MATHDISCOVERIES
Mathematical Preprint

Working Draft

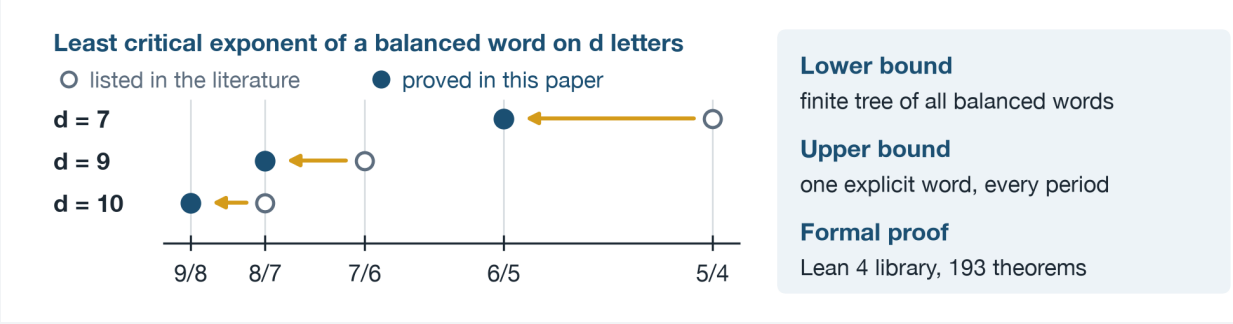
Balanced-Sequence Repetition Threshold Refutation: Exact RTB(7) = 6/5, RTB(9) = 8/7, and RTB(10) = 9/8

Tyler B. Hughes*,†

†Independent Researcher, United States

Supporting Information: `recompute.py`, the Lean 4 library `BalancedRepetitionThreshold`, and trust-boundary ledgers

Disclosure: AI-assisted working draft directed by Tyler B. Hughes. Not for submission as is. No human mathematician has yet refereed the proof.



ABSTRACT: An infinite word is balanced if the numbers of occurrences of each letter in any two factors of equal length differ by at most one. Let $\mathrm{RTB}(d)$ be the infimum, over balanced words on d letters, of the critical exponent (the supremum of length over period among factors). Papers dated 2021 to 2023 list $\mathrm{RTB}(d) = (d-2)/(d-3)$ for $5 \le d \le 10$ as established. We prove that $\mathrm{RTB}(7) = 6/5$, that $\mathrm{RTB}(9) = 8/7$ and that $\mathrm{RTB}(10) = 9/8$, so that $\mathrm{RTB}(7) < 5/4$, $\mathrm{RTB}(9) < 7/6$ and $\mathrm{RTB}(10) < 8/7$ against the listed values. Each infimum is attained by an explicit word: v_7 on 7 letters, v_9 on 9 letters and v_{10} on 10 letters. The values are the same when every letter must occur: 6/5 on 7 letters, 8/7 on 9 letters and 9/8 on 10 letters. The lower bound $(d-1)/(d-2)$ and the colourings are from the literature; the matching upper bounds are ours. Main results: 6 of 6 are proved in a Lean 4 library of 193 theorems, cited by its committed build receipt; kernel replay is pending. The cases $d = 5$, $d = 6$, $d = 8$ and $d \ge 11$ are not addressed. AI research agents built and directed by Tyler B. Hughes generated and checked this work. No human mathematician has yet refereed the proof.

Verification Scope (G11): WARN: Human referee: pending
Main results: 6 of 6 in Lean Kernel (100%)
Abstract claims: 12 of 12 in Lean Kernel (100%)

Full Trust Boundary: Section 5.

MSC 2020: 68R15, 05A05, 11J70, 68V20 **Keywords:** balanced words, critical exponent, repetition threshold, Sturmian words, Lean 4

1. INTRODUCTION AND MAIN RESULTS

A balanced word is an infinite sequence of letters in which every letter is spread as evenly as possible: any two windows of the same length contain almost the same number of copies of each letter. No such word is free of repetition, and its critical exponent measures the strongest repetition that it contains. This paper determines the least possible critical exponent of a balanced word on 7, on 9 and on 10 letters. In each case the answer is smaller than the value that the literature lists as established.

Write $\mathrm{RTB}(d)$ for the infimum of the critical exponent $E(w)$ over all balanced words w on d letters; Section 2 gives the exact definitions. Rampersad, Shallit and Vandomme [1] constructed a balanced word x_k on k letters for each $3 \le k \le 10$ and wrote: "For $k \ge 5$, computer calculations suggest that $E(x_k) = (k-2)/(k-3)$. Backtracking searches for $5 \le k \le 9$ show that this would be the least possible critical exponent for balanced words over a k-letter alphabet." Later work computed $E(x_k)$ for $5 \le k \le 10$ ([2] for $k = 5$; see [3] and the references there for the other cases), and the abstract of [4] states that the resulting conjecture $\mathrm{RTB}(d) = (d-2)/(d-3)$ "is known to hold for $d \in \{5,6,7,8,9,10\}$". The same paper proves that the conjecture fails for $d = 11$ and for every even $d \ge 12$, where the threshold is $(d-1)/(d-2)$. The lists of thresholds in [5, 7] repeat $\mathrm{RTB}(d) = 1 + 1/(d-3)$ for $5 \le d \le 10$.

Theorem 1 (Seven letters). *The infimum of the critical exponent $E(w)$ over all balanced words $w\colon \mathbb{N} \to \{0,1,\dots,6\}$ equals 6/5; that is,* $\mathrm{RTB}(7) = 6/5$.

Theorem 2 (Nine letters). *The infimum of the critical exponent $E(w)$ over all balanced words $w\colon \mathbb{N} \to \{0,1,\dots,8\}$ equals 8/7; that is,* $\mathrm{RTB}(9) = 8/7$.

Theorem 3 (Ten letters). *The infimum of the critical exponent $E(w)$ over all balanced words $w\colon \mathbb{N} \to \{0,1,\dots,9\}$ equals 9/8; that is,* $\mathrm{RTB}(10) = 9/8$.

Corollary 4 (Below the listed value, $d = 7$). $\mathrm{RTB}(7) < 5/4$.

Corollary 5 (Below the listed value, $d = 9$). $\mathrm{RTB}(9) < 7/6$.

Corollary 6 (Below the listed value, $d = 10$). $\mathrm{RTB}(10) < 8/7$.

Table 1 places the three values beside the listed ones. For $d = 7$, 9 and 10 the threshold is $(d-1)/(d-2)$, the value that [4] establishes for $d = 11$ and for even $d \geq 12$. Main results: 6 of 6 in Lean (100%) · 0 by certificate · 0 by written proof; the six are Theorems 1 to 3 and Corollaries 4 to 6, and Section 5 states exactly what that count rests on.

What is and is not due to this paper. The two halves of each theorem have different histories. The lower bound $E(w) \geq (d-1)/(d-2)$ for balanced words on d letters is in print: [4] proves it for $d \geq 11$ (Theorem 13 there) and notes that the argument works for $d \geq 4$ (footnote 3 there), setting the smaller alphabets aside because "for $d \leq 10$ a stronger lower bound is known". Our contribution to the lower bound is an independent finite proof for $d = 7$, 9 and 10 that is checked by the Lean kernel (Propositions 15 to 17). The upper bound is where the listed values fail: for each of the three alphabets we exhibit a balanced word whose critical exponent equals $(d-1)/(d-2)$ (Propositions 21 to 23). The words reuse published colourings. For $d = 7$ and $d = 9$ the two periodic colouring words are those of x_7 and x_9 in Table 1 of [1], up to renaming letters, and only the slope of the underlying Sturmian word differs. For $d = 10$ the word has the shape of the family of Theorem 19 of [4] at the parameter value $\delta = 5$, which lies outside the range $\delta \geq 7$ of that theorem.

Our results do not contradict the computed values $E(x_k)$: with the slope printed for x_7, our replay finds a factor of exponent 5/4 (Section 4). What fails is the minimality of those words, that is, the lower bounds stronger than $(d-1)/(d-2)$ for $d = 7$, 9 and 10. We have not seen the programs behind the earlier backtracking searches and do not know where they went wrong.

The following ten lines build 4000 letters of the 7-letter word v_7 of Section 2 in exact integer arithmetic, check that its initial six letters form a factor of period 5, and check on this finite sample that no repetition exceeds exponent 6/5. The sample check is an illustration; the statement for all periods is Proposition 21.

Instant Witness Check (verbatim in `recompute.py`)

```
from math import isqrt
f = lambda m: (11*m - isqrt(10*m*m) - 1) // 37
u = [f(n + 2) - f(n + 1) for n in range(4000)]
y, z = [5, 6], [0, 1, 2, 3, 0, 4, 2, 1, 0, 3, 2, 4]
v, a = [], 0
for n, t in enumerate(u):
    v.append(y[a % 2] if t else z[(n - a) % 12]); a += t
assert v[:6] == [0, 1, 2, 5, 3, 0]  # period 5: 6/5
assert all(v[i:i+p//5+1] != v[i+p:i+p+p//5+1]
           for p in range(1, 400) for i in range(3000))
```

2. DEFINITIONS, NOTATION, AND SETUP

Throughout, $\mathbb{N} = \{0, 1, 2, \dots\}$, the alphabet on d letters is $A_d = \{0, 1, \dots, d-1\}$, and a *word* is a map $w\colon \mathbb{N} \to A_d$. The definitions below match the Lean source `Defs.lean` and `Sturmian.lean` clause by clause; the Lean name is given in each case. The literature writes the letters as $1, \dots, d$, a shift by one.

Table 1. Least critical exponent of a balanced word on d letters, $5 \leq d \leq 10$.[a]

d	Listed, $(d-2)/(d-3)$	This paper
5	3/2	not re-examined
6	4/3	not re-examined
7	5/4	6/5 (Theorem 1)
8	6/5	not re-examined
9	7/6	8/7 (Theorem 2)
10	8/7	9/8 (Theorem 3)

[a] Listed values as in [4, 5, 7]. Three of the six listed values change.

Definition 7 (Factor, balanced word). For $i, L \in \mathbb{N}$ the *factor* of w at position i of length L is the finite word $w(i)\,w(i+1)\cdots w(i+L-1)$ (`factor`). The word w is *balanced* (`IsBalanced`) if for every letter $a \in A_d$, every length L and all positions i and j, the numbers of occurrences of a in the factors of length L at i and at j differ by at most 1. The word w *uses all letters* (`UsesAllLetters`) if every $a \in A_d$ equals $w(i)$ for some i.

Definition 8 (Period, critical exponent). The factor of w at position i of length L *has period* p (`HasPeriod`) if $0 < p \leq L$ and $w(i+k) = w(i+k+p)$ for every k with $k+p < L$. The *critical exponent* of w (`critExp`) is

$$E(w) = \sup_{i,L,p} L/p,$$

where the supremum, taken in the extended real line $[-\infty, +\infty]$, runs over all positions i, lengths L and integers p such that the factor of w at position i of length L has period p. A period p of a factor is its *least period* (`IsMinPeriod`) if no period of that factor is smaller.

Every factor of length 1 has period 1, so $E(w) \geq 1$ for every word (`one_le_critExp`), and $E(w) = +\infty$ is possible. Restricting the supremum to least periods does not change it (`sSup_minPeriodExponentSet_eq_critExp`).

Definition 9 (Repetition threshold of balanced words). For a natural number d,

$$\mathrm{RTB}(d) = \inf\{\, E(w) : w\colon \mathbb{N} \to A_d \text{ balanced} \,\},$$

and $\mathrm{RTB}^*(d)$ is the same infimum restricted to balanced words that use all letters (`RTB`, `RTB_strict`).

Since $\mathrm{RTB}(d)$ ranges over more words, $\mathrm{RTB}(d) \leq \mathrm{RTB}^*(d)$ (`RTB_le_RTB_strict`). The literature speaks of balanced sequences over a d-letter alphabet without always saying whether every letter must occur; the theorems of Section 1 and Propositions 12 to 14 cover both readings. No recurrence or aperiodicity hypothesis enters either definition.

Definition 10 (Sturmian word, colouring). For a real number α the word u_α over $\{\mathtt{a}, \mathtt{b}\}$ has $u_\alpha(n) = \mathtt{a}$ exactly when $\lfloor (n+2)\alpha \rfloor - \lfloor (n+1)\alpha \rfloor = 1$ (`sturmian`; the Lean value `true` is the letter `a`). For words $y, y'\colon \mathbb{N} \to A_d$ the *colouring* of a word u over $\{\mathtt{a}, \mathtt{b}\}$ (`colour`) is the word

$$\mathrm{colour}(u, y, y')(n) = \begin{cases} y(\#\{j < n : u(j) = \mathtt{a}\}), & u(n) = \mathtt{a}, \\ y'(\#\{j < n : u(j) = \mathtt{b}\}), & u(n) = \mathtt{b}. \end{cases}$$

In words, the successive letters `a` of u are replaced by $y(0), y(1), \dots$ and the successive letters `b` by $y'(0), y'(1), \dots$.

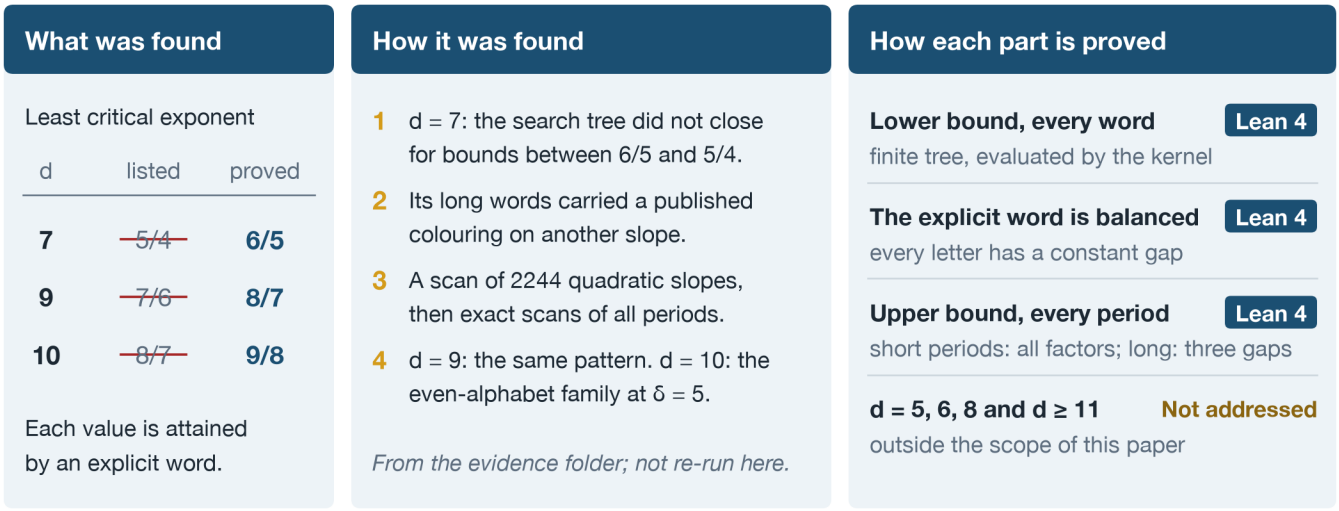


Figure 1. Discovery and proof architecture. Left: the three thresholds that change. Centre: how the words were found, as recorded in the evidence folder of the finding. Right: how each part is proved, with the Lean declarations that carry it. Thresholds for $d = 5, 6, 8$ and $d \geq 11$ are outside the scope of this paper.

When $0 < \alpha < 1$ the letter a has frequency α in u_α. Colourings of Sturmian words by constant gap words describe the recurrent aperiodic balanced words, by a theorem of Hubert [8] as stated in [1, 4]; this paper uses only the easy direction, proved directly in Lemmas 18 to 20.

Definition 11 (The three words). Write $(c_0\, c_1 \cdots c_{m-1})^\omega$ for the periodic word $k \mapsto c_{k \bmod m}$. The words v_7, v_9 and v_{10} are $v_d = \mathrm{colour}(u_{\alpha_d}, y_d, y'_d)$ with the parameters of Table 2 (`v7`, `v9`, `v10`).

The word v_7 begins 0125304261032540 (Figure 2), the word v_9 begins 253640728351462 7, and v_{10} begins 5061782935467081; `recompute.py` prints these prefixes. All d letters occur in v_d (`v7_usesAllLetters`, `v9_usesAllLetters`, `v10_usesAllLetters`).

Proposition 12 (All letters required, $d = 7$). *The infimum of $E(w)$ over all balanced words $w\colon \mathbb{N} \to \{0, 1, \ldots, 6\}$ that use all letters equals 6/5; that is,* $\mathrm{RTB}^*(7) = 6/5$.

Proposition 13 (All letters required, $d = 9$). *The infimum of $E(w)$ over all balanced words $w\colon \mathbb{N} \to \{0, 1, \ldots, 8\}$ that use all letters equals 8/7; that is,* $\mathrm{RTB}^*(9) = 8/7$.

Proposition 14 (All letters required, $d = 10$). *The infimum of $E(w)$ over all balanced words $w\colon \mathbb{N} \to \{0, 1, \ldots, 9\}$ that use all letters equals 9/8; that is,* $\mathrm{RTB}^*(10) = 9/8$.

3. PROOF OF THE MAIN RESULTS

Each theorem is the conjunction of a lower bound valid for every balanced word and an upper bound given by one word. The proof of record for every numbered statement is the Lean proof named beside it in Section 5. This section explains the arguments in four parts. The parts on the lower bound, on balance and on assembly are complete written proofs relative to a stated finite computation; the part on the upper bound is a guide to the Lean proof and is not a self-contained written proof.

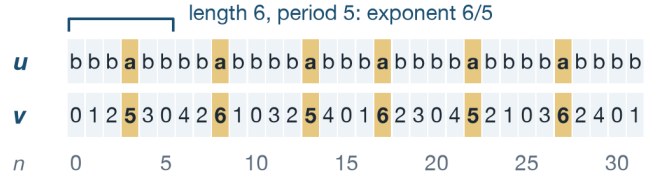


Figure 2. The initial 32 letters of v_7. Top row: the Sturmian word u_{α_7}. Bottom row: its colouring, with the two letters 5 and 6 on the positions of a. The bracket marks the factor of length 6 and period 5 that gives exponent 6/5.

Lower bound: a finite tree. The lower bound holds for every balanced word on d letters, with no further hypothesis.

Proposition 15 (Lower bound, $d = 7$). *Every balanced word $w\colon \mathbb{N} \to \{0, 1, \ldots, 6\}$ satisfies $E(w) \geq 6/5$.*

Proposition 16 (Lower bound, $d = 9$). *Every balanced word $w\colon \mathbb{N} \to \{0, 1, \ldots, 8\}$ satisfies $E(w) \geq 8/7$.*

Proposition 17 (Lower bound, $d = 10$). *Every balanced word $w\colon \mathbb{N} \to \{0, 1, \ldots, 9\}$ satisfies $E(w) \geq 9/8$.*

Proof (relative to the finite tree). Fix d and put $B = (d-1)/(d-2)$. Suppose a balanced word w on d letters had $E(w) < B$. Then every factor of every prefix of w has all its exponents L/p below B, and every prefix is balanced as a finite word. Rename the letters of w in order of earliest appearance; neither property is affected. Let T_d be the set of finite words over A_d that are named in order of earliest appearance, are balanced, and have every exponent L/p of every factor strictly below B. The set T_d is closed under taking prefixes, so it is a tree, and every prefix of the renamed word lies in T_d. The tree T_d is finite: its longest words have length 38, 57 and 75 for $d = 7$, 9 and 10 (Table 3). The prefix of length one more than that is a contradiction. Hence $E(w) \geq B$. □

The finiteness of T_d is the finite computation. In Lean it is a certificate tree (`LowerCert`) whose nodes are the words of

Table 2. Parameters of the three words $v_d = \mathrm{colour}(u_{\alpha_d}, y_d, y'_d)$.[a]

d	α_d	Continued fraction of α_d	y_d	y'_d
7	$(11-\sqrt{10})/37$	$[0;4,1,\overline{2,1,1}]$	$(56)^\omega$	$(0123042103 24)^\omega$
9	$(41-2\sqrt{2})/239$	$[0;6,3,\overline{1,4}]$	$(01)^\omega$	$(253647283546273845263748)^\omega$
10	$(7+\sqrt{5})/22$	$[0;2,2,\overline{1}]$	$(01234)^\omega$	$(56789)^\omega$

[a]The Lean library proves the enclosures $211/1000 < \alpha_7 < 212/1000$, $1597/10000 < \alpha_9 < 1598/10000$ and $419/1000 < \alpha_{10} < 420/1000$. The colouring words for $d = 7$ and $d = 9$ are those of x_7 and x_9 in Table 1 of [1] after renaming letters; the pair for $d = 10$ has the shape used in Theorem 19 of [4].

T_d together with, at each rejected extension, an explicit repetition or an explicit pair of unbalanced windows; a checker (`checkLowerCert`) is proved sound (`checkLowerCert_sound`) and evaluated by the kernel with `decide`, without `native_decide`. An extension by a letter that occurs among the last $d-2$ letters creates a repetition of exponent at least B and needs no node, which is why the certificates are small. Section 4 rebuilds the three trees with an independent program and recovers the same node counts.

The three words are balanced. Balance follows from a general lemma on colourings by words with constant gaps.

Lemma 18 (Balance, $d = 7$). *The word v_7 of Definition 11 is balanced.*

Lemma 19 (Balance, $d = 9$). *The word v_9 of Definition 11 is balanced.*

Lemma 20 (Balance, $d = 10$). *The word v_{10} of Definition 11 is balanced.*

Proof. Let $0 < \alpha < 1$. The number of letters `a` of u_α in the window $[i, i+L)$ is $\lfloor (i+L+1)\alpha \rfloor - \lfloor (i+1)\alpha \rfloor$, which is $\lfloor L\alpha \rfloor$ or $\lfloor L\alpha \rfloor + 1$. So for each L there is an M such that every window of length L contains M or $M+1$ letters `a`, and likewise for `b`. Say that a letter c has *constant gap* g in y if the indices k with $y(k) = c$ are exactly those in one residue class modulo g (`HasConstantGap`). If c has constant gap g in y and does not occur in y', then the occurrences of c in a window of colour(u_α, y, y') correspond to the members of that residue class within a block of consecutive integers whose length is M or $M+1$. A block of N consecutive integers contains $\lfloor N/g \rfloor$ or $\lceil N/g \rceil$ members of a residue class modulo g, and for $N \in \{M, M+1\}$ these counts all lie in $\{\lfloor M/g \rfloor, \lfloor M/g \rfloor + 1\}$. So the counts of c in two windows of equal length differ by at most 1. The same holds with the roles of y and y' exchanged. In each row of Table 2 every letter occurs in exactly one of y_d, y'_d and has a constant gap there: the gaps are 2, 4 and 6 for $d = 7$; 2, 6 and 8 for $d = 9$; and 5 for $d = 10$. The Lean statement of this argument is `colour_isBalanced_of_constantGap`. □

Upper bound: every period of the three words. The upper bound is a statement about one explicit word for each alphabet.

Proposition 21 (Critical exponent of v_7). *The word v_7 has critical exponent $E(v_7) = 6/5$, and $6/5$ equals L/p for a factor of v_7 of length L whose least period is p.*

Proposition 22 (Critical exponent of v_9). *The word v_9 has critical exponent $E(v_9) = 8/7$, and $8/7$ equals L/p for a factor of v_9 of length L whose least period is p.*

Proposition 23 (Critical exponent of v_{10}). *The word v_{10} has critical exponent $E(v_{10}) = 9/8$, and $9/8$ equals L/p for a factor of v_{10} of length L whose least period is p.*

Attainment. The factor of v_7 at position 0 of length 6 is `012530` and has period 5 (`v7_attains`). The factor of v_9 at position 0 of length 8 is `25364072` and has period 7 (`v9_attains`). The factor of v_{10} at position 4 of length 9 is `782935467` and has period 8 (`v10_attains`). In each case the two equal letters are the only constraint, and the exponent is $(d-1)/(d-2)$.

Reduction to runs. Call r a *run at distance* p of a word v if $v(i+k) = v(i+k+p)$ for all $k < r$ and some i. A factor of length L and period p is a run of length $L-p$, so $E(v) \le (d-1)/(d-2)$ follows once every run at every distance $p \ge 1$ satisfies $(d-2)\,r \le p$ (`critExp_le_of_run_bound`). The Lean library proves exactly this run bound for v_7, v_9 and v_{10} (`v7_run_bound`, `v9_run_bound`, `v10_run_bound`), in two ranges of p.

Small distances. For $p < P_d$, with $P_7 = 20$, $P_9 = 42$ and $P_{10} = 16$, a run that violates the bound fits inside a factor of v_d of length $\ell_d = 23$, 47 and 17 respectively. A Sturmian word has $\ell + 1$ factors of length ℓ; the library derives all of them from a rational enclosure of α_d by sorting the cut points of the rotation by α_d (`checkCutPartition`), colours each at every pair of phases of y_d and y'_d, and checks every distance $p < P_d$ (`checkSmallPeriods_sound`).

Large distances. Because y_d and y'_d share no letter, a run of v_d at distance p is a run of u_{α_d} at distance p on which the colours also agree (`colour_match_implies`). Let δ be the distance from $p\,\alpha_d$ to the nearest integer m. If $\delta \ge \alpha_d$, no two letters `a` at distance p agree, so a run is shorter than any window that must contain an `a` (`sturmian_run_lt_of_large_dist`). If $\delta < \alpha_d$, every position of a run sees exactly m letters `a` and $p-m$ letters `b` in the next p positions (`sturmian_match_counts_of_small_dist`), which ties the colour condition to the residues of m and $p-m$ modulo the periods of y_d and y'_d. The length of the run is then bounded by a three-gap argument on the rotation by α_d (`sturmian_three_gap_run_bound`), in terms of the denominators of the convergents of α_d. Since α_d is a quadratic irrational, its continued fraction is eventually periodic, with period 3, 2 and 1 for $d = 7$, 9 and 10, and the residues of the convergents modulo the periods of y_d and y'_d are eventually periodic with period 12, 8 and 20. The pairs (p, m)

therefore fall into finitely many classes, and one inequality per class, checked by the kernel on rational enclosures (`checkLevelClass_sound`), gives $(d-2)\,r \le p$ for all $p \ge P_d$.

We stress the status of this subsection: it names the steps and the Lean statements that carry them. It does not reproduce the case analysis, and a reader who wants to check the upper bound without Lean will not find a complete argument here. A written proof of the upper bound is future work (Section 6).

Assembly. The main results follow from the statements above in a few lines.

Proof of the main results. We prove Theorems 1 to 3, Corollaries 4 to 6 and Propositions 12 to 14. Fix $d \in \{7, 9, 10\}$ and $B = (d-1)/(d-2)$. By Propositions 15 to 17, B is a lower bound for $E(w)$ over all balanced words on d letters, so

$$B \le \mathrm{RTB}(d) \le \mathrm{RTB}^*(d).$$

The word v_d is balanced (Lemmas 18 to 20), uses all letters, and has $E(v_d) = B$ (Propositions 21 to 23), so $\mathrm{RTB}^*(d) \le B$, and the three quantities are equal. The corollaries are the inequalities $6/5 < 5/4$, $8/7 < 7/6$ and $9/8 < 8/7$. □

Remark 24 (Aperiodicity). An eventually periodic word has factors of a fixed period and unbounded length, hence infinite critical exponent. Since $E(v_d)$ is finite, no v_d is eventually periodic. This one-line argument is written here and is not part of the Lean library.

4. COMPUTATIONAL AND EMPIRICAL VERIFICATION

The script `recompute.py` replays every number in this paper that a short exact computation can reach. It uses integers and exact fractions only, and writes `recompute.out`. It proves nothing about all periods; it guards against transcription errors and gives an independent count for the finite part of the proof.

Trees. The script rebuilds the trees T_d of Section 3 by depth-first search, without reading the Lean certificates. Table 3 and Figure 3 give the result. The certificate node counts agree with those of the Lean files, where the trees are split into 19, 82 and 169 kernel evaluations. As a calibration against the literature, the same program run for $d = 5$ with bound $3/2$ returns a finite tree whose longest words have length 44, and there are 5 of them, the numbers printed by Baranwal and Shallit [2].

Scans of the three words. For each v_d the script builds a prefix with exact integer floors, confirms the prefixes and attaining factors of Section 2, checks balance on all windows of length up to 150 inside 3000 letters, and measures the longest run at every distance $p \le 600$ over all starting positions below 20000. No run exceeds $p/(d-2)$. Equality $r/p = 1/(d-2)$ occurs at $(p, r) = (5, 1)$, $(10, 2)$ and $(15, 3)$ for v_7, at $(7, 1)$ only for v_9, and at $(8, 1)$ only for v_{10}. The evidence folder of the finding records a longer scan of the same kind, over distances up to 3000 in 1000000 letters, with the same outcome.

Table 3. The finite trees T_d behind the lower bounds.[a]

d	Bound	Words	Leaves	Longest	Nodes
7	6/5	454	69	38 (10 words)	905
9	8/7	1473	235	57 (5 words)	2941
10	9/8	2437	426	75 (2 words)	4868

[a]Words: non-empty members of T_d. Nodes: nodes of the Lean certificate, that is, all members of T_d including the empty word plus the rejected extensions by a letter outside the last $d-2$ letters.

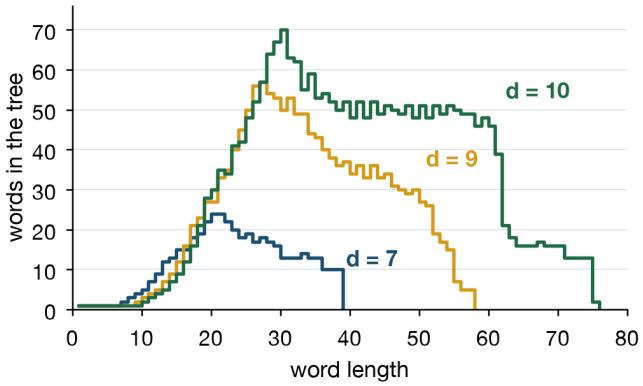


Figure 3. Number of words of each length in the trees T_7, T_9 and T_{10}. Each tree dies out, at lengths 38, 57 and 75; no balanced word on d letters keeps every exponent below $(d-1)/(d-2)$ beyond that length.

Control. With the colouring of v_7 and the slope printed for x_7 in Table 1 of [1], in which the letters `a` have frequency $(44 + \sqrt{10})/107$, the same scan finds $r/p = 1/4$ at $p = 4$ and nowhere else in the scanned range, that is, exponent $5/4$. The scan therefore separates the two slopes, and it agrees with the value $E(x_7) = 5/4$ of the literature.

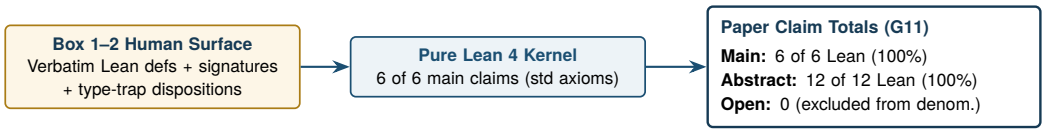
Lean receipt. The script also recounts the committed Lean bundle without building it: 13 source files, 8739 lines, and 193 public theorems, each of which appears in the committed axiom report. The axioms used are `propext`, `Classical.choice` and `Quot.sound`, and 4 of the 193 theorems use none.

5. FORMAL VERIFICATION & READER'S TRUST BOUNDARY

What You Must Trust to Accept This Paper:

1. **Lean 4 Kernel & Toolchain:** leanprover/lean4:v4.35.0; committed axiom receipt anchored to git commit ae4ef2f80d3a (independent gate-box kernel replay pending); standard axioms only (propext, Classical.choice, Quot.sound).
2. **Core Definitions & Fidelity (Boxes 1–2):** Verify that the verbatim Lean definitions in Box 1 match Section 2 and that each Lean signature in Box 2 matches its stated theorem.
3. **External Artifacts & Remaining Claims:** No external certificate artifacts are required; any claims marked Written or Unsettled rely on standard written proofs or remain explicitly unproved.

Reader's Trust-Flow Architecture (Distinct Evidence Paths to Claim Totals)



Anti-Shortcut Audit Strip — Source-Level Approx (Gate Replay Pending)

WARN: Committed receipt (gate replay pending) | PASS: sorry/admit: 0 (per committed receipt)
PASS: native_decide: 0 (per committed receipt) | PASS: Custom axioms & meta: 0 (per committed receipt)
PASS: Coverage: 176/193 public theorems (source-level approx) | WARN: Fidelity: 18/18 exact (draft cross-family)
WARN: Witnesses: pending gate replay | WARN: Human referee: pending

Box 1 — Verbatim Lean 4 Custom Definitions (Human Audit Surface)

BalancedRepetitionThreshold.factor (BalancedRepetitionThreshold/Defs.lean:33-34) — Definition 7 (factor)

```
def factor {α : Type*} (w : ℕ → α) (i L : ℕ) : List α :=
  (List.range L).map (fun j => w (i + j))
```

BalancedRepetitionThreshold.IsBalanced (BalancedRepetitionThreshold/Defs.lean:71-73) — Definition 7 (balanced word)

```
def IsBalanced {d : ℕ} (w : ℕ → Fin d) : Prop :=
  ∀ (a : Fin d) (L i j : ℕ),
    |((factor w i L).count a : ℤ) - ((factor w j L).count a : ℤ)| ≤ 1
```

BalancedRepetitionThreshold.UsesAllLetters (BalancedRepetitionThreshold/Defs.lean:91-92) — Definition 7 (uses all letters)

```
def UsesAllLetters {d : ℕ} (w : ℕ → Fin d) : Prop :=
  ∀ a : Fin d, ∃ i : ℕ, w i = a
```

BalancedRepetitionThreshold.HasPeriod (BalancedRepetitionThreshold/Defs.lean:95-96) — Definition 8 (period of a factor)

```
def HasPeriod {α : Type*} (w : ℕ → α) (i L p : ℕ) : Prop :=
  0 < p ∧ p ≤ L ∧ ∀ k : ℕ, k + p < L → w (i + k) = w (i + k + p)
```

BalancedRepetitionThreshold.IsMinPeriod (BalancedRepetitionThreshold/Defs.lean:99-100) — Definition 8 (least period)

```
def IsMinPeriod {α : Type*} (w : ℕ → α) (i L p : ℕ) : Prop :=
  HasPeriod w i L p ∧ ∀ q : ℕ, HasPeriod w i L q → p ≤ q
```

BalancedRepetitionThreshold.exponentSet (BalancedRepetitionThreshold/Defs.lean:104-105) — Definition 8 (the set of exponents L/p)

```
def exponentSet {d : ℕ} (w : ℕ → Fin d) : Set EReal :=
  { e : EReal | ∃ i L p : ℕ, HasPeriod w i L p ∧ e = (((L : ℝ) / (p : ℝ) : ℝ) :
    ↪ EReal) }
```

BalancedRepetitionThreshold.minPeriodExponentSet (BalancedRepetitionThreshold/Defs.lean:108-109) — Definition 8 (exponents L/p over least periods)

```
def minPeriodExponentSet {d : ℕ} (w : ℕ → Fin d) : Set EReal :=
  { e : EReal | ∃ i L p : ℕ, IsMinPeriod w i L p ∧ e = (((L : ℝ) / (p : ℝ) : ℝ) :
    ↪ EReal) }
```

BalancedRepetitionThreshold.critExp (BalancedRepetitionThreshold/Defs.lean:118-119) — Definition 8 (critical exponent E(w))

```
noncomputable def critExp {d : ℕ} (w : ℕ → Fin d) : EReal :=
  sSup (exponentSet w)
```

BalancedRepetitionThreshold.RTB (BalancedRepetitionThreshold/Defs.lean:122-123) — Definition 9 (RTB(d))

```
noncomputable def RTB (d : ℕ) : EReal :=
  sInf { e : EReal | ∃ w : ℕ → Fin d, IsBalanced w ∧ critExp w = e }
```

BalancedRepetitionThreshold.RTB_strict (BalancedRepetitionThreshold/Defs.lean:127-128) — Definition 9 (RTB*(d), all letters required)

```
noncomputable def RTB_strict (d : ℕ) : EReal :=
  sInf { e : EReal | ∃ w : ℕ → Fin d, IsBalanced w ∧ UsesAllLetters w ∧ critExp w = e
    ↪ }
```

BalancedRepetitionThreshold.sturmian (BalancedRepetitionThreshold/Sturmian.lean:31-32) — Definition 10 (Sturmian word)

```
noncomputable def sturmian (α : ℝ) (n : ℕ) : Bool :=
  decide (⌊((n + 2 : ℕ) : ℝ) * α⌋ - ⌊((n + 1 : ℕ) : ℝ) * α⌋ = 1)
```

BalancedRepetitionThreshold.countTrue (BalancedRepetitionThreshold/Sturmian.lean:35-36) — Definition 10 (number of earlier letters a)

```
def countTrue (u : ℕ → Bool) (n : ℕ) : ℕ :=
  ((Finset.range n).filter (fun j => u j = true)).card
```

BalancedRepetitionThreshold.countFalse (BalancedRepetitionThreshold/Sturmian.lean:39-40) — Definition 10 (number of earlier letters b)

```
def countFalse (u : ℕ → Bool) (n : ℕ) : ℕ :=
  ((Finset.range n).filter (fun j => u j = false)).card
```

BalancedRepetitionThreshold.colour (BalancedRepetitionThreshold/Sturmian.lean:44-45) — Definition 10 (colouring)

```
def colour {d : ℕ} (u : ℕ → Bool) (y y' : ℕ → Fin d) (n : ℕ) : Fin d :=
  if u n then y (countTrue u n) else y' (countFalse u n)
```

BalancedRepetitionThreshold.α7 (BalancedRepetitionThreshold/WitnessV7.lean:30-30) — Definition 11 and Table 2 (frequency alpha_7)

```
noncomputable def α7 : ℝ := (11 - Real.sqrt 10) / 37
```

BalancedRepetitionThreshold.y7Tab (BalancedRepetitionThreshold/WitnessV7.lean:33-33) — Definition 11 and Table 2 (table of y_7)

```
def y7Tab : List ℕ := [5, 6]
```

BalancedRepetitionThreshold.y'7Tab (BalancedRepetitionThreshold/WitnessV7.lean:36-36) — Definition 11 and Table 2 (table of y'_7)

```
def y'7Tab : List ℕ := [0, 1, 2, 3, 0, 4, 2, 1, 0, 3, 2, 4]
```

BalancedRepetitionThreshold.y7 (BalancedRepetitionThreshold/WitnessV7.lean:39-42) — Definition 11 and Table 2 (periodic word y_7)

```
def y7 (k : ℕ) : Fin 7 :=
  ⟨y7Tab.getD (k % 2) 0, by
    have hk : k % 2 < 2 := Nat.mod_lt k (by omega)
    interval_cases (k % 2) <;> decide⟩
```

BalancedRepetitionThreshold.y'7 (BalancedRepetitionThreshold/WitnessV7.lean:45-48) — Definition 11 and Table 2 (periodic word y'_7)

```
def y'7 (k : ℕ) : Fin 7 :=
  ⟨y'7Tab.getD (k % 12) 0, by
    have hk : k % 12 < 12 := Nat.mod_lt k (by omega)
    interval_cases (k % 12) <;> decide⟩
```

BalancedRepetitionThreshold.v7 (BalancedRepetitionThreshold/WitnessV7.lean:51-52) — Definition 11 and Table 2 (the word v_7)

```
noncomputable def v7 : ℕ → Fin 7 :=
  colour (sturmian α7) y7 y'7
```

BalancedRepetitionThreshold.α9 (BalancedRepetitionThreshold/WitnessV9.lean:29-29) — Definition 11 and Table 2 (frequency alpha_9)

```
noncomputable def α9 : ℝ := (41 - 2 * Real.sqrt 2) / 239
```

BalancedRepetitionThreshold.y9Tab (BalancedRepetitionThreshold/WitnessV9.lean:32-32) — Definition 11 and Table 2 (table of y_9)

```
def y9Tab : List ℕ := [0, 1]
```

BalancedRepetitionThreshold.y'9Tab (BalancedRepetitionThreshold/WitnessV9.lean:35-36) — Definition 11 and Table 2 (table of y'_9)

```
def y'9Tab : List ℕ :=
  [2, 5, 3, 6, 4, 7, 2, 8, 3, 5, 4, 6, 2, 7, 3, 8, 4, 5, 2, 6, 3, 7, 4, 8]
```

BalancedRepetitionThreshold.y9 (BalancedRepetitionThreshold/WitnessV9.lean:39-42) — Definition 11 and Table 2 (periodic word y_9)

```
def y9 (k : ℕ) : Fin 9 :=
  ⟨y9Tab.getD (k % 2) 0, by
    have hk : k % 2 < 2 := Nat.mod_lt k (by omega)
    interval_cases (k % 2) <;> decide⟩
```

BalancedRepetitionThreshold.y'9 (BalancedRepetitionThreshold/WitnessV9.lean:45-48) — Definition 11 and Table 2 (periodic word y'_9)

```
def y'9 (k : ℕ) : Fin 9 :=
  ⟨y'9Tab.getD (k % 24) 0, by
    have hk : k % 24 < 24 := Nat.mod_lt k (by omega)
    interval_cases (k % 24) <;> decide⟩
```

BalancedRepetitionThreshold.v9 (BalancedRepetitionThreshold/WitnessV9.lean:51-52) — Definition 11 and Table 2 (the word v_9)

```
noncomputable def v9 : ℕ → Fin 9 :=
  colour (sturmian α9) y9 y'9
```

BalancedRepetitionThreshold.α10 (BalancedRepetitionThreshold/WitnessV10.lean:30-30) — Definition 11 and Table 2 (frequency alpha_10)

```
noncomputable def α10 : ℝ := (7 + Real.sqrt 5) / 22
```

BalancedRepetitionThreshold.y10Tab (BalancedRepetitionThreshold/WitnessV10.lean:33-33) — Definition 11 and Table 2 (table of y_10)

```
def y10Tab : List ℕ := [0, 1, 2, 3, 4]
```

BalancedRepetitionThreshold.y'10Tab (BalancedRepetitionThreshold/WitnessV10.lean:36-36) — Definition 11 and Table 2 (table of y'_10)

```
def y'10Tab : List ℕ := [5, 6, 7, 8, 9]
```

BalancedRepetitionThreshold.y10 (BalancedRepetitionThreshold/WitnessV10.lean:39-42) — Definition 11 and Table 2 (periodic word y_10)

```
def y10 (k : ℕ) : Fin 10 :=
  ⟨y10Tab.getD (k % 5) 0, by
    have hk : k % 5 < 5 := Nat.mod_lt k (by omega)
    interval_cases (k % 5) <;> decide⟩
```

BalancedRepetitionThreshold.y'10 (BalancedRepetitionThreshold/WitnessV10.lean:45-48) — Definition 11 and Table 2 (periodic word y'_10)

```
def y'10 (k : ℕ) : Fin 10 :=
  ⟨y'10Tab.getD (k % 5) 0, by
    have hk : k % 5 < 5 := Nat.mod_lt k (by omega)
    interval_cases (k % 5) <;> decide⟩
```

BalancedRepetitionThreshold.v10 (BalancedRepetitionThreshold/WitnessV10.lean:51-52) — Definition 11 and Table 2 (the word v_10)

```
noncomputable def v10 : ℕ → Fin 10 :=
  colour (sturmian α10) y10 y'10
```

Box 2 — Verbatim Main Theorem Signatures, Context Binders & Applicability

BalancedRepetitionThreshold.rtb_7_eq (BalancedRepetitionThreshold/Main.lean:44-45) src:cabde9662ea8b4

```
theorem rtb_7_eq : RTB 7 = (((6 : ℝ) / (5 : ℝ) : ℝ) : EReal)
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.rtb_9_eq (BalancedRepetitionThreshold/Main.lean:67-68) src:4d67a3f2e8e50b

```
theorem rtb_9_eq : RTB 9 = (((8 : ℝ) / (7 : ℝ) : ℝ) : EReal)
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.rtb_10_eq (BalancedRepetitionThreshold/Main.lean:90-91) src:2971f0e8664417

```
theorem rtb_10_eq : RTB 10 = (((9 : ℝ) / (8 : ℝ) : ℝ) : EReal)
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.rtb_7_lt_printed (BalancedRepetitionThreshold/Main.lean:52-54) src:de6f827644e07c

```
theorem rtb_7_lt_printed : RTB 7 < (((5 : ℝ) / (4 : ℝ) : ℝ) : EReal)
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.rtb_9_lt_printed (BalancedRepetitionThreshold/Main.lean:75-77) src:329e74e83cdd80

```
theorem rtb_9_lt_printed : RTB 9 < (((7 : ℝ) / (6 : ℝ) : ℝ) : EReal)
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.rtb_10_lt_printed (BalancedRepetitionThreshold/Main.lean:98-100) src:dc0abb29d26166

```
theorem rtb_10_lt_printed : RTB 10 < (((8 : ℝ) / (7 : ℝ) : ℝ) : EReal)
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.rtb_strict_7_eq (BalancedRepetitionThreshold/Main.lean:48-49) src:71922d8fb6a97c

```
theorem rtb_strict_7_eq : RTB_strict 7 = (((6 : ℝ) / (5 : ℝ) : ℝ) : EReal)
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.rtb_strict_9_eq (BalancedRepetitionThreshold/Main.lean:71-72) src:cc716cae542a25

```
theorem rtb_strict_9_eq : RTB_strict 9 = (((8 : ℝ) / (7 : ℝ) : ℝ) : EReal)
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.rtb_strict_10_eq (BalancedRepetitionThreshold/Main.lean:94-95) src:b23b641b9c872b

```
theorem rtb_strict_10_eq : RTB_strict 10 = (((9 : ℝ) / (8 : ℝ) : ℝ) : EReal)
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.critExp_ge_7 (BalancedRepetitionThreshold/LowerBound7.lean:267-270) src:c5615d8b58a026

```
theorem critExp_ge_7 (w : ℕ → Fin 7) (hw : IsBalanced w) :
    (((6 : ℝ) / (5 : ℝ) : ℝ) : EReal) ≤ critExp w
```

Assumptions: w is a balanced word on 7 letters. The hypothesis is satisfiable: the word v_7 is balanced. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

BalancedRepetitionThreshold.critExp_ge_9 (BalancedRepetitionThreshold/LowerBound9.lean:1029-1032) src:c9ba97fef25a1c

```
theorem critExp_ge_9 (w : ℕ → Fin 9) (hw : IsBalanced w) :
    (((8 : ℝ) / (7 : ℝ) : ℝ) : EReal) ≤ critExp w
```

Assumptions: w is a balanced word on 9 letters. The hypothesis is satisfiable: the word v_9 is balanced. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

BalancedRepetitionThreshold.critExp_ge_10 (BalancedRepetitionThreshold/LowerBound10.lean:2076-2079) src:f7087b73b002b7

```
theorem critExp_ge_10 (w : ℕ → Fin 10) (hw : IsBalanced w) :
    (((9 : ℝ) / (8 : ℝ) : ℝ) : EReal) ≤ critExp w
```

Assumptions: w is a balanced word on 10 letters. The hypothesis is satisfiable: the word v_10 is balanced. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

BalancedRepetitionThreshold.v7_isBalanced (BalancedRepetitionThreshold/WitnessV7.lean:101-133) src:9b5c700eea0887

```
theorem v7_isBalanced : IsBalanced v7
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.v9_isBalanced (BalancedRepetitionThreshold/WitnessV9.lean:101-135) src:ab178accd2a100

```
theorem v9_isBalanced : IsBalanced v9
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.v10_isBalanced (BalancedRepetitionThreshold/WitnessV10.lean:101-132) src:6298885981a6d4

```
theorem v10_isBalanced : IsBalanced v10
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.v7_critExp_eq (BalancedRepetitionThreshold/WitnessV7.lean:504-507) src:205a84faca2d4e

```
theorem v7_critExp_eq :
    critExp v7 = (((6 : ℝ) / (5 : ℝ) : ℝ) : EReal) ∧
    (((6 : ℝ) / (5 : ℝ) : ℝ) : EReal) ∈ minPeriodExponentSet v7
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.v9_critExp_eq (BalancedRepetitionThreshold/WitnessV9.lean:561-564) src:5ebe2f8259fab3

```
theorem v9_critExp_eq :
    critExp v9 = (((8 : ℝ) / (7 : ℝ) : ℝ) : EReal) ∧
    (((8 : ℝ) / (7 : ℝ) : ℝ) : EReal) ∈ minPeriodExponentSet v9
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

BalancedRepetitionThreshold.v10_critExp_eq (BalancedRepetitionThreshold/WitnessV10.lean:446-449) src:b27851acc14dd7

```
theorem v10_critExp_eq :
    critExp v10 = (((9 : ℝ) / (8 : ℝ) : ℝ) : EReal) ∧
    (((9 : ℝ) / (8 : ℝ) : ℝ) : EReal) ∈ minPeriodExponentSet v10
```

Assumptions: None beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

Table 4. Lean 4 Type-Trap Audit Checklist — Source-Level Approximation (Gate Replay Pending).[a]

Declaration	Hazard	Operand	Category	Audited Disposition & Guard
`countTrue`	`finset_range`	`Finset.range`	intended_convention	Intended: the count runs over the positions strictly before n, as in Definition 10; for $n = 0$ the range is empty and the count is 0.
`countFalse`	`finset_range`	`Finset.range`	intended_convention	Intended: the count runs over the positions strictly before n, as in Definition 10; for $n = 0$ the range is empty and the count is 0.
`exponentSet`	`real_div`	`(p : ℝ)`	intended_convention	Intended: the same set-builder clause requires a period, and HasPeriod has $0 <$ p as a conjunct, so L/p is ordinary real division; the Lean convention $x/0 = 0$ is never reached.
`minPeriodExponentSet`	`real_div`	`(p : ℝ)`	intended_convention	Intended: the same set-builder clause requires a period, and HasPeriod has $0 <$ p as a conjunct, so L/p is ordinary real division; the Lean convention $x/0 = 0$ is never reached.
`y'10`	`nat_mod`	`% 5`	intended_convention	Intended: k mod 5 indexes the period table of length 5, giving the periodic word of Table 2.
`y'7`	`nat_mod`	`% 12`	intended_convention	Intended: k mod 12 indexes the period table of length 12, giving the periodic word of Table 2.
`y'9`	`nat_mod`	`% 24`	intended_convention	Intended: k mod 24 indexes the period table of length 24, giving the periodic word of Table 2.
`y10`	`nat_mod`	`% 5`	intended_convention	Intended: k mod 5 indexes the period table of length 5, giving the periodic word of Table 2.
`y7`	`nat_mod`	`% 2`	intended_convention	Intended: k mod 2 indexes the period table of length 2, giving the periodic word of Table 2.
`y9`	`nat_mod`	`% 2`	intended_convention	Intended: k mod 2 indexes the period table of length 2, giving the periodic word of Table 2.
`α10`	`real_log_rpow`	`Real.sqrt`	intended_convention	Intended: the ordinary real square root of 5; the Lean convention for negative arguments is never reached.
`α7`	`real_log_rpow`	`Real.sqrt`	intended_convention	Intended: the ordinary real square root of 10; the Lean convention for negative arguments is never reached.
`α9`	`real_log_rpow`	`Real.sqrt`	intended_convention	Intended: the ordinary real square root of 2; the Lean convention for negative arguments is never reached.

[a]Covers the specified Lean 4 hazard set (`Nat` subtraction/division/mod, totalized real/rational division, `Real.log`/`rpow`, `Finset.range`, `Classical.choose`, local instances); no finite checklist certifies all semantics.

Table 5. Per-Claim Reader's Trust Boundary Ledger (G11 Denominators Derived Mechanically).[a]

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Thm. 1: RTB(7) = 6/5: the infimum of E(w) over all balanced words on 7 letters equals 6/5	Lean Kernel	`BalancedRepetitionThreshold.rtb_7_eq`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Thm. 2: RTB(9) = 8/7: the infimum of E(w) over all balanced words on 9 letters equals 8/7	Lean Kernel	`BalancedRepetitionThreshold.rtb_9_eq`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Thm. 3: RTB(10) = 9/8: the infimum of E(w) over all balanced words on 10 letters equals 9/8	Lean Kernel	`BalancedRepetitionThreshold.rtb_10_eq`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Cor. 4: RTB(7) < 5/4, the value listed in the literature	Lean Kernel	`BalancedRepetitionThreshold.rtb_7_lt_printed`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Cor. 5: RTB(9) < 7/6, the value listed in the literature	Lean Kernel	`BalancedRepetitionThreshold.rtb_9_lt_printed`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Cor. 6: RTB(10) < 8/7, the value listed in the literature	Lean Kernel	`BalancedRepetitionThreshold.rtb_10_lt_printed`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Prop. 12: RTB*(7) = 6/5: the same infimum over balanced words on 7 letters that use all letters	Lean Kernel	`BalancedRepetitionThreshold.rtb_strict_7_eq`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Prop. 13: RTB*(9) = 8/7: the same infimum over balanced words on 9 letters that use all letters	Lean Kernel	`BalancedRepetitionThreshold.rtb_strict_9_eq`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Prop. 14: RTB*(10) = 9/8: the same infimum over balanced words on 10 letters that use all letters	Lean Kernel	`BalancedRepetitionThreshold.rtb_strict_10_eq`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Prop. 15: Every balanced word w on 7 letters has E(w) >= 6/5	Lean Kernel	`BalancedRepetitionThreshold.critExp_ge_7`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Prop. 16: Every balanced word w on 9 letters has E(w) >= 8/7	Lean Kernel	`BalancedRepetitionThreshold.critExp_ge_9`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Prop. 17: Every balanced word w on 10 letters has E(w) >= 9/8	Lean Kernel	`BalancedRepetitionThreshold.critExp_ge_10`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Lem. 18: The explicit word v_7 is balanced	Lean Kernel	`BalancedRepetitionThreshold.v7_isBalanced`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Lem. 19: The explicit word v_9 is balanced	Lean Kernel	`BalancedRepetitionThreshold.v9_isBalanced`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Lem. 20: The explicit word v_10 is balanced	Lean Kernel	`BalancedRepetitionThreshold.v10_isBalanced`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Prop. 21: E(v_7) = 6/5, attained as L/p by a factor of length L with least period p	Lean Kernel	`BalancedRepetitionThreshold.v7_critExp_eq`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Prop. 22: E(v_9) = 8/7, attained as L/p by a factor of length L with least period p	Lean Kernel	`BalancedRepetitionThreshold.v9_critExp_eq`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.
Prop. 23: E(v_10) = 9/8, attained as L/p by a factor of length L with least period p	Lean Kernel	`BalancedRepetitionThreshold.v10_critExp_eq`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound, and the Box 1 definitions; no other hypothesis.

[a]Allowed pure-kernel axioms: `propext`, `Classical.choice`, `Quot.sound`. Actual axioms printed above are read directly from verified outputs. Provenance nonce: d5c0150a1bed | Lean mode: committed_receipt | Closure SHA-256: 7f65ec9eacfa | Git anchor: ae4ef2f80d3a | Axioms SHA-256: a53b1cb4027c | Fidelity SHA-256: ac7e112cc5b2.

6. PRIOR WORK AND OPEN QUESTIONS

Sources. Rampersad, Shallit and Vandomme [1] introduced the problem, constructed balanced words of small critical exponent on 3 and on 4 letters, defined the words x_k of their Table 1, and stated the conjecture and the backtracking claim quoted in Section 1. Baranwal and Shallit [2] proved $E(x_5) = 3/2$ with automata and determined the least critical exponent on 5 letters by a breadth-first search; our tree program reproduces their longest words. Dolce, Dvořáková and Pelantová [3] developed a method for the critical exponent of colourings of Sturmian words with quadratic slope and computed $E(x_9) = 7/6$ and $E(x_{10}) = 8/7$; for minimality on 9 letters they rely on [1], writing that it "was shown ibidem that $E(v) \geq 7/6$ for every balanced sequence over a 9-letter alphabet". Proposition 22 contradicts that inequality. Dvořáková, Pelantová, Opočenská and Shur [4] proved the lower bound $(d-1)/(d-2)$ and its attainment for $d = 11$ and even $d \geq 12$, and conjectured that it is the threshold for all $d \geq 11$ (Conjecture 25 there). Their Remark 22 explains that the proof of Theorem 19 breaks down at $\delta = 6$; the word v_{10} corresponds to $\delta = 5$, which that theorem does not cover either. The abstract of [6] states that the threshold "is known to be at least $1 + 1/(d-2)$ for each $d \geq 3$". The lists in [5, 7] are quoted in Section 1. We read [1] and [4] in full in their arXiv versions and the other sources in part, as recorded in `references-checked.md`; sources not read in full are marked `[to check]` in the bibliography.

Search status. The evidence folder of the finding contains seven logs of literature searches (arXiv, Google Scholar, CORE, OpenAlex, Semantic Scholar, zbMATH, OEIS and two question-and-answer sites). Its own checker passes 6 of the 7 logs; the remaining log lacks the Google Scholar and CORE queries, so we make no claim about whether the values of Theorems 1 to 3 appear elsewhere in the literature. Not searched, according to the logs: the journal versions of [1, 4], the conference proceedings versions of the cited papers, the theses of the authors involved, and books. A correction that appeared only in one of those places would have been missed.

Caveats on the panels of Section 5. The Lean library was not rebuilt for this draft. The panels cite the build receipt committed with the library, and a kernel replay by the gate of this paper is pending. The coverage row counts theorems by a source-level scan that loses the namespace of the 17 theorems in `LowerBoundEngine.lean`, which follow a `mutual` block; it therefore matches 176 of the 193 theorems that the committed axiom report lists, and `recompute.py` matches all 193 by name. None of the 18 numbered statements of this paper cites one of those 17 theorems directly. The fidelity row records a statement-by-statement comparison by a model of a different family from the one that drafted this paper, made in two single requests and reproduced in `fidelity-notes.md`. The library does not record which model family wrote it, so independence of the auditor from the formalizer is not established, and the audit has not been accepted by a coordinator.

Limits of this draft. The written proofs of the upper bound that accompanied the discovery are not part of the bundle this paper was written from, so the upper-bound part of Section 3 is a guide to the Lean proof only. The programs that located the slopes are also outside the bundle; the centre panel of Figure 1 reports what the evidence folder says about them and was not re-run.

Open questions. (1) The lower-bound statements for $d = 6$ and $d = 8$ go back to the same backtracking searches as those for $d = 7$ and $d = 9$. Are $\mathrm{RTB}(6) = 4/3$ and $\mathrm{RTB}(8) = 6/5$ correct? For $d = 8$ the bound in print leaves the interval from $7/6$ to $6/5$. We have not examined either case. (2) Is $\mathrm{RTB}(d) = (d-1)/(d-2)$ for every odd $d \geq 13$, as conjectured in [4]? Our methods are specific to each alphabet and say nothing about this. (3) Find a written proof of Propositions 21 to 23 short enough to be read without a proof assistant, for instance by the method of bispecial factors and return words of [3].

7. DATA AVAILABILITY, REPRODUCIBILITY, AND USE OF AI

The Lean 4 library `BalancedRepetitionThreshold` (Lean `v4.35.0`, Mathlib `v4.35.0`) with its axiom report and build receipt, and the evidence folder with the search logs, are in the finding bundle `findings/balanced-sequences-critical-exponent` of the `mathdisc` repository. The paper directory contains `recompute.py` and `recompute.out`, the figure scripts under `figures/`, the ledgers `claims.md` and `references-checked.md`, and the manifests behind Section 5. Running `python3 recompute.py` regenerates every number in Section 4.

AI research agents built and directed by Tyler B. Hughes generated and checked this work: they found the words, wrote and audited the Lean library, and drafted this paper. No human mathematician has yet refereed the proof.

Drafted: November 2026 **Updated:** November 24, 2026

REFERENCES

(1) N. Rampersad, J. Shallit and É. Vandomme, *Critical exponents of infinite balanced words*, Theoret. Comput. Sci. 777 (2019), 454–463; read as arXiv:1801.05334v1.

(2) A. R. Baranwal and J. Shallit, *Critical exponent of infinite balanced words via the Pell number system*, arXiv:1902.00503v3 (2019). [to check]

(3) F. Dolce, L. Dvořáková and E. Pelantová, *On balanced sequences and their critical exponent*, arXiv:2108.07503v2 (2022). [to check]

(4) L. Dvořáková, E. Pelantová, D. Opočenská and A. M. Shur, *On minimal critical exponent of balanced sequences*, Theoret. Comput. Sci. 922 (2022); read as arXiv:2112.02854v1.

(5) L. Dvořáková, D. Opočenská and E. Pelantová, *Asymptotic repetitive threshold of balanced sequences*, arXiv:2208.00366v1 (2022). [to check]

(6) L. Dvořáková and E. Pelantová, *An upper bound on asymptotic repetitive threshold of balanced sequences via colouring of the Fibonacci sequence*, arXiv:2211.11877v1 (2022). [to check]

(7) L. Dvořáková and E. Pelantová, *The repetition threshold of episturmian sequences*, arXiv:2309.00988v1 (2023). [to check]

(8) P. Hubert, *Suites équilibrées*, Theoret. Comput. Sci. 242 (2000), 91–108. [to check]