EXACT MSD-FIRST STATE COMPLEXITY OF SHIFTS OF THE FIBONACCI WORD

KOREA SUPERINTELLIGENCE LABS

ABSTRACT. Let $\mathbf{f} = f(0)f(1)f(2)\cdots = 01001010\cdots$ be the Fibonacci word and let $\mathrm{sc}(c)$ be the number of states of the smallest deterministic finite automaton with output that reads a Zeckendorf representation most-significant-digit first and generates the shifted word $(f(i+c))_{i\ge 0}$. Moradi, Popoli, Shallit and Vukusic determined the least-significant-digit-first state complexity of these shifts exactly, and proved that $\mathrm{sc}(c) = O(\log c)$ with no explicit constant, no lower bound, and no table of values. We compute $\mathrm{sc}(c)$ exactly for 112 shifts — every $c \le 100$, together with eleven larger ones ranging up to $c = 20000$ — the values running from 2 to 26; for each of these c with $c \ge 2$ we show in addition that the automaton produced by their construction is already minimal, which they do not claim, and that

$$\mathrm{sc}(c) \;=\; |(c)_F| + \nu\big(F_{|(c)_F|+2} - 1 - c\big),$$

where $|(c)_F|$ is the number of Zeckendorf digits of c and $\nu(d)$ is the index of the lowest Fibonacci number occurring in $(d)_F$, with $\nu(0) := |(c)_F|$. An exact computation in $\mathbb{Z}[\varphi]$ confirms the identity for every $2 \le c \le 4000$, and its right-hand side against the size of their construction to $c = 20000$; beyond that it is a conjecture, whose missing half we isolate. The instrument is a change of coordinate: writing $\varepsilon(w) = \mathrm{sh}(w) - [w]_F\varphi \in \mathbb{Z}[\varphi]$ replaces the circle used in the earlier work by an interval of length exactly one on which appending a digit is a contraction, so that the correctness of an automaton for *all* i becomes a finite check in $\mathbb{Z}[\varphi]$ — in particular no appeal to an automated prover for automatic sequences is needed. All statements below are machine-checked in Lean 4.

1. INTRODUCTION

The *Fibonacci word*

$$\mathbf{f} \;=\; f(0)f(1)f(2)\cdots \;=\; 01001010\cdots$$

is one of the most studied infinite words in combinatorics on words [5]. Among its many equivalent descriptions — the fixed point of $0 \mapsto 01$, $1 \mapsto 0$; the limit of $X_i = X_{i-1}X_{i-2}$ with $X_0 = 1$, $X_1 = 0$; the characteristic Sturmian word of slope $2-\varphi$ — the one that governs everything below is arithmetic: *the i-th symbol of $\mathbf{f}$ is the last digit of the Zeckendorf representation of i* [1, Definition (d)].

Zeckendorf's theorem [7, 8] writes every $n \ge 0$ uniquely as $n = \sum_{2\le i\le t} a_i F_i$ with $a_i a_{i+1} = 0$, where $F_0 = 0$, $F_1 = F_2 = 1$ and $F_{i+2} = F_{i+1} + F_i$. Reading the digits from the top gives the word $(n)_F = a_t a_{t-1}\cdots a_2$ over $\{0,1\}$ with no factor 11; conversely a binary word $x = x_{\ell-1}\cdots x_0$ denotes the integer $[x]_F = \sum_i x_i F_{i+2}$. A *Fibonacci DFAO* is a deterministic finite automaton with output that is fed $(n)_F$ and whose output at the state reached is the n-th term of the sequence it generates; it is *msd-first* when the digits are read from the most significant end. As is standard, and as in [1], the output must not change when leading zeros are prepended, and words containing 11 are not required to reach any state at all. The number of states of the smallest msd-first Fibonacci DFAO generating a sequence g is its *msd-first state complexity*. Automatic sequences in this sense go back to Cobham [6]; the systematic reference is [4].

Shifting is the operation this paper is about. Write

$$\mathbf{f}_c \;=\; (f(i+c))_{i\ge 0}, \qquad \mathrm{sc}(c) \;=\; \text{the msd-first state complexity of } \mathbf{f}_c.$$

How expensive a shift is depends sharply on the sequence and on the digit order. For the Thue–Morse sequence the msd-first state complexity of the c-shift is at most $\frac{10}{3}c$ but exceeds $c^{0.694}$ infinitely often [3], whereas for the characteristic sequence of the powers of 2 it is $O(\log c)$ [3]; logarithmic growth is close to the least conceivable blow-up, since a counting argument shows that for an automatic sequence that is not eventually periodic the shift complexity exceeds $c'(\log c)/(\log\log c)$ infinitely often [1, §1].

Moradi, Popoli, Shallit and Vukusic [1] proved that $\mathbf{f}_c$ has state complexity $O(\log c)$ in both digit orders, which is close to the information-theoretic minimum for an aperiodic sequence, and that the two sides behave very differently. On the lsd-first side they are exact: for $c \geq 5$ the minimal lsd-first Fibonacci DFAO for $\mathbf{f}_c$ has

$$\mathrm{sc}_{\mathrm{lsd}}(c) \;=\; 2\big(|(c)_F| + 1\big) + 1 - g(c) \qquad \text{states}, \tag{1}$$

where $g(c) = 1$ if $f(c-1) = f(c-2) = 0$ and $g(c) = 0$ otherwise [1, Theorem 2]. On the msd-first side their result is one-sided:

> *For all integers $c \geq 0$ there is a Fibonacci DFAO of $O(\log c)$ states that generates the shifted Fibonacci word $(f(i+c))_{i\geq 0}$ in msd-first format* [1, Theorem 3].

No implied constant is stated, no lower bound is proved, and no msd-first value is tabulated anywhere in [1]; the only two msd-first counts printed there are the two states of the automaton for $\mathbf{f}$ itself [1, Fig. 1] and the eight intervals of the partition built for $c = 10$ [1, Example 3]. The companion paper of Moradi, Rampersad and Shallit on linear subsequences of Fibonacci-automatic sequences [2] repeats only the $O(\log c)$ statement. So on the msd-first side three questions are left open: what the values are; whether the construction that proves the upper bound is optimal — an optimality claim that [1] makes for the lsd-first construction only; and whether there is a closed form to match (1).

This paper answers all three on an explicit range, and identifies exactly what is missing for a proof in general.

Results. Let

$$\mathcal{W} = \{0, 1, \ldots, 100\} \cup \{144, 233, 610, 987, 1000, 1597, 2584, 4181, 6765, 10000, 20000\}, \qquad |\mathcal{W}| = 112.$$

(1) For every $c \in \mathcal{W}$ we determine $\mathrm{sc}(c)$ exactly (Theorem 5.1 and Table 1); the values run from 2 to 26. Each is proved from both sides: an explicit automaton correct for *every* i, and a Myhill–Nerode separation table valid against every competing automaton.

(2) For every $c \in \mathcal{W}$ with $c \geq 2$, the automaton that [1, proof of Theorem 3] constructs — its states being the intervals of the partition $\mathcal{P}(-(c+1)) \cup \mathcal{P}(-(c+2))$ — already has exactly $\mathrm{sc}(c)$ states, so that construction is optimal and not merely $O(\log c)$ (Theorem 5.2).

(3) For those same c, $\mathrm{sc}(c) = |(c)_F| + \nu(F_{|(c)_F|+2} - 1 - c)$ (Theorem 6.1); outside the formal development an exact computation confirms the identity for $2 \leq c \leq 4000$ and its right-hand side against the construction of [1] to $c = 20000$ (Remark 6.3); beyond that it is a conjecture (Conjecture 6.4). On that range it gives $|(c)_F| + 2 \leq \mathrm{sc}(c) \leq 2|(c)_F|$, the upper extreme occurring exactly at $c = F_k - 1$ (Corollary 6.2).

(4) The tool behind all of this is a coordinate ε (§3) in which the circle of [1] becomes an interval of length exactly one, appending a digit becomes a contraction, and the correctness of a candidate automaton for all i becomes a finite computation in $\mathbb{Z}[\varphi]$ (Theorem 3.1 and Theorem 3.4). In [1] the corresponding step is the consistency lemma [1, Lemma 5], whose proof there rests on two auxiliary lemmas discharged by the automated prover `Walnut` [10, 11].

Comparing (3) with (1): on $5 \leq c \leq 20000$ the msd-first complexity is always at least two below the lsd-first complexity, and the closed form of (3) attains $\mathrm{sc}_{\mathrm{lsd}}(c) - 2$ exactly at the eight values $c = 12, 33, 88, 232, 609, 1596, 4180, 10945$ (Remark 6.6). Msd-first is cheaper, and between about half and all of the lsd-first cost.

Method, in one paragraph. Everything rests on a single change of variable. For a msd-first digit word $w = w_{\ell-1} \cdots w_0$ put $\varepsilon(w) = \sum_i w_i \psi^{i+2}$ with $\psi = 1 - \varphi = -1/\varphi$. Because $\psi^i = F_{i+1} - F_i\varphi$, this is an element $\mathrm{sh}(w) - [w]_F\varphi$ of $\mathbb{Z}[\varphi]$ with $\mathrm{sh}(w) = \sum_i w_i F_{i+3}$; in particular $\varepsilon(w) \equiv -[w]_F\varphi \pmod 1$, so ε carries the same information as the point $\{n\varphi\}$ on which [1] works. What ε adds is that all valid words land in the interval $(\varphi - 2, \varphi - 1)$, whose length is exactly 1, and that appending a digit acts by the affine contraction $\varepsilon \mapsto \psi\varepsilon + a\psi^2$. Length exactly one makes the representative modulo 1 unique, so an automaton is nothing but a partition of $[\varphi - 2, \varphi - 1)$ closed under the two contractions, and its correctness for all i is a finite list of comparisons in $\mathbb{Z}[\varphi]$ — each of which is an integer comparison, since $a + b\varphi < a' + b'\varphi$ is decided by comparing $5B^2$ with D^2 for $B = b - b'$ and $D = 2(a' - a) - B$.

No Diophantine approximation and no external prover enter. Lower bounds come from the classical Myhill–Nerode argument, in the form appropriate to automata whose transition function may be partial.

Organisation. §2 fixes notation and the automaton model. §3 develops the ε coordinate and the certificate that upper-bounds $\mathrm{sc}(c)$; §4 gives the matching lower bound. §5 contains the exact values and the minimality of the construction of [1]; §6 the closed form, the elementary count that proves its upper half in general, and the numerical evidence. §7 describes the formal verification and says exactly which statements rest on exhaustive computation. §8 lists what is left open.

2. Preliminaries

Throughout, $\varphi = (1+\sqrt{5})/2$ and $\psi = 1-\varphi = -1/\varphi$, so that $\varphi^2 = \varphi+1$ and $\psi^2 = 2-\varphi$. Fibonacci numbers are indexed by $F_0 = 0$, $F_1 = F_2 = 1$, $F_{i+2} = F_{i+1} + F_i$, and we use the classical identity

$$\psi^i \;=\; F_{i+1} - F_i\varphi \qquad (i \geq 0). \tag{2}$$

A *word* is a finite binary word $w = w_{\ell-1}\cdots w_0$, written with the most significant digit on the left; it is *valid* if it contains no factor 11. Leading zeros are permitted, so a valid word need not be the Zeckendorf representation of its value

$$[w]_F = \sum_{i=0}^{\ell-1} w_i F_{i+2}.$$

We write $(n)_F$ for the Zeckendorf representation of n, that is the unique valid word with no leading zero and value n, and $|(n)_F|$ for its length; thus $|(n)_F| = L$ where L is least with $n < F_{L+2}$, and $|(0)_F| = 0$. The Fibonacci word is

$$f(n) \;=\; \text{the last digit of } (n)_F,$$

with $f(0) = 0$, and $\mathbf{f}_c = (f(i+c))_{i\geq 0}$.

Definition 2.1. A *msd-first Fibonacci DFAO* on n states is a tuple $M = (Q, \delta, \tau, q_0)$ with $|Q| = n$, a *partial* transition function $\delta\colon Q \times \{0,1\} \to Q$, an output $\tau\colon Q \to \{0,1\}$ and an initial state $q_0 \in Q$. For a word w we write $\delta(q,w)$ for the state reached by reading w from q one digit at a time, undefined as soon as some transition on the way is undefined. We say M *generates* the sequence $g\colon \mathbb{N} \to \{0,1\}$ if for *every* valid word w the state $\delta(q_0,w)$ is defined and $\tau(\delta(q_0,w)) = g([w]_F)$. The *msd-first state complexity* $\mathrm{sc}(g)$ is the least n for which such an M exists, and we abbreviate $\mathrm{sc}(c) = \mathrm{sc}(\mathbf{f}_c)$.

Two features of Definition 2.1 follow [1] and matter for the exactness of what follows. First, δ is partial: *"our convention is that inputs that form invalid Zeckendorf representations (i.e., those with two consecutive* 1*'s) do not reach any state and do not produce any output"* [1, §2], so an automaton is not penalised for having no transition on a digit that would create a factor 11, and states it does not have are not counted. Second, the requirement is over *all* valid words, not only over the words $(n)_F$; this is exactly the leading-zero convention of [1, §1], which asks that "all DFAO's give the same correct output no matter how many leading zeros there are in the input". Since $\mathrm{sc}(g)$ is defined as a least element, an upper bound and a lower bound of the same size pin it exactly, and both a too-small and a too-large claim about it are refutable; see Proposition 5.4.

Finally, for $d \geq 1$ let

$$\nu(d) \;=\; \min\{\, i \;:\; F_i \text{ occurs in } (d)_F \,\} \;\geq\; 2$$

be the index of the lowest Fibonacci number in the Zeckendorf representation of d.

3. The coordinate ε and interval certificates

3.1. **An interval of length one.** For a word $w = w_{\ell-1}\cdots w_0$ set $\mathrm{sh}(w) = \sum_i w_i F_{i+3}$ — the value of w with every digit moved one Fibonacci place up — and

$$\varepsilon(w) \;=\; \sum_{i=0}^{\ell-1} w_i \psi^{i+2} \;\overset{(2)}{=}\; \mathrm{sh}(w) - [w]_F\,\varphi \;\in\; \mathbb{Z}[\varphi].$$

We record ε as the pair $(\mathrm{sh}(w), -[w]_F) \in \mathbb{Z}^2$; every quantity in this section is an element of $\mathbb{Z}[\varphi]$, and every comparison between two of them is exact: $a + b\varphi < a' + b'\varphi$ holds if and only if $B\sqrt{5} < D$ where $B = b - b'$ and $D = 2(a' - a) - B$, which is decided by comparing the integers $5B^2$ and D^2 together with the signs of B and D.

Because $\mathrm{sh}(w) \in \mathbb{Z}$, we have $\varepsilon(w) \equiv -[w]_F\varphi \pmod 1$: the coordinate ε is the point $\{n\varphi\}$ of [1] up to sign and a choice of representative. The choice is what makes it useful. Appending a digit a on the right turns ε into

$$\varepsilon(wa) \;=\; T_a(\varepsilon(w)), \qquad T_a(x) = \psi x + a\psi^2, \tag{3}$$

which in coordinates is the integral map $(x_1, x_2) \mapsto (x_1 - x_2 + 2a,\ -x_1 - a)$; this is the operation $n \mapsto n^{(a)}$ of [1, §4] transported to ε. Since $\psi < 0$, each T_a reverses order.

Theorem 3.1 (The interval invariant)**.** *For every valid word w,*

$$\varphi - 2 \;<\; \varepsilon(w) \;<\; \varphi - 1,$$

and $\varepsilon(w) > 2\varphi - 3$ holds if and only if the last digit of w is 1 (the empty word counting as last digit 0). Consequently, for every $n \geq 0$,

$$f(n) = 1 \iff \varepsilon((n)_F) > 2\varphi - 3.$$

Proof sketch. Induction on the length of w, appending one digit at a time. For the empty word $\varepsilon = 0$ and $\varphi - 2 < 0 < 2\varphi - 3$. A direct computation with $\varphi^2 = \varphi + 1$ gives

$$T_0(\varphi - 1) = \varphi - 2, \quad T_0(\varphi - 2) = 2\varphi - 3, \quad T_1(2\varphi - 3) = 2\varphi - 3, \quad T_1(\varphi - 2) = \varphi - 1,$$

so, T_a being order-reversing, T_0 maps $(\varphi - 2, \varphi - 1)$ onto $(\varphi - 2, 2\varphi - 3)$ and T_1 maps $(\varphi - 2, 2\varphi - 3)$ onto $(2\varphi - 3, \varphi - 1)$: the two images tile the interval, the lower half being where a word ends in 0 and the upper half where it ends in 1. Validity is used exactly once, in the step appending 1: it forces the previous digit to be 0, hence $\varepsilon(w) < 2\varphi - 3$ by the inductive hypothesis, which is the hypothesis T_1 needs. The last sentence is the case $w = (n)_F$, since $f(n)$ is by definition the last digit of $(n)_F$. □

The interval $E = [\varphi - 2, \varphi - 1)$ has length exactly 1. Hence:

Corollary 3.2. *If $x, y \in \mathbb{Z}[\varphi]$ both lie in E and $x - y \in \mathbb{Z}$, then $x = y$. In particular, $\varepsilon((n)_F)$ is the unique element of E congruent to $-n\varphi$ modulo 1.*

This is the entire content of the reduction: on the circle of [1] one must argue that a construction is "consistent with the msd-first Fibonacci representation" [1, Definition 2, Lemma 5], a statement about all n at once; on the interval E the same statement is a containment of intervals under the two affine maps T_0, T_1, and containments of intervals are finitely many comparisons in $\mathbb{Z}[\varphi]$.

3.2. **Interval certificates.**

Definition 3.3. An *interval certificate* for the shift c consists of points

$$\varphi - 2 = p_0 < p_1 < \cdots < p_{m-1} < \varphi - 1 \quad \text{in } \mathbb{Z}[\varphi], \qquad p_m := \varphi - 1,$$

which cut E into the intervals $I_j = [p_j, p_{j+1})$, $0 \leq j < m$; together with maps $d_0, d_1 \colon \{0, \ldots, m-1\} \to \{0, \ldots, m-1\}$, outputs $\tau_j \in \{0, 1\}$, integers $k_j \in \mathbb{Z}$, and an initial index j_0. It *passes* if

- (C1) $p_{j_0} < 0 < p_{j_0+1}$;
- (C2) $T_0(I_j) \subseteq [p_{d_0(j)}, p_{d_0(j)+1}]$ for every j;
- (C3) $T_1(I_j) \subseteq [p_{d_1(j)}, p_{d_1(j)+1}]$ for every j with $p_j < 2\varphi - 3$;
- (C4) for every j, the translate $I_j - k_j - c\varphi$ lies in $[\varphi - 2, \varphi - 1]$ and lies entirely on the side of $2\varphi - 3$ named by τ_j: above it if $\tau_j = 1$, below it if $\tau_j = 0$.

Its *automaton* $M(K)$ has state set $\{0, \ldots, m-1\}$, initial state j_0, output τ, and transitions $j \overset{0}{\mapsto} d_0(j)$ always and $j \overset{1}{\mapsto} d_1(j)$ only when $p_j < 2\varphi - 3$.

Each of (C1)–(C4) is a finite conjunction of comparisons in $\mathbb{Z}[\varphi]$, so passing is decided exactly by integer arithmetic. Note where the partial transition function of Definition 2.1 is used: if $p_j \geq 2\varphi - 3$ then by Theorem 3.1 every word landing in I_j ends in the digit 1, so appending another 1 would produce a factor 11 and no 1-transition out of j is required.

Theorem 3.4 (Soundness of the certificate)**.** *If an interval certificate K for c with m intervals passes, then $M(K)$ is a msd-first Fibonacci DFAO on m states generating $\mathbf{f}_c$; that is, for* every *valid word w the run of $M(K)$ on w is defined and outputs $f([w]_F + c)$. In particular* $\mathrm{sc}(c) \leq m$.

Proof sketch. One shows by induction on w, using (3) and (C1)–(C3), that after reading a valid word w the automaton is in the state j with $p_j < \varepsilon(w) < p_{j+1}$; this is where the order-reversal of T_a and the interval containments are used, and it is the analogue of the consistency lemma [1, Lemma 5]. For the output, fix such a j and put $x = \varepsilon(w) - k_j - c\varphi$. By (C4), x lies strictly inside $(\varphi - 2, \varphi - 1)$, and its φ-coefficient is $-([w]_F + c)$, so x and $\varepsilon(([w]_F + c)_F)$ are two elements of E differing by an integer; Corollary 3.2 makes them equal. Theorem 3.1 now reads $f([w]_F + c)$ off the position of x relative to $2\varphi - 3$, and (C4) says that position is constant on I_j and equal to τ_j. □

4. The lower bound

The upper bound above is matched by the Myhill–Nerode argument, in the form that survives both conventions of Definition 2.1.

Theorem 4.1 (Myhill–Nerode for partial msd-first DFAOs)**.** *Let $g\colon \mathbb{N} \to \{0,1\}$ and let $w_1, \ldots, w_r$ be valid words such that for all $s \neq t$ there is a word z with $w_s z$ and $w_t z$ both valid and $g([w_s z]_F) \neq g([w_t z]_F)$. Then every msd-first Fibonacci DFAO generating g has at least r states.*

Proof sketch. If M generates g then $\delta(q_0, w_s)$ is defined for every s, and if two of these states coincided then reading the separating z from that common state would produce a single output for two different values of g. So $s \mapsto \delta(q_0, w_s)$ is injective into the state set. □

Two points make Theorem 4.1 the right statement here. The separating word z is required to be valid *after both* prefixes, so only outputs on valid words are used; consequently the bound applies verbatim to an automaton whose transition function is total and which does whatever it likes on inputs containing 11, and no strength is gained or lost by the partiality convention. And the hypothesis is a finite table — a list of prefixes, a list of candidate suffixes, and the check that some suffix separates each pair — so it is verifiable by exhaustive evaluation.

5. Exact values

Theorem 5.1. *For each of the* 112 *shifts*

$$c \in \mathcal{W} = \{0, 1, \ldots, 100\} \cup \{144, 233, 610, 987, 1000, 1597, 2584, 4181, 6765, 10000, 20000\},$$

the msd-first state complexity $\mathrm{sc}(c)$ *of $\mathbf{f}_c$ is the value listed in Table 1. The values run from* $\mathrm{sc}(0) = 2$ *to* $\mathrm{sc}(10000) = 26$.

Proof sketch. For each $c \in \mathcal{W}$ two finite objects are exhibited and checked by exhaustive evaluation. The first is an interval certificate K_c (Definition 3.3) with $\mathrm{sc}(c)$ intervals; checking (C1)–(C4) costs a bounded number of comparisons of elements of $\mathbb{Z}[\varphi]$ per interval, and by Theorem 3.4 it gives an automaton correct for every i, hence the upper bound. The second is a table of $\mathrm{sc}(c)$ valid prefixes together with a list of candidate suffixes; the check is that every one of the $\binom{\mathrm{sc}(c)}{2}$ pairs of prefixes is separated by some suffix in the list, in the sense of Theorem 4.1, which gives the matching lower bound. Nothing about how the two objects were found enters the proof: the certificates were produced from the construction of [1, Theorem 3] transported to the ε coordinate, and the prefixes by a breadth-first search of the ε dynamics, but both are re-verified from scratch, and a wrong certificate is rejected (Proposition 5.4). The largest instance is $c = 20000$, with 23 intervals and separating suffixes of length up to 20. □

	0	1	2	3	4	5	6	7	8	9
0	2	3	4	5	6	7	6	8	7	9
10	8	7	10	9	8	11	8	10	9	8
20	12	9	11	10	9	13	10	9	12	9
30	11	10	9	14	11	10	13	10	12	11
40	10	15	10	12	11	10	14	11	10	13
50	10	12	11	10	16	11	13	12	11	15
60	12	11	14	11	13	12	11	17	12	11
70	14	11	13	12	11	16	11	13	12	11
80	15	12	11	14	11	13	12	11	18	13
90	12	15	12	14	13	12	17	12	14	13
100	12									

c	$\mathrm{sc}(c)$
144	13
233	15
610	17
987	17
1000	18
1597	19
2584	19
4181	21
6765	21
10000	26
20000	23

TABLE 1. $\mathrm{sc}(c)$ for $c \le 100$ (row = tens, column = units) and for the eleven larger shifts. Every entry is a theorem.

The values $\mathrm{sc}(0) = 2$ and $\mathrm{sc}(10) = 8$ recover the only two msd-first numbers that appear in [1] — the 2-state msd-first automaton for $\mathbf{f}$ itself [1, Fig. 1] and the eight intervals of the worked example [1, Example 3] — and prove in addition that both are minimal. None of the four sequences $(\mathrm{sc}(c))_{c\ge 0}$, $(\mathrm{sc}(c))_{c\ge 6}$, $\big(\nu(F_{|(c)_F|+2} - 1 - c)\big)_{c\ge 2}$ and that last one shifted down by 2, each queried by its first twenty terms, is in the On-Line Encyclopedia of Integer Sequences [12] (four queries, 3 September 2026, all empty); the analogous Thue–Morse sequence is there, as `A382296`.

Theorem 5.2. *For every $c \in \mathcal{W}$ with $c \ge 2$, the msd-first Fibonacci DFAO constructed in the proof of* [1, Theorem 3] *— whose states are the intervals of the partition of the circle obtained by combining the endpoints of $\mathcal{P}(-(c+1))$ and $\mathcal{P}(-(c+2))$ — has exactly $\mathrm{sc}(c)$ states. That construction is therefore already minimal, and not merely $O(\log c)$.*

Proof sketch. The certificate K_c used in Theorem 5.1 *is* that automaton: its partition points are the representatives in E of the endpoints $\{r\varphi\}$ of $\mathcal{P}(-(c+1)) \cup \mathcal{P}(-(c+2))$ under $\varepsilon \equiv -n\varphi \pmod 1$, and its number of intervals is the number of states of M_c. What Theorem 5.1 proves is that this number is the state complexity. The identification of the certificate's endpoint list with the endpoint list of [1, Definition 3] is an exact computation in $\mathbb{Z}[\varphi]$ and is not part of the formal development; it was carried out for every $c \in \mathcal{W}$ with $c \ge 1$: in each case the partition points of K_c are exactly the representatives in E of the points $\{r\varphi\}$, $r \in \mathcal{P}(-(c+1)) \cup \mathcal{P}(-(c+2))$ — their φ-coefficients are the negatives of those integers — and the transitions, outputs and initial state of K_c are those of M_c. For $c = 10$ it can be read off directly: the φ-coefficients of the eight partition points of K_{10} are, as a set, $\{1, 2, 3, 5, 7, 8, 11, 12\}$, exactly the negatives of the eight integers $-12, -11, -8, -7, -5, -3, -2, -1$ listed in [1, Example 3]. An independent exact computation in $\mathbb{Z}[\varphi]$ (§6) confirms for every $2 \le c \le 20000$ that no two states of this automaton are merged by partition refinement and that all of them are reachable. □

Remark 5.3. The restriction $c \ge 2$ in Theorem 5.2 is a restriction on the construction, not on the values: $\mathcal{P}(r)$ is defined in [1, Definition 3] only for $r \le -2$, so $c = 0$ falls outside it, and $c = 0, 1$ lie outside the range of the independent computation quoted above. Their state complexities 2 and 3 are part of Theorem 5.1 all the same.

Proposition 5.4 (Controls)**.** *The following hold, and none of the checks above is vacuous.*

1. *No msd-first Fibonacci DFAO on 7 states generates $\mathbf{f}_{10}$; in particular neither 7 nor 9 is the msd-first state complexity of $\mathbf{f}_{10}$.*
2. *The certificate check of Definition 3.3 rejects the certificate K_{10} with all its outputs flipped, and rejects it again with its 0-transitions cyclically rotated.*

(3) *The separation check of Theorem 4.1 fails when a ninth prefix, namely the word* 0*, is appended to the eight prefixes used for* $c = 10$ *— as it must, that word having the same residual as the empty word.*
(4) *No sequence has msd-first state complexity* 0*.*

Item (1) is the point of proving each value from both sides: because sc is defined as a least element, a claim that is too small and a claim that is too large are both refuted, so Table 1 is not an artefact of a one-sided argument.

6. A closed form

Theorem 6.1. *For every* $c \in \mathcal{W}$ *with* $c \geq 2$ *— that is, for* 110 *of the* 112 *shifts of Theorem 5.1 —*

$$\mathrm{sc}(c) \;=\; |(c)_F| + \nu\big(F_{|(c)_F|+2} - 1 - c\big), \tag{4}$$

where $\nu(0) := |(c)_F|$*, a case that occurs exactly when* $c = F_{|(c)_F|+2} - 1$*.*

Proof sketch. The right-hand side of (4) is a finite computation on the Zeckendorf representations of c and of $F_{|(c)_F|+2} - 1 - c$; it is evaluated for each of the 110 shifts and compared with the number of intervals of the corresponding certificate K_c. By Theorem 5.1 that number is $\mathrm{sc}(c)$. So the theorem is an identity between two numbers each of which is known exactly, checked one shift at a time. □

Write $k = |(c)_F| + 2$ and $d = F_k - 1 - c$, so that $F_{k-1} \leq c < F_k$ for $c \geq 1$.

Corollary 6.2. *For* $c \in \mathcal{W}$ *with* $c \geq 2$*,*

$$|(c)_F| + 2 \;\leq\; \mathrm{sc}(c) \;\leq\; 2|(c)_F|,$$

and the upper extreme occurs exactly when $c = F_k - 1$*.*

Proof. It suffices to bound the right-hand side of (4). Always $\nu \geq 2$, since Zeckendorf indices start at 2. If $d \geq 1$ then $d \leq F_k - 1 - F_{k-1} = F_{k-2} - 1 < F_{k-2}$, so $(d)_F$ uses only $F_2, \ldots, F_{k-3}$ and $\nu(d) \leq k - 3 = |(c)_F| - 1$; while $d = 0$, which happens exactly for $c = F_k - 1$, gives $\nu = |(c)_F|$ by convention. □

Beyond the tabulated shifts.

Remark 6.3 (Exact computation outside the formal development)*.* All of the following were computed exactly in $\mathbb{Z}[\varphi]$ — elements $a + b\varphi$ as pairs of integers, comparisons by the $5B^2$ versus D^2 rule of §3, never in floating point — and are *not* part of the formal development.

(1) The automaton of [1, Theorem 3] has exactly $|(c)_F| + \nu(F_{|(c)_F|+2} - 1 - c)$ states for every $2 \leq c \leq 20000$: 19999 shifts, 0 mismatches.
(2) For every $2 \leq c \leq 20000$ no two states of that automaton are merged by partition refinement, and every state is reachable.
(3) For every $2 \leq c \leq 4000$, any two of its states are separated by a suffix that is a valid continuation of *both*, so the lower bound of Theorem 4.1 applies on that range in the strong form that also binds automata with total transition functions.
(4) The exact lsd-first formula (1) of [1, Theorem 2] was reproduced independently, by reversing and minimising the msd-first automaton, for every $5 \leq c \leq 20000$, with 0 mismatches; it fails at $c = 0, 1$, exactly as its hypothesis $c \geq 5$ allows.

It is item (3) rather than item (2) that turns item (1) into a statement about sc: two states kept apart by partition refinement need not be separated by any suffix valid for both, and Definition 2.1 permits an automaton to identify such a pair. So outside the formal development the identity (4) is confirmed for sc on $2 \leq c \leq 4000$, while on the rest of the range items (1) and (2) confirm the size of the construction of [1] and its minimality under refinement. On $\mathcal{W}$ itself none of this is needed.

Conjecture 6.4. Identity (4) holds for every $c \geq 2$.

The upper-bound half of Conjecture 6.4 can be proved outright, by counting the states of the construction of [1]; we record the argument, which is elementary but does not appear in [1], where no count of $|\mathcal{P}|$ finer than $O(\log c)$ is given.

Remark 6.5 (Outside the formal development: the size of the partition). *For every* $c \geq 2$,

$$\big|\mathcal{P}(-(c+1)) \cup \mathcal{P}(-(c+2))\big| \;=\; |(c)_F| + \nu\big(F_{|(c)_F|+2} - 1 - c\big),$$

and hence, with [1, Theorem 3], $\mathrm{sc}(c) \leq |(c)_F| + \nu(F_{|(c)_F|+2} - 1 - c)$ *for every* $c \geq 2$. This statement is proved here in prose only; it has not been formalised, and it is not used anywhere above. Each step of the count below, and both values of λ, were checked against the chains of [1, Definition 3] for every $2 \leq c \leq 20000$.

Sketch. By [1, Definition 3] the endpoints of $\mathcal{P}(-n)$ are the successive right-truncations of the negative Fibonacci representation of $-n$ used in [1, §2] — a special case of a representation of Rockett and Szüsz [9] — so $|\mathcal{P}(-n)| = k(n) - 1$ where $F_{k(n)}$ is the least Fibonacci number $\geq n$, and the two endpoint sets are chains of truncations meeting exactly in a common tail. Write $L_i = k(c+i) - 2$, so that $|\mathcal{P}(-(c+i))| = L_i + 1$, and let $\lambda + 1$ be the number of truncations the two chains share; then the union has $L_1 + L_2 + 1 - \lambda$ elements. If $c \leq F_k - 2$ then $k(c+1) = k(c+2) = k$ and $L_1 = L_2 = k - 2$, and the two truncation chains are those of $m = F_k - c - 1 \geq 1$ and $m - 1$; if F_j is the lowest Fibonacci number in $(m)_F$ then $m - 1 = (m - F_j) + (F_j - 1)$ with $F_j - 1 = F_{j-1} + F_{j-3} + \cdots$ using only indices $< j$, so the two words agree above position j and differ at j, whence $\lambda = k - 1 - j$ and the union has $k + j - 2 = |(c)_F| + \nu(F_k - 1 - c)$ elements. If $c = F_k - 1$ then the first chain is 0^{k-2} while $c + 2 = F_k + 1$ forces index $k + 1$ and a chain with two leading zeros, so $\lambda = 2$ and the union has $2k - 4 = 2|(c)_F|$ elements, which is the value of the right-hand side at $d = 0$. □

What Conjecture 6.4 still needs is therefore the matching *lower* bound in general: that the $|(c)_F| + \nu$ intervals are all reachable from $\varepsilon = 0$ and pairwise separated by a commonly valid suffix. Both halves are concrete in the ε picture — reachability is the forward orbit of 0 under T_0, T_1, separation a breadth-first search on pairs — and both have been verified exactly on the ranges of Remark 6.3.

Remark 6.6 (Msd-first versus lsd-first). Comparing (4) with (1): for every $5 \leq c \leq 20000$ the right-hand side of (4) is at most $\mathrm{sc}_{\mathrm{lsd}}(c) - 2$, with equality exactly at $c = 12, 33, 88, 232, 609, 1596, 4180, 10945$, that is at the eight values of the form $c = F_k - 1$ with k odd in range. At $c = F_k - 1$ with k even the gap is 3 instead, because the correction term $g(c)$ of (1) vanishes there. Since that right-hand side is realised by an automaton (Remark 6.3(1)), $\mathrm{sc}(c) \leq \mathrm{sc}_{\mathrm{lsd}}(c) - 2$ on the whole of that range. Since $\mathrm{sc}(c) \geq |(c)_F| + 2$ where (4) holds and $\mathrm{sc}_{\mathrm{lsd}}(c) = 2|(c)_F| + 3 - g(c)$, msd-first costs between roughly half of lsd-first and all but two states of it. This is a computation outside the formal development, in the sense of Remark 6.3.

Remark 6.7 (The distribution of ν). Over $2 \leq c \leq 20000$ the value $\nu(F_{|(c)_F|+2} - 1 - c)$ takes the values $2, 3, 4, \ldots$ with frequencies

$$7640,\ 4722,\ 2918,\ 1803,\ 1115,\ 689,\ 426,\ 263,\ 162,\ 101,\ 62,\ 38,\ \ldots$$

Successive ratios are close to φ, as they should be if the lowest Fibonacci index of a "random" Zeckendorf word is geometric with ratio $1/\varphi$ on $\{2, 3, \ldots\}$; such a law has mean $2 + \varphi = 3.6180\ldots$, against an observed mean of 3.6157. So by (4) the excess of the state count over $|(c)_F|$ is on average about $2 + \varphi$ on that range. The comparison is with $|(c)_F|$, not with $\log_\varphi c$: those differ by an oscillating amount (about $+0.65$ at $c = 1000$, mean 0.18 over the range), so the average excess over $\log_\varphi c$ is 3.796 instead.

7. Verification

Every theorem and proposition of §§3–6 has been formally verified in Lean 4 with its mathematical library Mathlib, in a development of eight modules that is free of `sorry`; the exceptions are flagged where they occur: Corollary 6.2, proved in full above; the identification of the certificates with the construction of [1] in Theorem 5.2; and the remarks labelled as computations outside the formal development. The reasoning layer — the exact order on $\mathbb{Z}[\varphi]$, the interval invariant of Theorem 3.1 and its corollary on unique representatives, the soundness of interval certificates (Theorem 3.4), the Myhill–Nerode bound (Theorem 4.1) and the combination of the two into a least element — is proved by ordinary induction

and depends on no axioms beyond `propext`, `Classical.choice` and `Quot.sound`; note in particular that the only real number ever used is $\sqrt{5}$, through the identity that turns a comparison in $\mathbb{Z}[\varphi]$ into a comparison of two integers. What is delegated to compiled evaluation, through Lean's `native_decide`, is exactly the finite searches: for each of the 112 shifts of Theorem 5.1, the certificate check (C1)–(C4) and the pairwise separation check, three compiled evaluations per shift, 336 in all, costing 2 minutes 9 seconds of wall time and 4.08 GB of peak memory; and the evaluation of the closed form (4) against all 110 certificates with $c \geq 2$. The definition of the Fibonacci word used is the source's "last digit of the Zeckendorf representation" [1, Definition (d)]; that it agrees with the printed prefix 01001010 and with the morphism and concatenation definitions on the first $F_{13} = 233$ letters is itself checked in the development, so the object studied here is not a variant of the intended one. The following are recorded computations *outside* the formal development and are identified as such where they occur: everything in Remark 6.3; the identification of the certificate endpoints with the partition of [1] used in Theorem 5.2; the prose count of Remark 6.5; and the statistics of Remarks 6.6 and 6.7. Repository reference to be added at submission.

8. Open questions

(1) *Prove Conjecture 6.4.* By Remark 6.5 the upper half is settled; what is missing is a proof, for all c, that the $|(c)_F| + \nu$ intervals of the construction of [1] are pairwise inequivalent. In the ε picture this asks for two things about the orbit of 0 under the contractions T_0, T_1: that it meets every interval, and that any two intervals admit a commonly valid separating suffix. Both have been verified exactly to $c = 20000$ and $c = 4000$ respectively.

(2) *A second proof of the consistency lemma.* The two auxiliary lemmas [1, Lemmas 7 and 8] that carry the proof of the consistency lemma [1, Lemma 5] are established there by `Walnut` scripts. In the ε coordinate both become finite statements about $\mathbb{Z}[\varphi]$ — the relevant one being $T_a^{-1}(S) \cap E \subseteq S$ for the two orbits — and a self-contained proof along those lines would remove the external prover from the argument entirely.

(3) *Is* sc *Fibonacci-regular?* For the lsd-first side this is part of [1, Theorem 2]. Formula (4) expresses $\mathrm{sc}(c)$ through two quantities, $|(c)_F|$ and $\nu(F_{|(c)_F|+2} - 1 - c)$, that are both computable by finite automata reading $(c)_F$, which suggests the answer is yes; we have not proved it.

(4) *Tribonacci.* The closing question of [1] asks the same questions for the Tribonacci word. The ε machinery generalises to any Pisot numeration system whose companion has all conjugates inside the unit circle, but for Tribonacci the conjugates are complex, so the interval E becomes a region bounded by a Rauzy fractal and the finite checks used here would have to be replaced.

References

[1] D. Moradi, P. Popoli, J. Shallit and I. Vukusic, *State Complexity of Shifts of the Fibonacci Word*, arXiv:2603.18858v1, 19 March 2026, cs.FL, cs.DM and math.NT; conference version in *Descriptional Complexity of Formal Systems (DCFS 2026)*, Lecture Notes in Computer Science 16681, Springer, pp. 133–147, `doi:10.1007/978-3-032-32016-2_11`. Numbered results cited here follow the arXiv PDF of v1 (Theorems 1–3, Lemmas 1–8, Definitions 1–3, Examples 1–4, Figures 1–5); its HTML rendering numbers the theorems 3.1, 3.2, 4.1 instead.

[2] D. Moradi, N. Rampersad and J. Shallit, *Complexity of Linear Subsequences of Fibonacci-Automatic Sequences*, arXiv:2603.21645v1, 23 March 2026, cs.FL, cs.DM and math.CO.

[3] D. Moradi, N. Rampersad and J. Shallit, *Complexity of Linear Subsequences of k-Automatic Sequences*, arXiv: 2512.10017, first posted 10 December 2025, cs.FL. The facts quoted in §1 were read in v5 (2 April 2026).

[4] J.-P. Allouche and J. Shallit, *Automatic Sequences: Theory, Applications, Generalizations*, Cambridge University Press, Cambridge, 2003.

[5] J. Berstel, *Fibonacci words—a survey*, in: G. Rozenberg and A. Salomaa (eds.), The Book of L, Springer-Verlag, 1986, pp. 13–27.

[6] A. Cobham, *Uniform tag sequences*, Math. Systems Theory 6 (1972), 164–192.

[7] E. Zeckendorf, *Représentation des nombres naturels par une somme de nombres de Fibonacci ou de nombres de Lucas*, Bull. Soc. Roy. Sci. Liège 41 (1972), 179–182.

[8] C. G. Lekkerkerker, *Voorstelling van natuurlijke getallen door een som van getallen van Fibonacci*, Simon Stevin 29 (1952), 190–195.

[9] A. M. Rockett and P. Szüsz, *Continued Fractions*, World Scientific, Singapore, 1992.

[10] H. Mousavi, *Automatic Theorem Proving in* `Walnut`, arXiv:1603.06017 (v1 2016, v2 2021).

[11] J. Shallit, *The Logical Approach to Automatic Sequences: Exploring Combinatorics on Words with* `Walnut`, London Mathematical Society Lecture Note Series 482, Cambridge University Press, 2022.
[12] OEIS Foundation Inc., *The On-Line Encyclopedia of Integer Sequences*, `oeis.org`, consulted 3 September 2026.