String Attractors and Combinatorics on Words*

Sabrina Mantaci[1], Antonio Restivo[1], Giuseppe Romana[1], Giovanna Rosone[2], and Marinella Sciortino[1]✉

[1] University of Palermo, Italy,
sabrina.mantaci@unipa.it, antonio.restivo@unipa.it,
giuseppe.romana01@community.unipa.it, marinella.sciortino@unipa.it
[2] University of Pisa, Italy,
giovanna.rosone@unipi.it

Abstract. The notion of *string attractor* has recently been introduced in [Prezza, 2017] and studied in [Kempa and Prezza, 2018] to provide a unifying framework for known dictionary-based compressors. A string attractor for a word $w = w[1]w[2]\cdots w[n]$ is a subset Γ of the positions $\{1, \ldots, n\}$, such that all distinct factors of w have an occurrence crossing at least one of the elements of Γ. While finding the smallest string attractor for a word is a NP-complete problem, it has been proved in [Kempa and Prezza, 2018] that dictionary compressors can be interpreted as algorithms approximating the smallest string attractor for a given word.

In this paper we explore the notion of string attractor from a combinatorial point of view, by focusing on several families of finite words. The results presented in the paper suggest that the notion of string attractor can be used to define new tools to investigate combinatorial properties of the words.

Keywords: String attractor, Burrows-Wheeler transform, Lempel-Ziv encoding, run-length encoding, Thue-Morse word, de Brujin word

1 Introduction

The notion of *String Attractor* has been recently introduced and studied in [25, 11] to find a common principle underlying the main techniques constituting the fields of dictionary-based compression. It is defined as a subset of the text's positions such that all distinct factors have an occurrence crossing at least one of the string attractor's elements. From one hand the problem of finding the smallest string attractor of a word has been proved to be NP-complete, on the other hand dictionary compressors can be interpreted as algorithms approximating the smallest string attractor for a given word [11]. Moreover, approximation rates with respect to the smallest string attractor can be derived for most known compressors. In particular compressors based on the Burrows-Wheeler Transform and the dictionary-based compressors are considered.

The Burrows-Wheeler Transform (BWT) is a reversible transformation that was introduced in 1994 in the field of Data Compression and it also largely used for self-indexing data structures. It has several combinatorial properties that make it a versatile tool in several contexts and applications [26, 16, 20, 28, 19, 8].

Dictionary-based compressors are mainly based on a technique originated in two theoretical papers of Ziv and Lempel [30, 31]. Such compressors, that are able to combine compression power and compression/decompression speed, are based on a paper in which combinatorial properties of word factorization are explored [14]. The relationship between LZ77 and BWT has been investigated from the algorithmic point of view in [24].

In this paper we explore the notion of string attractor from a combinatorial point of view. In particular, we compute the size of a smallest string attractor for infinite families of words that are well known in the field of Combinatorics on Words: standard Sturmian words (and some their extension to bigger alphabets), Thue-Morse words and de Brujin words. In particular, we show that the size of the smallest string attractor for standard Sturmian words is 2 and it contains two consecutive positions. For the de Brujin words the size of the smallest string attractor grows asintotically as $\frac{n}{\log n}$, where n is the length of the word. We show a string attractor of size $\log n$ for Thue-Morse words and we conjecture that this size is minimum. From the results presented in the paper, we believe that the distribution of the position in the smallest string attractor of a word, in addition to its size, can provide some interesting information about the combinatorial properties of the word itself. For this reason the notion of string attractor can provide hints for defining new methods and measures to investigate the combinatorial complexity of the words.

2 Preliminaries

Let $\Sigma = \{a_1, a_2, \ldots, a_\sigma\}$ be a finite ordered alphabet with $a_1 < a_2 < \ldots < a_\sigma$, where $<$ denotes the standard lexicographic order. We denote by Σ^* the set of words over Σ. Given a finite word $w = w_1 w_2 \cdots w_n \in \Sigma^*$ with each $w_i \in \Sigma$, the length of w, denoted $|w|$, is equal to n.

Given a finite word $w = w_1 w_2 \cdots w_n$ with each $w_i \in \Sigma$, a *factor* of a word w is written as $w[i, j] = w_i \cdots w_j$ with $1 \leq i \leq j \leq n$. A factor of type $w[1, j]$ is called a *prefix*, while a factor of type $w[i, n]$ is called a *suffix*. We also denote by $w[i]$ the i-th letter in w for any $1 \leq i \leq n$.

We denote by $\tilde{w}$ the reversal of w, given by $\tilde{w} = w_n \cdots w_2 w_1$. If w is a word that has the property of reading the same in either direction, i.e. if $w = \tilde{w}$, then w is called a *palindrome*.

We say that two words $x, y \in \Sigma^*$ are *conjugate*, if $x = uv$ and $y = vu$, where $u, v \in \Sigma^*$. Conjugacy between words is an equivalence relation over Σ^*.

Given a finite word w, w^k denotes the word obtained by concatenating k copies of w. A nonempty word $w \in \Sigma^+$ is *primitive* if $w = u^h$ implies $w = u$ and $h = 1$. A word x is *periodic* if there exists a positive integer p such that $x[i] = x[j]$ if $i = j \mod p$. The integer p is called *period* of x.

A *Lyndon word* is a primitive word which is the minimum in its conjugacy class, with respect to the lexicographic order relation. We call *Lyndon conjugate* of a primitive word w the conjugate of w that is a Lyndon word.

The Burrows-Wheeler Transform is a permutation $bwt(v)$ of the symbols in v, obtained as the concatenation of the last symbol of each conjugate in the list of the lexicographically sorted conjugates of v.

The LZ-factorization of a word w is its factorization $s = p_1 \cdots p_z$ built left to right in a greedy way by the following rule: each new factor (also called an LZ-phrase) p_i is either the leftmost occurrence of a letter in w or the longest prefix of $p_i \cdots p_z$ which occurs, as a factor, in $p_1 \cdots p_{i-1}$.

3 String Attractor of a word

In this section we describe the notion of string attractor that is a combinatorial object introduced in [25, 10] to obtain a unifying framework for dictionary compressors.

Definition 1. *A* string attractor *of a word $w \in \Sigma^n$ is a set of γ positions $\Gamma = \{j_1, \ldots, j_\gamma\}$ such that every factor $w[i,j]$ has an occurrence $w[i',j'] = w[i,j]$ with $j_k \in [i',j']$, for some $j_k \in \Gamma$.*

Simply put, a string attractor for a word w is a set of positions in w such that all distinct factors of w have an occurrence crossing at least one of the attractor's elements. Note that, trivially, any set that contains a string attractor for w, is a string attractor for w as well. Note also that a word can have different string attractors that are not included into each other. We are interested in finding a *smallest string attractor*, i.e. a string attractor with a minimum number of elements. We denote by $\gamma^*(w)$ the size of the smallest string attractor for w. Note that all the factors made of a single letter should be covered, and therefore $\gamma^*(w) \geq |\Sigma|$.

Example 1. Let $w = adcbaadcbadc$ be a word on the alphabet $\Sigma = \{a, b, c, d\}$. A string attractor for w is for instance $\Gamma = \{1, 4, 6, 8, 11\}$. Note that, in order to have a string attractor, position 1 can be removed from Γ, since all the factors that cross position 1 have a different occurrence that crosses a different position in Γ. Therefore $\Gamma' = \{4, 6, 8, 11\}$ is also a string attractor for w with a smaller number of elements. The positions of Γ' are underlined in

$$w = adc\underline{b}a\underline{a}d\underline{c}ba\underline{d}c.$$

Γ' is also a smallest string attractor since $|\Gamma'| = |\Sigma|$. Then $\gamma^*(w) = 4$. Remark that the sets $\{3, 4, 5, 11\}$ and $\{3, 4, 6, 7, 11\}$ are also string attractors for w. It is easy to verify that the set $\Delta = \{1, 2, 3, 4\}$ is not a string attractor since, for instance, the factor aa does not intersect any position in Δ.

The following two propositions, proved in [25], are useful to derive a lower bound on the value of γ^*.

Proposition 1. *Let Γ be a string attractor for the word w. Then, w contains at most $|\Gamma|k$ distinct factors of length k, for every $1 \leq k \leq |w|$.*

Proposition 2. *Let $w \in \Sigma^*$ and let r be the length of its longest repeated factor. Then it holds $\gamma^*(w) \geq \frac{|w|-r}{r+1}$.*

When a word w is obtained as a concatenation of two factors u and v, an upper bound for $\gamma^*(w)$ can be expressed in terms of $\gamma^*(u)$ and $\gamma^*(v)$, as stated in the following theorem.

Proposition 3. *Let u and v two words, then $\gamma^*(uv) \leq \gamma^*(u) + \gamma^*(v) + 1$.*

Example 2. The bound defined in the previous proposition is tight. In fact, let $u = \underline{b}aa\underline{a}ba$ and $v = c\underline{d}cc\underline{c}d$ be two words in which the positions of the smallest string attractors are underlined. If we consider $uv = \underline{b}aa\underline{a}ba\underline{cd}cc\underline{c}d$, the underlined positions represent one of the smallest string attractors for uv, as one can verify.

The following proposition gives an upper and lower bound for γ^*, when a power of a given word is considered.

Proposition 4. *Let $w = u^n$. Then $\gamma^*(u) \leq \gamma^*(u^n) \leq \gamma^*(u) + 1$.*

Example 3. The upper bound given by Proposition 4 is tight. In fact consider the word $u = abbaab$. It is easy to check that the only smallest string attractors for u are $\Gamma_1 = \{2, 4\}$ and $\Gamma_2 = \{3, 5\}$. In order to find the smallest string attractor for $u^2 = abbaababbaab$, we remark that neither Γ_1 nor Γ_2 (neither any string attractor obtained from them by moving some position from the first to the second occurrence of u) cover all the new factors that appear after the concatenation. A way to get the smallest string attractor for u^2 is to add to Γ_1 or Γ_2, the position corresponding either to the end of the first occurrence of u or the beginning of the second occurrence. For instance, $\Gamma^* = \{2, 4, 6\}$ is a smallest string attractor for u^2.

Example 4. Remark that $\gamma^*(u^n)$ can be equal to $\gamma^*(u)$ although different points for the string attractor could be chosen. For instance, let $u = a\underline{ba}b\underline{c}bc$ be a word whose smallest string attractor is $\{2, 3, 5\}$ (the underlined letters). Then $u^2 = \underline{ab}abcb\underline{c}ababcbc$ has a string attractor $\{3, 6, 7\}$ of cardinality 3. Remark that $\{2, 3, 5\}$ is not a string attractor for u^2.

A straightforward consequence of Proposition 4 is the following:

Corollary 1. *If u and v are conjugate words, then $|\gamma^*(u) - \gamma^*(v)| \leq 1$.*

Example 5. Consider the word $w = babbaaa$. Then a smallest string attractor for w is $\{3, 5\}$, i.e. $\gamma^*(w) = 2$. Consider its conjugate $u = a\underline{b}ab\underline{b}a\underline{a}$. Its smallest string attractor is $\{2, 4, 6\}$, i.e. $\gamma^*(u) = 3$. Note that the Lyndon word does not have necessarily the smallest γ^* among the conjugates. In fact, for instance, the Lyndon conjugate of w is $aaababb$, and it is easy to verify that one of the smallest string attractor is $\{3, 4, 6\}$.

4 Approximating a string attractor via compressors

In [11] the authors show that many of the most well-known compression schemes reducing the texts size by exploiting its repetitiveness can induce string attractors whose sizes are bounded by the repetitiveness measures associated to such compressors. In particular, straight-line programs, Run-Length Burrows-Wheeler transform, macro schemes, collage systems, and the compact directed acyclic word graph are considered. Here we report some results related to the Burrows-Wheeler transform, collage systems, and Lempel-Ziv 77 (that is a particular macro-scheme) that provide upper bounds on the size of the smallest string attractor for a given word. Such bounds will be used in next sections to compute the string attractors for known families of finite words.

The first theorem, proved in [11], states a connection between a string attractor of a word w and the runs of equal letters in the output produced by the BWT.

Theorem 1. *Let Σ be a finite alphabet, $w \in \Sigma^*$ and r be the number of equal-letter runs in the $\mathtt{bwt}(w\$)$, where $\$ \notin \Sigma$ is a symbol smaller than any symbol in Σ. Then, w has a string attractor of size r.*

In particular, in the proof of Theorem 1 the string attractor is constructed by considering the position of the symbols in w that correspond, in the output of the transformation, to the first occurrence of a symbol in each run (or, equivalently, the last occurrence of a symbol in each run).

The following result, proved in [11], states the relationship between a string attractor of a word w and the number of phrases in the LZ-parsing of w.

Theorem 2. *Given a word w, there exists a string attractor of w of size equal to the number of phrases of its LZ-parsing.*

By using the previous result, a string attractor can be constructed by considering the set of positions at the end of each phrase.

The following theorem in [11] gives a connection between a particular class of grammars, called *collage systems* [12], and string attractors.

Definition 2. *A collage system is a set of c rules of four possible types:*

- $X \to a$: *nonterminal X expands to a terminal a;*
- $X \to AB$: *nonterminal X expands to AB, with A and B nonterminals different from X;*
- $X \to R^\ell$: *nonterminal X expands to nonterminal $R \neq X$ repeated ℓ times;*
- $X \to K[l,r]$: *nonterminal X expands to a substring of the expansion of nonterminal $K \neq X$.*

Theorem 3. *Let $G = \{X_i \to a_i, i = 1, \ldots, g'\} \cup \{X_i \to A_iB_i, i = 1, \ldots, g''\} \cup \{Y_i \to Z_i^{l_i}, l_i \geq 2, i = 1, \ldots, g'''\} \cup \{W_i \to K_i[l_i, r_i], i = 1, \ldots, g''''\}$ be a collage system of size $g = g' + g'' + g''' + g''''$ generating a word w. Then w has a string attractor of size at most g.*

By using this theorem one can easily find the minimum string attractor for a particular class of very "regular" strings, as shown in the following corollary.

Corollary 2. *Let $u \in \Sigma^*$ be a word. If u is the form of $u = \sigma_{i_1}^{n_1}\sigma_{i_2}^{n_2}\cdots\sigma_{i_k}^{n_k}$ (where σ_{i_j}'s are different symbols in Σ), then $\Gamma = \{n_1, n_1{+}n_2, \ldots, n_1{+}\ldots{+}n_k\}$ is a string attractor of minimum size for u. Therefore $\gamma^*(u) = \sigma$.*

5 Minimum size string attractors

In this section, we analyze the words whose smallest string attractor has size equal to the size of the alphabet, that is the minimum possible size. In the following two subsections we distinguish the case of binary words and the case of words over alphabets with cardinality greater than 2.

5.1 Binary words

In this subsection we focus on an infinite family of finite binary words whose minimum string attractor has size 2.

Standard Sturmian words is a very well known family of binary words that are the basic bricks used for the construction of infinite Sturmian words, in the sense that every characteristic Sturmian word is the limit of a sequence of standard words (cf. Chapter 2 of [15]). These words have a multitude of characterizations and appear as extreme case in a very great range of contexts [13, 5, 6, 17, 18]. More formally, standard Sturmian words can be defined in the following way which is a natural generalization of the definition of the Fibonacci word. Let $q_0, q_1, \ldots q_n, \ldots$ any sequence of natural integers such that $q_0 \geq 0$ and $q_i > 0$ $(i = 1, \ldots, n)$, called *directive sequence*. The sequence $\{s_n\}_{n\geq 0}$ can be defined inductively as follows: $s_0 = b$, $s_1 = a$, $s_{n+1} = (s_n)^{q_{n-1}} s_{n-1}$, for $n > 1$. We denote by *Stand* the set of all words s_n, $n \geq 0$, constructed for any directive sequence of integers.

Furthermore, another characterization of standard Sturmian words is related to the Burrows Wheeler transform (BWT) since, for binary alphabets, the application of the BWT to standard Sturmian words produces a total clustering of all the instances of any character (cf. [20]), as reported in the following theorem.

Theorem 4 ([20]). *Let $w \in \{a, b\}^*$. Then w is a conjugate of a word in Stand if and only if $bwt(w) = b^p a^q$ with $gcd(p, q) = 1$, where p and q are the number of b's and a's in w, respectively.*

In the following theorem, for each standard Sturmian word, we individuate a string attractor, whose positions are strictly related with particular decompositions of such words depending on their periodicity. In particular, we recall that $Stand = \{a, b\} \cup PER\{ab, ba\}$ (cf. [17]), where PER is the set of all words v having two periods p and q such that $gcd(p, q) = 1$ and $|v| = p + q - 2$. Given a word $w \in Stand$, we denote by $\pi(w)$ its prefix of length $|w| - 2$, belonging to the set PER, uniquely defined by using previous equality. By using a property of words in PER (cf. [17]), $\pi(w) = QxyP = PyxQ$, where $x \neq y$ are characters and Q and P are uniquely determined palindromes. So, a standard Sturmian word $w = \pi(w)ba$ can be decomposed as $w = QxyPba = PyxQba$. We call PER-decompositions such factorizations of w.

Theorem 5. *For each $w \in Stand$ with $|w| \geq 2$, let η be the length of the longest palindromic proper prefix of $\pi(w)$, the set $\Gamma_1 = \{\eta + 1, \eta + 2\}$ or the set $\Gamma_2 = \{|w| - \eta - 3, |w| - \eta - 2\}$ is a minimum string attractor for w.*

Proof. Let us suppose that $w = \pi(w)ba$. By using PER-decompositions, a Standard sturmian word can be decomposed as $w = QxyPba = PyxQba$, $\pi(w) = QxyP = PyxQ$, where $x \neq y$ are characters and Q and P are uniquely determined palindromes. Let us suppose that $|Q| > |P|$. So, $\eta = |Q|$. Firstly we suppose that $x = b$. This means that $w = QbaPba = PabQba$. From a result in [1] $aPabQb$ and $bQbaPa$ are the smallest and the greatest conjugates in the lexicographic order, respectively. By Theorem 4 and by using an argument similar to the proof of Theorem 1, a string attractor can be constructed by considering the positions corresponding to the end of each run. It is possible to see that such positions correspond to the two characters following the prefix P of length $|w| - \eta - 4$. If $x = a$, then $w = QabPba = PbaQba$. In this case $aQabPb$ and $bPbaQa$ are the smallest and the greatest conjugates in the lexicographic order, respectively. In this case the ending positions of each run in the output of BWT correspond to the two characters following the prefix Q. So, the positions in the string attractor are $\{\eta + 1, \eta + 2\}$. The case $w = \pi(w)ab$ can be proved analogously by considering the starting characters of each run in the clustered output of BWT. □

Example 6. Given the standard Sturmian word $w = ababaababaabababa$, the PER-decompositions of w are $ababaababa.ab.aba.ba = aba.ba.ababaababa.ba$, then $\{11, 12\}$ is a (smallest) string attractor for w, since $\eta = 10$. Given $v = abaababaababa$, its PER-decompositions are $abaaba.ba.aba.ba = aba.ab.abaaba.ba$ and $\eta = 6$. So, $\{4, 5\}$ is a smallest string attractor for v.

The previous theorem shows an infinite family of finite binary words such that the size of the smallest string attractor is minimum. We remark that, in general, this is not the only family of words having a smallest string attractor with size 2, as shown in the following example.

Example 7. A possible set of smallest string attractor for the words $u = a^n b^m$ or $w = b^n a^m$ is $\{n, n + m\}$. Moreover, words of the form $u = (ab)^{n_1}(ba)^{n_2}$ or $v = (ba)^{n_1}(ab)^{n_2}$ have a smallest string attractor of the form $\{2n_1, 2n_1 + 2n_2\}$.

An open question is to characterize all the binary words with a string attractor of size 2. Furthermore, we can remark that in particular for the standard Sturmian words one can always find smallest string attractors whose positions are consecutive. This fact is related to the number of distinct factors appearing in the words. In fact, it is possible to give the following characterization.

Proposition 5. *A word w is a conjugate of a standard Sturmian word if and only if each conjugate of w admits a smallest string attractor containing exactly two consecutive positions.*

The proof uses some results about conjugacy classes of standard Sturmian words given in [2]. The details will be given in the full version of the paper.

5.2 The case of alphabets with cardinality greater than 2

We now analyze the string attractors for words over an alphabet with cardinality greater than 2 that generalize the standard Sturmian words. Numerous generalizations of Sturmian sequences have been introduced for an alphabet with more than 2 letters. Among them, one natural generalization are the episturmian sequences that are defined by using the palindromic closure property of Sturmian sequences (cf. [22]). Here we consider some special prefixes of episturmian sequences that are balanced, called *finite epistandard words* [23]. We recall that, a word w is *balanced* if, for any symbol $a \in \Sigma$, the numbers of a's in any couple of equal length factors of w differ at most by 1. A word is *circularly balanced* if each of its conjugate is balanced. In the case of a binary alphabet such property characterizes standard Sturmian words.

We focus on the circularly balanced epistandard words defined in Theorem 6 (see [23, 28]), that can be built via the *iterated palindromic closure* function. The *iterated palindromic closure* function [9], denoted by Pal, is defined recursively as follows. Set $Pal(\varepsilon) = \varepsilon$ and, for any word w and letter x, define $Pal(wx) = (Pal(w)x)^{(+)}$, where $w^{(+)}$, the *palindromic right-closure* of w, is the (unique) shortest palindrome having w as a prefix (see [18]). Circularly balanced epistandard words (up to letter permutation), like the standard Sturmian words, are *BWT-perfectly clustered words*, i.e. the BWT produces as output a word that has the minimum number $|\Sigma|$ of equal-letter runs (cf. [28, 29, 27]).

Theorem 6. *Any circularly balanced finite epistandard word t belongs to one of the following three families (up to letter permutation):*

(i) $t = pa_2$, *with* $p = Pal(a_1^m a_k a_{k-1} \cdots a_3)$, *where* $k \geq 3$ *and* $m \geq 1$;
(ii) $t = pa_2$, *with* $p = Pal(a_1 a_k a_{k-1} \cdots a_{k-\ell} a_1 a_{k-\ell-1} a_{k-\ell-2} \cdots a_3)$, *where* $0 \leq \ell \leq k-4$ *and* $k \geq 4$;
(iii) $t = Pal(a_1 a_k a_{k-1} \cdots a_2)$, *where* $k \geq 3$.

We observe that the words of the last family of Theorem 6 correspond to the *Fraenkel's sequence*, that are words related to a important conjecture [7].

For each epistandard word, we find a possible string attractor and show that its size is $|\Sigma|$.

Theorem 7. *If w is a circularly balanced epistandard words, then the minimum size of string attractor of w is $|\Sigma|$.*

The proof of the theorem uses similar arguments as in Theorem 5 and the notion of palindromic closure. It will be detailed in the full version of the paper.

Example 8. We consider some circularly balanced epistandard words and their corresponding Lyndon conjugates. A smallest string attractor of

- (type (i)) $w = (aaa\underline{ad}aaaa\underline{c}aaaadaaaa)\underline{b}$, obtained by $p = Pal(a^4dc)$, is $\{4, 5, 10, 20\}$, whereas for its Lyndon conjugate $aaaa\underline{b}aaa\underline{ad}aaaa\underline{c}aaaad$ it is $\{5, 9, 10, 15\}$.
- (type (ii) for $k = 4$ and $\ell = 0$) $u = (a\underline{d}a\underline{c}ada\underline{a}dacada)\underline{b}$, obtained by $p = Pal(adca)$, is $\{2, 4, 8, 15\}$ and for its Lyndon conjugate $a\underline{ad}a\underline{c}ada\underline{b}adacad$ is $\{2, 3, 5, 9\}$.
- (type (ii) for $k = 5$ and $\ell = 0$) $u' = (a\underline{e}a\underline{a}ea\underline{d}aeaaea\underline{c}aeaaeadaeaaea)\underline{b}$, obtained by $p = Pal(aeadc)$, is $\{2, 4, 7, 14, 28\}$ and for its Lyndon conjugate $aaea\underline{b}aea\underline{ae}a\underline{d}aeaaea\underline{c}aeaaeadae$ is $\{5, 9, 10, 12, 19\}$.
- (type (ii) for $k = 5$ and $\ell = 1$) $u'' = (a\underline{e}a\underline{d}aea\underline{a}eadaea\underline{c}aeadaeaaeadaea)\underline{b}$, obtained by $p = Pal(aedac)$, is $\{2, 4, 8, 15, 30\}$ and for its Lyndon conjugate $aaeadaea\underline{b}aeadaea\underline{ae}a\underline{d}aea\underline{c}aeadae$ is $\{9, 17, 18, 20, 24\}$.
- (type (iii)) $v = c\underline{acb}cac$, obtained by $p = Pal(cab)$, is $\{2, 3, 4\}$ and for its Lyndon conjugate $acbcacc$ is $\{3, 5, 6\}$.

Note that, in this example, we have an attractor for each symbol a of the alphabet and the position of such attractor in the epistandard word coincides with the position where the last occurrence of the letter $a \in p$ appears during the palindromic right-closure. Note also that, in the case of Lyndon words, for each letter a, we have an attractor at the position corresponding to the last occurrence of a in the run of the output of the BWT.

However, the authors in [28] show that there exist BWT-perfectly clustered words that do not belong to the families defined in Theorem 6. For instance, the BWT-perfectly clustered word $u = a\underline{b}bbbb\underline{ac}ac$ is not epistandard and $\{2, 7, 8\}$ is a string attractor. Moreover, the BWT-perfectly clustered word $v = aacaabaac$ is epistandard but it is not a balanced and a possible string attractor is $\{2, 3, 6\}$.

Also for alphabets with more than two letters, it remains open the problem of characterizing all words having the smallest string attractor with minimum size.

6 String Attractors in Thue-Morse Words

In this section we consider the problem of finding a smallest string attractor for the family of finite binary Thue-Morse words. Thue-Morse words are a sequence of words obtained by the iterated application of a morphism as described below.

Definition 3. *Let us consider the alphabet $\Sigma = \{a, b\}$ and the morphism $\varphi : \Sigma^* \mapsto \Sigma^*$ such that $\varphi(a) = ab$ and $\varphi(b) = ba$. Let us denote by $t_n = \varphi^n(a)$ the n-th iterate of the morphism φ that is called the n-th Thue-Morse word.*

Note that at each iteration of φ the length of the word is doubled, therefore the n-th Thue Morse word has length 2^n. The n-th Thue-Morse words for $n = 3, 4, 5$ are shown in Figure 1.

By using a result in [3] on the enumeration of factors in Thue-Morse words the following lower bound is proved.

Proposition 6. *Let $t_n = \varphi^n(a)$ be the n-th Thue-Morse word with $n > 2$. Then $\gamma^*(t_n) \geq 3$.*

The following proposition provides the recursive structure of the n-th Thue-Morse words by using the rules of a context-free grammar.

Proposition 7. *The n-th Thue-Morse word is obtained by the following grammar:*

$$\{A_0 \to a, B_0 \to b\} \bigcup_{i=1}^{n-1} \{A_i \to A_{i-1}B_{i-1}, B_i \to B_{i-1}A_{i-1}\} \bigcup \{A_n \to A_{n-1}B_{n-1}\}$$

by taking as axiom the non terminal symbol A_n.

The grammar described in Proposition 7 contains $2n+1$ rules for the Thue Morse word of length 2^n. Therefore, by using a result proved in [11], reported here as Theorem 3, it is possible to construct a string attractor for t_n having size $2n+1$. In the last part of this section we exhibit a string attractor Γ_n for t_n of size n. Our conjecture is that $\gamma^*(t_n) = n$.

Theorem 8. *A string attractor of the n-th Thue Morse word, with $n \geq 3$ is*

$$\Gamma_n = \{2^{n-1}+1\} \bigcup_{i=2}^{n} \{3 \cdot 2^{i-2}\}$$

Lemma 1. *Let x be a factor of $t_n = \varphi^n(a)$, with $n \geq 3$. Then, t_{n+1} admits an occurrence of x crossing a position in the set $\bigcup_{i=2}^{n+1}\{3 \cdot 2^{n+1-i}\}$.*

Proof (Theorem 8). The statement is proved by induction on n. If $n = 3$, it is easy to check (see Fig. 1) that $\Gamma_3 = \{3, 5, 6\}$ is a string attractor for t_3.

Let us suppose that Γ_n is a string attractor for t_n. We show that a string attractor for t_{n+1} can be obtained by applying to Γ_n the following two operations:

- ADD($2^n + 1$), that adds the new position $2^n + 1$
- MOVE($2^{n-1}+1$,$3 \cdot 2^{n-1}$) that replaces the position $2^{n-1}+1$ with $3 \cdot 2^{n-1}$.

Such operations are described, for $n = 3, 4, 5$, in Fig. 1. By using Proposition 7, it is easy to verify that for each $n \geq 2$, $t_n = u_n v_n v_n u_n$, with $|u_n| = |v_n| = 2^{n-2}$. Moreover, for each $n \geq 0$, t_n is prefix of t_{n+1}. Let x be a factor of t_{n+1}. If x is also a factor of t_n, then by Lemma 1 has at least an occurrence crossing a position in the set $\Delta = \bigcup_{i=2}^{n+1}\{3 \cdot 2^{n+1-i}\}$. We can suppose that x is factor of t_{n+1} that does not appear in t_n and that does not cross any position in Δ. It means that x has to be factor of $u_n v_{n+1}$. In particular, such a factor exists. In fact, let a and b be the first and the last character of u_n, then av_nb has only one occurrence in t_{n+1}. If follows from the fact that $t_{n+1} = u_n v_n v_n u_n v_n u_n u_n v_n$ and that the Thue-Morse words has no overlapping factors. Moreover, the occurrence of av_nb crosses the position $2^n + 1$. □

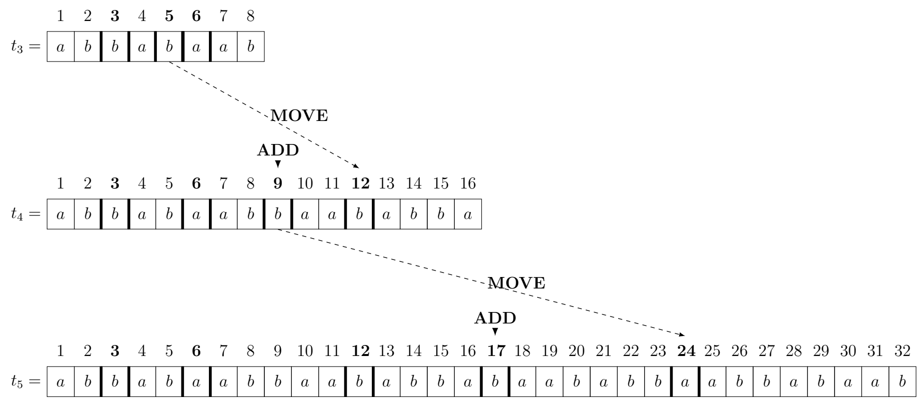

Fig. 1. String attractor Γ_n for the word $t_n = \varphi^n(a)$, with $n = 3, 4, 5$ (the positions in Γ_n are in bold), i.e. $\Gamma_3 = \{3, 5, 6\}$, $\Gamma_4 = \{3, 6, 9, 12\}$, $\Gamma_5 = \{3, 6, 12, 17, 24\}$. Note that in Γ_4 the position 9 is obtained by ADD(9) operation, the position 12 is obtained by the operation MOVE$(5, 3\cdot 4)$. In Γ_5 the position 17 is obtained by ADD(17) operation, the position 24 is obtained by the operation MOVE$(9, 3 \cdot 8)$.

7 Attractors in de Brujin words

A *de Bruijn* sequence (or words) B of order k on an alphabet Σ of size σ, is a circular sequence in which every possible length-k string on Σ occurs exactly once as a substring.

De Brujin words are widely studied in combinatorics on words, and all of them can be constructed by considering all the Eulerian walks on de Brujin graphs. All the de Brujin sequences of order k over an alphabet of size σ have length σ^k. For instance the (circular) word $w = aaaababbbbabaabb$ is a de Brujin word of order 4 over the alphabet $\{a, b\}$. In fact one can verify that all strings of length 4 over $\{a, b\}$ appear as factor of w just once.

Since we are here interested to linear and not to circular words, it is easy to verify that in order to have linear words containing all the k-length factors exactly once, it is sufficient to consider any linearization of the circular de Brujin word of order k (that is, we cut the circular word in any position to get a linear word) and concatenate it with a word equal to its own prefix of length $k - 1$. Therefore its length is $\sigma^k + k - 1$. We call such words *linear de Brujin sequences (or words)*. For instance the linear de Brujin word corresponding to the circular one in the above example is the word $w' = aaaababbbbabaabbaaa$. Remark that the length of w' is $2^k + k - 1$. In [14] the following theorem is proved.

Theorem 9. *The number of phrases $c(n)$ in a LZ parsing of a sequence of length n over an alphabet of size σ satisfies:*

$$c(n) < \frac{n}{(1 - \epsilon_n) \log_\sigma n}$$

where $\epsilon_n = 2\frac{1+\log_\sigma(\log_\sigma(\sigma n))}{\log_\sigma n}$.

When a linear de Brujin word B of order k is considered, by combining Theorem 9, Proposition 1 and the fact that the prefix and the suffix of B of length $k-1$ are equal, we get the following upper and lower bounds for a smallest string attractor of B.

Proposition 8. *Let B be a linear de Brujin sequence of order k and length $n+k-1$ over an alphabet of size σ ($n = \sigma^k$). Then the cardinality γ^* of a smallest string attractor for B satisfies:*

$$\frac{n}{\log_\sigma n} \leq \gamma^* < \frac{n}{(1-\epsilon_n)\log_\sigma n}$$

where $\epsilon_n = 2\frac{1+\log_\sigma(\log_\sigma(\sigma n))}{\log_\sigma n}$.

This means that γ^* for a linear de Brujin word of length n grows asintotically as $\frac{n}{\log n}$, corresponding to the worst case for the size of a smallest string attractor of any word over the constant alphabet Σ. Notice that the lower bound is somehow intuitively expected, since all the words of length k appear only once in B, therefore two consecutive positions in any string attractor cannot be farther than k. For instance one can verify that for $w' = aaaababbbbabaabbaaa$ a smallest string attractor is $\{4, 8, 12, 16\}$.

8 Conclusion and Open Problems

In this paper we have studied the notion of string attractor from a combinatorial point of view. We have given an explicit construction of the string attractor for the infinite families of standard Sturmian words and Thue-Morse words. For standard Sturmian words, by using their combinatorial properties, the construction gives a smallest string attractor whose size is 2. String attractors of minimum size can be also constructed for circularly balanced epistandard words. It is open the question to characterize all the words whose smallest string attractor has size equal to the cardinality of the alphabet. For Thue-Morse words, the size of the attractor is logarithmic with respect to the length of the word. We conjecture that such size is minimum and we leave it as an open problem. Based on the results presented in the paper two research directions could be explored. Some standard Sturmian words and Thue-Morse words are generated by morphisms. It could be interesting to find which properties of the morphism determine a smallest string attractor of constant size. Moreover, we plan to study how the distribution of the positions in the smallest string attractor is related to the combinatorial structure of the words.

Finally, the size of smallest string attractor could be used to define a new function to measure the complexity of infinite words. It could be interesting to investigate how such a measure is related with other known complexity measures, such as the factor complexity [21], or the cyclic complexity [4].

Acknowledgements

S. Mantaci, G. Rosone and M. Sciortino are partially supported by the project MIUR-SIR CMACBioSeq ("Combinatorial methods for analysis and compression of biological sequences") grant n. RBSI146R5L.

References

1. Berstel, J., de Luca, A.: Sturmian words, Lyndon words and trees. Theoret. Comput. Sci. **178**(1), 171 – 203 (1997)
2. Borel, J.P., Reutenauer, C.: On Christoffel classes. RAIRO - Theoretical Informatics and Applications **40**(1), 1527 (2006)
3. Brlek, S.: Enumeration of factors in the Thue-Morse word. Discrete Applied Mathematics **24**(1-3), 83–96 (1989)
4. Cassaigne, J., Fici, G., Sciortino, M., Zamboni, L.Q.: Cyclic complexity of words. J. Comb. Theory, Ser. A **145**, 36–56 (2017)
5. Castiglione, G., Restivo, A., Sciortino, M.: Circular sturmian words and Hopcroft's algorithm. Theoretical Computer Science **410**(43), 4372–4381 (2009)
6. Castiglione, G., Restivo, A., Sciortino, M.: On extremal cases of hopcroft's algorithm. Theoret. Comput. Sci. **411**(38-39), 3414–3422 (2010)
7. Fraenkel, A.S.: Complementing and exactly covering sequences. Journal of Combinatorial Theory, Series A **14**(1), 8 – 20 (1973)
8. Guerrini, V., Rosone, G.: Lightweight Metagenomic Classification via eBWT. In: Algorithms for Computational Biology. vol. 11488 LNBI, pp. 112–124. Springer International Publishing (2019)
9. Justin, J.: Episturmian morphisms and a galois theorem on continued fractions. RAIRO-Theor. Inf. Appl. **39**(1), 207–215 (2005). https://doi.org/10.1051/ita:2005012, https://doi.org/10.1051/ita:2005012
10. Kempa, D., Prezza, N.: At the roots of dictionary compression: String attractors. CoRR **abs/1710.10964** (2017), http://arxiv.org/abs/1710.10964
11. Kempa, D., Prezza, N.: At the roots of dictionary compression: string attractors. In: Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018. pp. 827–840. ACM (2018)
12. Kida, T., Matsumoto, T., Shibata, Y., Takeda, M., Shinohara, A., Arikawa, S.: Collage system: a unifying framework for compressed pattern matching. Theoret. Comput. Sci. **298**(1), 253 – 272 (2003)
13. Knuth, D., Morris, Jr., J., Pratt, V.: Fast pattern matching in strings. SIAM Journal on Computing **6**(2), 323–350 (1977)
14. Lempel, A., Ziv, J.: On the complexity of finite sequences. IEEE Transactions on Information Theory **22**(1), 75–81 (1976). https://doi.org/10.1109/TIT.1976.1055501
15. Lothaire, M.: Applied Combinatorics on Words (Encyclopedia of Mathematics and its Applications). Cambridge University Press, New York, NY, USA (2005)
16. Louza, F.A., Telles, G.P., Gog, S., Zhao, L.: Computing Burrows-Wheeler Similarity Distributions for String Collections. In: SPIRE 2018. Lecture Notes in Computer Science, vol. 11147, pp. 285–296. Springer (2018)
17. de Luca, A., Mignosi, F.: Some combinatorial properties of sturmian words. Theoretical Computer Science **136**(2), 361–385 (1994)
18. de Luca, A.: Combinatorics of standard sturmian words. LNCS, vol. 1261, pp. 249–267. Springer (1997)

19. Mantaci, S., Restivo, A., Rosone, G., Sciortino, M., Versari, L.: Measuring the clustering effect of BWT via RLE. Theor. Comput. Sci. **698**, 79–87 (2017)
20. Mantaci, S., Restivo, A., Sciortino, M.: Burrows-Wheeler transform and Sturmian words. Information Processing Letters **86**, 241–246 (2003)
21. Morse, M., Hedlund, G.A.: Symbolic dynamics. Amer. J. Math. **60**, 1–42 (1938)
22. Paquin, G.: On a generalization of Christoffel words: epichristoffel words. Theoretical Computer Science **410**(38), 3782 – 3791 (2009). https://doi.org/https://doi.org/10.1016/j.tcs.2009.05.014
23. Paquin, G., Vuillon, L.: A characterization of balanced episturmian sequences. The Electronic Journal of Combinatorics **14**(1), Research paper R33, 12 p.–Research paper R33, 12 p. (2007)
24. Policriti, A., Prezza, N.: LZ77 Computation Based on the Run-Length Encoded BWT. Algorithmica **80**(7), 1986–2011 (2018). https://doi.org/10.1007/s00453-017-0327-z
25. Prezza, N.: String attractors. CoRR **abs/1709.05314** (2017), `http://arxiv.org/abs/1709.05314`
26. Prezza, N., Pisanti, N., Sciortino, M., Rosone, G.: SNPs detection by eBWT positional clustering. Algorithms for Molecular Biology **14**(1), 3:1–3:13 (2019)
27. Restivo, A., Rosone, G.: Burrows-Wheeler transform and palindromic richness. Theoretical Computer Science **410**(30-32), 3018 – 3026 (2009)
28. Restivo, A., Rosone, G.: Balancing and clustering of words in the Burrows-Wheeler transform. Theoretical Computer Science **412**(27), 3019 – 3032 (2011)
29. Simpson, J., Puglisi, S.J.: Words with simple Burrows-Wheeler transforms. Electronic Journal of Combinatorics **15** (article R83, 2008)
30. Ziv, J., Lempel, A.: A universal algorithm for sequential data compression. IEEE Trans. Inf. Theor. **23**(3), 337–343 (1977)
31. Ziv, J., Lempel, A.: Compression of individual sequences via variable-length coding. IEEE Trans. Inf. Theor. **24**, 530–536 (1978)