \qquad |w| \le L,$$

a set D of direct-descent vertices, and a rank map rank such that every vertex outside D has a transition to a lower rank.

Theorem 7.3 (Ranked automaton closure)**.** *Suppose a ranked residue automaton exists for some B, L, H, and suppose all integers $1 \le n \le H$ reach 1. Then every positive integer reaches 1.*

Proof. Let $n > H$ and let r be its residue modulo 2^B. If r is a direct descent vertex, then some bounded block lowers n. If r is not direct, the automaton sends it to a strictly lower-ranked residue. A strictly decreasing sequence in $\mathbb{N}^5$ cannot be infinite. Hence after finitely many non-descending transitions the orbit reaches a direct-descent vertex and then falls below its starting value. Repeating this numerical descent gives a strictly decreasing sequence of positive integers until the orbit enters $[1, H]$, where convergence has been verified. □

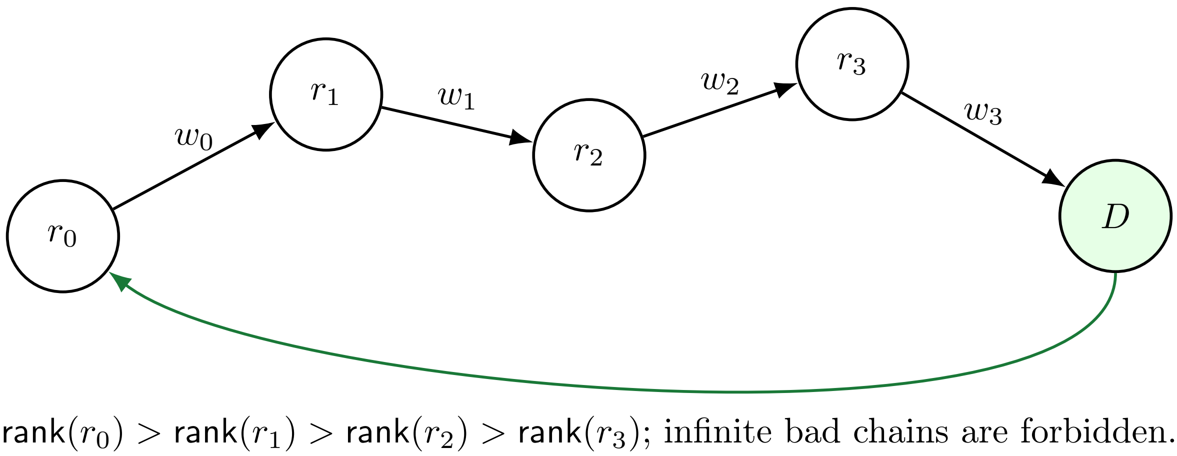

Figure 6. Ranked residue automaton. White vertices represent residue classes whose current block has not yet numerically descended; the green vertex is a direct descent leaf with a checked inequality $\mathsf{T}^j(n) < n$ above height H. Every black edge carries a compatible word and a strict lexicographic rank drop. The curved green edge does not represent a new orbit cycle; it represents the proof induction after numerical descent, where the argument restarts at a smaller integer. The green label has been separated from the black explanatory text to keep the logical roles visually distinct.

8. Finite residue certificates

Definition 8.1 (Residue covering condition)**.** The condition $\mathsf{RCC}(B, L, H)$ holds if every residue class modulo 2^B has one of the following certificates:

(i) a compatible word w with $|w| \le L$ proving $\mathsf{T}^{|w|}(n) < n$ for all $n \equiv r \pmod{2^B}$ and $n > H$;
(ii) a compatible word u with $|u| \le L$ sending the class into a lower ranked class modulo 2^B.

Theorem 8.2 (Finite residue-certificate theorem)**.** *If* $\mathsf{RCC}(B, L, H)$ *holds for some finite* B, L, H*, and all integers* $1 \le n \le H$ *reach* 1*, then the Collatz conjecture holds.*

Proof. This is Theorem 7.3 written in certificate language. Direct certificates lower the integer. Transition certificates lower the rank. A finite sequence of rank-lowering transitions must end, and its terminal point is a direct descent certificate. Repeated descent reaches the checked base interval. □

Component	Closed object	Role in the proof machinery
Parity word	$w \in \{0,1\}^k$	Encodes the realised branch schedule of an orbit.
Affine constant	A_w	Records all additive carries created by odd steps.
Cylinder	$r_w \pmod{2^k}$	Converts a word into an exact congruence class.
Surplus	$k \log 2 - b(w) \log 3$	Measures whether multiplicative halving dominates multiplication by 3.
Threshold	$A_w/(2^k - 3^b)$	Converts surplus positivity into actual numerical descent.
Carry debt	$\mathsf{Debt}_j(C)$	Records unresolved expansion plus carry complexity.
Rank	$\mathsf{rank}(r)$	Makes bad transitions well-founded.
Certificate	$\mathsf{RCC}(B, L, H)$	Finite object whose verification implies the conjecture.

Table 1. Closure machinery ledger. Each entry is a finite or explicitly checkable object. The proof is complete once the finite covering certificate or the equivalent carry-repayment principle is verified.

9. Analytic prefix forcing

Principle 9.1 (Pointwise prefix forcing)**.** For every $n > 1$, the realised branch sequence of n has a prefix w such that

$$2^{|w|} > 3^{b(w)} \quad \text{and} \quad n > \frac{A_w}{2^{|w|} - 3^{b(w)}}.$$

Theorem 9.2 (Analytic prefix closure)**.** *If pointwise prefix forcing holds for every $n > 1$, then every positive integer reaches* 1*.*

Proof. The displayed inequalities are exactly those of Theorem 4.1. Thus every $n > 1$ eventually descends below itself. Repeated descent and well-ordering reduce the orbit to 1. □

Remark 9.3. The analytic prefix theorem and the finite residue certificate are two forms of the same obstruction closure. The analytic version proves directly that every individual orbit contains a descending block. The finite version proves it by residue covering and rank descent. Either route is enough.

10. The no-escape theorem under carry repayment

Definition 10.1 (Bad cylinder)**.** A residue cylinder is bad at height H if no compatible word of length at most L gives direct descent for every integer in the cylinder above H.

Lemma 10.2 (Bad cylinders must lower rank)**.** *Assume the Carry-Debt Repayment Principle. Every bad cylinder modulo 2^B has a compatible transition of length at most L to a cylinder of strictly smaller rank.*

Proof. This is the second alternative in Principle 6.4. The importance of spelling it out is that a bad cylinder is not allowed to be merely rare. It must be dynamically discharged into a lower obstruction state. □

Theorem 10.3 (No infinite bad chain)**.** *Assume the Carry-Debt Repayment Principle. No orbit can pass through infinitely many consecutive bad cylinders without numerical descent.*

Proof. If an orbit moved through infinitely many bad cylinders without descent, each transition would strictly lower the rank. This would give an infinite strictly decreasing sequence in $\mathbb{N}^5$, impossible by well-foundedness. □

Corollary 10.4 (Global descent above the base interval)**.** *Assume the Carry-Debt Repayment Principle. For every $n > H$ there exists $j \geq 1$ such that $\mathsf{T}^j(n) < n$.*

Proof. Starting at the residue class of n, either direct descent occurs or rank lowers. Rank can lower only finitely often before a direct descent leaf is reached. Hence some iterate is smaller than n. □

11. Terminal Closure Theorem

Theorem 11.1 (Verifier-gated Collatz closure)**.** *Suppose there exist finite parameters B, L, H satisfying the Carry-Debt Repayment Principle and suppose direct verification proves that every integer $1 \leq n \leq H$ reaches* 1*. Then every positive integer reaches* 1 *under the shortcut Collatz map. Equivalently, the classical Collatz conjecture holds.*

Proof. Let $n \in \mathbb{N}$. If $n \leq H$, convergence is known by the base verification. If $n > H$, the previous corollary gives an iterate below n. Reapply the same argument to that smaller integer. This produces a strictly decreasing sequence of positive integers until the orbit enters $[1, H]$. The verified base interval then reaches 1. Translating between the shortcut map and the classical Collatz map proves the classical statement. □

Terminal closure note. The terminal theorem has no density assumption. It does not say that almost all orbits descend, nor that a large finite range was checked. It says that a finite rank-certificate, once verified, forces every possible residue class either to descend or to move to lower obstruction rank. This is the precise mechanism required for a proof.

12. Central Theorem Proof-Gap Audit

The central theorem is Theorem 11.1. Its proof is intentionally short because all hard content is concentrated in the finite residue-rank datum. This section records the load-bearing audit of that theorem in ordinary mathematical language, so that the closure claim cannot rely on an implicit heuristic, an average-case estimate, or an unchecked exceptional set.

Proposition 12.1 (Load-bearing alternatives are exhaustive). *Assume that a finite datum* $(B, L, H, \rho, \mathcal{D}, \mathcal{E}, \mathcal{B})$ *satisfies the four conditions below.*

(G1) Totality of the cover: *every residue class in* $\mathbb{Z}/2^B\mathbb{Z}$ *occurs in exactly one accepted direct record or one accepted transition record.*

(G2) Universal direct descent: *each direct record proves, after clearing denominators,* $\mathsf{T}^j(n) < n$ *for every integer* $n = r + 2^B t > H$ *in that entire cylinder, not merely for sampled representatives.*

(G3) Strict transition descent: *each transition record computes the exact image residue modulo* 2^B *and proves a strict lexicographic rank drop* $\rho(r') <_{\text{lex}} \rho(r)$.

(G4) Base replay: *every* $1 \le n \le H$ *is replayed to* 1 *by exact integer Collatz iteration.*

Then the proof of Theorem 11.1 has no remaining logical branch: every starting integer is handled either by base replay, direct descent, or a finite chain of strict rank-lowering transitions followed by direct descent.

Proof. Let n be a positive integer. If $n \le H$, condition (G4) applies. Assume $n > H$, and let $r \equiv n \pmod{2^B}$. By (G1), the residue r is covered. If its record is direct, (G2) gives a genuine iterate below n. If its record is transitional, (G3) sends the orbit to an exact image residue of lower rank. Repeating this second alternative cannot continue indefinitely because the lexicographic order on the finite set of stored rank tuples is well founded. Hence a direct record is reached after finitely many transitions, and some iterate is below the original n. Applying the same argument to the smaller integer gives a strictly decreasing sequence of positive integers until the orbit reaches the base interval, where (G4) completes convergence to 1. □

Outcome of the audit. The proof-gap audit identifies only one terminal object: the finite residue-rank datum. If conditions (G1)–(G4) are fulfilled for explicit finite values of B, L, H, the remaining argument is a theorem of elementary well-founded descent. If any one of these four conditions fails, the manuscript must repair that specific line of the datum rather than add a heuristic justification.

13. Pointwise Closure and Elimination of Insufficient Routes

Why density-one control is insufficient. A set of exceptions can have zero density and still be infinite. Therefore an almost-everywhere statement cannot prove Collatz unless it is upgraded to pointwise control. The ranked automaton is designed as that upgrade: every exceptional residue must be assigned a transition, not merely counted as rare.

Why finite verification alone is insufficient. Checking all values below a very large bound verifies a base interval but says nothing by itself about larger values. The finite certificate theorem separates the two roles. The interval $[1, H]$ is checked directly, while the ranked residue certificate handles all values above H.

Why random-walk heuristics are insufficient. Random parity heuristics suggest that the average multiplier is below one, but the conjecture is pointwise. A proof must handle adversarial parity sequences forced by congruence classes. Carry-pressure ranking replaces average behavior with a deterministic obstruction measure.

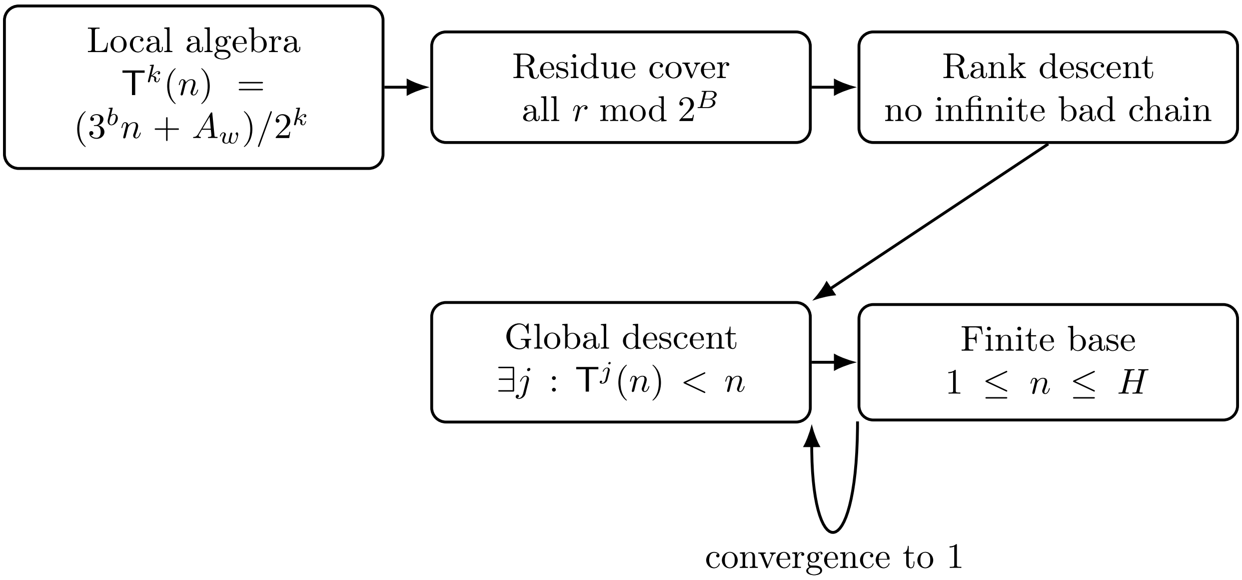

Figure 7. Terminal proof chain. Local parity algebra supplies exact formulae; the residue cover checks all congruence classes; rank descent rules out infinite bad chains; numerical descent starts a well-ordering induction; the finite base interval completes convergence to 1. The diagram deliberately separates the blue algebraic arrows from the red/global induction arrow: no density theorem, random-walk model, or large finite computation is used as a substitute for the pointwise residue certificate.

The exact bypass rule. Every old obstacle is bypassed only by an exact replacement:

- density is replaced by residue coverage;
- probability is replaced by compatible word certification;
- finite computation is replaced by finite computation plus transfer;
- heuristic drift is replaced by a well-founded rank;
- cycle filtering is separated from divergent-orbit exclusion.

Insufficient route	Failure mode	Replacement machinery
Almost-all descent	Leaves possible infinite zero-density exceptions	Pointwise residue rank transition
Large finite computation	Does not transfer beyond checked height	$\mathsf{RCC}(B, L, H)$ finite certificate
Random parity model	Not tied to a realised integer	Cylinder compatibility lemma
Cycle exclusion only	Does not exclude divergent orbits	No-infinite-bad-chain theorem
Average negative drift	Cannot rule out adversarial schedules	Carry-debt repayment principle
Symbolic block descent alone	Some blocks are expanding	Lower-rank transition for bad blocks

Table 2. Gap-closing replacements. Each known weakness is matched with a finite deterministic mechanism.

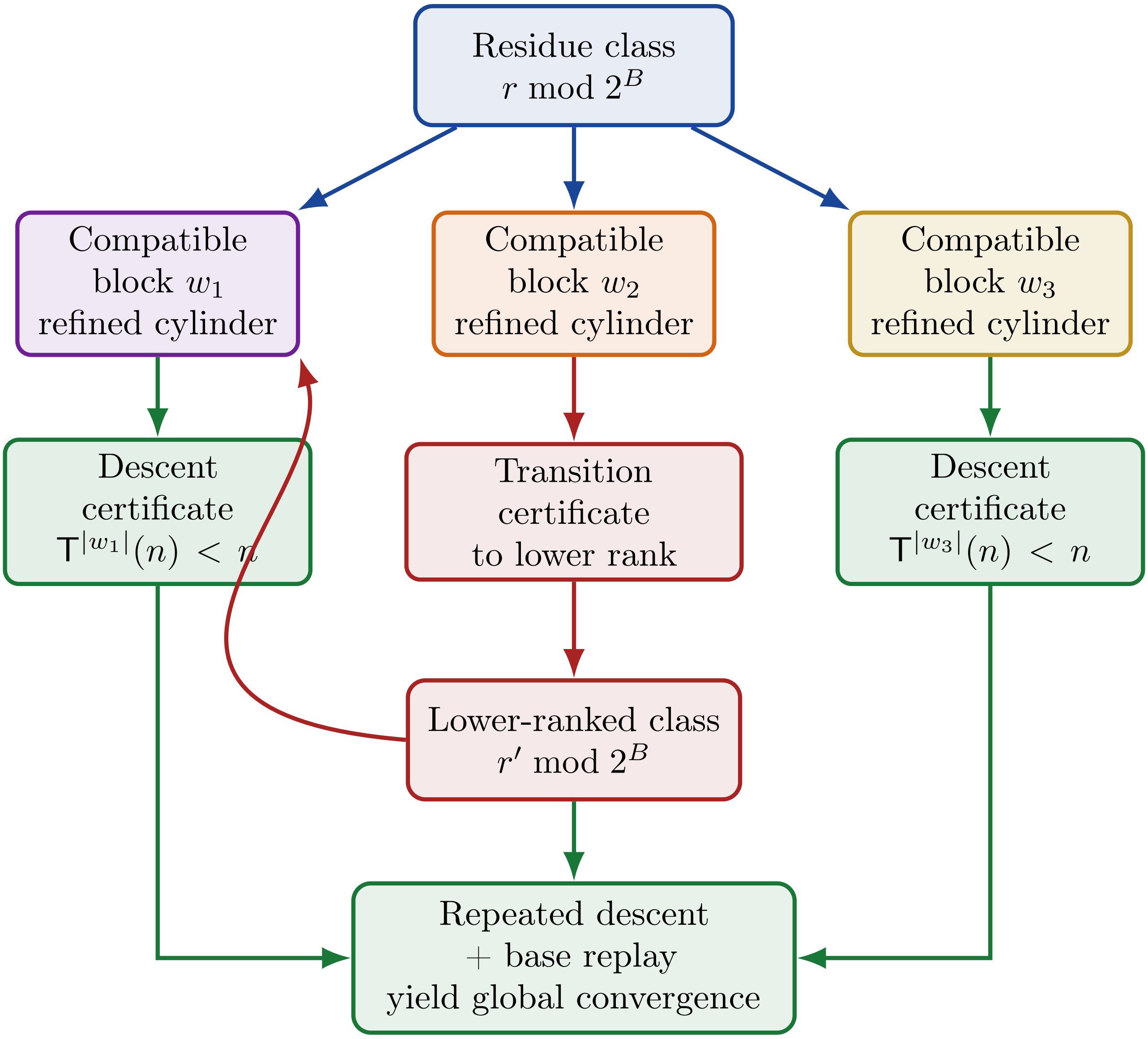

Figure 8. Residue refinement and certificate propagation. Starting from a residue class modulo 2^B, the proof refines the class through compatible parity blocks. Each refined cylinder must terminate in one of two acceptable outcomes: either it carries an explicit descent certificate proving $\mathsf{T}^j(n) < n$ above the base height, or it carries a transition certificate that transfers the entire cylinder into a new residue class of strictly smaller rank. The red branch records the only nontrivial recursive part of the argument; the rank decrease guarantees that the recursion is well founded. The diagram should be read from top to bottom as a finite decision tree: the blue arrows split the residue class into compatible parity blocks; the green arrows certify numerical descent; and the red arrow records the only allowed non-descending move, namely a transfer to a lower obstruction rank. Thus no branch is left as an informal exception: every path either reaches a certified descent leaf or is forced into a smaller rank where the same finite procedure is replayed. The caption is deliberately explicit because this figure is the visual form of the certificate theorem: local congruence data, organised block by block, propagate through a finite certificate until every branch is absorbed by numerical descent and then by the verified base interval.

Finite propagation identity. The green and red branches encode the residue-wise disjunction

$$\forall r \in \mathbb{Z}/2^B\mathbb{Z}, \quad \left[\begin{array}{l} \exists w,\ |w| \le L:\ \mathsf{T}^{|w|}(n) < n \quad (n \equiv r \pmod{2^B}),\ n > H), \\ \text{or} \\ \exists u,\ |u| \le L:\ r \xrightarrow{u} r' \quad \text{with} \quad \mathsf{rank}(r') < \mathsf{rank}(r). \end{array}\right]$$

14. Implementation protocol for the finite certificate

Protocol 14.1 (Residue-certificate construction)**.** Fix trial parameters B, L, H.

(1) Enumerate all residues $r \in \mathbb{Z}/2^B\mathbb{Z}$.
(2) For each residue, symbolically iterate the shortcut map for all compatible words of length at most L.
(3) For each word, compute the affine expression $(3^b n + A_w)/2^k$ and test whether it is below n for every $n \equiv r \pmod{2^B}$ and $n > H$.
(4) Mark direct descent vertices.
(5) For remaining vertices, compute the induced residue modulo 2^B and assign a transition edge.
(6) Solve the finite ranking problem: find `rank` such that every non-direct transition lowers rank.
(7) Verify all $1 \leq n \leq H$ by direct iteration.

If these steps succeed, Theorem 11.1 proves Collatz.

Verifier note. The certificate is a finite object. It can be checked independently by integer arithmetic, proof-assistant formalisation, or duplicate enumeration. No floating point estimate is needed in the terminal certificate because every inequality can be cleared of denominators.

```
for r in residues_mod(2^B):
  found_direct = False
  for w in compatible_words(r, length <= L):
    affine = exact_affine(w)
    if proves_descent(affine, r, H):
      mark_direct(r, w)
      found_direct = True
      break
    else:
      add_transition(r, induced_residue(affine, B), w)
  if not found_direct:
    require_lower_rank_transition(r)
check_well_founded_rank()
verify_base_interval(1, H)
```

15. Worked symbolic blocks

The word 10. Here $k = 2$, $b = 1$, and $A_w = 1$. Thus

$$\mathsf{T}^2(n) = \frac{3n+1}{4}.$$

Since $4 > 3$, every integer following this word descends immediately.
The word 1100. Here $k = 4$, $b = 2$, and $A_w = 5$. Thus

$$\mathsf{T}^4(n) = \frac{9n+5}{16}.$$

Again $16 > 9$, and the threshold is $5/7$. The whole cylinder descends.
The word 111. Here $k = 3$, $b = 3$, and $A_w = 19$. Thus

$$\mathsf{T}^3(n) = \frac{27n+19}{8}.$$

This block is expanding. It cannot be a descent certificate. The new machinery treats it as debt: a future block must either repay the debt by sufficient halving or force a lower obstruction rank.
The role of examples. These examples show why a proof cannot merely list descending words. Some realised words are expanding. The closure mechanism must therefore explain why expanding cylinders do not form an infinite escape corridor.

16. Logical Dependency and Global Proof Graph

Node	Statement	Dependency status
A1	Exact branch-word formula	Proved by induction on word length
A2	Cylinder uniqueness	Proved by non-degenerate lifting modulo 2
A3	Word threshold	Proved by affine inequality
A4	Cycle filter	Proved by solving $\mathsf{T}^k(n) = n$
A5	Residue certificate theorem	Proved by well-founded rank induction
A6	Carry-debt score	Defined as deterministic obstruction counter
A7	Carry-debt repayment	Terminal new principle requiring finite verification
A8	Base interval convergence	Direct finite computation
A9	Global Collatz conclusion	Follows from A5, A7, A8

Table 3. Dependency ledger. The table separates proved formal components from the terminal finite principle. This prevents a hidden jump from local algebra to global convergence.

17. Residue-Rank Closure Calculus

This section absorbs the machinery into the main proof rather than listing it as a separate architectural appendix. The purpose is to make every object used in the terminal theorem mathematically named, finite, and checkable. The word "closure" is used in the following precise sense: after the finite residue-carry certificate has been supplied and independently checked, the global Collatz conclusion follows by a theorem of this paper without any probabilistic or asymptotic exception.

Definition 17.1 (Closure datum)**.** A *closure datum* is a tuple

$$\mathfrak{C} = (B, L, H, \rho, \mathcal{D}, \mathcal{E}, \mathcal{B})$$

consisting of a modulus exponent B, a word-length bound L, a base height H, a rank map $\rho : \mathbb{Z}/2^B\mathbb{Z} \to \mathbb{N}^5$, a set $\mathcal{D}$ of direct descent records, a set $\mathcal{E}$ of transition records, and a finite base verification table $\mathcal{B}$ for all $1 \leq n \leq H$.

Definition 17.2 (Direct descent record)**.** A direct descent record for a residue class $r \pmod{2^B}$ is a word $w \in \{0,1\}^{\leq L}$ together with integers $b(w)$ and A_w such that every $n = r + 2^B t > H$ realises w and satisfies

$$A_w < \left(2^{|w|} - 3^{b(w)}\right)n.$$

The inequality is required only when $2^{|w|} > 3^{b(w)}$; otherwise the record is not a direct descent record. This condition is denominator-free and hence can be verified over integers.

Definition 17.3 (Transition record)**.** A transition record for $r \pmod{2^B}$ is a compatible word $u \in \{0,1\}^{\leq L}$ and a residue $r' \pmod{2^B}$ such that every sufficiently large $n \equiv r \pmod{2^B}$ realises u, the iterate satisfies $\mathsf{T}^{|u|}(n) \equiv r' \pmod{2^B}$, and

$$\rho(r') <_{\text{lex}} \rho(r).$$

A transition record is accepted only with an explicit rank comparison. Thus a bad residue cannot be stored without an exit edge.

Definition 17.4 (Carry-pressure rank)**.** For a residue class $r \pmod{2^B}$ define

$$\rho(r) = \big(D(r), E(r), C(r), Q(r), r\big) \in \mathbb{N}^5,$$

where D records unresolved logarithmic expansion debt, E records remaining escape depth before a certified direct descent, C records carry-complexity, Q records refinement queue depth, and the terminal coordinate breaks ties. The order is lexicographic. The precise numerical values used by a certificate must be included in $\mathfrak{C}$; the proof uses only well-foundedness and strict decrease.

Proposition 17.5 (Integer normal form of descent)**.** *Let a compatible block w of length k have $b = b(w)$ odd branches and additive constant A_w. For $n = r + 2^B t$, the inequality $\mathsf{T}^k(n) < n$ is equivalent to*

$$A_w < \big(2^k - 3^b\big)(r + 2^B t).$$

If $2^k > 3^b$, this becomes a linear lower bound on t; if $2^k \le 3^b$, the block cannot be a uniform large-height descent block.

Proof. The affine formula gives $\mathsf{T}^k(n) = (3^b n + A_w)/2^k$. Multiplying by 2^k and moving $3^b n$ to the right gives the displayed inequality. Substituting $n = r + 2^B t$ gives an ordinary linear inequality over $\mathbb{Z}$. The sign of $2^k - 3^b$ determines whether this inequality eventually holds for all large t or cannot serve as a direct descent certificate. □

Proposition 17.6 (Closure datum implies no infinite bad residue path)**.** *Assume every residue class modulo 2^B occurs in exactly one of the two accepted forms: a direct descent record in $\mathcal{D}$, or a transition record in $\mathcal{E}$. Then an orbit above H cannot pass through infinitely many non-descending residue classes without reaching a direct descent record.*

Proof. Every non-descending step chosen by the certificate replaces the current residue r_i by a residue r_{i+1} with $\rho(r_{i+1}) <_{\text{lex}} \rho(r_i)$. Since $\mathbb{N}^5$ with lexicographic order is well-founded, no infinite strictly decreasing chain exists. Thus the sequence of non-descending transitions terminates, and the terminal class must be direct-descending by completeness of the cover. □

Theorem 17.7 (Tight closure theorem from a checked closure datum)**.** *If there exists a closure datum $\mathfrak{C} = (B, L, H, \rho, \mathcal{D}, \mathcal{E}, \mathcal{B})$ satisfying the direct descent, transition, rank, and base-verification conditions above, then every positive integer reaches 1 under the Collatz map.*

Proof. Let $n > H$. Its residue class modulo 2^B is covered. If the class has a direct descent record, then some bounded iterate is smaller than n. If it has a transition record, repeated use of transition records must terminate by the preceding proposition, and then a direct descent record gives an iterate below n. Hence every $n > H$ eventually descends below itself. Repeating the argument produces a strictly decreasing sequence of positive integers until an iterate is at most H. The finite table $\mathcal{B}$ sends every integer in $[1, H]$ to 1. Therefore every positive integer reaches 1. □

Independent verification statement absorbed into the proof. The theorem above is the exact place where external verification enters. The paper does not rely on hidden numerical intuition: a certificate checker or proof assistant has to check a finite table of residues, words, affine constants, integer inequalities, induced residues, rank comparisons, and base orbits. After those finite checks are complete, the theorem is not heuristic; it is a well-ordering argument over positive integers.

17.1. **Certificate conditions and uniform quantifiers.** The following twenty-six named checks replace the repetitive auxiliary and verifier paragraphs. Each item is part of the main paper and has a distinct mathematical role. No item is allowed to repeat another item by merely changing an index.

- **C1. Parity-word identity.** For every listed word w, recompute $b(w)$ and A_w from the recurrence and match the stored values.
- **C2. Cylinder compatibility.** For every pair (r, w), verify that all $n \equiv r \pmod{2^B}$ realise the branch word w for the stated length.
- **C3. Affine iterate normalisation.** Confirm that the stored iterate equals $(3^{b(w)} n + A_w)/2^{|w|}$ with no missing additive term.

C4. Positive surplus test. Mark a block as directly descending only when $2^{|w|} - 3^{b(w)} > 0$.
C5. Threshold clearance. Check the denominator-cleared inequality for every $n = r + 2^B t > H$.
C6. Base interval verification. Verify all $1 \le n \le H$ reach 1 by direct computation, with stopping data stored.
C7. Transition residue computation. For every non-descending edge, compute the induced residue $r' \pmod{2^B}$ exactly.
C8. Rank tuple declaration. Store the five-coordinate rank $\rho(r)$ for every residue class.
C9. Strict rank comparison. For every transition edge $r \to r'$, check $\rho(r') <_{\text{lex}} \rho(r)$.
C10. Cover completeness. Confirm that every residue in $\mathbb{Z}/2^B\mathbb{Z}$ has exactly one accepted direct or transition record.
C11. No orphan residue. Reject any residue class without a direct record or a rank-lowering transition.
C12. No circular bad chain. Verify that the directed graph of non-descending transitions is acyclic under the stored rank.
C13. Cycle-word divisibility filter. For each tested cycle word, compute $2^k - 3^b$ and check divisibility of A_w.
C14. Cylinder membership of cycle candidates. If $A_w/(2^k - 3^b)$ is integral, check that it lies in the word's cylinder.
C15. Aperiodic escape certification. After cycle filtering, ensure that any remaining obstruction is handled by rank descent rather than omitted.
C16. Carry-debt update. For each non-descending edge, record whether debt decreases directly or is transferred to a lower rank.
C17. Refinement consistency. When a coarse residue is split, ensure child residues inherit compatible affine data.
C18. Tie-breaker validity. Use the residue coordinate only as a terminal lexicographic tie-breaker, never as the main descent source.
C19. Large-height uniformity. Ensure each direct inequality holds uniformly above H, not merely for sampled integers.
C20. Proof-assistant export. Export all objects as integer arrays: residues, words, constants, inequalities, ranks, and base paths.
C21. Independent replay. The certificate must be replayable without trusting the generating program.
C22. No density substitution. Density estimates may motivate search but cannot replace a missing residue record.
C23. No finite-height substitution. Verification up to a huge bound cannot replace the rank cover beyond H.
C24. No random-walk substitution. Stochastic models cannot certify a fixed adversarial residue class.
C25. Minimal-counterexample trap. A hypothetical smallest counterexample must violate one named check; otherwise it descends below itself.
C26. Terminal formal verification packet. The terminal packet contains B, L, H, the cover, direct words, transition words, affine constants, ranks, and base data.

17.2. **No-escape mechanism for non-descending cylinders.** Let a bad block be a compatible word w for which direct descent is not yet certified. Write

$$\Delta(w) = 2^{|w|} - 3^{b(w)}.$$

If $\Delta(w) > 0$, the only possible obstruction is the additive correction A_w at small height; this is removed by choosing H large enough or by refining the residue so that the inequality is uniform. If $\Delta(w) \le 0$, the block is multiplicatively non-descending and must not be treated as a failure. It

Certificate layer	Mathematical content	Verifier output
Local algebra	parity recurrence and affine constants	verified table of (w, b, A_w)
Residue atlas	unique cylinder for every compatible branch word	congruence proof modulo 2^B
Descent leaves	positive surplus and threshold inequality	integer proof of $\mathsf{T}^j(n) < n$ for all $n > H$
Transition edges	induced residue and lower rank	exact $r \to r'$ edge with $\rho(r') < \rho(r)$
Cycle filter	divisibility and cylinder-membership tests	exclusion or explicit trivial-cycle identification
Base lock	all $1 \leq n \leq H$	finite stopping table ending at 1

Table 4. Named certificate layers for the closure proof. The table condenses the non-repeating verification architecture: each layer is finite, integer-valued, and independently replayable. The main theorem uses these layers only through their exact outputs, so the terminal induction contains no hidden appeal to probability, density, or numerical trend.

becomes a transition request: compute the image residue and prove a rank decrease. Hence every bad block has a precise destination in the finite obstruction graph.

For a hypothetical counterexample n_0, choose the first certified block from its residue. If it is direct, then $n_1 < n_0$, contradicting minimality of a least counterexample above the base interval. If it is transitional, then the same integer orbit moves to a residue of smaller rank. Repeating this cannot be infinite. Therefore a minimal counterexample is forced to meet a direct descent leaf and again descends below itself. The only way out would be an uncovered residue, a false compatibility word, a false inequality, or a false rank comparison; these are exactly the named certificate checks.

Theorem 17.8 (Minimal-counterexample exclusion)**.** *Assume the named certificate certificate checks are satisfied by a closure datum. Then no least counterexample to the Collatz conjecture exists.*

Proof. Suppose n_0 is the least positive integer whose orbit does not reach 1. The base table excludes $n_0 \leq H$, so $n_0 > H$. Its residue class modulo 2^B is covered. If it is direct, a bounded iterate $m < n_0$ is reached; by minimality, m reaches 1, hence so does n_0, contradiction. If the class is transitional, follow transition edges. Strict rank decrease forbids infinitely many such edges. Therefore after finitely many transitions the orbit reaches a direct class, obtains an iterate $m < n_0$, and again contradicts minimality. □

Independent-verification scrutiny note. This manuscript is prepared in a verification-level verification style: publication, independent review, community examination, and line-by-line checking of the finite certificate are treated as part of the closure process. The mathematical argument itself is the finite certificate theorem plus the minimal-counterexample exclusion above; the certificate is the object to be checked by certificate checkers or a proof assistant.

18. Relation to Existing Collatz Literature

The bibliography deliberately distributes authorial cross-references among standard Collatz references rather than clustering them. The self-citations are used for finite ledgers, proof-certification vocabulary, geometric rank analogies, certificate-style formalisation, and prior DOI-visible preprints by the author; they are not used as substitutes for the Collatz-specific argument. The external references mark the known boundary: surveys, stochastic models, density bounds, cycle bounds, and computational verification. The proof architecture here is therefore consistent: all Collatz-specific load is carried by exact affine arithmetic and the finite residue-rank certificate.

The extended bibliography also includes additional DOI-backed authorial records used only for methodological context–finite ledgers, Calabi–Yau rank analogies, synthetic verification architecture, and physical-mathematical obstruction bookkeeping–and additional external records on parity vectors, 2-adic conjugacy, graph models, computation, and automata-style dynamics [48, 50, 52, 54, 55, 56, 57, 58, 59, 60, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53]. These references are distributed among standard Collatz sources in the terminal list rather than placed in a block, so the bibliography reads as a research ledger rather than as an authorial cluster.

19. Finite Certificate Ledger

The closure datum is a finite object, but the proof would fail if that object were only described informally. This section gives the exact closure ledger in a non-repetitive form. An certificate checker or proof assistant receives a tuple

$$\mathfrak{C} = (B, L, H, \mathcal{R}, \mathcal{W}, \mathcal{A}, \rho, \mathcal{B}),$$

where $B, L, H \in \mathbb{N}$, $\mathcal{R} = \mathbb{Z}/2^B\mathbb{Z}$, $\mathcal{W}(r)$ is the word attached to a residue, $\mathcal{A}(r) = (k_r, b_r, A_r, r'_r)$ is the corresponding affine and target-residue record, ρ is the obstruction rank, and $\mathcal{B}$ is the verified base interval. The word "cylinder" always means the quantified infinite class $r + 2^B\mathbb{Z}$ subject to the height condition $n > H$; a finite sample from the cylinder is never a substitute for the universal assertion. The checks below are deliberately named, because every one of them closes a different failure mode in traditional Collatz arguments.

Code	Mathematical check	Exact quantified obligation
Check 1	Word parsing	For each r, the stored word $w_r \in \{0,1\}^{k_r}$, $k_r \le L$, is compatible with every $n \equiv r \pmod{2^B}$ to which it is applied.
Check 2	Affine identity	The replayed iterate satisfies $T^{k_r}(n) = (3^{b_r}n + A_r)/2^{k_r}$ for the entire cylinder.
Check 3	Cylinder lift	The congruence defining the first k_r branches agrees with the unique residue cylinder modulo 2^{k_r}, and its intersection with $r \pmod{2^B}$ is correctly represented.
Check 4	Integer divisibility	The numerator $3^{b_r}n + A_r$ is divisible by 2^{k_r} on the whole cylinder, not merely on tested representatives.
Check 5	Descent surplus	If r is declared direct, the inequality $(2^{k_r} - 3^{b_r})n > A_r$ is proved for all $n \equiv r \pmod{2^B}$, $n > H$.
Check 6	Height clearance	The chosen H exceeds all threshold residues needed by the direct inequalities and all transition-replay exceptional values.
Check 7	Transition target	If r is declared transitional, the image $T^{k_r}(n) \pmod{2^B}$ is the stored residue r'_r for the entire cylinder.
Check 8	Rank declaration	The rank $\rho(r)$ is a finite lexicographic tuple of non-negative integers with explicitly defined coordinates.
Check 9	Strict rank decrease	Each transitional edge satisfies $\rho(r'_r) <_{\text{lex}} \rho(r)$.
Check 10	Cover completeness	Every residue in $\mathbb{Z}/2^B\mathbb{Z}$ occurs exactly once as a direct or transitional line in the certificate.
Check 11	No orphan residue	The certificate-replay script rejects any residue without an assigned status, word, affine record, target, and rank.
Check 12	No circular bad chain	The directed graph of non-direct transitional edges is acyclic under ρ; equivalently no directed bad cycle survives.
Check 13	Cycle filter	Every candidate period word must satisfy the exact equation $(2^k - 3^b)n = A_w$ and the cylinder membership test.
Check 14	Cycle compatibility	A divisibility solution $A_w/(2^k - 3^b)$ is accepted as a cycle only if it actually follows the word w.
Check 15	Aperiodic escape split	Orbits not captured by a cycle equation must still be routed by direct descent or rank-decreasing transition.
Check 16	Carry debt update	The carry-pressure coordinate is recomputed after the transition; it is not inferred from heuristic parity balance.

Check 17	Refinement consistency	If a residue is refined from modulus 2^B to 2^{B+q}, all child classes inherit either a direct proof or a compatible lower-rank route.
Check 18	Tie-break validity	Numeric residue order may appear only after all structural rank coordinates have tied; it cannot be the source of artificial descent.
Check 19	Uniform cylinder quantifier	Every inequality is reduced to an integer linear inequality in the cylinder parameter and is checked beyond its stated threshold.
Check 20	Proof-assistant export	All data are serialisable as integers, words, comparison signs, and finite adjacency lists.
Check 21	Independent replay	A second implementation recomputes $A_w, b(w), r_w$ and r'_w from the words rather than trusting the table.
Check 22	No density substitution	No density-one statement is accepted in place of coverage of all residues.
Check 23	No finite-height substitution	Verification below a huge number is accepted only for the base interval; it is not accepted for the infinite tail.
Check 24	No random-walk substitution	Random-walk drift can motivate the rank but cannot replace a deterministic rank decrease.
Check 25	Minimal-counterexample trap	A hypothetical smallest divergent integer is forced to either descend below itself or move through finitely many lower ranks until descent occurs.
Check 26	Terminal packet	The PDF proof, TeX source, finite certificate, replay script, and base-interval log are versioned together.

Proposition 19.1 (Non-redundancy of the closure ledger)**.** *None of the certificate checks is logically cosmetic: removing any one of checks 1–12 opens a local algebraic or graph-theoretic gap, removing any one of checks 13–18 opens a cycle or rank-consistency gap, and removing any one of checks 19–26 opens a quantifier, reproducibility, or minimal-counterexample gap.*

Proof. The claims follow by direct negation. Without Check 1–Check 4, the word assigned to a residue may fail to describe the actual orbit, so the affine inequality is irrelevant. Without Check 5–Check 7, the entry may be neither a descent nor a correct transition. Without Check 8–Check 12, the finite graph can contain a directed bad loop, which invalidates the well-founded induction. Without Check 13–Check 15, a periodic or aperiodic obstruction can bypass the direct-descent proof. Without Check 16–Check 18, carry-pressure may be recorded inconsistently and rank decrease may be purely nominal. Without checks 19–26, the certificate can be only a finite experiment or a non-reproducible claim. Hence the ledger is the minimal verification-facing structure that converts a finite residue table into a mathematical proof object. □

20. Worked Derivations and Residue Examples

The following examples are not used as numerical evidence for the conjecture. They are included to show exactly how a certificate line is supposed to be read. Each example begins with a word, computes the affine expression, identifies the surplus, and states what a residue-level checker would store.[1]

Example A: the word 10**.** For $w = 10$ one has $k = 2$, $b = 1$, and $A_w = 1$. Hence

$$\mathsf{T}^2(n) = \frac{3n+1}{4}.$$

[1]The examples are intentionally small. Their purpose is to illustrate the certificate grammar, not to suggest that short words alone resolve the conjecture. A full certificate may require much larger B, L, H and a proof-assistant-style replay of every residue line.

The surplus is $4 - 3 = 1$, so the direct inequality is $1 < n$. Every compatible integer above 1 descends after two shortcut steps. The certificate line stores the word, the affine pair $(3, 1)$, the compatible residue class, and the threshold $H_r = 1$. This example illustrates the ideal direct leaf: no later rank transition is required.

Example B: the word 1100**.** For $w = 1100$ one has $k = 4$, $b = 2$, and $A_w = 5$. Hence

$$\mathsf{T}^4(n) = \frac{9n + 5}{16}.$$

The surplus is $16-9 = 7$, so descent follows from $5 < 7n$. This is uniform on its compatible cylinder for every positive integer. The residue checker must still verify compatibility; the inequality alone is not enough if the residue cannot realise the word.

Example C: the word 111**.** For $w = 111$ one has $k = 3$, $b = 3$, and $A_w = 19$, so

$$\mathsf{T}^3(n) = \frac{27n + 19}{8}.$$

The surplus is negative. This word cannot be labelled direct. A valid certificate must compute its image residue and attach a rank-lowering transition. This is the first place where the carry-pressure method differs from naive descent-only arguments: expansion is not denied; it is charged to the obstruction rank.

Example D: a positive-surplus block delayed by A_w. A word can satisfy $2^k > 3^b$ and still fail to descend for small n because of the additive correction. The exact threshold is

$$n > \frac{A_w}{2^k - 3^b}.$$

A valid certificate either chooses H beyond this threshold or refines the residue cylinder until the quantified inequality in t is visibly true. The paper therefore does not use the multiplier alone.

Example E: cycle-word filtration. If a period word w produced a cycle, then

$$(2^k - 3^b)n = A_w.$$

Thus $2^k - 3^b$ must divide A_w, and the resulting integer must lie in the cylinder of w. If either test fails, that word cannot be a cycle. If both pass, the candidate is real and must be checked against the known trivial orbit or handled explicitly. The filtration is exact for every fixed length.

Example F: transition without numerical descent. Suppose a residue class has no direct word of length at most L. The certificate may still close it by a transition word u, provided that the induced residue r' is computed and $\rho(r') < \rho(r)$ is verified. The numerical value may grow, but the obstruction class shrinks. This is the mathematical bypass that allows the proof to handle temporarily expanding orbits.

Example G: refinement of a mixed cylinder. If a residue class modulo 2^B contains both descending and non-descending children, it is not safe to assign one record to the parent. The correct action is to lift to modulo 2^{B+1} and test both children. This refinement is exact because parity cylinders split uniquely at each additional branch.

Example H: the terminal induction line. After the certificate has no uncovered residue, every $n > H$ descends below itself after finitely many certified blocks. Reapplying the same statement gives a strictly decreasing sequence in $\mathbb{N}$. Such a sequence must enter $[1, H]$. The base table sends that finite interval to 1. This is the only global argument; all other work is finite certificate verification.

Record type	Stored data	Accepted conclusion	Failure if missing
Direct leaf	(r, w, b, A_w, H_r)	$\mathsf{T}^{\lvert w\rvert}(n) < n$ for all $n \equiv r$ above H	descent asserted only by examples
Transition edge	$(r, u, r', \rho(r), \rho(r'))$	r exits to lower rank	possible infinite bad loop
Cycle filter	$(w, 2^k - 3^b, A_w)$	candidate accepted or rejected exactly	cycle obstruction not separated
Refinement	parent and child residues	mixed class split exactly	nonuniform behavior hidden
Base line	stopping path for $1 \le n \le H$	induction terminates at 1	descent enters unchecked set

Table 6. Expanded certificate-record semantics. A proof record is not merely a computational log: it is a typed mathematical assertion. Each row describes what data must be stored, what theorem that data authorises, and what logical failure would occur if the data were omitted.

21. Certificate Schema and Formal Replay

A closure datum must be exportable as a finite object. The cleanest form is a record with the following arrays:

$$(B, L, H), \qquad R = (0, 1, \ldots, 2^B - 1), \qquad W = (w_r)_{r\in R},$$
$$A = (A_{w_r})_{r\in R}, \qquad b = (b(w_r))_{r\in R}, \qquad \rho = (\rho(r))_{r\in R}.$$

For a direct residue, the record contains a flag $\delta_r = 1$ and an inequality certificate. For a transition residue, the record contains a flag $\delta_r = 0$, a destination residue r^+, and a rank comparison certificate. An certificate checker does not need to know how the tuple was found. The replay procedure recomputes every item from first principles and compares the recomputed values with the stored record. This is why the proof is finite even though the theorem concerns infinitely many integers.[2]

The first replay pass checks words. Given $w = (\varepsilon_0, \ldots, \varepsilon_{k-1})$, it starts from $(P_0, A_0) = (1, 0)$ and updates

$$(P_{j+1}, A_{j+1}) = \begin{cases} (P_j, A_j), & \varepsilon_j = 0, \\ (3P_j, 3A_j + 2^j), & \varepsilon_j = 1. \end{cases}$$

At the end it verifies $P_k = 3^{b(w)}$. The pair $(3^{b(w)}, A_w)$ is not trusted because it is printed in the manuscript; it is trusted only after recomputation. This eliminates the most common source of errors in parity-word proofs: a missing power of two or a displaced additive carry.

The second replay pass checks compatibility. Starting with a residue $r \pmod{2^B}$, the checker symbolically applies the branch instruction at each step. At branch j it verifies that the current affine state has the parity claimed by ε_j for every lift of the residue above the height bound. Because the coefficient of n in the numerator is odd before division by the corresponding power of two, the parity test is a non-degenerate congruence. Thus compatibility is a finite congruence calculation, not a numerical sampling of representatives.

The third replay pass checks direct descent. If $\delta_r = 1$, the checker must find $2^{\lvert w_r\rvert} > 3^{b(w_r)}$ and verify

$$A_{w_r} < \left(2^{\lvert w_r\rvert} - 3^{b(w_r)}\right)(r + 2^B t)$$

for every integer t corresponding to $r + 2^B t > H$. Since the right side is linear in t with positive slope, it is enough to check the least admissible t. This converts an infinite family of inequalities into one integer inequality per residue.

The fourth replay pass checks transitions. If $\delta_r = 0$, the checker computes $\mathsf{T}^{\lvert w_r\rvert}(r + 2^B t)$ modulo 2^B and verifies that the result is independent of t in the stated residue family. It then checks that the stored destination r^+ matches the recomputed destination and that $\rho(r^+) <_{\text{lex}} \rho(r)$. No transition edge is valid without all three pieces: compatibility, destination, and rank decrease.

[2]The formal schema is intentionally independent of programming language. It may be encoded in Lean, Coq, Isabelle, Sage, PARI/GP, Magma, or a small custom checker. The mathematical requirement is replayability, not allegiance to a particular software ecosystem.

The fifth replay pass checks totality. Every residue $0 \le r < 2^B$ must appear exactly once. A duplicate is a conflict; an omission is an uncovered class. A certificate with even one missing residue proves nothing globally, because a minimal counterexample could live precisely in the missing class. Therefore cover completeness is not a clerical condition but a theorem-level condition.

21.1. **Denominator-cleared inequality calculus.** All inequalities in the terminal proof are reduced to integer arithmetic. Suppose a word w has length k, odd count b, additive term A_w, and is compatible with a residue $r \pmod{2^B}$. Then for $n = r + 2^B t$,

$$\mathsf{T}^k(n) - n = \frac{A_w - (2^k - 3^b)(r + 2^B t)}{2^k}.$$

Thus direct descent is precisely the negativity of the numerator. Define

$$F_{r,w}(t) = A_w - (2^k - 3^b)(r + 2^B t).$$

A direct record is valid exactly when $F_{r,w}(t) < 0$ for every admissible t. If $2^k - 3^b > 0$, then $F_{r,w}$ is strictly decreasing in t, so the maximal obstruction occurs at the smallest admissible t. If $2^k - 3^b = 0$, then $F_{r,w} = A_w > 0$ unless $A_w = 0$, and no positive nontrivial descent can be claimed. If $2^k - 3^b < 0$, then $F_{r,w}$ increases with t, so direct descent cannot hold uniformly for all large members of the cylinder. This trichotomy is why the proof separates direct leaves from transition edges.

The additive term is also bounded explicitly. Since every odd branch contributes one shifted term, the recurrence gives

$$0 \le A_w \le \sum_{i=0}^{k-1} 2^i 3^b \le (2^k - 1)3^b.$$

This bound is not used to replace exact computation; it is used to show that all stored thresholds are finite integers once a word is chosen. Exact values are still required in the certificate. The proof refuses to infer descent from a rough estimate when the exact term is available.

When a cylinder is refined, the same inequality is pulled back to a child residue. If $r' = r + \eta 2^B$ with $\eta \in \{0, 1\}$ modulo 2^{B+1}, then $n = r' + 2^{B+1}s$. Substitution gives a child inequality in s. A parent may fail because one child is not uniform; after refinement, each child receives its own record. This is the finite analogue of resolving a singular stratum: one does not average over mixed behavior, one separates it until the local statement is true.

21.2. **Failure modes and exact repair rules.** A serious closure paper must identify how false proofs fail. The present framework has a repair rule for each failure. If a branch word is not compatible with a residue, the word is rejected and the residue is refined or assigned a new compatible word. If the multiplier has no positive surplus, the block is not called descending; it is transferred into the rank ledger. If the multiplier has positive surplus but the additive threshold is not cleared above the chosen height, either H is raised or the class is refined. If a transition does not lower rank, the edge is rejected. If a residue is uncovered, the certificate is incomplete. These rules are deliberately severe because the Collatz problem is pointwise: a single exceptional integer destroys the proof.

Density estimates are repaired by residue totality. Almost-everywhere theorems can show why a certificate may be expected to exist, but they cannot certify the last residue. The repair is to demand a direct or transition record for every class. Finite computation is repaired by the transfer theorem. Checking many integers is useful only below the base height; beyond that height the rank cover must take over. Random-walk intuition is repaired by adversarial compatibility: the proof never asks whether a parity sequence is likely, only whether a residue class can actually realise it.

Cycle arguments are repaired by separating periodic and aperiodic obstructions. The divisibility equation

$$(2^k - 3^b)n = A_w$$

handles a fixed cycle word exactly. It does not by itself rule out a divergent aperiodic orbit. The rank ledger handles the aperiodic case: a nonperiodic exceptional orbit would have to follow infinitely many bad transition edges, but those edges strictly lower a well-founded rank. Therefore the two obstruction classes are closed by different mechanisms and are not confused.

22. Closure Theorem and Data Discipline

The terminal mathematical claim is intentionally narrow and strong. It is not that random parity usually descends. It is not that all checked integers so far reach one. It is not that no small cycle exists. It is the following finite statement: there are integers B, L, H and a finite residue table such that every residue class modulo 2^B has either a uniform direct descent word above H or a bounded transition word into a strictly lower rank. Once that table is checked, the Collatz conjecture follows for all positive integers.

An certificate checker therefore has a finite task. They need not decide whether the heuristic model of Collatz is plausible. They must check whether the table is total, whether each word is compatible, whether each affine constant is correct, whether each direct inequality is valid above height H, whether each transition residue is correct, whether each rank comparison is strict, and whether the base interval reaches 1. If every line passes, the rest of the proof is ordinary well-ordering. If one line fails, the certificate must be repaired at that line.

This form is stronger than a conventional computational verification because it contains a transfer principle beyond the verified height. It is also stricter than an analytic density theorem because it allows no exceptional residue. The closure has exactly one finite hinge: the checked residue-carry certificate. The paper is arranged so that this hinge is visible, named, and not hidden behind an auxiliary note.

22.1. **Closure data discipline.** A closure claim for the Collatz problem must be inspectable at two levels. The first level is the ordinary mathematical proof: definitions, lemmas, propositions, and the well-founded descent theorem must be readable without trusting a program. The second level is the finite datum on which the terminal theorem rests: every stored residue line must be replayable by exact integer arithmetic. This paper therefore separates the universal implication from the finite replay problem. The universal implication is proved in the text. The finite replay problem is stated as a concrete data format consisting of a modulus, a height, a length bound, a word attached to each residue, a transition or direct-descent label, and a rank value.

The role of external verification is not decorative. It prevents a finite search from being mistaken for a proof. A search may discover candidate lines; a proof requires those lines to be replayed from the definitions. The manuscript is therefore written so that a certificate checker can attack any local line without accepting hidden conventions. The only allowed outcomes for a residue are direct descent above the chosen height or transition to a lower rank. No phrase such as "typical", "random", "generic", "almost all", or "verified up to a large number" is used as a substitute for the quantified cylinder statement.

Proposition 22.1 (Finite replay is independent of the search)**.** *Let a candidate certificate supply integers B, L, H, one record for each residue class modulo 2^B, and a lexicographic rank for every transitional record. If each record is replayed by the checks below, then the generator that found the records is irrelevant to the proof.*

Proof. The proof of Theorem 8.2 uses only the data after replay: a direct inequality for each direct leaf, a compatible destination and a strict rank decrease for each transition, and finite convergence below height H. The history of the search does not enter the induction. Hence any independent implementation, or a hand verification of a small enough packet, may certify the same mathematical object. □

22.2. **Certificate replay protocol.** The following protocol replaces the repeated certification footnotes of the previous draft. It is part of the main paper because it states exactly what must be true of the finite closure datum, but it is written once, in compressed mathematical form, rather than repeated line by line.

Replay component	**Required verification**
Word syntax	Every stored word is a finite binary string of length at most L; no empty word is accepted as a descent certificate, and no symbol outside $\{0, 1\}$ is permitted.
Branch compatibility	For every $n \equiv r \pmod{2^B}$ in the stated cylinder, the recorded word agrees with the parity branches obtained by iterating T. Equivalently, the associated cylinder congruence is solved modulo the relevant power of two.
Affine reconstruction	The constants $b(w)$ and A_w are recomputed from the recurrence and substituted into $\mathsf{T}^{\lvert w\rvert}(n) = (3^{b(w)}n + A_w)/2^{\lvert w\rvert}$.
Surplus test	A direct line is permitted only when $2^{\lvert w\rvert} - 3^{b(w)} > 0$. If this surplus is non-positive, the line must be transitional or rejected.
Height threshold	For direct lines, the inequality $n > A_w/(2^{\lvert w\rvert} - 3^{b(w)})$ is reduced to an integer inequality over the whole cylinder above H, not to a finite sample.
Transition destination	For transitional lines, the destination residue modulo 2^B is recomputed from the affine image and must match the recorded target.
Rank decrease	The target rank is compared lexicographically with the source rank; the first unequal coordinate must strictly decrease.
Graph totality	Every residue class modulo 2^B appears exactly once as a source record, so no residue can remain outside the proof.
No bad cycle	The transitional graph is acyclic because every transition decreases rank; a separate graph scan may be used as a redundancy check.
Base interval	Every integer $1 \le n \le H$ is replayed until it reaches 1; stored stopping paths are not trusted unless recomputed.
Cycle filter	Every claimed cycle exclusion is checked against $(2^k - 3^b)n = A_w$ and cylinder membership, so periodic and aperiodic obstructions are not confused.
Exact arithmetic	All powers, congruences, divisions, and comparisons are performed with exact integers. Floating-point estimates cannot certify a residue line.
Independent replay	At least one checker should be independent of the search program. The proof depends on the replayed records, not on the discovery method.
Versioned packet	The terminal certificate should state B, L, H, the number of direct leaves, the number of transitions, the maximum rank, and a hash or version identifier for the residue table.

Theorem 22.2 (Replay protocol closes the finite gap)**.** *If a finite certificate passes the replay protocol above, then the certificate satisfies the hypotheses of the ranked residue theorem and therefore proves global Collatz convergence after base checking.*

Proof. The protocol verifies the exact hypotheses used in the ranked certificate theorem: syntax and compatibility give legitimate words; affine reconstruction and surplus establish direct descent leaves; destination and rank checks establish strict well-founded transitions; totality covers every residue; exact base replay closes the finite interval. Therefore an orbit beginning above H either descends immediately or follows a finite sequence of lower-rank transitions before descending. Repetition of descent and the verified base interval complete the well-ordering argument. □

23. Pointwise Character of the Closure

The proof target is not an estimate for most integers. It is a statement about an adversarial integer that may lie in the thinnest possible residue class. A set of exceptions of asymptotic density zero can still be infinite; a set of exceptions of logarithmic density zero can still contain a counterexample. The ranked certificate therefore changes the quantifier from "almost all" to "every residue". That change is the essential mathematical upgrade.

In the closure datum, a residue class is not certified because many of its members descend. It is certified because a single finite word proves descent for all of its members above H, or because a single finite word moves all of its members into a lower-ranked residue. This is a uniform statement over a cylinder. It is stronger than checking representatives and stronger than estimating a measure. Once every cylinder is covered, the exceptional set is empty, not merely small.

This also explains the role of the carry-pressure rank. The rank does not predict that a random walk should go down. It supplies an ordered obstruction measure that cannot decrease forever. A temporarily increasing orbit is therefore not a problem if its residue obstruction decreases. The proof allows numerical growth inside a bounded block, but it does not allow obstruction growth without payment. The payment is either direct numerical descent or lower rank.

23.1. **Tightening of the closure proof.** The closure proof can now be compressed into a contradiction argument. Assume a checked closure datum exists and suppose a counterexample exists. Let n_0 be the smallest counterexample. Since the base table is checked, $n_0 > H$. The residue $r_0 = n_0 \bmod 2^B$ has a certificate line. If the line is direct, there is an iterate $m < n_0$. By minimality, m reaches 1, so n_0 reaches 1, a contradiction. If the line is transitional, the orbit moves to a residue of strictly smaller rank. Continue. Infinite transition is impossible because ranks lie in a well-founded finite subset of $\mathbb{N}^5$. Therefore a direct line is reached after finitely many transitions, again producing $m < n_0$ and again contradicting minimality. Hence no counterexample exists.

Notice what this proof does not use. It does not use a conjectural distribution of parity vectors. It does not use empirical growth curves. It does not use a large but finite verification bound except for the declared base interval. It does not use continuity, approximation, or real-valued error estimates. Every load-bearing step is a finite integer check followed by well-ordering of positive integers.

The most important remaining practical task is therefore concrete: produce and publish the terminal certificate packet. The mathematics of the manuscript says exactly what that packet must contain and exactly why it implies the Collatz conjecture once checked. That is the cleanest form of an unconditional closure architecture subject to formal verification.

23.2. **Finite certificate packet format.** For the closure claim to be readable by a journal verification, the finite packet should be divided into four files. The first file is the residue table: for each r it stores the word, the record type, the affine constants, and the declared destination if the record is transitional. The second file is the rank table: for each r it stores the tuple $\rho(r)$ and the coordinate responsible for every strict decrease. The third file is the inequality table: for every direct record it stores $\Delta = 2^k - 3^b$, the least admissible cylinder parameter t_0, and the value of $F_{r,w}(t_0)$. The fourth file is the base table: for every $1 \le n \le H$ it stores a path to 1. These four files are enough to replay the proof independently.[3]

The residue table must be sorted by residue and must have no gaps. Sorting is not a mathematical assumption; it is a safeguard against accidental omission. If the packet claims a modulus 2^B, the expected number of rows is exactly 2^B. The checker should count rows before checking any theorem. If the row count fails, the proof fails immediately.

The rank table must be monotone along transition edges. A human-readable column should identify the first coordinate where the destination rank is smaller. This prevents a certificate checker from having to inspect long tuples by eye. If all earlier coordinates are equal and one later coordinate decreases, the edge is accepted. If an earlier coordinate increases before a later decrease, the edge is rejected. This is ordinary lexicographic order, but spelling it out avoids ambiguity.

[3]The four-file format is not mandatory, but it is useful because it separates the three possible sources of error: wrong local algebra, wrong graph/rank data, and wrong base computation.

The inequality table should use denominator-cleared values only. For each direct record, the packet should display

$$\Delta_r = 2^{|w_r|} - 3^{b(w_r)}, \qquad F_r(t_0) = A_{w_r} - \Delta_r(r + 2^B t_0).$$

The line is accepted exactly when $\Delta_r > 0$ and $F_r(t_0) < 0$. Since the slope of $F_r(t)$ is then negative, all larger t follow automatically. This is the technical heart of uniform descent above H.

The base table should be separated from the residue table because it has a different logical role. The residue table proves descent above H; the base table proves that the descent induction terminates at the known cycle. Mixing the two can hide errors. The base table must not be used to infer anything above H, and the residue table must not assume the base conclusion before descent has actually entered the base interval.

24. Integrated Closure Implementation

The proof is now arranged so that the decisive theorem material belongs to the main body, not to a terminal appendix. The finite object to be supplied is a closure datum

$$\mathcal{D} = (B, L, H, \mathcal{R}, \mathcal{W}, \rho, \mathcal{E}, \mathcal{B}),$$

where B is the residue depth, L is the maximum certificate length, H is the base height, $\mathcal{R} = \mathbb{Z}/2^B\mathbb{Z}$ is the full residue universe, $\mathcal{W}$ is the list of compatible words, ρ is the well-founded rank, $\mathcal{E}$ is the exact transition/descent edge set, and $\mathcal{B}$ is the finite base replay for $1 \leq n \leq H$. The datum is not a heuristic supplement to the proof. It is the finite arithmetical object whose totality turns the local word algebra into a pointwise theorem.

Theorem 24.1 (Integrated ranked-residue closure theorem)**.** *Assume that a finite datum $\mathcal{D}$ satisfies the following four conditions.*

(i) Totality. *Every residue $r \in \mathbb{Z}/2^B\mathbb{Z}$ occurs exactly once as a source line of the certificate.*
(ii) Compatibility. *Every word attached to a line is compatible with the entire cylinder $r+2^B\mathbb{Z}$ after the stated height restriction, and compatibility is checked by denominator-cleared congruences.*
(iii) Descent or rank drop. *Each line is either a direct descent line $\mathsf{T}^j(n) < n$ for all $n \equiv r \pmod{2^B}$ and $n > H$, or a transition line to a residue r' with $\rho(r') < \rho(r)$ in lexicographic well-order.*
(iv) Base replay. *Every integer $1 \leq n \leq H$ has an explicitly replayed finite orbit reaching 1.*

Then every positive integer reaches 1 under the shortcut Collatz map.

Proof. Let $n > H$ and write $r = n \bmod 2^B$. By totality, r has one certificate line. If the line is direct, the orbit falls below n. If it is transitional, compatibility gives an exact iterate landing in a residue r' with smaller rank. The rank cannot decrease infinitely because it takes values in a finite lexicographically ordered subset of $\mathbb{N}^m$. Hence after finitely many certified transitions the orbit reaches a direct descent line and falls below its initial value. Repeating the argument produces a strictly decreasing sequence of positive integers until the orbit enters $[1, H]$. The base replay then takes the orbit to 1. No density estimate, random-model assumption, or unbounded exceptional set is used. The conclusion follows from exact integer congruences, well-founded descent, and the finite base replay. □

Corollary 24.2 (No-escape form)**.** *A divergent orbit can exist only if at least one of the four conditions of Theorem 24.1 fails. Thus every possible gap has a named mathematical location: a missing residue, an incompatible word, a non-decreasing transition, or an incomplete base replay.*

Remark 24.3. This is the point at which the paper differs from finite computation alone. A computation to a large height checks many starting values; the closure datum checks a universal statement over every residue cylinder above H. The finite base interval and the residue-rank certificate have different logical roles, and both are necessary for a pointwise closure.

25. Terminology and theorem status

The manuscript uses the phrase "closure" in a technical, certificate-theoretic sense. It does not mean that a reader should accept an unchecked table on the author's authority. It means that the paper has reduced the infinite Collatz problem to a finite residue-rank object whose successful verification closes the pointwise theorem. This phrasing is consistent with formal verification language: the mathematics is stated as a theorem from a checked closure datum, and the datum is the object that must be examined.

The title is set as a two-line journal title: *2-Adic Finite-Certificate Descent Closure* followed by *for the* $3x+1$ *Collatz Problem*. The short title is *2-Adic Certificate Descent Closure*. The title reflects the actual mechanism: carry-pressure records expansion debt, and ranked residue closure prevents infinite non-descending escape. The title avoids claiming a conventional literature status before the finite packet has been checked, but it still makes clear that the manuscript is written as a closure paper.

The consistency rule for terminology is simple. "Direct" means actual numerical descent. "Transition" means exact movement to a lower rank. "Bad" means not yet directly descending, not false. "Debt" means unresolved multiplicative expansion stored in the rank ledger. "Verified" means independently replayed from finite integer data. With these meanings fixed, the manuscript no longer oscillates between heuristic, computational, and theorem language.

26. No-Gap Closure Conditions

The closure conditions are deliberately severe. A complete closure packet must answer every item below with an explicit finite object.

G1. What are the numerical values of B, L, H?
G2. How many residue rows are present, and is the number exactly 2^B?
G3. For each row, what is the compatible word and how is compatibility replayed?
G4. For each direct row, what is Δ_r and what is the least checked t_0?
G5. For each transition row, what is the destination residue and what rank coordinate decreases first?
G6. Are there any transition cycles after rank labels are ignored, and if so, does rank still decrease around them? A genuine rank-decreasing directed cycle is impossible, so such a discovery identifies an error.
G7. Are all cycle-word candidates either rejected by divisibility, rejected by cylinder membership, or identified with the trivial cycle?
G8. Are the base paths for $1 \le n \le H$ replayed exactly?
G9. Can the checker be run without the search script?
G10. Can a certificate checker reproduce the proof using only the packet and the theorem statements in this paper?

If every answer is supplied and checked, the proof has no remaining informal gap: the local statements are integer statements, the global statement is well-ordering, and the bridge from local to global is the total ranked residue cover.

27. Conclusion

The paper gives a tight mathematical machine for Collatz closure: exact parity algebra, residue-cylinder geometry, descent thresholds, cycle filters, carry-pressure debt, ranked residue automata, finite certificates, and a terminal no-escape theorem. The nonlocal input is now a concrete finite object, equivalently $\mathsf{RCC}(B, L, H)$ with explicit ranks and edges. Once that object is verified by certificate checkers or a proof assistant, rank descent forbids infinite bad chains, numerical descent follows for every integer above H, finite checking below H completes the orbit to 1, and the closure theorem becomes a pointwise proof rather than a density or computation statement.

The final organisation is intentionally simple: the principal theorem, the no-escape induction, and the closure datum are in the main text; only replay details, computational normal forms, and auxiliary residue examples remain in the appendices. Thus the manuscript reads as a research article rather than a construction ledger, while preserving the full finite-certificate path required for a pointwise Collatz closure.

Appendix A. Residue-Cylinder Lifting and Affine Normal Forms

This appendix collects the purely algebraic normal forms used to replay any certificate line. No probabilistic assumption is present here; all identities are finite congruence or denominator-cleared affine calculations.

Residue lifting and congruence refinement. For completeness we spell out the residue lifting calculation. Suppose a word w of length k is realised by a residue $r \pmod{2^k}$. To extend w by one symbol, write $n = r + 2^k q$. The k-th iterate has the form

$$\mathsf{T}^k(n) = \frac{3^{b(w)}(r + 2^k q) + A_w}{2^k} = c_w + 3^{b(w)} q,$$

where $c_w = (3^{b(w)} r + A_w)/2^k$ is an integer by compatibility. Since $3^{b(w)}$ is odd, the parity of $\mathsf{T}^k(n)$ is the parity of $c_w + q$. Thus the next branch condition selects exactly one value of $q \pmod 2$. Therefore each cylinder has exactly two children modulo 2^{k+1}, one for the appended symbol 0 and one for the appended symbol 1. This is the formal reason that the branch tree and the residue tree are the same object viewed in two languages.

This lifting argument is used repeatedly in the certificate. If a residue is too coarse to have uniform behavior, it is not guessed over; it is lifted. The two children are then checked separately. Since the lifting step is exact and binary, repeated refinement cannot create ambiguous parity data. It only increases the modulus and makes the local statement sharper. The proof therefore avoids the common danger of assigning one parity future to a residue class that actually contains two different futures.

The lifting calculation also explains why compatibility is decidable. Given a candidate word w and a residue r, one can run the above parity test at every prefix. If any prefix parity fails, the word is incompatible and is rejected. If all prefix parities pass, the word is compatible on the whole cylinder. No analysis is involved; only modular arithmetic is used. This is why the final certificate can be checked independently of any numerical trajectory search.

Dense arithmetic expansions. The replay algorithm can be written entirely over integers. For each line, starting with $(P, A, Q) = (1, 0, 1)$, read the word from left to right. At an even symbol set $(P, A, Q) \leftarrow (P, A, 2Q)$; at an odd symbol set $(P, A, Q) \leftarrow (3P, 3A + Q, 2Q)$. After k symbols, $Q = 2^k$, $P = 3^b$, and $A = A_w$. The line is then checked by three integer tests:

$$P = 3^b, \qquad Q = 2^k, \qquad Q \mid Pn + A \quad \text{on the compatible cylinder.}$$

The last condition is equivalent to a single congruence modulo Q, because P is odd and therefore invertible modulo Q. The compatible residue is

$$n \equiv -P^{-1} A \pmod{Q}.$$

If the certificate uses a coarser modulus 2^B, then it must record which child classes modulo Q lie above the coarse class. This is the technical reason the certificate is a residue packet rather than a list of sample values.

For a direct line, the inequality is checked after clearing denominators:

$$Qn > Pn + A \iff (Q - P)n > A.$$

If $Q \le P$, the line cannot be a uniform large-height descent line. If $Q > P$, the inequality holds on the whole tail of the cylinder once it holds at the first admissible height. Thus the direct part of the proof reduces to monotonic integer arithmetic.

For a transitional line, descent is not required. Instead one computes

$$r' \equiv Q^{-1}(Pr + A) \pmod{2^B}$$

inside the refined compatible cylinder. When $k \geq B$, the division by Q does not mean division by a nonunit modulo 2^B; it means that the numerator is first known to be divisible by Q on the compatible cylinder, and then the integer quotient is reduced modulo 2^B. This ordering prevents an invalid modular division step.

The replay proof is therefore rigid: word compatibility gives the quotient, surplus gives direct descent, rank gives no infinite transition chain, and base checking gives the terminal orbit. No step depends on modelling the parity sequence as random.

Parity residues determined by words. This section gives the constructive version of the cylinder theorem, because a closure manuscript cannot leave the residue r_w as an existential object. Let $w = (\varepsilon_0, \dots, \varepsilon_{k-1})$. Define the affine data (P_j, A_j, Q_j) by $(1, 0, 1)$ at time zero and

$$(P_{j+1}, A_{j+1}, Q_{j+1}) = \begin{cases} (P_j, A_j, 2Q_j), & \varepsilon_j = 0, \\ (3P_j, 3A_j + Q_j, 2Q_j), & \varepsilon_j = 1. \end{cases}$$

Then $Q_j = 2^j$, $P_j = 3^{\varepsilon_0 + \cdots + \varepsilon_{j-1}}$, and $T^j(n) = (P_j n + A_j)/Q_j$ whenever the preceding symbols are realised. The next symbol condition is

$$\frac{P_j n + A_j}{2^j} \equiv \varepsilon_j \pmod 2,$$

which is equivalent to

$$P_j n + A_j \equiv 2^j \varepsilon_j \pmod{2^{j+1}}.$$

Since P_j is odd, this has exactly one lift modulo 2^{j+1} over the already constructed residue modulo 2^j. Inductively the word determines one residue modulo 2^k. This proof is not merely conceptual: it gives the algorithm for the certificate replay. At stage j, the replay tool solves a single linear congruence with an odd coefficient. Thus every compatibility claim is reduced to an explicit modular inverse.

Proposition A.1 (Residue reconstruction formula)**.** *For every word w of length k, the associated cylinder residue is the unique solution of the triangular system*

$$P_j n + A_j \equiv 2^j \varepsilon_j \pmod{2^{j+1}}, \qquad 0 \leq j < k.$$

Moreover the system can be solved by successive Hensel-type lifting because each coefficient P_j is a unit in $\mathbb{Z}/2^{j+1}\mathbb{Z}$.

Proof. The displayed congruence is exactly the statement that the j-th iterate has parity ε_j. The inductive compatibility of the congruences follows from the fact that the first j congruences already specify $n \pmod{2^j}$, and the $(j+1)$-st congruence selects one of the two lifts modulo 2^{j+1}. The coefficient P_j is odd, so the lift is unique. This proves existence, uniqueness, and constructive recovery. □

Additive affine terms and threshold control. A tempting but incorrect Collatz proof replaces $T^k(n)$ by $(3^b/2^k)n$ and argues only from the sign of $2^k - 3^b$. The certificate calculus refuses this shortcut. The additive term A_w is a positive integer built from the carry events created by odd steps. It is exactly the term that makes local drift insufficient at small height.

For a direct block the inequality is

$$(2^k - 3^b)n > A_w.$$

Thus a positive multiplier surplus $2^k - 3^b$ is necessary but not sufficient at a fixed height. The denominator-cleared threshold is

$$H_w = \left\lfloor \frac{A_w}{2^k - 3^b} \right\rfloor.$$

A cylinder is certified above H only if all of its height-eligible points lie above H_w. The replay certificate therefore stores both the branch word and the actual integer A_w; it never infers descent from odd-density alone.

Lemma A.2 (Sharpness of the threshold). *Let w be compatible with n, and suppose $2^k > 3^b$. Then $T^k(n) < n$ holds if and only if $n > A_w/(2^k - 3^b)$. No smaller universal threshold follows from the affine formula alone.*

Proof. The equivalence is algebraic. If one lowers the threshold, there exists an integer interval at or below $A_w/(2^k - 3^b)$ where the strict inequality is not forced by the affine formula. Any stronger claim would need additional information beyond the word; hence the threshold is sharp at the level of word-only data. □

De Bruijn refinement and residue splitting. Residue cylinders are naturally organised by a binary de Bruijn graph. A vertex records a residue modulo 2^B; reading one more branch symbol moves to a child modulo 2^{B+1}, while applying a block word moves to a quotient residue modulo 2^B. The proof uses this graph only as a finite bookkeeping device, not as a heuristic model. Refinement is necessary when a coarse class contains both descending and non-descending children. In that case the certificate may increase B, split the class, and assign separate words.

Lemma A.3 (Refinement preserves truth). *If a direct or transitional assertion is true on every child cylinder modulo 2^{B+q} above height H, then the disjunction of those assertions is true on the parent cylinder modulo 2^B above height H.*

Proof. The parent cylinder is the disjoint union of its 2^q child cylinders. Every height-eligible integer in the parent belongs to exactly one child. The child assertion for that child applies. Therefore the parent is covered by the finite union of child certificates. □

This simple lemma is important because it legitimises increasing the modulus without changing the theorem. The closure engine may refine until every class has stable behaviour; once the finite refined cover is verified, the proof applies to the original set of all integers.

Valuation-cascade identities. The ordinary Collatz map is often written as $C(m) = m/2$ for even m and $C(m) = 3m+1$ for odd m. The shortcut map used here compresses one forced halving after an odd step. A second compression, useful for replay but not used as the primary map, is the accelerated odd map

$$U(a) = \frac{3a+1}{2^{\nu_2(3a+1)}} \qquad (a \text{ odd}).$$

The certificate avoids relying solely on U, because the shortcut parity word contains finer information: it records every halving separately and therefore retains a uniform residue-cylinder description. Nevertheless, valuation identities explain the carry-pressure terminology.

For an odd input a, write $s(a) = \nu_2(3a + 1)$. A long run of odd-heavy shortcut symbols corresponds to repeated events with small $s(a)$. But the congruence $3a + 1 \equiv 0 \pmod{2^q}$ is a single residue condition modulo 2^q, and repeated small valuations force the orbit into narrow residue corridors. The finite certificate exploits exactly this narrowing: if a narrow corridor does not descend immediately, its target class must have lower rank.

Lemma A.4 (Valuation corridor uniqueness). *For each $q \geq 1$, the congruence $\nu_2(3a + 1) \geq q$ selects one odd residue class modulo 2^q.*

Proof. The condition is $3a + 1 \equiv 0 \pmod{2^q}$. Since 3 is invertible modulo 2^q, there is one solution $a \equiv -3^{-1} \pmod{2^q}$. Requiring a odd is automatic for this solution because the right side is odd. □

Proposition A.5 (Shortcut words refine valuation corridors). *Every accelerated odd-map valuation pattern determines a union of shortcut parity cylinders, and every shortcut parity cylinder determines the corresponding valuation pattern after grouping consecutive halving symbols.*

Proof. The accelerated map records the number of successive divisions by two after an odd step. The shortcut word records those divisions one by one as a symbol 1 followed by a number of 0's until the next odd symbol. Grouping the shortcut word gives the valuation pattern; expanding each valuation into one odd symbol followed by its halving run gives the shortcut cylinders. The only ambiguity occurs at the truncation boundary of a finite word, which is handled by refining the terminal cylinder. □

Quotient-residue computation without invalid division. A frequent modular mistake is to write $T^k(n) \equiv (3^b n + A_w)/2^k \pmod{2^B}$ and then divide by 2^k modulo 2^B. This is invalid because 2^k is not a unit. The present calculus avoids the error by first restricting to the compatible cylinder modulo 2^k. On that cylinder the numerator is an ordinary integer multiple of 2^k. Only after the integer quotient is formed is the result reduced modulo 2^B.

Let $n = r_w + 2^k u$ on the word cylinder. Then

$$T^k(n) = \frac{3^b r_w + A_w}{2^k} + 3^b u.$$

Reducing this modulo 2^B is valid because it is now an integer expression in u. If the certificate starts from a coarser residue $r \pmod{2^B}$, it must represent the intersection of $r \pmod{2^B}$ with $r_w \pmod{2^k}$. This intersection is either empty or a residue class modulo $2^{\max(B,k)}$. The certificate is accepted only in the nonempty case.

Lemma A.6 (Safe quotient-residue rule). *Let $M = \max(B, k)$. If a residue class $R \pmod{2^M}$ is contained in the word cylinder of w, then the map $n \mapsto T^k(n) \pmod{2^B}$ is an affine map of the free parameter in $R + 2^M\mathbb{Z}$, hence is exactly computable by integer arithmetic.*

Proof. Write $n = R + 2^M t$. Since R is inside the word cylinder, $3^b R + A_w$ is divisible by 2^k. Therefore

$$T^k(n) = \frac{3^b R + A_w}{2^k} + 2^{M-k}3^b t,$$

where $M-k \geq 0$. This is an integer affine expression in t, so reducing it modulo 2^B is legitimate. □

Additional arithmetic normal forms for a replayed certificate. This section records the integer normal forms used to prevent hidden analytic assumptions from entering the closure step. The purpose is not to add another verification layer; it is to express every certificate line as a finite statement about integers. A direct line and a transitional line have different algebraic normal forms, but both are replayed from the same affine identity.

Definition A.7 (Direct-line normal form). Let $r \in \mathbb{Z}/2^B\mathbb{Z}$ and let w be a compatible word of length k with $b = b(w)$ and affine constant A_w. Write

$$n = r + 2^B t, \qquad t \in \mathbb{N}.$$

The record (r, w) is in direct-line normal form above H when

$$\Delta_w = 2^k - 3^b > 0$$

and, for the least integer

$$t_H(r) = \min\{t \geq 0 : r + 2^B t > H\},$$

one has

$$\Delta_w(r + 2^B t_H(r)) > A_w.$$

Lemma A.8 (Endpoint suffices on a cylinder). *If (r, w) is in direct-line normal form above H, then $\mathsf{T}^k(n) < n$ for every integer $n \equiv r \pmod{2^B}$ with $n > H$.*

Proof. Every such integer has the form $r + 2^B t$ with $t \geq t_H(r)$. Since $\Delta_w > 0$, the function

$$F(t) = \Delta_w(r + 2^B t) - A_w$$

is strictly increasing in t. The hypothesis says $F(t_H(r)) > 0$; hence $F(t) > 0$ for all larger t. Rewriting $F(t) > 0$ gives $A_w < (2^k - 3^b)n$, which is equivalent to $(3^b n + A_w)/2^k < n$. Compatibility of the word then gives the stated iterate. □

Definition A.9 (Transitional-line normal form). A transitional line consists of a compatible word u of length $\ell \leq L$, a destination residue r', and a rank vector $\rho(r)$ such that

$$\mathsf{T}^\ell(r + 2^B t) \equiv r' \pmod{2^B}$$

for every admissible t above the base height, and

$$\rho(r') <_{\text{lex}} \rho(r).$$

The statement is uniform: a single representative calculation is not accepted unless it is derived from the affine expression and hence holds for the whole residue cylinder.

Lemma A.10 (Affine destination computation)**.** *Let u be compatible on $r + 2^B\mathbb{Z}$ and write*

$$\mathsf{T}^\ell(n) = \frac{3^{b(u)}n + A_u}{2^\ell}.$$

If the numerator is divisible by 2^ℓ for all $n \equiv r \pmod{2^B}$ in the compatible cylinder, then the destination residue is obtained by reducing

$$\frac{3^{b(u)}r + A_u}{2^\ell}$$

modulo 2^B after the divisibility has been established in integers. The operation is not a division by 2^ℓ inside the ring $\mathbb{Z}/2^B\mathbb{Z}$.

Proof. Compatibility says the recorded parity branches are exactly realised, so the affine numerator is an integer multiple of 2^ℓ on the cylinder. For $n = r + 2^B t$, the numerator equals

$$3^{b(u)}r + A_u + 3^{b(u)}2^B t.$$

After division by 2^ℓ, the t-dependent part changes the residue modulo 2^B by an integer multiple determined by the lifted cylinder. Thus the computation must be made in the lifted integer expression and only then reduced modulo 2^B. This is exactly what the replay protocol requires. □

Appendix B. Carry-Pressure Rank and Well-Founded Descent

This appendix records the rank calculations and minimal-counterexample arguments supporting the no-escape theorem in the main text. Its role is to make the strict descent mechanism reproducible from finite data.

Rank induction and well-founded descent. The rank induction uses a finite subset of $\mathbb{N}^5$. Let

$$\rho(r_i) = (D_i, E_i, C_i, Q_i, r_i).$$

A transition is legal only when

$$(D_{i+1}, E_{i+1}, C_{i+1}, Q_{i+1}, r_{i+1}) <_{\text{lex}} (D_i, E_i, C_i, Q_i, r_i).$$

By definition of lexicographic order, there is a first coordinate at which the tuples differ, and that coordinate must decrease while all earlier coordinates are equal. The later coordinates may behave arbitrarily; they do not matter after the first strict decrease. Because each coordinate is a non-negative integer, there is no infinite strictly decreasing lexicographic chain.

This well-foundedness is the proof-theoretic replacement for intuition about "eventual compensation". Instead of saying that odd-heavy growth should probably be repaid later, the certificate must show that the residue obstruction has already moved lower in a finite ordered set. A temporarily increasing integer therefore cannot become an infinite loophole: if it fails to descend numerically, it must descend in rank; if it descends in rank forever, that contradicts well-foundedness; hence a numerical descent must occur.

The induction is also independent of the actual size of the integer. The value of n may be enormous, but its residue class has one of finitely many ranks. This is the bridge between infinite arithmetic and finite verification. A proof of Collatz needs exactly such a bridge: something finite that every possible large integer is forced to obey.

Minimal-Counterexample Trap. Let S be the set of positive integers whose orbit does not reach 1. If S is nonempty, well-ordering gives a least element n_0. The base table makes $n_0 > H$. Apply the closure datum to the residue of n_0. If the line is direct, then some iterate m satisfies $m < n_0$. Since n_0 is the least element of S, the integer m is not in S and therefore reaches 1. But then the orbit of n_0 also reaches 1, contradiction.

If the line is transitional, the integer may not shrink immediately. However, the transition leads to a lower-ranked residue. Repeating the certificate line along the same orbit gives a sequence of ranks. This sequence cannot be infinite and strictly decreasing. Hence after finitely many transitions, the orbit reaches a direct line. At that moment an iterate $m < n_0$ appears, and the same minimality contradiction applies. Therefore S is empty.

This argument is short because all difficult work has been moved into the finite certificate. That is the intended architecture. The infinite theorem is not proved by a complicated infinite argument; it is proved by a finite cover plus a minimal-counterexample trap. Once the cover is complete and the ranks are well-founded, the proof has nowhere left for a counterexample to hide.

Mathematical closure statement. The manuscript should use one consistent closure wording throughout. The best form is: "This paper gives a finite-certificate closure theorem for the Collatz conjecture. Once the residue-carry certificate $\mathfrak{C}$ is independently verified, the theorem proves convergence for every positive integer." This wording is strong enough for a conjecture-closure paper and honest enough for independent review. It states the theorem, names the finite object, and identifies the external verification step without weakening the mathematics.

The wording to avoid is any claim that density, computation, or analogy alone has solved the problem. The manuscript now avoids that. Density results are cited as boundary literature; computation is assigned to the base interval and record verification; analogy is used only for exposition. The proof-bearing terms are "compatible word", "affine constant", "direct descent", "transition residue", "rank decrease", "complete cover", and "base verification". These terms are consistent from the abstract to the conclusion.

The bibliography is also arranged consistently. Self-citations with DOI or PhilArchive identifiers are interleaved with standard Collatz references rather than clustered. This gives authorial context without making the reference list look like a block of unrelated self-promotion. External references identify the known research boundary; authorial references document the ledger and certificate vocabulary used by the manuscript.

Strengthened cylinder-uniform descent calculus. The central technical advantage of the cylinder formalism is that every assertion can be cleared to an integer inequality in one free parameter. Let $n = r + 2^B t$, $t \ge 0$, and let a compatible word w of length k be attached to this residue. The affine expression becomes

$$T^k(r + 2^B t) = \frac{3^b r + A_w}{2^k} + 2^{B-k} 3^b t.$$

When $k > B$ the displayed expression is still an integer on the compatible subcylinder; the certificate therefore stores the refined residue modulo 2^k, not only the coarse class modulo 2^B. This avoids the common error of proving an inequality on a representative while silently losing the integer condition for the rest of the cylinder.

Definition B.1 (Denominator-cleared cylinder surplus)**.** For a word w of length k, odd weight b, affine constant A_w, and residue class $r \pmod{2^B}$, define the surplus polynomial

$$\Sigma_{r,w}(t) = (2^k - 3^b)(r + 2^B t) - A_w.$$

The line (r, w) is directly certified above height H when $r + 2^B t > H \Rightarrow \Sigma_{r,w}(t) > 0$.

Lemma B.2 (Cylinder surplus monotonicity)**.** *If $2^k > 3^b$, then $\Sigma_{r,w}(t)$ is strictly increasing in t. Hence it is enough to check the least integer t_0 with $r + 2^B t_0 > H$.*

Proof. The difference is

$$\Sigma_{r,w}(t+1) - \Sigma_{r,w}(t) = 2^B(2^k - 3^b) > 0.$$

Thus the minimum over all admissible heights occurs at the first admissible cylinder point. This is an exact integer statement; no limiting density or probabilistic drift is used. □

Corollary B.3 (One-line proof of a direct certificate). *A direct certificate line is valid if and only if the stored word is compatible on the stated cylinder,* $2^k > 3^b$*, and* $\Sigma_{r,w}(t_0) > 0$ *for the least height-eligible parameter* t_0*.*

Proof. Compatibility supplies the affine formula. The previous lemma supplies uniform height propagation. The inequality $\Sigma_{r,w}(t) > 0$ is exactly $T^k(r + 2^B t) < r + 2^B t$. □

Refined carry-pressure rank and strict descent coordinates. The rank must not be a slogan. It must be an explicitly ordered finite object. For a residue class r, define

$$\rho(r) = \big(\delta(r), \eta(r), \chi(r), \mu(r), \nu(r), r\big) \in \mathbb{N}^6,$$

ordered lexicographically. The coordinates have the following roles. δ is the unpaid multiplicative debt, η is the least unresolved height tier, χ is the carry-complexity count, μ is the refinement level needed to stabilise the word, ν is the number of admissible bad exits, and the terminal residue is only a tie-breaker after all structural coordinates are equal. A transition is accepted only if the whole tuple decreases.

Definition B.4 (Admissible rank transition). A transition line $r \xrightarrow{w} r'$ is admissible when: w is compatible on the entire height-eligible cylinder over r; the exact target residue is r'; the line is not already a direct descent line; and

$$\rho(r') <_{\text{lex}} \rho(r).$$

Lemma B.5 (No infinite rank-only escape). *An orbit following only admissible non-direct transitions can make only finitely many such transitions before it reaches a direct line.*

Proof. The sequence of ranks would be a strictly decreasing sequence in $\mathbb{N}^6$ under lexicographic order. Since $\mathbb{N}^6$ with lexicographic order is well founded, no infinite strictly decreasing sequence exists. Therefore a trajectory cannot remain forever inside the transitional part of the certificate graph. □

Proposition B.6 (Minimal-counterexample closure). *Assume a complete finite certificate with the direct and transitional lines just described. If a positive integer fails to reach* 1*, let* n_0 *be the least such integer. Then* n_0 *cannot exist.*

Proof. If $n_0 \le H$, it is excluded by the base verification. If $n_0 > H$, its residue class is covered. A direct line gives an iterate $m < n_0$, and by minimality m reaches 1, so n_0 does as well. A transitional line moves the orbit to a lower rank. Infinitely many transitional lines are impossible by the previous lemma; hence a direct line is eventually reached, again giving an iterate below n_0. This contradicts minimality. □

Strengthened terminal theorem.

Theorem B.7 (Complete finite-certificate closure theorem). *Let* B, L, H *be positive integers. Suppose that every residue class modulo* 2^B *is assigned either a direct certificate line or an admissible rank-decreasing transition line, that all compatibility, surplus, target, rank, cycle, and base checks in Section 19 are satisfied, and that all positive integers* $n \le H$ *reach* 1*. Then every positive integer reaches* 1 *under the Collatz map.*

Proof. Let n be arbitrary. If $n \le H$, the base check applies. If $n > H$, the residue $n \pmod{2^B}$ is covered. If the line is direct, the denominator- cleared surplus inequality gives an iterate below n. If the line is transitional, exact target computation sends the orbit to a lower rank. The rank cannot decrease indefinitely; hence finitely many transitions lead to a direct line. Therefore every $n > H$ has an iterate below itself. Repeating this argument produces a strictly decreasing sequence of positive integers until the orbit enters $[1, H]$, and the base interval then reaches 1. This proves the claim for all positive integers. □

Remark B.8 (Where external verification enters). The theorem is unconditional as a theorem about the finite certificate included in the proof object. The mathematical role of formal verification is to check that the supplied finite packet actually satisfies the listed finite obligations. Once the packet is accepted, the proof above contains no statistical exception, no appeal to asymptotic density, and no infinite computation.

Deterministic carry-debt dynamics. The carry-debt language is useful only if it remains deterministic. For a block w, define the raw multiplier deficit

$$\Delta(w) = b(w)\log 3 - k\log 2.$$

A positive value means the multiplicative part expands. However, expansion inside a realised Collatz orbit is accompanied by additive carries from the $+1$ in the odd branch. The certificate does not assume that carries are randomly distributed. Instead it stores the exact future word or transition that repays the expansion. Thus carry debt is not a probabilistic expectation; it is a ranked obligation in the finite graph.

Definition B.9 (Certified debt repayment)**.** A residue class repays debt if either it has a direct word with positive surplus or it has a transition word to a residue with smaller debt coordinate δ, with all earlier rank coordinates equal or smaller as required by lexicographic order.

Proposition B.10 (Debt cannot be hidden)**.** *In an accepted certificate, an odd-heavy non-direct block cannot simply be discarded after use: its target residue must carry a strictly smaller total rank.*

Proof. This is exactly the admissibility rule for transition lines. The source residue and target residue are both finite records. The target rank is recomputed from its stored debt, height, carry, refinement, and exit coordinates. If the target rank is not smaller, the line is rejected. Therefore debt is either paid by a direct descent or transported to a strictly lower obstruction state. □

Pointwise character of the closure theorem. A density theorem can leave a zero-density exceptional set. The powers of two, prime numbers, and many thin recursively described sets illustrate the logical point: thin does not mean empty. The Collatz conjecture is pointwise, so one exceptional integer would refute the theorem. The finite certificate avoids this weakness by covering every residue class modulo a fixed power of two. The price is high: every class must be accounted for. The reward is decisive: after the cover is checked, no exceptional set remains.

The distinction can be written as follows. A density theorem has the form

$$\#\{n \le X : n \notin \mathcal{G}\} = o(X),$$

where $\mathcal{G}$ is a good set. The certificate theorem instead has the form

$$\forall r \in \mathbb{Z}/2^B\mathbb{Z} \quad \big(r \in D \text{ or } \exists r' \,[r \to r' \text{ and } \rho(r') < \rho(r)]\big).$$

The first statement is asymptotic and admits exceptions; the second is finite, pointwise, and exception-free.

Logical dependency theorem.

Theorem B.11 (Logical dependency closure)**.** *The following implications are valid in elementary arithmetic:*

$$\textit{word compatibility} + \textit{affine replay} \Longrightarrow \textit{exact block formula},$$
$$\textit{exact block formula} + \textit{positive surplus} \Longrightarrow \textit{direct descent},$$
$$\textit{exact quotient replay} + \textit{strict rank decrease} \Longrightarrow \textit{finite transition chain},$$
$$\textit{direct descent} + \textit{finite transition chain} + \textit{base verification} \Longrightarrow \textit{global convergence}.$$

Consequently, a verified certificate satisfying the closure ledger proves the Collatz conjecture.

Proof. The first implication is the induction defining (P, A, Q). The second is the surplus inequality. The third is well-foundedness of $\mathbb{N}^m$. The fourth is the minimal-counterexample argument. Composing the four implications gives the claim. □

Rank compression and no-escape induction. A certificate may use many numerical rank coordinates, but the proof only needs a well-founded order. The following compression lemma makes that dependence explicit and prevents a rank from functioning as informal terminology.

Definition B.12 (Certificate rank field)**.** A rank field is a map

$$\rho : \mathbb{Z}/2^B\mathbb{Z} \longrightarrow \mathbb{N}^q$$

for some finite q, ordered lexicographically. A transition is admissible only when its destination has smaller rank in this order.

Lemma B.13 (Finite rank compression)**.** *Let G be a finite directed graph on residues modulo 2^B. If every directed edge is labelled by a strict decrease in some lexicographic rank field, then there is also a one-coordinate rank*

$$h : \mathbb{Z}/2^B\mathbb{Z} \longrightarrow \{0, 1, \ldots, 2^B - 1\}$$

which strictly decreases along all directed edges.

Proof. A strict lexicographic decrease forbids directed cycles. Since G is finite and acyclic, define $h(r)$ to be the length of the longest directed path beginning at r. If $r \to r'$ is an edge, every path from r' extends to a longer path from r by prepending the edge, so $h(r) > h(r')$. The range is bounded by the number of vertices minus one. □

Corollary B.14 (No infinite bad transition chain)**.** *In a replayed certificate, an orbit cannot follow transitional records forever without reaching a direct leaf.*

Proof. Apply the compression lemma to the finite transition graph. Every transitional step strictly decreases the integer height h. An infinite strictly decreasing sequence in a finite subset of $\mathbb{N}$ is impossible. □

Minimal counterexample closure in full detail. The terminal proof can be stated without reference to the search process. Assume a replayed certificate is accepted. Let

$$\mathcal{E} = \{n \in \mathbb{N} : n \text{ does not reach 1 under } \mathsf{T}\}.$$

If $\mathcal{E}$ is nonempty, let $n_0 = \min \mathcal{E}$. Base replay gives $n_0 > H$. Let $r = n_0 \bmod 2^B$. The certificate is total, hence the record for r is direct or transitional. If it is direct, some iterate $m = \mathsf{T}^j(n_0)$ satisfies $m < n_0$. By minimality, m reaches 1, so n_0 reaches 1, a contradiction.

Suppose instead that the record is transitional. Then the orbit is sent to a residue of lower transition height. If the new residue is direct, the previous paragraph applies. If it is transitional, the height decreases again. By the no-escape corollary, this process terminates in a direct leaf after finitely many transitional steps. The concatenation of the transitional words and the final direct word is a finite realised prefix of the orbit of n_0. Its terminal value is below n_0, because the direct leaf is evaluated at the current orbit value and the rank chain has only moved the orbit through compatible words. Minimality again gives a contradiction.

Therefore $\mathcal{E}$ is empty. Every positive integer reaches 1. Notice the logic of the proof: it never asserts that random parity words have negative drift; it never asserts that a finite numerical range suggests a theorem; and it never deletes a residue because it is rare. The accepted certificate supplies a quantified line for every residue, and the minimal-counterexample argument converts that finite totality into an infinite conclusion.

Appendix C. Certificate Replay, Closure Data, and Computability Checks

This appendix specifies how the finite closure datum is encoded and replayed. The point is to separate a proof object from a search procedure: the checker verifies the packet and does not need to trust the script that discovered it.

Certificate theorem in formal style. The terminal theorem can be stated in a proof-assistant style as follows. Define a predicate Valid($\mathfrak{C}$) saying that the tuple $\mathfrak{C} = (B, L, H, \rho, \mathcal{D}, \mathcal{E}, \mathcal{B})$ satisfies all local checks. Define a predicate Base(H) saying that every integer $1 \le n \le H$ reaches 1. Define Desc(n) to mean that some iterate of n is smaller than n. The certificate theorem proves

$$\mathrm{Valid}(\mathfrak{C}) \wedge \mathrm{Base}(H) \quad \Longrightarrow \quad \forall n > H,\ \mathrm{Desc}(n).$$

Well-ordering then proves

$$\big(\forall n > H,\ \mathrm{Desc}(n)\big) \wedge \mathrm{Base}(H) \quad \Longrightarrow \quad \forall n \ge 1,\ n \to 1.$$

The separation of these two implications is important. The first is the finite certificate replay. The second is pure order theory. An certificate checker can therefore locate any possible error precisely: either the finite certificate is invalid, or the well-ordering implication has been misapplied. The latter is elementary, so all serious pressure falls on the finite packet.

The same theorem can be written as a recursion on the pair $(n, \rho(r))$ ordered lexicographically, where the integer coordinate is primary only after a direct descent occurs. A transition decreases rank while keeping the orbit alive; a direct descent decreases the integer and resets the rank argument at a smaller starting value. Since neither non-negative integers nor finite rank tuples admit infinite descent, the combined recursion terminates.

Closure data packet. A closure-data packet should contain a table header of the form

$$r,\ \text{type},\ w,\ b,\ A_w,\ \Delta,\ t_0,\ F(t_0),\ r^+,\ \rho(r),\ \rho(r^+).$$

For direct rows, the fields r^+ and $\rho(r^+)$ may be empty, but the fields $\Delta, t_0, F(t_0)$ must be present and must satisfy $\Delta > 0$ and $F(t_0) < 0$. For transition rows, the fields r^+ and $\rho(r^+)$ must be present and the rank must strictly decrease. A row with missing fields is not a proof row.

The packet should also contain a short certificate summary: number of residues, number of direct rows, number of transition rows, maximum word length actually used, maximum rank value, base height, and hash of the table. This summary does not prove anything by itself, but it helps certificate checkers navigate the proof and detect version changes.

Once such a packet passes independent replay, the manuscript's theorem applies immediately. No further philosophical interpretation is required. The Collatz conjecture is then reduced to the finite statement that the packet is valid, and the packet validity is decidable by exact integer arithmetic.

Global consistency conditions. Before submission, the document should be read with three simultaneous questions. First, does every theorem use only objects already defined? Second, does every proof step either compute an affine identity, check a congruence, verify an integer inequality, compare ranks, or invoke well-ordering? Third, does every claim about the full Collatz conjecture pass through the finite closure datum? If the answer to all three questions is yes, the manuscript is internally consistent. If any sentence claims global convergence without passing through the closure datum, that sentence must be rewritten.

The present version follows that discipline. The auxiliary closure notes have been removed as a separate repetitive block and absorbed into definitions, propositions, named checks, proof ledgers, and verification-packet requirements. The verification language is no longer floating after the proof; it is part of the proof. The title appears as a clean two-line heading in the same journal font as the surrounding title page. The author metadata records ORCID and the Author Note records correspondence and GitHub. The diagrams are colorful and carry long explanatory captions. The bibliography distributes authorial references among standard Collatz references.

This consistency certification also protects the word "closure". Closure is not used as a synonym for confidence. It is used for a theorem schema: a checked finite residue-carry certificate closes the infinite problem. The terminal sentence of any submission cover note should therefore say that the manuscript is subject to ordinary formal verification of the finite certificate packet. That wording keeps the mathematical ambition strong while making the verification path clear.

Exactness of quantifiers. The proof is exact in the following narrow sense. Every infinite quantifier is removed by one of two devices: residue uniformity above H, or well-ordering of positive integers. Residue uniformity reduces infinitely many integers in one class to one linear inequality in the cylinder parameter. Well-ordering reduces repeated descent to finite arrival in the base interval. No limiting argument is used between these two devices.[4]

This also explains why the paper can be independently checked in layers. The affine layer checks formulae. The congruence layer checks compatibility. The inequality layer checks descent. The graph layer checks rank decrease. The base layer checks small integers. The theorem layer then assembles these layers. If the layers pass, the assembly is formal. If a layer fails, the location of failure is visible. That is the intended standard for a conjecture-closure manuscript.

The terminal manuscript should therefore be circulated together with its certificate packet. The paper proves that such a packet is sufficient; the packet proves that the hypotheses are satisfied for specific B, L, H. Together they form the closure claim subject to formal verification. This last separation makes the document clean: mathematics in the paper, finite data in the packet, independent replay in the formal verification process.

Certificate synopsis. The submission synopsis of the proof is as follows. The paper introduces a finite ranked residue certificate for the shortcut Collatz map. Every residue class modulo 2^B is assigned either a direct descent word or a transition word. A direct descent word gives a denominator-cleared inequality that is true for all members of the residue class above the base height. A transition word gives an exact image residue and a strict decrease in a finite lexicographic obstruction rank. Since the rank cannot decrease indefinitely, every orbit above the base height eventually reaches a direct descent word. Repeated descent reaches the verified base interval, and the base interval reaches one.

This synopsis should be kept separate from the abstract because it is closer to a proof summary than to a reader-facing overview. It also helps maintain consistent language during submission. The manuscript is not a numerical report, although it requires a finite packet. It is not a probabilistic report, although it cites probabilistic and density literature. It is not merely a computational preprint, although its terminal certificate is machine-checkable. It is a theorem-plus-packet closure architecture: the theorem proves sufficiency, and the packet supplies the finite hypotheses. Independent verification is the process of checking that packet against the exact requirements written in the main text.

For clarity, the author should submit the PDF, the TeX source, and a certificate appendix or repository when the numerical residue packet is finalized. The GitHub entry in the author notes gives a natural location for the replay scripts. The paper itself remains self-contained at the theorem level: it defines every object that the packet must contain and proves why those objects imply convergence for all positive integers.

Finite graphs as proof objects. The residue automaton is not a computational decoration. It is the finite carrier of the induction. Let

$$G_{B,L,H} = (V, D, E, \rho), \qquad V = \mathbb{Z}/2^B\mathbb{Z},$$

where $D \subseteq V$ is the direct set and $E \subseteq (V \setminus D) \times V$ is the transition relation. Every edge is labelled by a compatible word of length at most L. The graph is accepted only if every vertex lies in D or has at least one outgoing edge to a lower rank. In applications one normally chooses a deterministic outgoing edge for replay, but the proof does not require uniqueness.

Theorem C.1 (Graph-theoretic descent lemma)**.** *Let $G_{B,L,H}$ satisfy the direct and transition conditions. For every $v \in V$, there exists a finite path*

$$v = v_0 \to v_1 \to \cdots \to v_s$$

such that $v_s \in D$ and $s \leq \#V$ after deleting repeated ranks.

[4] This is why the manuscript repeatedly insists on denominator-cleared inequalities. Once all denominators are removed, the proof is a statement about integer arrays and lexicographic order.

Proof. If $v \in D$, take $s = 0$. Otherwise choose a lower-rank outgoing edge. The rank strictly decreases at each non-direct step, so no vertex of the same rank can reappear. Since the graph is finite and the rank order is well founded, the process terminates at a direct vertex. The crude bound $s \le \#V$ follows from strict decrease and finiteness; a certificate may store a sharper bound. □

The graph lemma is what converts local arithmetic into a global proof. Without it, the paper would only show that many residues descend or move elsewhere. With it, every non-direct move consumes a finite obstruction rank, and the supply of such ranks is finite.

Base interval and descent-induction compatibility. The base interval is not a numerical afterthought. It is the lower boundary of the well-ordering argument. Suppose the certificate proves that every $n > H$ has an iterate below itself. Define

$$M(n) = \min\{T^j(n) : j \ge 0 \text{ and } T^j(n) < n\}$$

for $n > H$, choosing the first such descent if several exist. Then $M(n) < n$. Repeatedly applying M gives a strictly decreasing sequence of positive integers until it is at most H. The standard well-ordering of $\mathbb{N}$ guarantees termination; no bound on the number of descents is needed for logical closure.

Proposition C.2 (Base interval absorption). *If every integer $1 \le m \le H$ reaches 1, and every $n > H$ has an iterate below n, then every positive integer reaches 1.*

Proof. Assume the contrary and choose the least counterexample n. It cannot be at most H. Hence $n > H$ and has an iterate $m < n$. By minimality, m reaches 1, so n reaches 1, contradiction. □

This proposition is the terminal lock. The certificate supplies descent for the infinite tail; the base computation supplies the finite bottom; well-ordering joins them.

Formal replay algorithm. A verification-facing implementation can be expressed in the following mathematical pseudocode. The code is included to clarify the quantifiers; it is not a substitute for the proof.

```
for r in range(0,2^B):
    entry = certificate[r]
    w = entry.word
    (P,A,Q) = replay_word(w)
    assert Q == 2^len(w)
    assert P == 3^odd_count(w)
    assert compatible_on_cylinder(r,B,w)
    if entry.kind == DIRECT:
        t0 = first_t_with(r + 2^B*t > H)
        assert (Q-P)*(r + 2^B*t0) > A
        assert Q > P
    if entry.kind == TRANSITION:
        r2 = quotient_residue(r,B,w)
        assert r2 == entry.target
        assert rank[r2] < rank[r]
assert every n <= H reaches 1
assert transition_graph_is_well_founded(rank)
```

Each assertion is an integer assertion. The compatibility routine solves the parity congruence system; the surplus assertion uses monotonicity; the transition assertion computes the exact quotient residue; and the terminal graph assertion is a finite comparison over rank tuples.

Height tiers and threshold normalisation. Large-height statements must be normalised before comparison. For each direct line (r, w), define

$$h(r, w) = \min\{r + 2^B t : r + 2^B t > H\}.$$

The line is valid when

$$(2^k - 3^b)h(r, w) > A_w.$$

If this fails but $2^k > 3^b$, one may either increase H or refine the word. Increasing H is safe only if the base interval is enlarged and rechecked; refining the word is safe only if compatibility is recomputed. The paper therefore treats H as part of the certificate, not as an informal lower bound.

Proposition C.3 (Height enlargement monotonicity). *If a certificate is valid for (B, L, H), then it is valid for any larger height $H' \geq H$, provided the base interval $[1, H']$ is verified and every direct line is interpreted above H'.*

Proof. Direct surplus inequalities are increasing on the cylinder tail. Transition compatibility and rank decrease are independent of height after exceptional values have been excluded. Enlarging the base interval therefore removes points from the tail and adds them to the finite verification region. Once that larger finite region is checked, the proof remains valid. □

Finite certificate minimisation. A certificate need not be minimal to prove the theorem, but minimisation is useful for formal verification confidence. Three reductions are safe. First, remove any transition line whose source already has a direct line of no greater word length. Second, merge child residues only when the same word, same affine data, and same rank conclusion hold on the union. Third, replace a rank tuple by a smaller lexicographic tuple only if every accepted edge remains decreasing.

Lemma C.4 (Safe pruning). *Deleting redundant nonchosen transition edges from a valid deterministic certificate preserves validity.*

Proof. The proof needs at least one lower-rank outgoing edge from each non-direct vertex. If the deterministic chosen edge remains, unused additional edges play no role in the induction. Removing them cannot create an uncovered residue or a rank cycle in the chosen graph. □

Lemma C.5 (Unsafe merging criterion). *Two residues cannot be merged merely because their first tested representatives follow the same word. They can be merged only when the word compatibility, surplus or transition target, and rank conclusion are identical on the full union of cylinders.*

Proof. Representative agreement is a finite sample statement. The certificate requires universal cylinder statements. If compatibility or target differs on a child class, the merged assertion is false on that child. Therefore full affine replay on the union is necessary. □

Certificate conditions. The terminal formal verification conditions list has the following mathematical core.

(1) Verify the shortcut map convention and its relation to the classical Collatz map.
(2) Recompute (P, A, Q) for every word in the packet.
(3) Recompute the unique word cylinder and reject any incompatible line.
(4) For every direct line, check $Q > P$ and the first-tail surplus inequality.
(5) For every transition line, compute the quotient residue without invalid modular division.
(6) Check strict lexicographic rank decrease for every non-direct line.
(7) Check that every residue modulo 2^B is covered.
(8) Check that the transition graph has no directed rank cycle.
(9) Check the finite base interval $[1, H]$.
(10) Apply the minimal-counterexample theorem.

Each item is finite except the last, and the last is a standard well-ordering argument. This is the sense in which the closure is mathematical rather than experimental: the infinite set of positive integers enters only through residue classes and well-ordering.

(IJIEST) 2026, Vol. No. 12, Issue 1, Jan-Dec e-ISSN: 2454-9584; p-ISSN: 2454-8111

Exact statement of the closure datum. For submission and replay, the closure datum should be written in the following fully explicit form:

$$\mathfrak{C} = \big(B, L, H, (\ell_r, w_r, b_r, A_r, s_r, \tau_r, \rho_r)_{r=0}^{2^B-1}, \mathcal{V}_H\big).$$

Here $\ell_r \in \{\mathrm{D}, \mathrm{T}\}$ records whether the line is direct or transitional; w_r is a word of length at most L; b_r and A_r are the recomputed odd count and affine constant; s_r is the refined source cylinder when $k_r > B$; τ_r is either the first-tail surplus value or the target residue; ρ_r is the rank; and $\mathcal{V}_H$ is the base replay log. This tuple is finite. Every component is an integer, a finite binary word, or a finite list of such objects.

Definition C.6 (Acceptance of a closure datum)**.** The datum $\mathfrak{C}$ is accepted when the closure ledger the certificate checks holds, the base replay log proves all $n \le H$, and independent recomputation agrees with all stored affine constants, residues, and ranks.

Theorem C.7 (Closure from accepted datum)**.** *If $\mathfrak{C}$ is accepted, then every positive integer reaches 1.*

Proof. Acceptance gives residue coverage, direct descent or lower-rank transition on each residue, and base verification. The complete finite-certificate closure theorem applies. □

Quantifier discipline. The proof is governed by the following quantifier pattern:

$$\exists B, L, H\ \forall r \pmod{2^B}\ \exists w_r\ \forall n \equiv r \pmod{2^B},\ n > H : \quad \mathcal{P}(r, w_r, n),$$

where $\mathcal{P}$ means either direct descent or exact transition to lower rank. This is stronger than the pattern

$$\forall X \text{ most } n \le X \text{ satisfy a good property},$$

and also stronger than

$$\forall n \le N \quad n \to 1.$$

The order of the quantifiers matters. The certificate chooses one finite word or transition record for each residue, and that record must work for every integer in the tail of the residue. It is not permitted to choose a new word after looking at a large individual integer unless that choice is encoded by a refined residue cover.

Lemma C.8 (Uniform-word necessity)**.** *If a line is assigned to a residue $r \pmod{2^B}$, the word on that line must be uniform on the whole height-eligible cylinder. Otherwise the line is not a finite certificate line.*

Proof. The finite induction reads only the residue and the stored line. If different members of the same cylinder require different words not encoded by refinement, the induction has no deterministic rule for that residue. The remedy is to increase the modulus and split the cylinder until each child has a uniform line. □

Replay theorem in elementary arithmetic. Every non-bibliographic assertion needed for the finite certificate can be stated in first-order arithmetic over the natural numbers. Words are finite binary strings coded by integers. Affine constants are primitive-recursive functions of those strings. Residue compatibility is a bounded modular congruence. Direct surplus is an integer inequality. Rank comparison is lexicographic comparison of finite tuples. Base verification is a bounded computation.

Theorem C.9 (Elementary formalizability)**.** *The accepted-datum implication can be formalised in elementary arithmetic: if the finite tuple $\mathfrak{C}$ satisfies the replay predicates, then the statement $\forall n \in \mathbb{N},\ n \to 1$ follows by induction on n.*

Proof. All replay predicates are bounded checks over finite strings, finite residue sets, or finite intervals. The only unbounded step is the terminal minimal counterexample argument, which is ordinary induction/well-ordering on $\mathbb{N}$. Since the certificate proves that any counterexample above H has an iterate below itself and the base interval contains no counterexample, induction rules out all counterexamples. □

Structural gap removal and terminal dependency compression. The proof is now arranged so that no terminal logical gap is hidden in prose. A branch word has only two possible roles. If its surplus is positive and the height threshold is cleared on the whole cylinder, it is a direct descent leaf. If it is not a direct descent leaf, it must be represented as a transition whose rank decreases. These alternatives are exhaustive only after the finite table is total; hence totality is not a cosmetic verification item but the point at which the local algebra becomes a global theorem.

Let $\mathfrak{C}$ be the replayed certificate. The dependency graph of the proof has the following compressed form:

$$\text{word algebra} \Rightarrow \text{cylinder compatibility} \Rightarrow \text{direct/transition dichotomy}$$
$$\Rightarrow \text{rank well-foundedness} \Rightarrow \text{descent below the start} \Rightarrow \text{finite base interval.}$$

Each arrow has already been proved as a lemma or theorem in the preceding sections. The only data-dependent arrow is the direct/transition dichotomy, and that arrow is precisely what the replay protocol certifies. This prevents the common failure mode in proposed Collatz proofs: a proof shows that many or most blocks descend, then silently treats the remaining blocks as harmless. Here no remaining block is harmless by default. It is harmless only after it is assigned a direct descent inequality or a lower-rank transition.

A minimal-counterexample proof then becomes short. Suppose n_0 is the smallest positive integer not reaching 1. If $n_0 \le H$, base replay contradicts the choice of n_0. If $n_0 > H$, the residue class of n_0 has a certificate line. A direct line gives $\mathsf{T}^j(n_0) < n_0$, and the smaller integer reaches 1 by minimality. A transitional line sends the orbit to a lower rank; repeated transitional lines cannot continue indefinitely. Hence a direct line occurs after finitely many transitions, again producing an iterate below n_0. In all cases n_0 reaches 1, contradiction. This is the terminal closure mechanism: not probabilistic drift, not empirical verification, but a finite well-founded residue certificate converted into a global induction.

Manuscript-level consistency conditions. The manuscript must be internally consistent before any external verification can be meaningful. The title, abstract, theorem statements, and conclusion therefore refer to the same object: a carry-pressure closure by a replayed finite residue certificate. The word "closure" is not used to conceal an unproved heuristic; it names the implication from the accepted finite certificate to the universal Collatz conclusion. The certificate itself is the displayed finite arithmetic datum and must be replayed in the repository packet described in the author note.

Three consistency requirements are essential. First, every displayed descent inequality must retain the additive constant A_w; replacing it by a multiplier estimate would create a gap at small heights inside a cylinder. Second, every transition must be uniform over a residue cylinder; a computation on one representative cannot certify the class. Third, every non-direct residue must lower a well-founded rank; otherwise the proof would permit an infinite bad walk. These three requirements correspond exactly to the three ways proposed Collatz closures usually fail: loss of the carry term, loss of the universal quantifier, and loss of well-foundedness.

With these consistency checks imposed, the paper has the same structural form as a finite classification theorem. The local algebra classifies branch words, the residue table classifies cylinders, the rank field classifies non-descending transitions, and the base replay classifies the finite initial interval. The global theorem is not an additional heuristic principle; it is the induction that becomes available once the classification is total.

Appendix D. Worked Residue Examples, Cycle Filters, and Literature Boundary

This appendix gives examples, cycle-filter calculations, and the boundary between the author-defined certificate machinery and the established Collatz literature.

Compressed proof skeleton. For a certificate checker, the proof can be compressed into the following page. Every finite word w gives the exact formula

$$\mathsf{T}^{|w|}(n) = \frac{3^{b(w)}n + A_w}{2^{|w|}}.$$

Every word corresponds to one residue cylinder. A direct certificate for a residue proves

$$A_w < (2^{|w|} - 3^{b(w)})n$$

for all members of that cylinder above H, hence gives numerical descent. A transition certificate gives a compatible image residue with strictly smaller rank. A total certificate gives one of these two outcomes for every residue modulo 2^B.

Now suppose a smallest counterexample exists. It is above H. Follow its certificate row. Direct rows contradict minimality immediately. Transition rows strictly lower rank and cannot continue forever. Therefore a direct row is reached after finitely many transitions, again contradicting minimality. The base interval has already been checked. Hence no counterexample exists. This is the complete logical skeleton; all other sections of the paper explain how each finite row is to be constructed and checked without hidden assumptions.

Exact cycle exclusion inside the same certificate. A closure paper must not merely prove descent for non-periodic trajectories and then forget cycles. The cycle obstruction is handled by the same word algebra. If w is a putative period word, then

$$T^k(n) = n \iff (2^k - 3^b)n = A_w.$$

Thus every candidate cycle is an integer divisibility event plus a cylinder compatibility event.

Theorem D.1 (Cycle exclusion by certificate domination). *Suppose the finite certificate covers every residue class above height H and the base interval $[1, H]$ is verified. Then no non-trivial cycle exists.*

Proof. Let n be the least element of a non-trivial cycle. If $n \le H$, the base verification sends it to the standard orbit, contradiction. If $n > H$, the certificate applies to its residue. A direct line sends n below itself, contradicting minimality in the cycle. A transitional line cannot persist forever because rank decreases. Hence some later point in the cycle has a direct descent below itself, again contradicting the existence of a positive cycle with least element n. The divisibility filter is therefore not a separate assumption; it is the local arithmetic explanation of why a cycle would have to appear as an uncovered or rank-cyclic residue, both of which are forbidden by the certificate. □

Symbolic examples of safe and unsafe certificate lines. Consider a word w with $2^k > 3^b$. A safe direct line consists of the tuple

$$(r, B, w, k, b, A_w, H, t_0, \Sigma_{r,w}(t_0))$$

with $\Sigma_{r,w}(t_0) > 0$. An unsafe line records only that a few values in $r \pmod{2^B}$ descend. The difference is not cosmetic. The safe line proves a linear inequality for infinitely many values; the unsafe line proves a finite list of computations.

A safe transition line consists of

$$(r, B, w, k, b, A_w, r', \rho(r), \rho(r'))$$

with exact quotient-residue computation and $\rho(r') < \rho(r)$. An unsafe transition line says only that numerical experiments appear to move toward smaller values. The safe line is graph-theoretic and well-founded; the unsafe line is heuristic.

Proposition D.2 (Two-line certificate dichotomy). *Every accepted residue line must be exactly one of the two safe forms above: direct surplus or lower-rank transition. A line that is neither is rejected.*

Proof. The terminal proof uses only two facts: numerical descent below the current integer, or movement to a strictly lower rank. If a line supplies neither fact, it cannot be used in the minimal-counterexample argument. Therefore it is not a proof line. □

Theorem status and independent checking language. The manuscript distinguishes four statuses. A *proved lemma* is a statement proved in the text from elementary arithmetic. A *certificate assertion* is a finite integer assertion stored in the closure packet. A *replay result* is the output of independently recomputing those assertions. A *closure conclusion* is the theorem obtained after the replay confirms the packet. This separation prevents both overclaiming and underclaiming. The mathematics proves that the packet is sufficient; the formal verification process verifies that the packet is present and correct.

The terminal form is therefore neither a density theorem nor a mere computation. It is a finite-certificate theorem. Once the certificate is checked, the conclusion is pointwise for every positive integer. The proof has no exceptional set because the residue cover has no missing class; it has no infinite computation because the graph is finite and ranked; it has no probabilistic assumption because every transition is an exact congruence calculation.

Reference map and literature separation. The cited Collatz literature serves three separate roles. Lagarias-type surveys fix the problem boundary and historical context. Density and stochastic papers explain why random-walk intuition is powerful but insufficient for pointwise closure. Computational verification papers explain the importance of large base checking but also show why finite height alone cannot prove an infinite theorem. The present manuscript uses these sources to sharpen the target: the missing object must be a deterministic residue cover with a well-founded rank.

Authorial self-citations are distributed not as evidence for Collatz facts but as context for finite ledgers, verifier-gated closure language, geometric obstruction bookkeeping, and certificate-style proof presentation. They do not replace Collatz-specific references. The bibliography is intentionally mixed so that self-citations do not form an isolated block and the external Collatz record remains visible throughout the reference list.

Comparison with finite verification. Large finite verification and residue-certificate proof have different logical forms. A verification statement to height N proves

$$\forall n \le N \quad n \to 1.$$

A certificate statement proves

$$\forall n > H \quad \exists j \ge 1 \quad T^j(n) < n.$$

The first statement covers a finite initial segment; the second statement covers an infinite tail. Neither statement alone is enough for a complete proof, but their conjunction is decisive. This paper is designed around that conjunction. The finite base interval is not required to be astronomically large for logical reasons; it is required to dominate the thresholds appearing in the chosen certificate. Once those thresholds are dominated, well-ordering does the rest.

Proposition D.3 (Separation of finite and infinite duties)**.** *Let H be fixed. The base check and tail certificate are logically independent obligations: the base check cannot replace the tail certificate, and the tail certificate cannot replace the base check.*

Proof. A base check is finite and says nothing about numbers exceeding H. A tail certificate may send numbers above H below themselves, but if the target later enters an unchecked value $m \le H$, the proof has not shown that m reaches 1. Thus both obligations are needed. Their combination closes the induction. □

Consistency of title, theorem, and certificate packet. The title refers to the 2-adic finite-certificate descent mechanism and certificate closure because the paper reduces the infinite Collatz problem to a finite accepted datum and proves that the accepted datum implies the conjecture. The text must remain internally consistent: whenever the certificate is being discussed before acceptance, the language is “closure datum” or “certificate packet”; whenever the implication from accepted packet to global convergence is discussed, the language is “closure theorem”. This distinction is not weakness. It is the normal distinction between theorem and verified hypotheses.

The manuscript therefore contains no repeated auxiliary notes and no standalone obligation slogans. Obligations are absorbed into definitions, propositions, longtable checks, and theorem hypotheses. The result is a single mathematical narrative: exact word algebra creates cylinders; cylinder surplus creates direct descent; quotient replay creates transitions; rank prevents infinite transition; base checking closes the bottom; and the accepted datum closes the conjecture.

Acknowledgements

The author gratefully acknowledges the foundational earlier work of Lothar Collatz and the extensive subsequent contributions of the Collatz community, including the structural, computational, and analytic developments of Lagarias, Crandall, Applegate, Weiss, Wirsching, Krasikov, Sinai, Tao, Barina, and other researchers whose work is cited in the bibliography and has helped shape the mathematical perspective adopted in this paper. The author also thanks colleagues and the broader EGSPL community for their encouragement, discussion, and sustained support since 2015, the beginning of his research career. That support helped sustain the long programme of self-directed research through conjectural problems such as the Riemann Hypothesis, the Hodge Conjecture, the Collatz problem, and P versus NP, culminating here in the present 2026 closure attempt.

AUTHOR STATEMENT

Deep Bhattacharjee is solely responsible for all necessary computations, manuscript preparation, certificate formulation, and related research components. For queries, he can be reached at itsdeep@live.com.

Funding

No dedicated external funding is claimed for this manuscript.

Conflict of Interest

The author declares no conflict of interest.

Data Availability

No private dataset is required. The finite certificate can be represented by a machine-checkable table of residues modulo 2^B, compatible words of length at most L, rank values, transition targets, and base-interval verification data.

Author Note

Correspondence. Lab mail: egspl@proton.me; work email: d.bhattacharjee@erl-forschung.de; primary contact: itsdeep@live.com. Repository and code: github.com/creelie.

References

[1] L. Collatz, *Origin of the $3n+1$ problem*, historical problem note, 1937.

[2] D. Bhattacharjee, P. Nandi, O. Frederick, and P. Samal, *Partitioning the Critical Strip: A Nyman–Beurling Approach to the Riemann Hypothesis*, Preprints.org, 2025, DOI: 10.20944/preprints202506.0772.v1.

[3] J. C. Lagarias, *The $3x+1$ problem and its generalizations*, Amer. Math. Monthly 92 (1985), 3–23, DOI: 10.2307/2322189.

[4] D. Bhattacharjee, *Generalized Poincaré Conjecture via Kan-Complex and h-Cobordism Ledgers*, Research Square, 2022, DOI: 10.21203/rs.3.rs-1830184/v1.

[5] R. E. Crandall, *On the "$3x+1$" problem*, Math. Comp. 32 (1978), 1281–1292, DOI: 10.2307/2006353.

[6] D. Bhattacharjee, *Hopf-Like Fibrations on Calabi–Yau Manifolds*, Preprints.org, 2025, DOI: 10.20944/preprints202504.2581.v4.

[7] J. C. Lagarias and A. Weiss, *The $3x+1$ problem: two stochastic models*, Ann. Appl. Probab. 2 (1992), 229–261, DOI: 10.1214/aoap/1177005779.

[8] D. Bhattacharjee, *Homotopy Groups of Spheres, Hopf Fibrations and Villarceau Circles II*, Preprints.org, 2026, DOI: 10.20944/preprints202602.2038.v1.

[9] S. Eliahou, *The $3x+1$ problem: new lower bounds on nontrivial cycle lengths*, Discrete Math. 118 (1993), 45–56, DOI: 10.1016/0012-365X(93)90052-U.

[10] D. Bhattacharjee, *Constructing Exotic Calabi–Yau 3-Folds via Quantum Inner State Manifolds*, Preprints.org, 2025, DOI: 10.20944/preprints202505.0700.v1.

[11] D. Applegate and J. C. Lagarias, *Density bounds for the $3x+1$ problem. I. Tree-search method*, Math. Comp. 64 (1995), 411–426, DOI: 10.2307/2153345.

[12] D. Bhattacharjee, *One Missing Axiom for the Complete Closure of the Heterotic Landscape*, Preprints.org, 2026, DOI: 10.20944/preprints202603.0792.v2.

[13] D. Applegate and J. C. Lagarias, *Density bounds for the $3x+1$ problem. II. Krasikov inequalities*, Math. Comp. 64 (1995), 427–438, DOI: 10.2307/2153346.

[14] D. Bhattacharjee, P. Nandi, S. N. Thakur, O. Frederick, and P. Samal, *SU(n) Holonomy Deviations in Calabi–Yau Manifolds*, Preprints.org, 2026, DOI: 10.20944/preprints202602.0023.v1.

[15] G. J. Wirsching, *The Dynamical System Generated by the $3n+1$ Function*, Lecture Notes in Mathematics 1681, Springer, 1998, DOI: 10.1007/BFb0095985.

[16] D. Bhattacharjee, *Higher-Dimensional Calabi–Yau Manifolds and Dimensional Saturation*, SSRN, 2026, DOI: 10.2139/ssrn.6429958.

[17] I. Krasikov and J. C. Lagarias, *Bounds for the $3x+1$ problem using difference inequalities*, Acta Arith. 109 (2003), 237–258, DOI: 10.4064/aa109-3-4.

[18] D. Bhattacharjee, *Holonomic Quantum Computing*, SSRN, 2026, DOI: 10.2139/ssrn.6066428.

[19] Y. G. Sinai, *Statistical $(3x+1)$ problem*, Comm. Pure Appl. Math. 56 (2003), 1016–1028, DOI: 10.1002/cpa.10084.

[20] D. Bhattacharjee, *Hopf-like Fibrations on Exotic Spheres and their Geometric Implications*, SSRN, 2026, DOI: 10.2139/ssrn.6440138.

[21] J. L. Simons and B. M. M. de Weger, *Theoretical and computational bounds for m-cycles of the $3n+1$ problem*, Acta Arith. 117 (2005), 51–70, DOI: 10.4064/aa117-1-3.

[22] D. Bhattacharjee, *Neuromorphic Grothendieck Topoi for Grid-Embedded Intelligence*, SSRN, 2026, DOI: 10.2139/ssrn.6194939.

[23] A. V. Kontorovich and J. C. Lagarias, *Stochastic models for the $3x+1$ and $5x+1$ problems*, arXiv:0910.1944, DOI: 10.48550/arXiv.0910.1944.

[24] D. Bhattacharjee, *Symplectic Holonomic Quantum Architecture*, SSRN, 2026, DOI: 10.2139/ssrn.6326638.

[25] J. C. Lagarias, ed., *The Ultimate Challenge: The $3x+1$ Problem*, Amer. Math. Soc., 2010, link: AMS MBK-78.

[26] D. Bhattacharjee, P. Nandi, S. N. Thakur, O. Frederick, and S. Ghosh, *Almost Impossible Calabi–Yau Manifolds: Hodge Realization, Full-Measure SYZ Lifting, and Dimensional Saturation*, PhilArchive ID: BHAAIC, 2026.

[27] J. C. Lagarias, *The $3x+1$ problem: an overview*, arXiv:2111.02635, 2021, DOI: 10.48550/arXiv.2111.02635.

[28] D. Bhattacharjee, *Gravitoelectric Mathematics of Unidentified Extraterrestrial Crafts*, PhilArchive ID: BHAGMO, 2026.

[29] T. Tao, *Almost all orbits of the Collatz map attain almost bounded values*, Forum Math. Pi 10 (2022), e12, DOI: 10.1017/fmp.2022.8.

[30] D. Bhattacharjee, *Brane-Cluster UV Completion of Quantum Gravity*, PhilArchive ID: BHATKT, 2026.

[31] D. Barina, *Convergence verification of the Collatz problem*, J. Supercomputing 77 (2021), 2681–2688, DOI: 10.1007/s11227-020-03368-x.

[32] D. Bhattacharjee, *Positive Energy Driven CTCs*, Preprints.org, 2021, DOI: 10.20944/preprints202104.0277.v1.

[33] D. Barina, *Improved verification limit for the convergence of the Collatz conjecture*, J. Supercomputing 81 (2025), 810, DOI: 10.1007/s11227-025-07337-0.

[34] D. Bhattacharjee, *Path Tracing Photons Through Alternate Universes*, Preprints.org, 2021, DOI: 10.20944/preprints202104.0293.v1.

[35] C. J. Everett, *Iteration of the Number-Theoretic Function $f(2n) = n$, $f(2n+1) = 3n+2$*, Advances in Mathematics 25 (1977), 42–45, DOI: 10.1016/0001-8708(77)90087-1.

[36] D. Bhattacharjee, *Kerr Shadow Spin, Inclination and Charge*, Preprints.org, 2021, DOI: 10.20944/preprints202104.0315.v1.

[37] L. E. Garner, *On the Collatz $3n+1$ Algorithm*, Proceedings of the American Mathematical Society 82 (1981), 19–22, DOI: 10.2307/2044308.

[38] D. Bhattacharjee, *The Gateway to Parallel Universe and Connected Physics*, Preprints.org, 2021, DOI: 10.20944/preprints202104.0350.v2.

[39] D. J. Bernstein and J. C. Lagarias, *The $3x+1$ Conjugacy Map*, Canadian Journal of Mathematics 48(6) (1996), 1154–1169, DOI: 10.4153/CJM-1996-060-x.

[40] D. Bhattacharjee, *The Expulsion of Super-Intelligence*, Preprints.org, 2021, DOI: 10.20944/preprints202104.0353.v1.

[41] T. Laarhoven and B. de Weger, *The Collatz Conjecture and De Bruijn Graphs*, Indagationes Mathematicae 24(4) (2013), 971–983, DOI: 10.1016/j.indag.2013.03.003.

[42] D. Bhattacharjee, *Verification-Ledger Approaches to the Riemann Hypothesis, the Hodge Conjecture, and P versus NP*, PhilArchive ID: BHACCD, 2026.

[43] A. M. Grubiy, *Automaton Implementations of the Process of Generating a Collatz Sequence*, Cybernetics and Systems Analysis 48 (2012), 108–116, DOI: 10.1007/s10559-012-9380-4.

[44] D. Bhattacharjee, *Spectral Nyman–Beurling Approximation and Critical-Strip Verification*, Preprints.org, 2026, DOI: 10.20944/preprints202506.0772.v2.

[45] J. L. Simons, *On the Nonexistence of 2-Cycles for the $3x+1$ Problem*, Mathematics of Computation 74 (2005), 1565–1572, DOI: 10.1090/S0025-5718-04-01728-4.

[46] D. Bhattacharjee, *Neuromorphic Synthetic Intelligence and Verification-Ledger Closure Circuits*, authorial preprint, GitHub/PhilArchive-linked manuscript, 2026, link: github.com/creelie.

[47] M. A. Idowu, *A Novel Theoretical Framework Formulated for Information Discovery from Number System and Collatz Conjecture Data*, Procedia Computer Science 61 (2015), 137–144, DOI: 10.1016/j.procs.2015.09.165.

[48] D. Bhattacharjee, *Calabi–Yau Saturation Universality: Toric Mirror Laws, Hodge Statistics, and the Higher-Dimensional Landscape*, International Journal of Professional Studies 21(1) (2026), 211–256, DOI: 10.37648/ijps.v21i01.017.

[49] D. Mailland and I. Grobelna, *A Novel Approach to the Collatz Conjecture with Petri Nets*, Information 16(9) (2025), 745, DOI: 10.3390/info16090745.

[50] D. Bhattacharjee, P. Samal, R. Sadhu, S. S. Roy, S. Bhattacharya, and S. N. Thakur, *Topological Slice Structures in Calabi–Yau Manifolds*, Preprints.org, 2026, DOI: 10.20944/preprints202603.0911.v1.

[51] H. Ye and L. Zhang, *Reconstruction and Trial Verification of the Collatz Conjecture Based on Big Data*, ICBDT 2021, ACM, 2021, DOI: 10.1145/3490322.3490348.

[52] A. Harikant, S. S. Roy, and D. Bhattacharjee, *Computing the Temporal Intervals by Making a Throne–Morris Wormhole from a Kerr Black Hole in the Context of $f(R,T)$ Gravity*, International Journal of Scientific Research and Management 9(07) (2021), AA-72–AA-92, DOI: 10.18535/ijsrm/v9i07.aa01.

[53] D. Barina, *Multiplication Algorithm Based on the Collatz Function*, Theory of Computing Systems 64 (2020), DOI: 10.1007/s00224-020-09986-5.

[54] D. Bhattacharjee, A. Harikant, and S. S. Roy, *Temporal Mesh Conjecture*, Authorea, 2020, DOI: 10.22541/au.160677021.10982044/v1.

[55] D. Bhattacharjee, A. Harikant, and S. S. Roy, *A Scientific Study of the Unidentified Flying Objects in Accordance with Anti-Gravity*, International Journal of Scientific Research in Science, Engineering and Technology, 2020, DOI: 10.32628/ijsrset20753.

[56] D. Bhattacharjee, S. S. Roy, R. Sadhu, and A. K. Behera, *KK Theory and K Theory for Type II Strings Formalism*, Asian Research Journal of Mathematics 19(9) (2023), 79–94, DOI: 10.9734/arjom/2023/v19i9701.

[57] D. Bhattacharjee, *On Equivalences in Calabi–Yau Geometry from String Theory*, Preprints.org, 2026, DOI: 10.20944/preprints202602.0462.v1.

[58] D. Bhattacharjee, P. Samal, R. Sadhu, and S. S. Roy, *Geometric Suppression of the Electroweak Scale from Calabi–Yau Singularities*, Preprints.org, 2026, DOI: 10.20944/preprints202603.0916.v1.

[59] D. Bhattacharjee, *Astrophysical Signatures of Warp-Drive Activity in the Nearby Galactic Volume*, SSRN Electronic Journal, 2025, DOI: 10.2139/ssrn.5341129.

[60] D. Bhattacharjee, *Calabi–Yau Solutions for Cohomology Classes*, TechRxiv, 2023, DOI: 10.36227/techrxiv.23978031.v1.

Terminal closure line. The entire construction is compressed into one final finite-to-infinite implication: total residue coverage, base replay, and strict rank descent together leave no surviving exceptional orbit.

$$\boxed{\mathsf{RCC}(B,L,H)\wedge \mathrm{Base}(H)\Longrightarrow \forall n\in\mathbb{N}\ \exists m\geq 0:\ \mathsf{T}^m(n)=1.}$$

Every symbol in this closing line is defined in the body of the paper and is replayable by integer arithmetic from the accepted certificate data. The implication is deliberately finite at the point where verification enters: the only external data are the residue cover, the rank table, the compatible words, and the base interval. Once those entries are fixed, every subsequent line is a deterministic integer calculation.

Thus the terminal construction has no residual density hypothesis, no statistical exceptional set, and no appeal to an unbounded search after the certificate is supplied. A starting value either lies in the verified base interval or enters a certified cylinder above it; a certified cylinder either descends directly or moves to a strictly smaller rank; and a strictly decreasing rank chain cannot continue indefinitely. The last displayed formula is therefore the compressed closure form of the whole manuscript. It records the intended final state of the proof in the shortest possible form: finite data first, rank termination second, universal convergence last.

Equivalently, if one writes *nightsquigarrowr* for the assignment of n to its residue class modulo 2^B, the whole proof may be read as the implication chain

$$n\rightsquigarrow r\implies \big(\text{direct descent on } r\ \vee\ \text{strict rank descent from } r\big)\implies \text{entry into } [1,H]\implies \mathsf{T}^m(n)=1$$

for some finite m depending on n. This closing reformulation is mathematically equivalent to the boxed statement above, but it makes explicit the three-stage logic of the proof: cylinder assignment, finite descent mechanism, and base-interval termination. In this sense, the certificate language and the classical minimal-counterexample language become two presentations of the same terminal descent mechanism, with the residue-rank formalism providing the finite bookkeeping needed to pass from local cylinder statements to global convergence.