INSTITUT DE FRANCE
Académie des sciences

Comptes Rendus
Mathématique

Narad Rampersad

Prefixes of the Fibonacci word that end with a cube

Volume 361 (2023), p. 323-330

https://doi.org/10.5802/crmath.408

Les Comptes Rendus. Mathématique *sont membres du*
Centre Mersenne pour l'édition scientifique ouverte
www.centre-mersenne.org
e-ISSN : 1778-3569

Comptes Rendus Mathématique
2023, Vol. 361, p. 323-330
https://doi.org/10.5802/crmath.408

Combinatorics / *Combinatoire*

Prefixes of the Fibonacci word that end with a cube

Narad Rampersad [a]

[a] Department of Mathematics and Statistics, University of Winnipeg, 515 Portage Ave., Winnipeg, MB, R3B 2E9, Canada
E-mail: narad.rampersad@gmail.com

Abstract. The Fibonacci word $\mathbf{f} = 010010100100101\cdots$ is one of the most well-studied words in the area of combinatorics on words. It is not periodic, but nevertheless contains many highly periodic factors (contiguous subwords). For example, it contains many cubes (i.e., non-empty words of the form xxx). We study the prefixes of the Fibonacci word that end with a cube. Using the computer prover Walnut, we obtain an exact description of the positions of the Fibonacci word at which a cube ends. This gives a certain measure of how close the Fibonacci word is to being periodic.

2020 Mathematics Subject Classification. 68R15.

Funding. The author is supported by NSERC Discovery Grant RGPIN-2019-04111.

Manuscript received 29 November 2021, revised 13 July 2022, accepted 20 July 2022.

1. Introduction

Periodicity is a fundamental concept in the study of combinatorics on words. An infinite word $\mathbf{w}$ is *ultimately periodic* if it can be written in the form $\mathbf{w} = uvvvvvv\cdots$ for some finite word u and some non-empty finite word v. An infinite word that is not ultimately periodic is *aperiodic*. Many of the classical results concerning periodicity give equivalent characterizations of ultimate periodicity. For example, a classical result of Morse and Hedlund [8] states that $\mathbf{w}$ is ultimately periodic if and only if there is a constant C such that for every length n, the word $\mathbf{w}$ contains at most C factors (contiguous subwords) of length n.

This paper is motivated by another result of this type, which was originally conjectured by Jeffrey Shallit and proved by Mignosi, Restivo, and Salemi [7]:

> *An infinite word* $\mathbf{w}$ *is ultimately periodic if and only if every sufficiently long prefix of* $\mathbf{w}$ *ends with a suffix that has exponent at least* φ^2*, where* φ *is the golden ratio.*

Let $x = x_0x_1\cdots x_{n-1}$ be a finite word, with the x_i single letters. We write $|x|$ for the *length* of x; i.e., we have $|x| = n$. We say that x has *period* p if $1 \le p \le n$ and $x_i = x_{i+p}$ for $i = 0, \ldots, n-1-p$. A word may have several periods. The *exponent* of a word is the ratio of its length to its least period. For example, the word $aabaabaa$ has periods 3, 6, and 8 and exponent 8/3.

Since $\varphi^2 \approx 2.618$, the result of Mignosi, Restivo, and Salemi implies that no aperiodic infinite word can have every sufficiently long prefix end with a *cube* (a word with exponent 3). Counting the number of prefixes of an infinite word that end with cubes can therefore provide a measure, in some sense, of how close the infinite word is to being ultimately periodic.

The first candidate that one would choose to investigate in regards to this measure is the *Fibonacci word*

$$\mathbf{f} = 0100101001001010010100100\cdots$$

This word can be defined in terms of the following sequence of words, which are constructed by an analogue of the Fibonacci reccurrence: let $\Phi_0 = 0$, $\Phi_1 = 01$, and let $\Phi_n = \Phi_{n-1}\Phi_{n-2}$ for $n \geq 2$. Then

$$(\Phi_n)_{n\geq 0} = (0, 01, 010, 01001, 01001010, \ldots),$$

and $\mathbf{f}$ is the infinite word having Φ_n as a prefix for every $n \geq 0$.

The Fibonacci word is aperiodic, and, as shown by Mignosi et al., it witnesses the optimality of their result in the following sense:

> *For any $\epsilon > 0$, every sufficiently long prefix of the Fibonacci word ends with a suffix that has exponent at least $\varphi^2 - \epsilon$.*

In this paper we examine the positions at which a cube ends in the Fibonacci word (the starting positions of cubes in the Fibonacci word have been characterized by Mousavi, Schaeffer, and Shallit [10]). Let $\mathbf{cubes_f}$ be the infinite word whose n^{th} term (indexing starting from 0) is

$$\begin{cases} 1 \text{ if a cube ends at position } n \text{ of } \mathbf{f}, \\ 0 \text{ otherwise.} \end{cases}$$

Our main results are given by describing precisely the runs of 1's and 0's in $\mathbf{cubes_f}$. A *run* is an occurence of a factor containing only one distinct letter and which cannot be extended to the left or to the right without encountering a different letter.

We describe the positions and lengths of the runs in $\mathbf{cubes_f}$ using the Fibonacci (or Zeckendorf) numeration system, which we define later. For now, let us simply introduce some notation. Let F_n denote the n^{th} Fibonacci number and for any $n \geq 0$, let $(n)_F$ denote the canonical representation of n in the Fibonacci numeration system. We also use some notation from the theory of *regular expressions* (see any standard text, such as [3]): the symbol ϵ denotes the empty word; the operator + denotes *union*; and if w is a finite word, then w^* denotes the set $\{\epsilon, w, ww, www, \ldots\}$ and w^+ denotes the set $w^* \setminus \{\epsilon\}$.

Theorem 1. *There are arbitrarily long runs of* 1*'s in* $\mathbf{cubes_f}$. *More precisely, the runs of* 1*'s in* $\mathbf{cubes_f}$ *are characterized by the following: If* $(i)_F$ *has the form*

$$(i)_F \in (10)^+0(0+10)(00)^*0w,$$

where $w \in 0(10)^*(\epsilon + 1)$ *then* $\mathbf{cubes_f}$ *contains a run of* 1*'s of length*

- $F_{2n+2} - 1$, *if* $|w| = 2n$ *for some* $n \geq 0$,
- $F_{2n+3} - 1$, *if* $|w| = 2n+1$ *for some* $n \geq 0$,

beginning at position i.

Theorem 2. *The runs of* 0*'s in* $\mathbf{cubes_f}$ *have lengths* 1, 2, 3, 7, 8, *and* 13. *The only run of length* 13 *occurs at the beginning of* $\mathbf{cubes_f}$. *For each of the other lengths (*1, 2, 3, 7, *and* 8*), there are infinitely many runs of that length in* $\mathbf{cubes_f}$.

2. Walnut and automatic sequences

Our main results are all obtained by computer using a computer prover called Walnut [9], which provides an implementation of an algorithm for proving many combinatorial properties of infinite words such as the Fibonacci word. Walnut can be obtained from the website

https://cs.uwaterloo.ca/~shallit/walnut.html

which also contains a tutorial on how to use it and a list of papers that contain results proved using Walnut.

Walnut operates on a class of sequences known as *automatic sequences*. The following is a brief and somewhat informal overview of the basic definitions needed for this topic; for further details the reader can consult [1].

Let Σ be a finite alphabet and let Σ^* denote the set of all words over the alphabet Σ. A *deterministic finite automaton with output* (DFAO) is a 6-tuple $\Gamma = (Q, \Sigma, \delta, q_0, \Delta, \tau)$, where Q is a finite set of *states*, Σ is a finite *input alphabet*, $\delta : Q \times \Sigma$ is the *transition function*, Δ is a finite *output alphabet*, and $\tau : Q \to \Delta$ is the *output function*. A DFAO is often represented as an edge-labeled directed graph where the vertices are the states from Q, and for each $(q, a) \in Q \times \Sigma$ there is a directed edge from q to $\delta(q, a)$ labeled a. Given an input word $w \in \Sigma^*$, the computation of Γ on w is performed by starting in the initial state q_0 and then, as w is processed symbol-by-symbol, the computation moves from state to state based on the current state, the symbol being read, and the transition function δ. Finally, after reading w, the automaton ends up in some state q and outputs the value $\tau(q)$. This computation defines a map $f : \Sigma^* \to \Delta$.

Now suppose that $\Sigma = \{0, 1, \ldots, k-1\}$ and let $(n)_k$ denote the base-k expansion of n. Then such an $f : \Sigma^* \to \Delta$ can be used to define a sequence (or infinite word) $(a_n)_{n\geq 0}$ over Δ by taking $a_n = f((n)_k)$. In other words, Γ takes $(n)_k$ as input and outputs a_n. Such a sequence is called a *k-automatic sequence*. In this paper it is assumed that DFAO's read their input starting with the most significant digit first.

We can generalize this model to numeration systems beyond the usual base-k systems. In this paper, we will be using a numeration system based on the sequence of Fibonacci numbers. Let $F_0 = 0$, $F_1 = 1$, and $F_n = F_{n-1} + F_{n-2}$ for $n \geq 2$. Several authors (Ostrowski [11], Lekkerkerker [5], Zeckendorf [12]) showed that every non-negative integer has an essentially unique representation as a sum of Fibonacci numbers, subject to the constraint that one cannot use two consecutive Fibonacci numbers in the sum. Let $n = \Sigma_{i=1}^{\ell} a_i F_{\ell+2-i}$ be such a representation. If $a_1 \neq 0$, then we call $(n)_F := a_1 a_2 \cdots a_\ell$ the *canonical Fibonacci representation of n*.

We can now define *Fibonacci-automatic sequences* via DFAO's exactly as above, except that now our sequence $(a_n)_{n\geq 0}$ is defined by $a_n = f((n)_F)$; i.e., the DFAO takes $(n)_F$ as input and returns a_n as output. The Fibonacci word **f** is Fibonacci-automatic and is defined by the DFAO in Figure 1 (here the output associated with each state is simply its label). For example, the

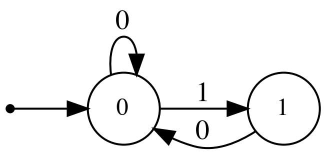

Figure 1. DFAO for **f**

Fibonacci representations of $0,1,2,3,4,5$ are $\epsilon,1,10,100,101,1000$ and we see that this DFAO outputs $0,1,0,0,1,0$ on these inputs, which are the first few values of **f**.

Now we can describe what Walnut does. Walnut is a software program that can prove or disprove certain statements about automatic sequences. It takes the following as input: a DFAO computing an automatic sequence **s** (it can handle k-automatic sequences, Fibonacci-automatic sequences, and sequences defined using a few other numeration systems), and a formula ψ in a certain first-order-logic that extends Presburger arithmetic. The formula ψ is a first-order formula that involves variables, quantifiers, addition and subtraction of natural numbers, and indexing into the sequence **s**. Arithmetic can only be done on indices, not the terms of **s**, and multiplication and division are not allowed, other than multiplication by a constant (understood as repeated addition). If the formula ψ has no free variables, then Walnut outputs either `TRUE` or `FALSE` and if ψ does contain free variables, then Walnut outputs an automaton that accepts the representations (in the appropriate numeration system) of the natural numbers (or tuples of natural numbers) that can be assigned to the free variables to satisfy the formula ψ.

For example, Walnut can verify that **f** is aperiodic. A formula ψ expressing the property of ultimate periodicity is

$$\psi : \exists\, p \geq 1,\, n \geq 0\ \forall\, i \geq n\, \mathbf{f}[i] = \mathbf{f}[i+p].$$

Translated into Walnut's syntax, the command to evaluate the truth of this formula is

```
eval f_ult_per "?msd_fib Ep En (p >= 1) & (Ai (i >= n) =>F[i] = F[i+p])":
```

and the resulting output is `FALSE`, so we conclude that **f** is aperiodic. The reader may also consult [10] to see a number of other properties of (or related to) **f** that can be proved using Walnut.

3. Walnut computations

In this section we give the Walnut-assisted proofs of Theorems 1 and 2. We begin with the command

```
eval fib_end_cubes "?msd_fib Ei En n > 1 & j = i+3*n-1 &(Ak k < 2*n
=> F[i+k] = F[i+k+n])":
```

which produces the automaton in Figure 2, which accepts the Fibonacci representations of the positions at which a cube ends in **f**.

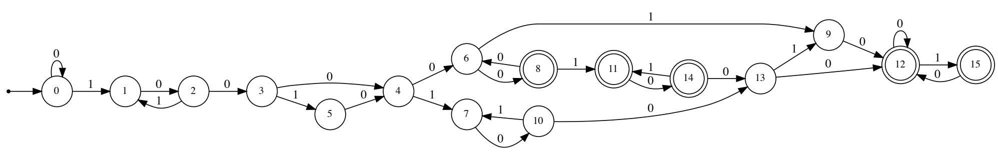

Figure 2. Automaton for ending positions of cubes in **f**

Proof of Theorem 1. To determine the lengths of the runs of 1's in $\mathbf{cubes_f}$, we use the command

```
eval fib_end_cubes_run "?msd_fib n>=1 & (At t<n =>$fib_end_cubes(i+t))
 &~$fib_end_cubes(i+n) & (i=0|~$fib_end_cubes(i-1))":
```

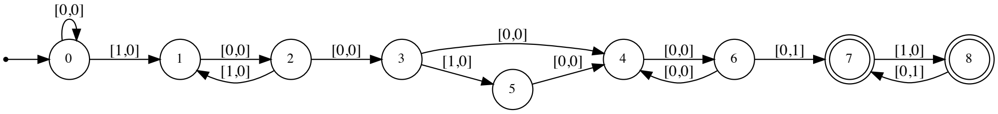

Figure 3. Automaton for runs of 1's in $\mathbf{cubes_f}$

which produces the automaton in Figure 3, which accepts the Fibonacci representations of pairs (i,ℓ) such that there is a run of 1's in $\mathbf{cubes_f}$ of length ℓ starting at position i.

By examining the structure of this automaton we see that for an accepted pair (i,ℓ), the representation $(i)_F$ has the form $(i)_F = (10)^+0(0+10)(00)^*0w$, where $w \in 0(10)^*(\epsilon+1)$. Furthermore, if $|w| = 2n$, then $(\ell)_F = (10)^n$ and if $|w| = 2n+1$, then $(\ell)_F = (10)^n1$. Now, let F_m denote the m^{th} Fibonacci number and recall the identities:

$$\sum_{j=0}^{n-1} F_{2j+1} = F_{2n} \quad \text{and} \quad \sum_{j=1}^{n} F_{2j} = F_{2n+1} - 1.$$

Hence, if $|w| = 2n$, we have

$$\ell = \sum_{j=1}^{n} F_{2j+1} = F_{2n+1} + F_{2n} - F_1 = F_{2n+2} - 1$$

and if $|w| = 2n+1$ we have

$$\ell = \sum_{j=1}^{n+1} F_{2j} = F_{2n+2} + F_{2n+1} - 1 = F_{2n+3} - 1.$$

□

Proof of Theorem 2. To determine the lengths of the runs of 0's in $\mathbf{cubes_f}$, we use the command

```
eval fib_no_cubes_run "?msd_fib n>=1 & (At t<n => ~$fib_end_cubes(i+t))
& $fib_end_cubes(i+n) &(i=0|$fib_end_cubes(i-1))":
```

which produces the automaton in Figure 4, which accepts the Fibonacci representations of pairs (i,ℓ) such that there is a run of 0's in $\mathbf{cubes_f}$ of length ℓ starting at position i.

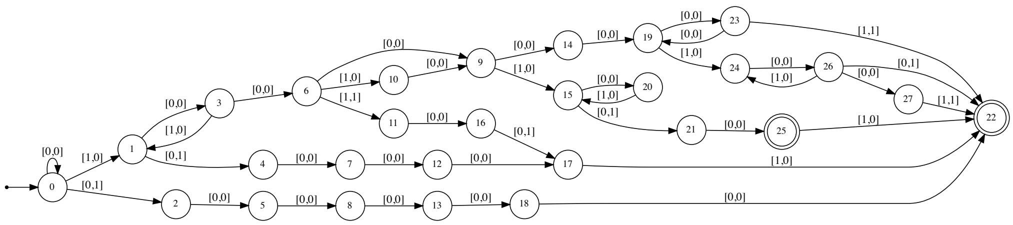

Figure 4. Automaton for runs of 0's in $\mathbf{cubes_f}$

We can project this automaton onto the second component of its input with the command

```
eval fib_no_cubes_run_length "?msd_fib Ei $fib_no_cubes_run(i,n)":
```

which produces the automaton in Figure 5. We see that the only possible run lengths are $\ell \in \{1,2,3,7,8,13\}$.

The command

```
eval tmp "?msd_fib Ai Ej j>i & $fib_no_cubes_run(j,1)":
```

evaluates to TRUE, indicating that there are infinitely many runs of 0's of length 1. This is also the case for run lengths 2, 3, 7, and 8. For length 13 however, we get a result of FALSE. □

The positions of the runs of length 7 and 8 have a simple structure, so we describe these next.

Theorem 3.

- *The runs of* 0*'s in* $\mathbf{cubes_f}$ *of length* 8 *begin at positions* i *where* $(i)_F \in (10)^+0001$.
- *The runs of* 0*'s in* $\mathbf{cubes_f}$ *of length* 7 *begin at positions* i *where* $(i)_F \in (10)^+01001$.

Proof. These are obtained via the commands

```
eval tmp "?msd_fib $fib_no_cubes_run(j,8)":
eval tmp "?msd_fib $fib_no_cubes_run(j,7)":
```

□

The descriptions of the starting positions for the other lengths of runs of 0's in $\mathbf{cubes_f}$ are a little more complicated, so we omit them here, but the reader can easily compute these with Walnut.

Theorem 4. *The density of* 0*'s in* $\mathbf{cubes_f}$ *is zero.*

Proof. We examine the complement of the automaton in Figure 2. The Walnut command

```
eval fib_no_end_cubes "?msd_fib~$fib_end_cubes(j)":
```

produces the automaton in Figure 6, which gives the positions in **f** where no cube ends.

To complete the proof, it suffices to show that there are only polynomially many strings of length n that are accepted by this automaton. This can be seen directly from the structure of the automaton: since this automaton does not have two cycles that can both mutually reach each other, we can conclude that the number of strings of length n accepted by this automaton is polynomially bounded (see, for example, [2]).

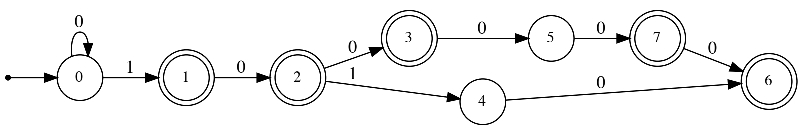

Figure 5. Automaton for lengths of runs of 0's in $\mathbf{cubes_f}$

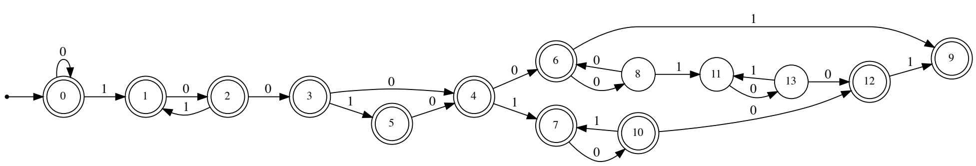

Figure 6. Automaton for positions in **f** where no cube ends.

□

4. Other Sturmian words

Although the Fibonacci word is "optimal" with respect to the result of Mignosi et al. mentioned in the Introduction, some computer calculations suggest that there may be other words that have even more prefixes that end with cubes than the Fibonacci word. The Fibonacci word is just one example of a much larger class of words known as *Sturmian words*. These are words with the property that they have $n+1$ factors of length n for all $n \geq 0$.

The structure of such words is determined by a parameter α, which is an irrational real number between 0 and 1, called the *slope*, and more specifically, by the continued fraction expansion $\alpha = [0, d_1, d_2, d_3, \ldots]$, where $d_i \in \mathbb{Z}$ and $d_i \geq 1$ for $i \geq 1$. The *characteristic Sturmian word with slope* α (see [1, Chapter 9]) is the infinite word $\mathbf{c}_\alpha$ obtained as the limit of the sequence of *standard words* s_n defined by

$$s_0 = 0, \quad s_1 = 0^{d_1-1}1, \quad s_n = s_{n-1}^{d_n} s_{n-2}, \quad n \geq 2.$$

So, for example, we have $\mathbf{f} = \mathbf{c}_\theta$, where $\theta := 1/\varphi^2 = [0, 2, \overline{1}]$.

For any infinite word $\mathbf{w}$, let us define $\mathbf{cubes_w}$ to be the binary word whose n^{th} term is 1 if $\mathbf{cubes_w}$ has a cube ending at position n, and 0 otherwise. Let max_no_cubes($\mathbf{w}$) denote the largest ℓ such that $\mathbf{cubes_w}$ contains infinitely many runs of 0's of length ℓ. So, for instance, Theorem 2 shows that max_no_cubes($\mathbf{f}$) = 8. Let us also define $S_\mathbf{w}(n)$ to be the sum of the first n terms of $\mathbf{cubes_w}$. That is, $S_\mathbf{w}(n)$ counts the number of positions $< n$ at which a cube ends in $\mathbf{w}$.

Problem 5. *Is it possible to determine* max_no_cubes($\mathbf{c}_\alpha$) *from the continued fraction expansion of* α*?*

Problem 6. *What is the least (resp. greatest) possible value of* max_no_cubes($\mathbf{c}_\alpha$) *over all* α*?*

Jeffrey Shallit (personal communication) has proved (also using Walnut) that

$$\text{max\_no\_cubes}\left(\mathbf{c}_{\sqrt{2}-1}\right) = 9,$$

and has empirical evidence that

$$\text{max\_no\_cubes}\left(\mathbf{c}_{\sqrt{3}-1}\right) = 3.$$

Perhaps these are the extremal values.

Regarding $S_{\mathbf{c}_\alpha}(n)$, we pose the following problem:

Problem 7. *Is there an* α *such that for all other* α' *the function* $S_{\mathbf{c}_\alpha}(n)$ *is eventually greater than* $S_{\mathbf{c}_{\alpha'}}(n)$*?*

Similarly, we can also ask:

Problem 8. *Can one prove that the density of* 0*'s in* $\mathbf{cubes}_{\mathbf{c}_\alpha}$ *is* 0 *for all* α*?*

One might also wish to investigate the relationship between the critical exponent of an infinite word $\mathbf{w}$ and the density of 0's in $\mathbf{cubes_w}$. The *critical exponent* of $\mathbf{w}$ is the quantity

$$\sup\{r : \mathbf{w} \text{ contains a factor with exponent } r\}.$$

Note that it is easy to construct an aperiodic word with unbounded critical exponent for which "almost all" positions are the ending position of a cube: for example, the infinite word

$$010^2 10^4 10^8 10^{16} 10^{32} 1 \cdots$$

has this property. So it is natural to restrict our attention to words with bounded critical exponent. The Fibonacci word has critical exponent $2 + \varphi \approx 3.618$ [6], and all Sturmian words have critical exponent at least this large. Are there words $\mathbf{w}$ with lower critical exponent for which the density of 0's in $\mathbf{cubes_w}$ is still 0? The answer is "yes". For instance, the fixed point $\mathbf{x}$ (starting with 0) of the morphism $0 \to 0001$, $1 \to 1011$ has critical exponent $10/3$ [4, p. 99], and just as we did for the

Fibonacci word, we can use Walnut to show that the density of 0's in $\mathbf{cubes_x}$ is 0 (after computing the automaton for the 0's in $\mathbf{cubes_x}$, one computes the eigenvalues of the adjacency matrix and finds that they are all strictly smaller than 4).

Problem 9. *What is the infimum of the critical exponents among all infinite words* $\mathbf{w}$ *for which the density of* 0*'s in* $\mathbf{cubes_w}$ *is* 0*? Is it* 3*?*

Acknowledgments

The idea for this work came from discussions with James Currie. We thank him for those stimulating conversations.

References

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

[2] P. Gawrychowski, D. Krieger, N. Rampersad, J. Shallit, "Finding the growth rate of a regular or context-free language in polynomial time", in *Developments in Language, Theory. 12th International Conference, DLT 2008, Kyoto, Japan, September 16–19, 2008. Proceedings*, Lecture Notes in Computer Science, vol. 5257, Springer, 2008, p. 339-358.

[3] J. E. Hopcroft, J. D. Ullman, *Introduction to Automata Theory, Languages and Computation*, Addison-Wesley Series in Computer Science, Addison-Wesley Publishing Group, 1979.

[4] D. Krieger, "Critical Exponents and Stabilizers of Infinite Words", PhD Thesis, University of Waterloo, Waterloo, Ontario, Canada, 2008, http://uwspace.uwaterloo.ca/handle/10012/3599.

[5] C. G. Lekkerkerker, "Voorstelling van natuurlijke getallen door een som van getallen van Fibonacci [Representation of natural numbers as a sum of Fibonacci numbers]", *Simon Stevin* **29** (1952), p. 190-195.

[6] F. Mignosi, G. Pirillo, "Repetitions in the Fibonacci infinite word", *RAIRO, Inform. Théor. Appl.* **26** (1992), p. 199-204.

[7] F. Mignosi, A. Restivo, S. Salemi, "Periodicity and the golden ratio", *Theor. Comput. Sci.* **204** (1998), p. 153-167.

[8] M. Morse, G. A. Hedlund, "Symbolic dynamics", *Am. J. Math.* **60** (1938), p. 815-866.

[9] H. Mousavi, "Automatic theorem proving in Walnut", Documentation (2016–2021) available at https://arxiv.org/abs/1603.06017, 2021.

[10] H. Mousavi, L. Schaeffer, J. Shallit, "Decision algorithms for Fibonacci-automatic words I: Basic results", *RAIRO, Theor. Inform. Appl.* **50** (2016), no. 1, p. 39-66.

[11] A. Ostrowski, "Bemerkungen zur Theorie der Diophantischen Approximationen", *Abh. Math. Semin. Univ. Hamb.* **1** (1922), p. 77-98, 250-251, reprinted in *Collected Mathematical Papers*, **Vol. 3**, pp. 57–80.

[12] É. Zeckendorf, "Représentation des nombres naturels par une somme de nombres de Fibonacci ou de nombres de Lucas", *Bull. Soc. Roy. Liège* **41** (1972), p. 179-182.