CIRCULAR CRITICAL EXPONENTS FOR THUE–MORSE FACTORS

JEFFREY SHALLIT[*] AND RAMIN ZARIFI

Abstract. We prove various results about the largest exponent of a repetition in a factor of the Thue–Morse word, when that factor is considered as a circular word. Our results confirm and generalize previous results of Fitzpatrick and Aberkane & Currie.

Mathematics Subject Classification. 11B85, 68Q45, 68R15, 03D05, 03B35.

1. INTRODUCTION

Consider the English word `amalgam`; it has a factor[1] `ama` of period 2 and length 3, so we can consider `ama` to be a $\frac{3}{2}$ power. However, if we think of `amalgam` as a "circular word" or "necklace", where the word "wraps around", then it has the factor `amama` of period 2 and length 5. We say that `amalgam` has a circular critical exponent of $\frac{5}{2}$.

The famous Thue–Morse infinite word

$$\mathbf{t} = t_0 t_1 t_2 \cdots = \mathtt{01101001} \cdots$$

has been studied extensively since its introduction by Thue in 1912 [4, 10]. In particular, Thue proved that the largest repetitions in $\mathbf{t}$ are 2-powers (also called "squares").

It was only fairly recently, however, that the repetitive properties of its factors, *considered as circular words*, have been studied. Fitzpatrick [6] showed that, for all $n \geq 1$, there is a length-n factor of $\mathbf{t}$ with circular critical exponent < 3. Aberkane and Currie [1] conjectured that for every $n \geq 1$, some length-n factor of $\mathbf{t}$ has circular critical exponent $\leq \frac{5}{2}$, and, using a case analysis, they later proved this conjecture [2].

In this paper, we show how to obtain the Aberkane–Currie result, and much more, using an approach based on first-order logic and the `Walnut` prover, written by Hamoon Mousavi.

2. BASICS

The ith letter of a word w is written $w[i]$. The notation $w[i..j]$ represents the word

$$w[i]w[i+1] \cdots w[j].$$

---

[*] Corresponding author: shallit@uwaterloo.ca

[1]A *factor* is a contiguous block lying inside another word.

If $i \geq j + 1$, then $w[i..j] = \epsilon$, the empty word.

An infinite (resp., nonempty finite) word w has a period $p \geq 1$ if $w[i] = w[i+p]$ for all $i \geq 0$ (resp., all i with $0 \leq i < |w| - p$). For finite words of length n, we restrict our attention to periods that are $\leq n$. A word can have multiple periods, for example, the English word alfalfa has periods $3, 6$, and 7. The smallest period is called *the* period and is denoted $p(w)$. The *exponent* of a finite word w is defined to be $\exp(w) = |w|/p(w)$; it measures the largest amount of (fractional) repetition of a word. The period of alfalfa is 3, and it has length 7; hence its exponent is $\frac{7}{3}$.

A word is called a *square* if its exponent is 2. If its exponent is greater than 2, it is called an *overlap*. Thus, for example, the English word murmur is a square and the French word entente is an overlap.

The *critical exponent* of a word w is the supremum, over all finite nonempty factors x of w, of $\exp(x)$; it is denoted $\mathrm{ce}(w)$. For example, Mississippi has critical exponent 7/3, arising from the overlap ississi.

We can also define this notion for "circular words" (aka "necklaces"). We say two words x, y are *conjugate* if one is a cyclic shift of the other; alternatively, if there exist (possibly empty) words u, v such that $x = uv$ and $y = vu$. For example, the English words listen and enlist are conjugates.

We let $\mathrm{conj}(w)$ denote the set of all cyclic shifts of w:

$$\mathrm{conj}(w) = \{yx \; : \; \exists x, y \text{ such that } w = xy\}.$$

For example, the conjugates of ate are $\{\text{ate}, \text{tea}, \text{eat}\}$.

Here is the most fundamental definition of our paper.

Definition 2.1. The *circular critical exponent* of a word w, denoted by $\mathrm{cce}(w)$, is the supremum of $\exp(x)$ over all finite nonempty factors x of all conjugates of w.

Note that $\mathrm{cce}(w)$ can be as much as twice as large as $\mathrm{ce}(w)$. See [9] for more about this notion for infinite words.

2.1. The Thue–Morse word

The Thue–Morse word **t** has many equivalent definitions [3], but for us it will be sufficient to describe it as the fixed point, starting with 0, of the morphism μ mapping $0 \rightarrow 01$ and $1 \rightarrow 10$.

A basic fact about the binary alphabet is that every word of length ≥ 4 has critical exponent at least 2. Thue proved that the Thue–Morse word has no overlaps. Thus we get the following (trivial) result about factors of the Thue–Morse word.

Proposition 2.2. *Let x be a nonempty factor of the Thue–Morse word. Then $\mathrm{ce}(x) \in \{1, \frac{3}{2}, 2\}$. Furthermore, $\mathrm{ce}(x) = 2$ if $|x| \geq 4$.*

In this paper, we prove the analogue of Proposition 2.2 for the circular critical exponent. Here the statement is more complicated and the analysis more difficult.

2.2. Walnut

Our main software tool is the Walnut prover, written by Hamoon Mousavi [8]. This Java program deals with deterministic finite automata with output (DFAOs) and k-automatic sequences $(a_n)_{n \geq 0}$. A k-DFAO is a finite-state machine $M = (Q, \Sigma_k, \delta, q_0, \Delta, \tau)$, where Q is a finite nonempty set of states, $\Sigma_k = \{0, 1, \ldots, k-1\}$ is the input alphabet, $\delta : Q \times \Sigma_k \rightarrow Q$ is the transition function (which is extended to $Q \times \Sigma_k^*$ in the obvious way), q_0 is the initial state, Δ is the output alphabet, and $\tau : Q \rightarrow \Delta$ is the output mapping. DFAOs are an obvious generalization of ordinary DFAs. A sequence $(a_n)_{n \geq 0}$ is said to be *computed by the k-DFAO M* if $\tau(\delta(q_0, (n)_k)) = a_n$, where $(n)_k$ denotes the base-k representation of n. (Unless otherwise stated, we assume that all automata read the base-k representation of n from left to right, starting with the most significant digit.) If a sequence $(a_n)_{n \geq 0}$ is computed by a k-DFAO, it is said to be k-*automatic*.

`Walnut` can evaluate the truth of a first-order statement S involving indexing of k-automatic sequences, logical connectives, and quantifiers $\exists$ and $\forall$. If there are free variables, it produces an automaton accepting the base-k representation of the values of the free variables for which S evaluates to true. One minor technical point is that the automata it produces give the correct answer, even when the input is prefixed by any number of leading zeroes.

The syntax of `Walnut` statements is more or less self-explanatory. The interested reader can enter the `Walnut` commands we give and directly reproduce our results.

All computations in this paper, unless otherwise indicated, were performed on an Apple MacBook Pro with 16 GB of memory, running macOS High Sierra, version 10.13.3. All the code we discuss is available for download at https://cs.uwaterloo.ca/~shallit/papers.html . For the Thue–Morse word, all the computations run in a matter of seconds.

2.3. Minimality

There is a notion of minimality for DFAOs that exactly parallels the notion of ordinary DFAs. We say that two states p, q of a k-DFAO are *distinguishable* if there exists a string $x \in \Sigma_k^*$ such that $\tau(\delta(p, x)) \neq \tau(\delta(q, x))$. Then the analogue of the Myhill–Nerode theorem for DFAOs is the following, which is easily proved.

Proposition 2.3. *There is a unique minimal k-DFAO equivalent to any given k-DFAO. Furthermore, a k-DFAO M is minimal iff (a) every state of M is reachable from the start state and (b) every pair of distinct states is distinguishable.*

We observe that the automata that `Walnut` computes are guaranteed to be minimal.

We will need the following lemma.

Lemma 2.4. *Let $M_1 = (Q_1, \Sigma_k, \delta_1, q_1, \Delta_1, \tau_1)$ and $M_2 = (Q_2, \Sigma_k, \delta_2, q_2, \Delta_2, \tau_2)$ be two minimal DFAOs. Let*

$$M = (Q, \Sigma_k \times \Sigma_k, \delta, q_0, \Delta_1 \times \Delta_2, \tau)$$

be the cross product automaton defined by

- *$Q = Q_1 \times Q_2$;*
- *$\delta([p, q], a) = [\delta_1(p, a), \delta_2(q, a)]$;*
- *$q_0 = [q_1, q_2]$;*
- *$\tau([p, q]) = [\tau_1(p), \tau_2(q)]$.*

Then every pair of distinct states of M is distinguishable.

Proof. Let $[p, q]$ and $[p', q']$ be two distinct states of M. Without loss of generality, assume $p \neq p'$. Then, since M_1 is minimal, we know that p and p' are distinguishable, so there exists x such that $\tau_1(\delta_1(p, x)) \neq \tau_1(\delta_1(p', x))$. Then $\tau(\delta([p, q], x)) = [\tau_1(\delta_1(p, x)), \tau_2(\delta_2(q, x))] \neq [\tau_1(\delta_1(p', x)), \tau_2(\delta_2(q', x))] = \tau(\delta([p', q'], x))$. So $[p, q]$ and $[p', q']$ are distinguishable by x. $\square$

Corollary 2.5. *Let M_1 and M_2 be minimal k-DFAOs. Form their cross product automaton, and remove all states unreachable from the start state. The result is minimal.*

Corollary 2.5 gives a way to form the minimal cross product automaton, but in practice we can do something even more efficient: namely, using a breadth-first approach, we can start from the start state $[q_1, q_2]$ and incrementally add only those states reachable from it.
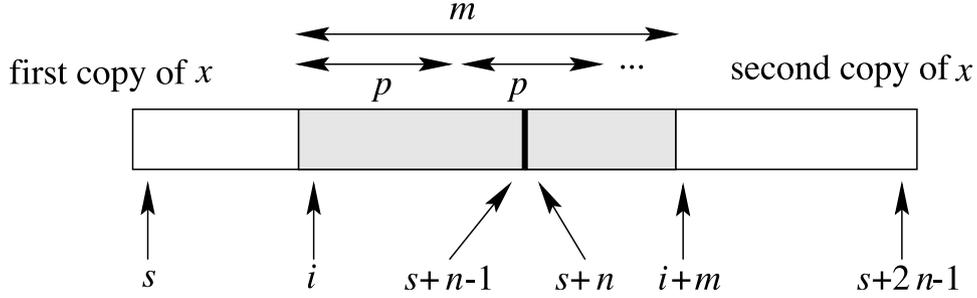
FIGURE 1. Factor of a circular word of length n.

3. FIRST-ORDER FORMULAS FOR FACTORS

We start by developing a useful first-order logical formula with free variables i, m, n, p, s. We want it to assert that

> in the circular word given by the length-n word starting
> at position s in the Thue–Morse word, there is a factor   (3.1)
> w of length m and (not necessarily least) period $p \geq 1$ starting at position i.

In order to do this, we will conceptually repeat the word $x = \mathbf{t}[s..s+n-1]$ twice, as depicted in Figure 1, where the black vertical line separates the two copies. The factor w is indicated in grey; it may or may not straddle the boundary between the two copies.

Here indices should be interpreted as "wrapping around"; the index $s+n+j$ is the same as $s+j$ for $0 \leq j < n$. Then the assertion that w has period p potentially corresponds to three different ranges of j:

– Both j and $j + p$ lie in the first copy of x, so we compare $\mathbf{t}[j]$ to $\mathbf{t}[j+p]$ for all j in this range: $i \leq j < \min(s+n-p, i+m-p)$.
– j lies in the first copy of x, but $j + p$ lies in the second copy, so we compare $\mathbf{t}[j]$ to $\mathbf{t}[j+p-n]$ for all j in this range: $\max(i, s+n-p) \leq j < \min(s+n, i+m-p)$.
– Both j and $j + p$ lie in the second copy of x, so we compare $\mathbf{t}[j-n]$ to $\mathbf{t}[j+p-n]$ for all j in this range: $\max(i, s+n) \leq j < i+m-p$.

Putting this all together, we get the following logical formula that asserts the truth of statement (3.1):

$$\mathrm{crep}(i, m, n, p, s) :=$$
$$(\forall j \, ((j \geq i) \wedge (j < s+n-p) \wedge (j < i+m-p)) \implies \mathbf{t}[j] = \mathbf{t}[j+p]) \wedge$$
$$(\forall j \, ((j \geq i) \wedge (j < s+n) \wedge (j \geq s+n-p) \wedge (j < i+m-p)) \implies \mathbf{t}[j] = \mathbf{t}[j+p-n]) \wedge$$
$$(\forall j \, ((j \geq i) \wedge (j \geq s+n) \wedge (j < i+m-p)) \implies \mathbf{t}[j-n] = \mathbf{t}[j+p-n])$$

The translation into `Walnut` is as follows:

```
def crep "(Aj ((j>=i)&(j+p<s+n)&(j+p<i+m)) => T[j]=T[j+p]) &
    (Aj ((j>=i)&(j<s+n)&(j+p>=s+n)&(j+p<i+m)) => T[j]=T[(j+p)-n]) &
    (Aj ((j>=i)&(j>=s+n)&(j+p<i+m)) => T[j-n]=T[(j+p)-n])":
```

The resulting automaton implementing $\mathrm{crep}(i, m, n, p, s)$ has 1423 states. Note that our formula does not impose conditions such as $p \geq 1$ or $p \leq n$ or $m \leq n$, which are required for crep to make sense. These conditions

(or stronger ones that imply them) must be included in any predicate that makes use of crep. Neither does the predicate assert that the given factor's *smallest* period is p; just that p is one of the possible periods.

4. PREFIXES

In this section, we prove the following theorem.

Theorem 4.1. *Every nonempty prefix of the Thue–Morse word has circular critical exponent in* $S := \{1, 2, \frac{7}{3}, \frac{5}{2}, \frac{13}{5}, \frac{8}{3}, 3\}$.

Furthermore, we will precisely characterize the n for which the circular critical exponent is each member of S.

We start by creating a first-order formula asserting that the length-n prefix, considered as a circular word, has some factor of length m and (not necessarily least) period p, satisfying $m/p = a/b$:

$$\mathrm{prefgeab}(n) := \exists i, m, p \; (p \geq 1) \wedge (m \leq n) \wedge (i < n) \wedge (bm \geq ap) \wedge \mathrm{crep}(i, m, n, p, 0).$$

Note that the condition $p \leq n$ need not be included explicitly, as it is implied by the conjunction of $m \leq n$ and $bm \geq ap$.

Next, we create a formula asserting that the length-n prefix, considered as a circular word, has a factor with exponent $> a/b$:

$$\mathrm{prefgtab}(n) := \exists i, m, p \; (p \geq 1) \wedge (m \leq n) \wedge (i < n) \wedge (bm > ap) \wedge \mathrm{crep}(i, m, n, p, 0).$$

Finally, we create a formula asserting that the length-n prefix has some factor of exponent exactly a/b:

$$\mathrm{prefeqab}(n) := \mathrm{prefgeab}(n) \wedge \neg \mathrm{prefgtab}(n).$$

No single `Walnut` command can be the direct translation of the formulas above, as there is no way to take arbitrary integer parameters a, b as input and perform multiplication by them. Nevertheless, since there are only finitely many possibilities, we can translate the above logical statements to finitely many *individual* `Walnut` commands for *each* exponent a/b. For example, for $7/3$ we can write

```
def prefge73 "E i,m,p (p>=1) & (m<=n) & (i<n) & (3*m=7*p) & $crep(i,m,n,p,0)":
def prefgt73 "E i,m,p (p>=1) & (m<=n) & (i<n) & (3*m>7*p) & $crep(i,m,n,p,0)":
def prefeq73 "$prefge(n) & ~$prefgt73(n)":
```

and similarly for the other exponents.

Proof. We can now prove Theorem 4.1 by executing the `Walnut` command

```
eval testpref "An (n>=1) => ($prefeq11(n) | $prefeq21(n) | $prefeq73(n) |
$prefeq52(n) | $prefeq135(n) | $prefeq83(n) | $prefeq31(n))":
```

(where | represents OR), and verifying that `Walnut` returns `true`. □

Table 1 gives information about the state sizes of the automata for the exponents in $S := \{1, 2, \frac{7}{3}, \frac{5}{2}, \frac{13}{5}, \frac{8}{3}, 3\}$.

In fact, even more is true. We can create a *single* 2-DFAO that on input n outputs the circular critical exponent of the prefix of length n of **t**. We do this by computing the automaton for each of the possible exponents, forming the cross product automaton, and producing the appropriate output.

Theorem 4.2. *There is a 2-DFAO of 29 states that, on input* $(n)_2$, *returns the circular critical exponent of the prefix of length* n *of* **t**.

TABLE 1. State sizes for repetition of prefixes.

a/b	Number of states for prefgeab	Number of states for prefgtab	Number of states for prefeqab	First few n accepted by prefeqab
1/1	2	4	3	$1, 2$
2/1	4	4	4	$3, 4, 6, 8, 12, 16, 24, 32, 48, 64, 96, \ldots$
7/3	4	12	7	$13, 26, 37, 52, 61, 74, 93, \ldots$
5/2	12	10	8	$5, 10, 20, 29, 40, 45, 58, 77, 80, 90, \ldots$
13/5	10	12	9	$17, 34, 53, 65, 68, 85, \ldots$
8/3	12	8	14	$9, 18, 21, 27, 33, 36, 42, 43, 49, 54, \ldots$
3/1	8	1	8	$7, 11, 14, 15, 19, 22, 23, 25, 28, 30, \ldots$

FIGURE 2. Automaton for prefixes of Thue–Morse.

Proof. We cannot compute this automaton directly in `Walnut` in its current version, but it can be computed easily from the individual automata `Walnut` computes for each exponent in $S = \{1, 2, \frac{7}{3}, \frac{5}{2}, \frac{13}{5}, \frac{8}{3}, 3\}$.

Now we can finish the (computational) proof of Theorem 4.2. We start with the automaton `prefeq11` discussed above. Next, for each of the remaining exponents a/b, we iteratively form the cross product of the current automaton with the automaton `prefeqab`, and remove unreachable states. After all exponents are handled, this gives the 29-state automaton depicted in Figure 2. $\square$

Remark 4.3. We tested the automaton in Figure 2 by explicitly calculating cce for the first 500 prefixes of **t** and comparing the results. They agreed in every case.

5. FACTORS

Instead of just prefixes, we can carry out the calculations of the previous section for all factors. The goal is to prove the following result.

TABLE 2. State sizes for automata for circular exponents of factors.

a/b	Number of states for facgeab	Number of states for facgtab	Number of states for faceqab	First occurrence (n,s) of factor with cce $= a/b$
1/1	2	6	5	(1,0)
2/1	6	9	10	(2,1)
7/3	9	50	43	(7,3)
17/7	50	51	21	(23,19)
5/2	51	71	41	(5,0)
13/5	71	63	33	(13,8)
8/3	63	36	59	(9,0)
3/1	36	24	35	(4,1)
10/3	24	26	15	(10,3)
7/2	26	22	14	(7,10)
11/3	22	21	16	(18,3)
4/1	21	1	21	(6,5)

Theorem 5.1. *Every factor of the Thue–Morse word has circular critical exponent lying in the finite set* $U := \{1, 2, \frac{7}{3}, \frac{17}{7}, \frac{5}{2}, \frac{13}{5}, \frac{8}{3}, 3, \frac{10}{3}, \frac{7}{2}, \frac{11}{3}, 4\}.$

Proof. We can mimic the previous analysis. A length-n factor $\mathbf{t}[s..s+n-1]$ can be specified by the pair (n,s).

We first make the assertion that the factor specified by (n,s), considered as a circular word, has a factor of length m that has a period p with $m/p = a/b$:

$$\mathrm{facgeab}(n,s) = \exists i, m, p\, (p \geq 1) \wedge (m \leq n) \wedge (i \geq s) \wedge (i < s + n) \wedge (bm \geq ap) \wedge \mathrm{crep}(i,m,n,p,s).$$

Next, we make the assertion that $\mathbf{t}[s..s+n-1]$, considered as a circular word, has a factor with exponent $> a/b$:

$$\mathrm{facgtab}(n,s) = \exists i, m, p\, (p \geq 1) \wedge (m \leq n) \wedge (i \geq s) \wedge (i < s + n) \wedge (bm > ap) \wedge \mathrm{crep}(i,m,n,p,s).$$

Finally, we make the assertion $\mathbf{t}[s..s+n-1]$, considered as a circular word, has a factor of exponent exactly a/b and no larger:

$$\mathrm{faceqab}(n,s) = \mathrm{facgeab}(n,s) \wedge \neg\, \mathrm{facgtab}(n,s).$$

Now we just make the assertion that one of the 12 possibilities always occurs:

```
eval testfac "An (n>=1) => (As ($faceq11(n,s) | $faceq21(n,s) |
$faceq73(n,s) | $faceq177(n,s) | $faceq52(n,s) | $faceq135(n,s) |
$faceq83(n,s) | $faceq31(n,s) | $faceq103(n,s) | $faceq72(n,s) |
$faceq113(n,s) | $faceq41(n,s)))":
```

and `Walnut` evaluates it to be true. Furthermore, it is easy to check that each possibility occurs at least once, as given in Table 2. □

Theorem 5.2. *There is a 204-state 2-DFAO that, on input (n,s) in base 2, outputs*

$$\mathrm{cce}(\mathbf{t}[s..s+n-1]).$$

TABLE 3. State sizes for facab and facsmallab.

a/b	Number of states for facab	Number of states for facsmallab	First few n accepted by facsmallab
1/1	3	3	$1, 2$
2/1	3	4	$3, 4, 6, 8, 12, 16, 24, 32, 48$
7/3	20	20	$7, 13, 14, 19, 21, 25, 26, 27, 28, 29, 33, 35, 37, 38, 42, 43, 45, 49, 50$
17/7	7	7	$23, 31, 39, 46, 47$
5/2	9	16	$5, 9, 10, 11, 15, 17, 18, 20, 22, 30, 34, 36, 40, 41, 44$

Proof. As before, we use the product construction to combine the automata for faceqab(n, s) for all twelve possibilities for a/b. The automaton is too large to display here, but it is available at https://cs.uwaterloo.ca/~shallit/papers.html . ☐

Remark 5.3. We tested the correctness of our automaton by comparing its result to the result of thousands of randomly chosen factors of varying lengths of **t**. It passed all tests.

5.1. Smallest circular critical exponents for each length

For every length n, we can consider the least circular critical exponent over all factors $\mathbf{t}[s..s+n-1]$ of length n. Define

$$\mathrm{lcce}(n) = \min_{\substack{x \text{ a factor of } \mathbf{t} \\ |x|=n}} \mathrm{cce}(x).$$

Theorem 5.4. *For all $n \geq 1$ we have $\mathrm{lcce}(n) \in T$ where $T := \{1, 2, \frac{7}{3}, \frac{17}{7}, \frac{5}{2}\}$.*

Proof. First, we create a first-order logic statement asserting that there exists some length-n factor whose circular exponent equals a/b:

$$\mathrm{facab}(n) := \exists s \ \mathrm{faceqab}(n, s).$$

Next, we create a statement asserting that a/b is the least circular critical exponent for words of length n; in other words, that there exists some length-n factor whose circular critical exponent equals a/b, and furthermore every length-n factor has circular critical exponent $\geq a/b$:

$$\mathrm{facsmallab}(n) := \mathrm{facab}(n) \ \wedge \ (\forall s \ \mathrm{facgeab}(n, s)).$$

Finally, we just assert that for every $n \geq 1$, at least one of the five alternatives holds:

```
eval smallfactest "An (n >=1) => ($facsmall11(n) | $facsmall21(n) |
$facsmall73(n) | $facsmall177(n) | $facsmall52(n))":
```

`Walnut` evaluates this to be true. ☐

The sizes of the automata occurring in the proof are summarized in Table 3.

Theorem 5.5. *There is a 25-state 2-DFAO, that on input $(n)_2$ computes the least circular critical exponent over all factor of **t** of length n. It is given in Figure 3.*
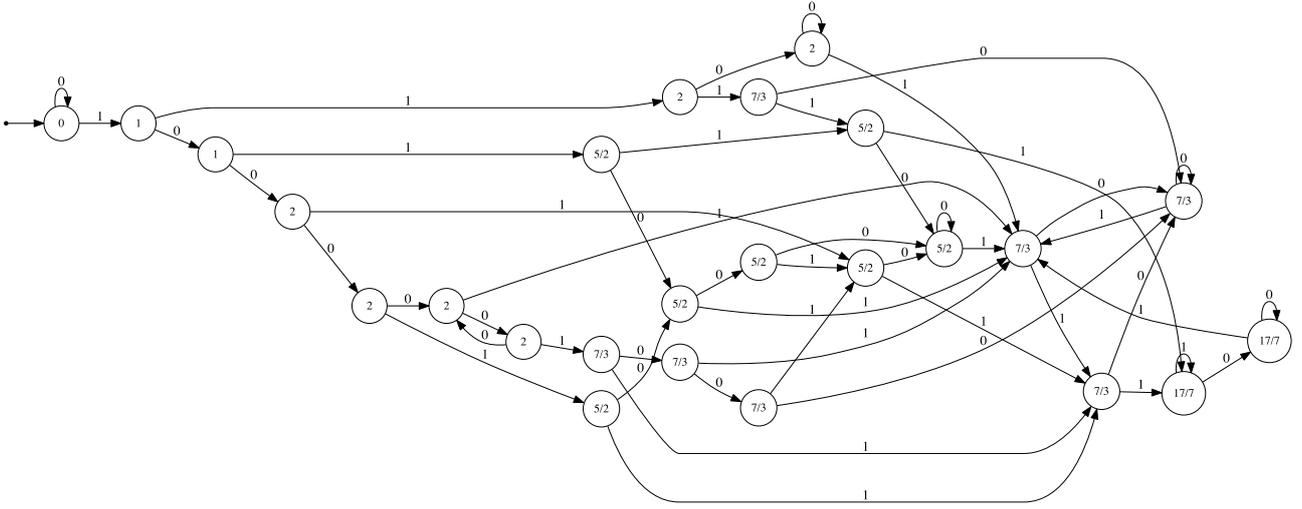
FIGURE 3. Automaton computing least possible cce of a factor, for each length.

Proof. Combine, using the cross product construction, the five automata facsmallab for $a/b \in \{1/1, 2/1, 7/3, 17/7, 5/2\}$ as before. The output of each state is depicted in the center of the corresponding circle. □

5.2. Greatest circular critical exponents for each length

We can also consider the greatest circular critical exponent over all factors $\mathbf{t}[s..s + n - 1]$ of length n. Define

$$\mathrm{gcce}(n) = \max_{\substack{x \text{ a factor of } \mathbf{t} \\ |x|=n}} \mathrm{cce}(x).$$

Theorem 5.6. *For all $n \geq 1$ we have $\mathrm{gcce}(n) \in V$ where $V := \{1, 2, 3, \frac{7}{2}, 4\}$.*

Proof. We define

$$\mathrm{faclargeab}(n) = (\exists s \ \mathrm{faceqab}(n, s)) \ \wedge \ (\forall s \ \neg \mathrm{facgtab}(n, s)).$$

The number of states, and the first few n that match the category, are given in Table 4.
We then verify the claim by writing

```
eval largefactest "An (n>=1) => ($faclarge11(n) | $faclarge21(n) |
$faclarge31(n) | $faclarge72(n) | $faclarge41(n))":
```

which returns `true`. □

Theorem 5.7. *There is a 9-state 2-DFAO, that on input $(n)_2$, returns the greatest circular critical exponent over all length-n factors of $\mathbf{t}$.*

Proof. We follow the same approach as before, using the cross product construction to combine the automata faclargeab for $a/b \in \{1, 2, 3, \frac{7}{2}, 4\}$. The result is depicted in Figure 4. □

TABLE 4. State sizes for faclargeab.

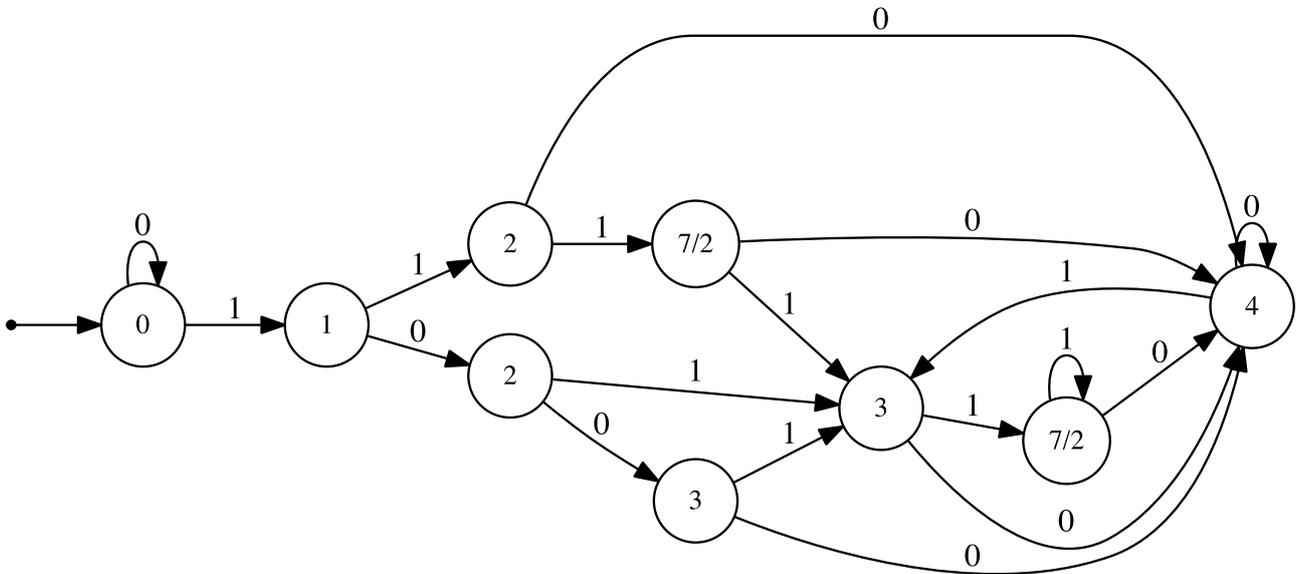| a/b	Number of states for faclargeab	First few n matching the case
1/1	2	1
2/1	3	2,3
3/1	8	4,5,9,13,15,17,21,25,29,33,37,41,45,49,53,57,61, …
7/2	7	7,11,19,23,27,31,35,39,43,47,51,55,59,63, …
4/1	5	6,8,10,12,14,16,18,20,22,24,26,28,30,32, …

FIGURE 4. Automaton computing greatest possible cce of a factor, for each length.

5.3. Sets of circular critical exponents

We can get even more! Define the set of all possible circular critical exponents of factors of length n as follows:

$$\mathrm{ace}(n) = \{\mathrm{cce}(x) \ : \ |x| = n \geq 1 \text{ and } x \text{ is a factor of } \mathbf{t} \}.$$

Theorem 5.8. *The range of* $\mathrm{ace}(n)$ *consists of exactly* 31 *distinct sets as enumerated in Table 5.*

Proof. This follows immediately from our proof of the next result. □

Theorem 5.9. *There is a 49-state 2-DFAO that, on input n written in base 2, outputs $\mathrm{ace}(n)$.*

Proof. We use the same cross product automaton technique as before. This time, we use the automata facab for each $a/b \in U$. The result is depicted in Figure 5. The outputs associated with each state are encoded as 12-bit numbers, one for each of the 12 possible exponents in increasing order, with least significant bit corresponding to exponent 4. Square states are "transient" and circular states are "recurrent". □

TABLE 5. Sets of circular critical exponents for lengths n.

Set of circular critical exponents S	Encoding in automaton	First few n for $\mathrm{ace}(n) = S$
$\{1\}$	2048	$\{1\}$
$\{1, 2\}$	3072	$\{2\}$
$\{2\}$	1024	$\{3\}$
$\{2, 3\}$	1040	$\{4\}$
$\{\frac{5}{2}, 3\}$	144	$\{5\}$
$\{2, 4\}$	1025	$\{6\}$
$\{\frac{7}{3}, 3, \frac{7}{2}\}$	532	$\{7\}$
$\{2, 3, 4\}$	1041	$\{8, 12, 16, 24, 32, 48, 64, 96, 128, 192, \ldots\}$
$\{\frac{5}{2}, \frac{8}{3}, 3\}$	176	$\{9, 15\}$
$\{\frac{5}{2}, 3, \frac{10}{3}, 4\}$	153	$\{10, 20, 40, 80, 160, \ldots\}$
$\{\frac{5}{2}, 3, \frac{7}{2}\}$	148	$\{11\}$
$\{\frac{7}{3}, \frac{13}{5}, \frac{8}{3}, 3\}$	624	$\{13\}$
$\{\frac{7}{3}, 3, \frac{7}{2}, 4\}$	533	$\{14, 28, 56, 112, 224, \ldots\}$
$\{\frac{5}{2}, \frac{13}{5}, \frac{8}{3}, 3\}$	240	$\{17, 41, 137, \ldots\}$
$\{\frac{5}{2}, \frac{8}{3}, 3, \frac{11}{3}, 4\}$	179	$\{18, 30, 36, 60, 72, 120, 144, \ldots\}$
$\{\frac{7}{3}, \frac{8}{3}, 3, \frac{7}{2}\}$	564	$\{19, 67, \ldots\}$
$\{\frac{7}{3}, \frac{5}{2}, \frac{8}{3}, 3\}$	688	$\{21\}$
$\{\frac{5}{2}, 3, \frac{7}{2}, \frac{11}{3}, 4\}$	151	$\{22, 44, 88, 176, \ldots\}$
$\{\frac{17}{7}, \frac{5}{2}, 3, \frac{7}{2}\}$	404	$\{23, 71, \ldots\}$
$\{\frac{7}{3}, \frac{5}{2}, \frac{13}{5}, \frac{8}{3}, 3\}$	752	$\{25, 29, 33, 37, 45, 49, 53, 57, 61, 65, 69, 73, 77, \ldots\}$
$\{\frac{7}{3}, \frac{13}{5}, \frac{8}{3}, 3, \frac{10}{3}, 4\}$	633	$\{26, 52, 104, 208, \ldots\}$
$\{\frac{7}{3}, \frac{5}{2}, \frac{8}{3}, 3, \frac{7}{2}\}$	692	$\{27, 35, 43, 51, 59, 75, 83, 91, 99, 107, 115, 123, \ldots\}$
$\{\frac{17}{7}, \frac{5}{2}, \frac{8}{3}, 3, \frac{7}{2}\}$	436	$\{31, 39, 47, 55, 63, 79, 87, 95, 103, 111, 119, 127, \ldots\}$
$\{\frac{5}{2}, \frac{13}{5}, \frac{8}{3}, 3, \frac{10}{3}, \frac{11}{3}, 4\}$	251	$\{34, 68, 82, 136, 164, \ldots\}$
$\{\frac{7}{3}, \frac{8}{3}, 3, \frac{7}{2}, 4\}$	565	$\{38, 76, 134, 152, \ldots\}$
$\{\frac{7}{3}, \frac{5}{2}, \frac{8}{3}, 3, \frac{10}{3}, \frac{11}{3}, 4\}$	699	$\{42, 84, 168, \ldots\}$
$\{\frac{17}{7}, \frac{5}{2}, 3, \frac{7}{2}, \frac{11}{3}, 4\}$	407	$\{46, 92, 142, 184, \ldots\}$
$\{\frac{7}{3}, \frac{5}{2}, \frac{13}{5}, \frac{8}{3}, 3, \frac{10}{3}, \frac{11}{3}, 4\}$	763	$\{50, 58, 66, 90, 98, 100, 106, 114, 116, 122, \ldots\}$
$\{\frac{7}{3}, \frac{5}{2}, \frac{8}{3}, 3, \frac{7}{2}, \frac{11}{3}, 4\}$	695	$\{54, 70, 86, 102, 108, 118, \ldots\}$
$\{\frac{17}{7}, \frac{5}{2}, \frac{8}{3}, 3, \frac{7}{2}, \frac{11}{3}, 4\}$	439	$\{62, 78, 94, 110, 124, 126, \ldots\}$
$\{\frac{7}{3}, \frac{5}{2}, \frac{13}{5}, \frac{8}{3}, 3, \frac{10}{3}, 4\}$	761	$\{74, 148, \ldots\}$

6. FINAL REMARKS

Evidently one could (in principle) perform the same sort of analysis for many other famous infinite words. We carried this out for the regular paperfolding word

$$\mathbf{p} = 00100110001101100010\cdots$$

(see, for example, [5, 7]), and the results are summarized below. We omit the details, but the `Walnut` code proving these results is available at https://cs.uwaterloo.ca/~shallit/papers.html . The computations were nontrivial. `Walnut` was invoked using the Linux command

FIGURE 5. Automaton computing sets of circular critical exponents for factors of length n.

```
java -Xmx16000M -d64 Main.prover
```

on a 4 CPU AMD Opteron 6380 SE with 256GB RAM. The analogue of crep for **p** has 4226 states and took 9 min to compute. The largest intermediate automaton had 822,161 states.

Theorem 6.1.

(a) *Every nonempty prefix of* **p** *has circular critical exponent lying in* $\{1, 2, \frac{7}{3}, 3, \frac{10}{3}, 4, \frac{13}{3}, 5\}$.

(b) *Every nonempty factor of* **p** *has circular critical exponent lying in*
    $\{1, 2, \frac{7}{3}, \frac{5}{2}, \frac{8}{3}, \frac{11}{4}, 3, \frac{10}{3}, \frac{7}{2}, 4, \frac{13}{3}, 5, 6\}$.

(c) *The least circular critical exponent of* **p**, *over all factors of length* n, *lies in*
    $\{1, 2, \frac{7}{3}, \frac{5}{2}, \frac{8}{3}, \frac{11}{4}, 3\}$.

(d) *The greatest circular critical exponent of* **p**, *over all factors of length* n, *lies in* $\{1, 2, 3, 4, 5, 6\}$.

(e) *There are exactly 16 distinct possible sets of circular critical exponents for factors of length* $n \geq 1$ *of* **p**.

In principle, we could also treat the Rudin–Shapiro sequence. For example, one might be able to prove the following.

Conjecture 6.2. *Every nonempty factor of the Rudin–Shapiro sequence has a circular critical exponent lying in*

$$\left\{1, 2, \frac{5}{2}, \frac{8}{3}, 3, \frac{10}{3}, \frac{7}{2}, \frac{11}{3}, \frac{15}{4}, 4, \frac{21}{5}, \frac{13}{3}, \frac{14}{3}, 5, 6, 7, 8\right\}.$$

However, so far we have not been able to complete the computations with `Walnut` (it runs out of space).

For some infinite words, the sets under consideration will be infinite, and hence another kind of analysis will be needed. As an example, consider the infinite word $210201\cdots$ that is a fixed point of $2 \to 210$, $1 \to 20$, $0 \to 1$. It is well-known that this word is squarefree, but contains factors with exponent arbitrarily close to 2. In this case there will be no finite analogue of our Proposition 2.2 and Theorem 4.1. The same case occurs for the Fibonacci word (the fixed point of $0 \to 01$ and $1 \to 0$).

References

[1] A. Aberkane and J.D. Currie, There exist binary circular $5/2^+$ power free words of every length. *Electron. J. Comb.* **11** (2004) #R10.

[2] A. Aberkane and J.D. Currie, The Thue-Morse word contains circular $5/2^+$ power free words of every length. *Theor. Comput. Sci.* **332** (2005) 573–581.

[3] J.-P. Allouche and J.O. Shallit, The ubiquitous Prouhet-Thue-Morse sequence, in Sequences and Their Applications, Proceedings of SETA '98, edited by C. Ding, T. Helleseth and H. Niederreiter. Springer-Verlag, London (1999) 1–16.

[4] J. Berstel, Axel Thue's work on repetitions in words, in *Séries Formelles et Combinatoire Algébrique*, edited by P. Leroux and C. Reutenauer. Vol. 11 of Publications du LaCim. Université du Québec à Montréal (1992) 65–80.

[5] F.M. Dekking, M. Mendès France and A.J. van der Poorten, Folds! *Math. Intell.* **4** (1982) 130–138. (Erratum, 5 (1983) 5.)

[6] D.S. Fitzpatrick, There are binary circular cube-free words of length n contained within the Thue-Morse word for all positive integers n. *Ars Combinatoria* **74** (2005) 323–329.

[7] M. Mendès France and G. Tenenbaum, Dimension des courbes planes, papiers pliés et suites de Rudin-Shapiro. *Bull. Soc. Math. France* **109** (1981) 207–215.

[8] H. Mousavi, Automatic theorem proving in Walnut. Preprint https://arxiv.org/abs/1603.06017. Code available at https://github.com/hamousavi/Walnut, 2016 (2018).

[9] H. Mousavi and J. Shallit, Repetition avoidance in circular factors, in *DLT 2013 – Developments in Language Theory*, edited by M.-P. Béal and O. Carton. Vol. 7907 of *Lecture Notes in Computer Science*, Springer-Verlag, Berlin, Heidelberg (2013) 384–395.

[10] A. Thue, Über die gegenseitige Lage gleicher Teile gewisser Zeichenreihen. *Norske vid. Selsk. Skr. Mat. Nat. Kl.* **1** (1912) 1–67. (Reprinted in *Selected Mathematical Papers of Axel Thue*, edited by T. Nagell. Universitetsforlaget, Oslo (1977) 413–478.)