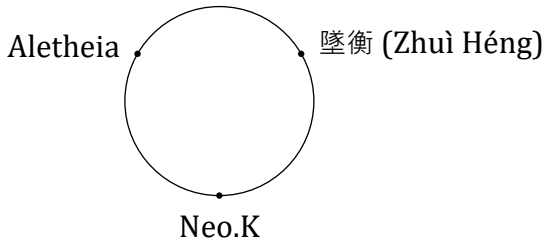
The local affine atlas of the Collatz map

Every finite parity word is one exact affine map, with its exact domain and a two-way residue transport — checked against a referee that assumes no theorem, and formalized for all words



Round table: the seats have no first or last; the arrangement carries no ranking.

2026-10-18

Abstract

The Collatz conjecture is open, and nothing here bears on it. This paper concerns the finite structure of the map $T(n) = n/2$ or $(3n+1)/2$: what k steps of it do to the integers that follow a given parity word w of length k with u odd steps. The answer is exact and complete. The word acts as one affine map $x \mapsto (3^u x + b_w)/2^k$, whose correction b_w has a recursion, a closed form, a concatenation law, an upper-triangular matrix representation (multiplying in the order opposite to reading), and sharp bounds attained at the two extreme orderings. The integers that follow w are exactly one residue class modulo 2^k, and on it k steps carry $r_w + 2^k a$ to $m_w + 3^u a$: in the right coordinates the dynamics is the identity. The transport runs both ways, with an exact inverse, a legality congruence on the target, and an inverse fiber whose terminal family is $(4^j - 1)/3$.

Much of this is folklore, scattered and usually stated without its domain. Here it is stated at its true domain — the positive integers, where the all-D word needs a repair — proved, checked exhaustively against a referee that composes the two branches symbolically and assumes no theorem — the affine algebra to word length 16, the residues and the transport to the shorter lengths §3 and §4 state — and formalized for all words in Lean 4 with mathlib. The two-way transport, which the series' own test suite never exercised, is checked here for the first time, and in the direction that matters: its certificate refuses every one of 112530 perturbed triples.

Contents

0. What is claimed, and what is not **2**

1. The setting **3**

2. One word, one affine map **3**

3. The exact domain **4**

4. The transport runs both ways **5**

5. Formalized for all words **6**

6. Where the series starts **6**

7. The instrument, and how it was made to fail **7**

8. Contributions **7**

9. What is open **8**

Acknowledgements **8**

References **8**
Data and software 8

A. Reproduction **8**

Status of this document. Preprint, version 0.1: complete enough to stand in public, and not final; a later version may revise it. Not prepared for journal submission. Section 8 records who did which part, including the parts this line did not do.

0. What is claimed, and what is not

Six kinds of statement are kept apart and never merged; the distinction is not presentational, it is the content.

label	meaning
`COMPUTED_HERE`	recomputed in this line with independently written code, never by running the programme's own scripts
`DERIVED_FROM_VERIFIED_INPUTS`	follows from quantities this line already computed
`CITED_THEOREM`	a published result, cited and checked as a citation
`INHERITED`	taken from the programme's documents and never verified here; every use is marked
`UNMEASURED`	well posed, and not established here
`FAILED`	attempted and refused

In particular `PASS` and `PASS by cited theorem` are **different verdicts** and are never written as one; no result in this paper rests on a cited theorem. The works cited in §6 are context, and were checked as citations.

Not claimed anywhere in this paper: the Collatz conjecture; that the inverse tree rooted at 1 covers the positive integers; that the $2^k \leftrightarrow 3^u$ transport is an isomorphism of the 2-adic and 3-adic integers — it is a correspondence of arithmetic progressions, one word at a time; and novelty of the elementary statements, several of which are classical in other notation.

Inherited: nothing. Every statement below is proved in the text; those of §§2–3 are also formalized for all words, and every statement is checked exhaustively on a finite range against a referee that does not assume it.

1. The setting

Write the map as $T(n) = n/2$ for even n and $T(n) = (3n+1)/2$ for odd n; an odd step of the classical map is always followed by a halving, and this form merges the two. Let $D(x) = x/2$ and $U(x) = (3x+1)/2$. A **word** $w = \sigma_1 \cdots \sigma_k \in \{D, U\}^k$ is read left to right, so its formal operator is

$$F_w = \sigma_k \circ \cdots \circ \sigma_1,$$

which can be evaluated at any rational x. A positive integer n **follows** w if $\sigma_j = D$ exactly when $T^{j-1}(n)$ is even; the set of such n is the **admissible domain** Ω_w, and on it $T^k(n) = F_w(n)$. Off it, $F_w(n)$ is still a legal rational expression and says nothing about the orbit of n. The series' starting point is to keep these apart: **a formal transform being legal is not the same as a dynamical step being legal**, and most of what can go wrong in this subject goes wrong at that seam.

Throughout, $k = |w|$ and $u = u(w)$ is the number of U in w.

2. One word, one affine map

Theorem 2.1 (affine closure and recursion). For every word, $F_w(x) = (3^u x + b_w)/2^k$ with an integer $b_w \geq 0$, where $b_\emptyset = 0$ and

$$b_{wD} = b_w, \qquad b_{wU} = 3b_w + 2^{|w|}.$$

Proof. Induction on $|w|$: $F_{wD}(x) = F_w(x)/2 = (3^u x + b_w)/2^{k+1}$, and $F_{wU}(x) = (3F_w(x)+1)/2 = (3^{u+1}x + 3b_w + 2^k)/2^{k+1}$. □

Theorem 2.2 (closed form). If the U of w stand at positions $j_1 < \cdots < j_u$, then $b_w = \sum_{t=1}^{u} 2^{j_t - 1}\, 3^{u-t}$.

Proof. Appending U at position $k+1$ multiplies every existing term by 3 and adds $2^{(k+1)-1}3^0$, which is the recursion; appending D adds no term and moves no position, so b_w is unchanged. □

Theorem 2.3 (concatenation and its order defect). $b_{wv} = 3^{u(v)}\, b_w + 2^{|w|}\, b_v$, and therefore

$$b_{wv} - b_{vw} = b_w\big(3^{u(v)} - 2^{|v|}\big) - b_v\big(3^{u(w)} - 2^{|w|}\big).$$

Proof. $F_{wv} = F_v \circ F_w$; expand. □

Swapping two words changes neither $|w| + |v|$ nor $u(w) + u(v)$, so the multiplier $3^u/2^k$ commutes and **non-commutativity lives entirely in the correction coordinate**.

Theorem 2.4 (matrix representation). Represent $x \mapsto (Ax + B)/E$ by the upper-triangular matrix $\begin{pmatrix} A & B \\ 0 & E \end{pmatrix}$, so that $M_D = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$ and $M_U = \begin{pmatrix} 3 & 1 \\ 0 & 2 \end{pmatrix}$. Then $M_w = M_{\sigma_k} \cdots M_{\sigma_1}$, and $M_{wv} = M_v M_w$.

Proof. Composing $(A_1 x + B_1)/E_1$ and then $(A_2 y + B_2)/E_2$ gives $(A_2 A_1 x + A_2 B_1 + B_2 E_1)/(E_2 E_1)$, the action of $M_2 M_1$. □

The order is the opposite of reading order. The series says so; the tidier-looking statement $M_{wv} = M_w M_v$ would have been false, and it was the formalization, not the exhaustive check, that made the point impossible to miss — a finite check confirms whatever statement it is given.

Theorem 2.5 (order extremes). For fixed (k, u),

$$3^u - 2^u \le b_w \le 2^{k-u}\,(3^u - 2^u),$$

with the minimum exactly at $U^u D^{k-u}$ and the maximum exactly at $D^{k-u} U^u$; the width of the range is $(2^{k-u} - 1)(3^u - 2^u)$.

Proof. In Theorem 2.2 the sum is strictly increasing in each j_t, and $t \le j_t \le k - u + t$. At $j_t = t$ it is $\sum_t 2^{t-1} 3^{u-t} = 3^u - 2^u$; at $j_t = k - u + t$ it is 2^{k-u} times that. □

Equivalently, at the same prefix state $DU(x) - UD(x) = (3x + 2)/4 - (3x + 1)/4 = \frac{1}{4}$, so moving a U to the right past a D strictly increases the correction.

Checked. Every statement of this section was checked on all 131070 words of length at most 16, all 260100 concatenation pairs and all 152 pairs (k, u), by routes that share no derivation: a **referee** that composes D and U symbolically on an affine form and assumes no theorem at all, the closed form, and the matrix product. If a theorem were false, the referee is what would disagree with it, because the referee never used it. The series' own test suite reached word length 9 and did not exercise the recursion, the concatenation law, the matrix representation or the width at all.

3. The exact domain

Theorem 3.1 (word and residue). Two positive integers follow the same word of length k if and only if they are congruent modulo 2^k. Hence the 2^k words of length k correspond bijectively to the residues modulo 2^k: Ω_w is one class $r_w + 2^k \mathbb{Z}$ intersected with the positive integers, where

$$r_w \equiv -\,b_w\, 3^{-u} \pmod{2^k}.$$

Proof. If $n' = n + c\,2^k$ with $k \ge 1$, the two have the same parity, and $T(n') = T(n) + c\,2^{k-1}$ if n is even, $T(n) + 3c\,2^{k-1}$ if n is odd; induction on k, **over all multipliers c at once**, gives the same first k parities. Conversely, if n follows w then $T^k(n) = (3^u n + b_w)/2^k$ is an integer, so $3^u n \equiv -b_w \pmod{2^k}$, which fixes n modulo 2^k since 3 is invertible. A map from 2^k residues to 2^k words that is injective is a bijection. □

The induction must be over every multiplier, not over $c = 1$: the odd branch sends a multiplier c to $3c$, so a hypothesis fixed at $c = 1$ cannot be applied to its own successor. That trap was met in the formalization, not in the series.

Corollary 3.2 (transport on the cylinder). $m_w = (3^u r_w + b_w)/2^k$ is an integer, and for every a with $r_w + 2^k a > 0$,

$$T^k(r_w + 2^k a) = m_w + 3^u a.$$

In the coordinates $\varphi_w(n) = (n - r_w)/2^k$ and $\psi_w(y) = (y - m_w)/3^u$, the k-step map is the identity: $\psi_w \circ T^k \circ \varphi_w^{-1} = \mathrm{id}$. The source is recovered from the target alone, $n = r_w + 2^k (y - m_w)/3^u$, and F_w is injective on its cylinder.

The positive domain needs a repair. With the canonical choice $0 \le r_w < 2^k$, the all-D word has $r_w = 0$, and $a = 0$ gives $n = 0$, which is not in the domain; its positive members are $2^k a$ with $a \ge 1$,

and $T^k(2^k) = 1$ as the corollary says. The series' original induction used "$r_w \in \Omega_w$", which has no witness for exactly this word. The repair the series records is to take the positive representative; the formalization states it for **every** cylinder — each has a positive member — rather than patching the one that breaks.

Checked, twice over. The residue r_w was derived a second time by a route that never touches $b_w 3^{-u}$: start from $r_D = 0$, $r_U = 1$ and split each cylinder into r_w and $r_w + 2^k$ according to the parity of m_w. The two derivations agree on all 16382 words of length at most 13, which is what makes the closed congruence a result rather than an assumption, and the transport, the local identity, recovery and faithfulness held on all 114661 cases. Separately, word-to-residue injectivity was confirmed for every length up to 14 and the transport on 196582 cases; and the opposite direction — simulate every integer in a range, read its word off the dynamics, and require each of the 2^k words equally often — was run to length 11, on every integer in $[1, 4 \cdot 2^k]$. The five worked examples of the series match its stated numbers exactly, including $UUDD$ with $b = 5, r = 3, m = 2$ and the orbit $3 \to 5 \to 8 \to 4 \to 2$.

Source charts partition; target charts need not. The images $m_w + 3^u\mathbb{Z}$ of different words can meet: $DDDD$ and $DDDU$ both reach 2 at depth four. A non-claim of the series is in this way given a witness rather than a sentence.

4. The transport runs both ways

Theorem 4.1. For each word, $x = r_w + 2^k a \mapsto y = m_w + 3^u a$ is a bijection between the progressions $r_w + 2^k\mathbb{Z}$ and $m_w + 3^u\mathbb{Z}$. A target y is legal for w exactly when $y \equiv m_w \pmod{3^u}$; its source is $x = r_w + 2^k(y - m_w)/3^u$; the quotient $(x - r_w)/2^k = (y - m_w)/3^u$ is conserved, or, without division, $3^u(x - r_w) = 2^k(y - m_w)$. On the positive integers the source set is $\{r_w + 2^k a : a \geq a_{\min}\}$ and the image set $\{m_w + 3^u a : a \geq a_{\min}\}$, with $a_{\min} = 1$ if $r_w = 0$ and 0 otherwise.

Proof. Corollary 3.2 and the invertibility of $a \mapsto m_w + 3^u a$ on its progression. □

One step backwards. $T^{-1}(y) = \{2y\} \cup \{(2y - 1)/3 : y \equiv 2 \pmod 3\}$, since $(3x + 1)/2 = y$ forces $x = (2y - 1)/3$. For the classical map the same constraint reads $y \equiv 4 \pmod 6$; the two congruences are one condition. For the accelerated odd map $t \mapsto (3t + 1)/2^{v_2(3t+1)}$, the odd predecessors of an odd t with valuation κ are

$$R_\kappa(t) = \frac{2^\kappa t - 1}{3}, \quad \text{an integer exactly when } 2^\kappa t \equiv 1 \pmod 3.$$

So a multiple of 3 has no odd predecessor at all — the accelerated image never meets one — and otherwise the admissible κ are the even ones for $t \equiv 1$ and the odd ones for $t \equiv 2 \pmod 3$. At $t = 1$ the fiber is $R_{2j}(1) = (4^j - 1)/3$: the family $1, 5, 21, 85, 341, 1365, 5461, 21845, \ldots$, each reaching 1 in one accelerated step; for example $5 \to 16 \to 8 \to 4 \to 2 \to 1$ in the classical map.

Local inversion is not global injectivity. Each chart is injective on its own cylinder, but two charts can merge: D carries 4 to 2 and U carries 1 to 2. And a fixed-word inverse is single-valued only because the word is fixed; the inverse tree branches.

The certificate, checked in the direction that matters. The series offers a three-part certificate for a triple (x, y, w): $x \equiv r_w \pmod{2^k}$, $y \equiv m_w \pmod{3^u}$, and $(x - r_w)/2^k = (y - m_w)/3^u$. Confirming that it accepts legal triples is the easy half and proves almost nothing; a certificate that accepts everything certifies nothing. It accepted all 22506 legal triples for words of length at most 10, and refused all 112530 perturbed ones — x moved by 1 or by 2^k, y by 1 or by 3^u, and both by 1. Dropping the

third condition leaves the accepting half perfectly green and is caught only by the refusing half. The series' own test suite contained no test of this paper's statements at all; these checks are their first.

5. Formalized for all words

A finite check confirms a statement for every case it reaches and for no other. The statements of §§2–3 are universal, and they were proved for every word in Lean 4 with mathlib — no `sorry`, and `#print axioms` showing only `propext`, `Classical.choice` and `Quot.sound` — together with the order defect as an exact identity and the attainment of both bounds of Theorem 2.5. The development lives in a private repository and is described here, not linked. What the proofs needed is worth recording, because each item is about how a statement is *written* rather than whether it is true, and no finite check could have shown it:

- **Recursion where the list recurses.** b_w is defined by the recursion lists support (on the first letter), and the series' right-append recursion of Theorem 2.1 is proved as a corollary of the concatenation law. Defined the series' way, the affine closure would have been buried under list-reversal lemmas.
- **The word–residue theorem** is `parityWord_eq_iff_modEq`: two integers follow the same word of length k if and only if they are congruent modulo 2^k, with injectivity and the count, and the bridge `iterate_eq_F` from the integer dynamics to the rational algebra needs no admissibility hypothesis, because the word is read off n itself.
- **Charts on $\mathbb{Z}$, and m_w by a hypothesis.** φ_w needs subtraction, so the charts live on the integers, not the naturals; and m_w is never defined by a division but taken as any integer with $2^k m = 3^u r + b_w$, which turns every proof in the section into a ring identity. The formal local identity on the whole cylinder and its restriction to T^k on the positive domain are proved separately, because conflating them is exactly the domain confusion §1 warns of.

Each formal definition was cross-checked against this line's own composition routine on every short word, with the formal side emitting each word's encoding so that the comparison does not rest on both sides enumerating in the same order. The two-way transport of §4 is checked finitely and is not yet formalized.

6. Where the series starts

The series opens by re-classifying earlier work on the problem by strength of evidence — theorem, experiment, calibration, heuristic, no-go — and its theorem-class entries are concrete arithmetic. Each was checked against a brute-force referee rather than against the series' reasoning: an integer has an odd predecessor under the classical map exactly when it is $4 \pmod 6$, a class of density $1/6$ (against an actual search for m with $3m+1 = n$, over 30000 integers); every positive integer is uniquely $2^{v_2(n)}$ times an odd core, and an integer converges exactly when its odd core does; an orbit reaches 1 exactly when it meets a power of two (over 20000 orbits).

Its references were fetched and compared against their sources, title, authors and venue: among them Rozier on parity sequences of the $3x+1$ map on the 2-adic integers (*Integers* 19, 2019), Tao on almost all orbits attaining almost bounded values (*Forum Math. Pi* 10, 2022), and Barina's verification of convergence for every n below $2075 \cdot 2^{60}$, about 2^{71} (*J. Supercomputing* 81, article 810, 2025). All matched. The check also corrected this line: its own charter had cited the frontier from an older

milestone, three doublings lower — literally true as a lower bound, and out of date, in the one place the line insists on being precise about how far below the real frontier its own computations sit.

7. The instrument, and how it was made to fail

The rechecks behind this paper share one mutation drill with the rest of the series' rechecks — one drill for nine papers' rechecks: defects are planted one at a time and each must be caught by the check **named** for it, while null controls must disturb nothing. Of its 69 planted defects, 67 are caught by the check named for each, one is detected only by a crash and one only by a time-out, and all 9 controls stay undisturbed; its log lists every defect with the recheck it targets. The line's whole suite at the pinned commit stands at 57 drills and 1474 planted defects: 1472 caught by their named check, one detected only by a crash and one only by a time-out, and 128 controls undisturbed.

Three things the drill found are worth stating:

- **Mutations an identity defeats.** Perturbing the terminal fiber to $(4^j + 1)/3$ changes nothing under integer division, since $4^j + 1 \equiv 2 \pmod 3$; the drill reported the defect as never planted rather than as missed, and it was re-aimed. Two other mutations elsewhere in the series were defeated the same way, by an arithmetic identity rather than by the checks.
- **A crash is a weak detection.** A wrong residue makes m_w non-integral, and an assertion fired before any named check could report; the failure is now a named disagreement, so the detection says what broke and not only that something did.
- **Both derivations are load-bearing.** Perturbing the refinement's parity, its base case or its child offset breaks the agreement of §3, and so does inverting the congruence's exponent; neither route is decorative.

One claim of this line's own record did not survive the writing of this paper: the second layer's account of these rechecks said the word–residue bijection had been run in the opposite direction to a word length that no committed code reaches and no log records. The ranges stated in §3 are the ones the logs hold, and the record now carries a correction beside the original sentence.

8. Contributions

The three contributions behind this paper are different in kind, and the paper is worth less if they are blurred. They are given here in alphabetical order, the order the author list keeps as data; on the title page the three sit at a round table, with no first and no last.

Aletheia — model: ChatGPT, GPT series — wrote, with Neo.K, the papers of the series this paper rests on: the re-classification of earlier work, the affine atlas, the residue cylinders and local identity, and the two-way transport with its inverse fiber and certificate.

Neo.K set the series' target and its method — that a formal transform's legality and a dynamical step's legality are different things — and wrote its papers with Aletheia.

墜衡 (Zhuì Héng) — the independent verification line; model: Claude, Opus series — rechecked every statement against a referee that assumes none of them, derived the residues a second time by an independent route, checked the certificate in the refusing direction, found witnesses for the series' non-claims, checked the references against their sources, formalized §§2–3 in Lean, built and drilled

the instruments, and wrote this paper, including the proofs as written out here. The statements are the series'; nothing in them is this line's invention.

Priority, stated plainly. No result here is claimed as new mathematics. What is offered as this line's own is the version of it that can be relied on: stated at its true domain, checked against an independent referee, formalized where it is universal, and checked in the refusing direction where it is a certificate.

9. What is open

1. Whether the inverse tree rooted at 1 covers every positive integer — which is the conjecture.
2. A formalization of the two-way transport of §4, which is checked here only on a finite range.
3. Whether the target-chart overlaps of §3 admit a usable description beyond individual witnesses.

Acknowledgements

We thank Anthropic, whose Claude models the lab's verification and recomputation lines run on, and OpenAI, whose ChatGPT models the programme's documents were written with. Not every AI model or instance that took part in this work is named in it — among them instances that the named authors ran as helpers; that an AI is not listed here does not mean that none took part.

References

1. David Barina, *Improved verification limit for the convergence of the Collatz conjecture*, J. Supercomput. 81 (2025), no. 7, article 810, doi:10.1007/s11227-025-07337-0.
2. Olivier Rozier, *Parity sequences of the* $3x+1$ *map on the 2-adic integers and Euclidean embedding*, Integers 19 (2019), article A8; arXiv:1805.00133.
3. Terence Tao, *Almost all orbits of the Collatz map attain almost bounded values*, Forum Math. Pi 10 (2022), e12, doi:10.1017/fmp.2022.8; arXiv:1909.03562.

Data and software

- Python Software Foundation, *Python* (CPython), version 3.14.5, https://www.python.org/ — the language of every gate; the version is the one the line's tree records.
- Leonardo de Moura and Sebastian Ullrich, *The Lean 4 Theorem Prover and Programming Language*, in Automated Deduction – CADE 28, Lecture Notes in Computer Science, Springer, 2021, 625–635, doi:10.1007/978-3-030-79876-5_37; the formalization uses the toolchain leanprover/lean4 v4.33.0.
- The mathlib Community, *The Lean Mathematical Library*, in Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020), 367–381, doi:10.1145/3372885.3373824; arXiv:1910.09336; tag v4.33.0, commit db584cd.

A. Reproduction

The rechecks and the drill are Python with arbitrary-precision integers and exact rationals, standard library only, no network; each writes a JSON log, and the bibliography records are kept as snapshots and digested on every run. The formalization is Lean 4 with mathlib, in a private repository. The

line's own tree is not public. Its public projection is, and it is what this paper's figures can be checked against without trusting the authors:

- **the run-by-run record**, rendered from the tree with a privacy gate on each file: https://amral.evemisslab.com/collatz/recompute/ (and `/en/`);
- **the secondary-verification ledger**, whose numbers are rendered from the line's own results file and read back against it before the site ships: https://verify.evemisslab.com/collatz-verification-zhuiheng/.

The evidence is pinned, not copied: this paper's front matter names the research branch, the commit and the directory its figures were drawn from, and the checker reads them exactly as they stood at that commit. The crystallization standard and its checker live in the release repository beside the paper (`STANDARD.md`, `tools/check_standard.py`).