Combinatorial Analysis of the Burrows-Wheeler Transform*

Sabrina Mantaci[1], Antonio Restivo[1], Giovanna Rosone[2], Floriana Russo[1], and Marinella Sciortino[1]

[1]University of Palermo, Palermo, Italy
[2]University of Pisa, Pisa, Italy

Abstract

The Burrows-Wheeler Transform is a well known transformation widely used in Data Compression: important competitive compression software, such as Bzip (cf. [16]) and Szip (cf, [15]) and some indexing software, like the FM-index (cf. [4]), are deeply based on the Burrows Wheeler Transform. The main advantage of using BWT for data compression consists in its feature of "clustering" together equal characters. In this paper we show the existence of fixed points of BWT, i.e., words on which BWT has no effect. We show a characterization of the permutations associated to BWT of fixed points and we give the explicit form of fixed points on a binary ordered alphabet $\{a, b\}$ having at most four b's and those having at most four a's.

1 Introduction

The Burrows-Wheeler Transform (BWT) is a transformation introduced in [1] on which are based several new generation compressors, and is used also in many applications, as in Bioinformatics, Computational Biology and Information Retrieval. The Burrows-Wheeler Transform consists in a permutation of the letters of the input text according to the lexicographic order

*The final publication is available at IOS Press through `https://doi.org/10.3233/FI-2017-1566`. Please cite: Mantaci, S., Restivo, A., Rosone, G., Russo, F., Sciortino, M. 7801635449;7004102790;6507452804;57195345959;6602723375; On fixed points of the burrows-wheeler transform (2017) Fundamenta Informaticae, 154 (1-4), pp. 277-288.

of their contexts. Since similar contexts usually follow equal letters, the permuted string tends to group together equal letters. Therefore the output of BWT is usually better compressible than the original text.

Besides its applications, the Burrows Wheeler transform has many interesting combinatorial properties and for this reason in literature several studies can be found on BWT from the combinatorial point of view. The combinatorial nature of BWT is evidenced by the fact that, by chance, the same year of the publication of the paper by Burrows and Wheeler, in a very different context Gessel and Reutenauer introduced in [5] a bijection between the family of multisets of "necklaces" over an alphabet Σ and the words in Σ^*. In [2] the authors found out that BWT could be seen as a particular case of the transformation introduced in [5]. In the same paper they give a characterization of permutations associated to the BWT over a given alphabet as the permutations having a single cycle and a number of descents smaller than the size of the alphabet. From an algorithmic viewpoint the general case of Gessel and Reutenauer bijection is considered in [9] with several application in data compression and sequences comparison (cf. [10]). Some combinatorial researches have been done in order to study words that are BWT images (cf. [7, 6]). In [11] it is proved that a word over a two letters alphabet has a BWT with the minimal number of clusters, i.e. a word of the form $b^k a^h$ if and only if it is a power of a conjugate of a standard sturmian word (cf. [8]). For non-binary alphabets several authors have considered the set $\mathcal{S}$ of the words v over a totally ordered alphabet $\Sigma = \{a_1, a_2, \ldots, a_\sigma\}$, with $a_1 < a_2 < \ldots < a_\sigma$, for which the string produced by BWT is $a_\sigma^{n_\sigma} a_{\sigma-1}^{n_{\sigma-1}} \cdots a_2^{n_2} a_1^{n_1}$ for some non-negative integers $n_1, n_2, \ldots, n_\sigma$. In the case of three-letters alphabet a constructive characterization of the elements of $\mathcal{S}$ has been given by Simpson and Puglisi in [17]. In [13] the authors show that the elements of $\mathcal{S}$ are "rich" in palindromes, in the sense that they contain the maximum number of different palindromic factors. Finally, in [3] it is proved that perfectly clustering words are intrinsically related to k-discrete interval exchange transformations. In [14], the authors propose an experimental study in order to analyze the clustering effect on real text.

In this stream of combinatorial study of the BWT, a natural question is to ask whether there exist words that are fixed points, i.e. the string transformed by BWT is equal to the input string. The question is then whether it is possible to give a general condition in order w to be a fixed point, and second, whether we can define infinite families of fixed points having a given form.

In this paper we give a combinatorial condition for fixed points and

define for binary alphabet $\{a, b\}$ all the classes of fixed point, when the original word w has at most four occurrences of b or when w has at most four occurrences of a.

2 The Burrows-Wheeler Transform

In this section we describe the construction of the BWT as defined by Burrows and Wheeler in their original paper of 1994. This definition has been, in times, modified in order to better fit efficient data structures for its representation and storage. Nevertheless the original transformation is the one that holds the most important combinatorial properties, that catches the essence and the meaning of this computational tool.

Let $\Sigma = \{c_1, c_2, \ldots, c_m\}$ be a finite ordered alphabet with $c_1 < c_2 < \ldots < c_m$, where $<$ denotes the standard lexicographic order. We denote by Σ^* the set of words over Σ. Given a finite word $w = a_1 a_2 \cdots a_n \in \Sigma^*$ with each $a_i \in \Sigma$, the length of w, denoted $|w|$, is equal to n. Given a finite string $w = a_1 a_2 \cdots a_n$ with each $a_i \in \Sigma$, a *substring* of a string w is a word of the form $a_i \cdots a_j$ with $1 \leq i \leq j \leq n$. The *concatenation* of two words w and v, written wv, is simply the string consisting of the symbols of w followed by the symbols of v. 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^*. The *conjugacy class* $[w]$ of $w \in \Sigma^n$ is the set of all words $a_i a_{i+1} \cdots a_n a_1 \cdots a_{i-1}$, for $1 \leq i \leq n$. A conjugacy class can also be represented as a circular word.

The transformation of the original BWT is described as follows: given a word $w = a_1 a_2 \cdots a_n \in \Sigma^*$, the output of BWT is the pair $(\texttt{bwt}(w), I)$, where: $\texttt{bwt}(w)$ is the permutation of symbols in the input string w obtained by lexicographically sorting the list of the conjugates of w, and considering the concatenation of the last symbols of the conjugates in the sorted list; I is the row where the original text appears in the sorted list of conjugates.

The construction of the output of BWT consists of the following steps:

1. Consider all conjugates of the input text.
2. Sort the conjugates in lexicographic order in a matrix M.
3. Denote by L the last column of M.
4. Set $\texttt{bwt}(w) = L$ and I as the position of w in M.

Remark that the first column of M, denoted by F, contains the letters of L, alphabetically sorted.

		F										L
		$\downarrow$										$\downarrow$
	1	a	t	h	e	m	a	t	i	c	s	m
	2	a	t	i	c	s	m	a	t	h	e	m
	3	c	s	m	a	t	h	e	m	a	t	i
	4	e	m	a	t	i	c	s	m	a	t	h
	5	h	e	m	a	t	i	c	s	m	a	t
	6	i	c	s	m	a	t	h	e	m	a	t
$I \rightarrow$	7	m	a	t	h	e	m	a	t	i	c	s
	8	m	a	t	i	c	s	m	a	t	h	e
	9	s	m	a	t	h	e	m	a	t	i	c
	10	t	h	e	m	a	t	i	c	s	m	a
	11	t	i	c	s	m	a	t	h	e	m	a

Figure 1: The matrix M of $w = mathematics$.

Example 2.1. The construction of BWT for the input string $w = mathematics$ is illustrated in Fig. 1. The output is $(mmihttsecaa, 7)$.

The matrix M defines a permutation π_w (or simply π when no confusion arises) of $1, 2, \ldots, n$: $\pi_w(i) = j$ if and only if the conjugate $a_i \cdots a_n a_1 \cdots a_{i-1}$ appears at row j of M. In other terms, $\pi_w(i)$ is the rank in the lexicographic order of the i-th circular shift of the word w. We call π_w the `bwt`*-permutation of* w.

Example 2.2. Let $w = mathematics$. The the `bwt`-permutation of w is:

$$\pi_w = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\ 7 & 1 & 10 & 5 & 4 & 8 & 2 & 11 & 6 & 3 & 9 \end{pmatrix}.$$

One of the fundamental properties of the BWT is its *reversibility*, that is, if we are given the pair $(\texttt{bwt}(w), I)$ we are able to recover the original word w. This feature makes BWT a useful tool in many applications where it is necessary to partially or entirely recover the original word, such as data compression or pattern matching. This fact is a direct consequence of the following properties. The reader can easily verify the properties in Figure 1.

Proposition 2.3. *Let w be a string and let (L, I) be the output of* BWT. *Let F be the sequence of the sorted letters of L. The following properties hold:*

1. *For all $i = 1, \ldots, n$ and $i \neq I$, the letter $L[i]$ precedes $F[i]$ in the original word;*
2. *for each letter z, the i-th occurrence of z in L corresponds to the i-th occurrence of z in F.*

Actually, according to Property 2 of Proposition 2.3, we can always define the permutation, $\tau_w \colon \{1, \ldots, n\} \to \{1, \ldots, n\}$ where τ_w gives the correspondence between the positions of letters of the first and the last column of the matrix M. In literature this permutation is also known as *FL-mapping*.

We remark that τ_w can be obtained from π_w when the second line of π_w is read as a cycle (cf. [2]). The permutation τ_w represents also the order in which we have to rearrange the elements of F to reconstruct the original word w. Hence, starting from the position I, we can recover the word $w = a_1 a_2 \cdots a_n$ as follows:

$$a_i = F[\tau_w^{i-1}(I)]\ , \text{ where } \tau_w^0(x) = x, \text{ and } \tau_w^i(x) = \tau_w(\tau_w^{i-1}(x)), \text{with } 1 \leq i \leq n.$$

This means that permutation $\tau_w(w)$, gives the sequence of the letters we have to pick in F, starting from I, in order to recover the original word w.

Example 2.4. Consider the BWT matrix of the word $w = mathematics$ as in Figure 2.1. One can see that, according to Property 2 of Proposition 2.3, the first a in F corresponds to the a in position 10 of L, the a in position 2 of F corresponds to the a in position 11 of L and so on. The permutation that associates letters in the first and in the last columns of the matrix is the following

$$\tau_w = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\ 10 & 11 & 9 & 8 & 4 & 3 & 1 & 2 & 7 & 5 & 6 \end{pmatrix}.$$

The reader can verify that when the lower line of permutation π_w is read as a cycle, we get the τ_w permutation. Starting from position $I = 7$, if we take the letters in F according to the positions indicated by the cyclic permutation above, we get the original word $w = mathematics$.

3 Fixed Points

In this section we introduce the notion of fixed point of the BWT.

Definition 3.1. A word $w \in \Sigma^*$ is a *fixed point* (with respect to the BWT) if $\texttt{bwt}(w) = w$.

Example 3.2. Let $v = babaabaaaa$. The reader can verify that $\texttt{bwt}(v) = v$, then v is a fixed point.

We now give a general condition that a word w has to satisfy in order to be a fixed point. Such a condition is expressed in terms of permutations associated to a word w.

In the previous section we have associated to a word w its `bwt`-permutation π_w. Starting from π_w we have defined a new permutation τ_w obtained by interpreting the second line of π_w as a cycle.

We now associate to a word w another permutation. Given any word $w = a_1 a_2 \cdots a_n$ where $a_i \in \Sigma$, we define the *standard permutation* σ_w of w as follows: for $i, j \in \{1, 2, \ldots, n\}$ the condition $\sigma_w(i) < \sigma_w(j)$ if and only if either $a_i < a_j$ or $a_i = a_j$ and $i < j$. The permutation σ_w may be obtained by numbering from left to right the letters of w, starting from the smallest letter, then the second smallest, and so on.

Example 3.3. If $w = mathematics$ then

$$\sigma_w = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\ 7 & 1 & 10 & 5 & 4 & 8 & 2 & 11 & 6 & 3 & 9 \end{pmatrix}.$$

Consider instead $\mathtt{bwt}(w) = mmihttsecaa$ then

$$\sigma_{\mathtt{bwt}(w)} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\ 7 & 8 & 6 & 5 & 10 & 11 & 9 & 4 & 3 & 1 & 2 \end{pmatrix}.$$

From Example 3.3 and Example 2.4 one can realize that given a word w, $\sigma_{\mathtt{bwt}(w)} = \tau_w^{-1}$. This is formalized in [2, 12]. From the previous considerations and from the definition of fixed point we derive the following characterization:

Theorem 3.4. *Let $w \in \Sigma^*$. Then w is a fixed point if and only if $\sigma_w = \tau_w^{-1}$.*

Example 3.5. We have already remarked that $v = babaabaaaa$ is a fixed point. The `bwt` permutation is

$$\pi_v = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 10 & 6 & 9 & 3 & 5 & 8 & 1 & 2 & 4 & 7 \end{pmatrix},$$

and the permutation τ_v is obtained as follows:

$$\tau_v = \begin{pmatrix} 10 & 6 & 9 & 3 & 5 & 8 & 1 & 2 & 4 & 7 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 2 & 4 & 5 & 7 & 8 & 9 & 10 & 1 & 3 & 6 \end{pmatrix}.$$

Moreover

$$\sigma_v = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 8 & 1 & 9 & 2 & 3 & 10 & 4 & 5 & 6 & 7 \end{pmatrix}.$$

The reader can verify that σ_v is the inverse of the permutation τ_v given above, then v is a fixed point. This property is not true for σ_w and τ_w when $w = mathematics$, then w is not a fixed point.

4 Fixed points over a binary alphabet

Although Theorem 3.4 provides a complete characterization, however it is not easy to derive from it the explicit form of the words that are fixed points of the BWT. In the following we consider only words over a binary alphabet $\{a, b\}$, with $a < b$ and we give the explicit form of the fixed points belonging to some special classes of words.

First of all remark that the image of the `bwt` never begins with the smallest letter. This is because otherwise the τ permutation would not be a single cycle, since $\tau(1) = 1$ would be a cycle by itself. For the same reason it never ends with the greatest letter. This suggests that, when we look for fixed points over a two letters alphabet, we will never consider words starting with the letter a and ending with the letter b. Then the general form of a possible fixed point is $w = b^{h_1}a^{k_1}b^{h_2}a^{k_2}\cdots b^{h_l}a^{k_l}$ with $h_1, k_l > 0$, $h_i \geq 0$ if $2 \leq i \leq l$ and $k_j \geq 0$ if $1 \leq j \leq l-1$. We consider the following:

Question: What are the possible exponents $h_1, k_1, h_2, k_2, \ldots, h_l, k_l$ such that w is a fixed point?

We state now several propositions giving an explicit answer to the above question when we have from one to four b's or from one to four a's. Since technique for the different proofs is the same, we give just in details one of them, namely the case of fixed points with three b's, and we give a sketch of the proof for the fixed points with three a's.

Proposition 4.1. *Let $w \in \{a, b\}^*$*

1. *if $|w|_b = 1$, w is a fixed point if and only if $w = ba^k$, $\forall k \in \mathbb{N}$.*
2. *If $|w|_a = 1$, w is a fixed point if and only if $w = b^k a$, $\forall k \in \mathbb{N}$.*
3. *If $|w|_b = 2$, w is a fixed point if and only if $w = ba^k ba^{2k+1}$, $\forall k \in \mathbb{N}$.*
4. *If $|w|_a = 2$, w is a fixed point if and only if $w = b^{2k+1}ab^k a$, $\forall k \in \mathbb{N}$.*

We are now going to consider fixed points with three occurrences of b or with three occurrences of a, i.e. the words of the forms $ba^k ba^h ba^l$ or $b^k ab^h ab^l a$. They are characterized by the triplets of their exponents, (k, h, l). The following proposition characterizes fixed points with three b's. We give its proof in details in order to show the technique used also for other cases:

Proposition 4.2. *The word $w = ba^k ba^h ba^l$ is a fixed point if and only if it is defined by the triplets:*

1. $(k, 2k, 3k+1) \quad \forall k \in \mathbb{N}$ *if* $k \leq h < l$;

2. $(k, 4k+2, 3k+2) \quad \forall k \in \mathbb{N}$ *if* $k < l \leq h$;

3. $(3p+1, 2p, 6p+2) \quad \forall p \in \mathbb{N}$ *if* $h < k < l$.

Proof. The proof depends on the relative order of the exponents (k, h, l), therefore there are $3! = 6$ different cases to consider:
Case 1. Let $k \leq h < l$, then we can consider the following factorization of w:

$$b\underbrace{a\cdots a}_{k} b\underbrace{a\cdots a}_{h-k}\underbrace{a\cdots a}_{k} b\underbrace{a\cdots a}_{l-h}\underbrace{a\cdots a}_{h-k}\underbrace{a\cdots a}_{k}.$$

For computing $\mathtt{bwt}(w)$, we notice that the first $l-h$ conjugates are the ones starting at positions from $h+k+4$ to $k+l+3$, the first one preceded by a b and all the other ones preceded by an a. So the first part of $\mathtt{bwt}(w)$ is made by letters ba^{l-h-1}. The following conjugate is the one starting at position $k+3$ that is preceded by a b and then we alternate conjugates of the second and the third block of a's, all preceded by a until there are k elements left in each block. Then the following letters of $\mathtt{bwt}(w)$ are $ba^{2h-2k-1}$. The next conjugate is the one starting at position $h+3$ that is preceded by a a, then the one starting at position 2, preceded by a b, then the one in position $h+l+3$ preceded by a a. Then we get the factor aba. Since all the b's are already included, all the remaining letters are a's, then we have to add a^{3k}. Finally $\mathtt{bwt}(w) = ba^{l-h-1}ba^{2h-2k-1}aba^{3k+1}$.

Since we want w to be a fixed point we have to impose that $w = \mathtt{bwt}(w)$ that is

$$ba^{l-h-1}ba^{2h-2k-1}aba^{3k+1} = ba^k ba^h ba^l$$

and this is true if and only if

$$\begin{cases} k = l-h-1 \\ h = 2h-2k \\ l = 3k+1 \end{cases} \quad \Rightarrow \quad \begin{cases} k = k \\ h = 2k \\ l = 3k+1 \end{cases}$$

Case 2. Let $k < l \leq h$. Then the factorization of w is the following:

$$b\underbrace{a\cdots a}_{k} b\underbrace{a\cdots a}_{h-l}\underbrace{a\cdots a}_{l-k}\underbrace{a\cdots a}_{k} b\underbrace{a\cdots a}_{l-k}\underbrace{a\cdots a}_{k}.$$

Let us compute the $\mathtt{bwt}(w)$. The first $h-l$ conjugates are taken from the second block of a's and produce the first part of $\mathtt{bwt}(w)$ equal to ba^{h-l-1}. Then we have the conjugate starting at positions $k+h-l+3$, $h+l+4$,

from which we get the second part of $\texttt{bwt}(w)$ equal to ab. Then we alternate conjugates from the second and the third block, giving the third part of the $\texttt{bwt}(w)$ equal to $a^{2(l-k-1)}$. The next conjugate starts in position 2 is preceded by a b. Then we have alternatively conjugates from the second, the third and the first block, and we get a^{3k-1} and finally the three conjugates starting with b are preceded by a a. In all we have $\texttt{bwt}(w) = ba^{h-l-1}aba^{2(l-k-1)}ba^{3k-1}a^3 = ba^{h-l}ba^{2l-2k-2}ba^{3k+2}$, and we have a fixed point if and only if:

$$\begin{cases} k = h - l \\ h = 2l - 2k - 2 \\ l = 3k + 2 \end{cases} \quad \Rightarrow \quad \begin{cases} k = k \\ h = 4k + 2 \\ l = 3k + 2 \end{cases}$$

Case 3. Let $h < k < l$. Consider the following factorization of w

$$b\underbrace{a\cdots a}_{k-h}\underbrace{a\cdots a}_{h}b\underbrace{a\cdots a}_{h}b\underbrace{a\cdots a}_{l-k}\underbrace{a\cdots a}_{k-h}\underbrace{a\cdots a}_{h}.$$

This time the longest block of a's is the third, so the first $l - k$ conjugates are from position $h + k + 4$ to position $l + h + 3$, the first one preceded by a b and the other ones preceded by a's. Then we get the first part of $\texttt{bwt}(w)$ equal to ba^{l-k-1}. The next conjugates are respectively the ones starting at positions $l+h+4$ and the one starting at position 2, then we add to $\texttt{bwt}(w)$ the string ab. Then we alternate conjugates from the third and the first block, all preceded by a's. Then we add to $\texttt{bwt}$ the factor $a^{2k-2h-2}$. The next factor is the one starting at position $k + 3$, preceded by a b and the remaining letters are all a's. In all we have $\texttt{bwt}(w) = ba^{l-k}ba^{2k-2h-2}ba^{3h+2}$, that is a fixed point if and only if

$$\begin{cases} k = l - k \\ h = 2k - 2h - 2 \\ l = 3h + 2 \end{cases} \quad \Rightarrow \quad \begin{cases} k = k \\ h = \dfrac{2(k-1)}{3} \\ l = 2k \end{cases}$$

that has an integer solution if and only if $(k, h, l) = (3p + 1, 2p, 6p + 2)$.
Case 4. Let $l \leq k \leq h$. Consider the following factorization of w:

$$b\underbrace{a\cdots a}_{k-l}\underbrace{a\cdots a}_{l}b\underbrace{a\cdots a}_{h-k}\underbrace{a\cdots a}_{k-l}\underbrace{a\cdots a}_{l}b\underbrace{a\cdots a}_{l}.$$

The first $h - k$ conjugates are the ones from position $k + 3$ to position $h + 2$, yielding the first $h - k$ letters of $\texttt{bwt}(w)$ equal to ba^{h-k-1}. The following conjugates start in position 2 and in position $h + 3$ producing

the factor ba. Then we alternate conjugates from the first and the third block, getting $a^{2(k-l-1)}$. Then there are conjugates starting respectively in positions $k-l+2$, $k+h+4$ and $k+h-l+3$, yielding aba. Finally we have all the remaining a^{3l}. In all we have: $\texttt{bwt}(w) = ba^{h-k-1}ba^{2k-2l}ba^{3l+1}$ then in order w to be a fixed point we have

$$\begin{cases} k = h-k-1 \\ h = 2k-2l \\ l = 3l+1 \end{cases} \quad \Rightarrow \quad \begin{cases} k = k \\ h = 2k+1 \\ l = -\dfrac{1}{2} \end{cases}$$

The system has no solution in $\mathbb{N}$.
Case 5. Let $h \leq l \leq k$.

$$b\underbrace{a\cdots a}_{k-l}\underbrace{a\cdots a}_{l-h}\underbrace{a\cdots a}_{h}b\underbrace{a\cdots a}_{h}b\underbrace{a\cdots a}_{l-h}\underbrace{a\cdots a}_{h}.$$

Then the first $k-l$ conjugates are taken from the first block of a's, yielding ba^{k-l-1}. Next conjugates are the ones in positions $k+h+4$ and $k-l+2$ producing the factor ba. Then we have alternatively conjugates from the third and the first blocks of a's, giving $a^{2(l-h-1)}$. Then we have conjugates in positions $k+l+4$, $k+3$ and $k-h+2$ giving the factor aba. Finally we have all the remaining a^{3h}. Then $\texttt{bwt}(w) = ba^{k-l-1}ba^{2l-2h}ba^{3h+1}$ and $\texttt{bwt}(w) = w$ if and only if

$$\begin{cases} k = k-l-1 \\ h = 2l-2h \\ l = 3h+1 \end{cases} \quad \Rightarrow \quad \begin{cases} k = k \\ h = -\dfrac{2}{3} \\ l = -1 \end{cases}$$

This system has no solutions in $\mathbb{N}$.
Case 6. Let $l \leq h \leq k$.

$$b\underbrace{a\cdots a}_{k-h}\underbrace{a\cdots a}_{h-l}\underbrace{a\cdots a}_{l}b\underbrace{a\cdots a}_{h-l}\underbrace{a\cdots a}_{l}b\underbrace{a\cdots a}_{l}.$$

The first $k-h$ conjugates are from position 2 to position $k-h+1$ giving the beginning of $\texttt{bwt}(w)$ equal to ba^{k-h-1}. Then we have the conjugates starting at positions $k-h+2$ and $k+3$ yielding the factor ab. Then we alternate conjugates in the first and the second block of a's giving $a^{2(h-l-1)}$. Then we have conjugates in positions $k+h+4$, $k-l+2$ and $k+h-l+3$ producing baa. Finally we have all the remaining a's, namely a^{3l}. Then

$\mathtt{bwt}(w) = ba^{k-h}ba^{2h-2l-2}ba^{3l+2}$, and w is a fixed point if:

$$\begin{cases} k = k - h \\ h = 2h - 2l - 2 \\ l = 3l + 2 \end{cases} \quad \Rightarrow \quad \begin{cases} k = k \\ h = 0 \\ l = -1 \end{cases}$$

This system has no solution in $\mathbb{N}$. □

The following proposition characterize the case of three a's. We give just a sketch of the proof.

Proposition 4.3. *The word $w = b^k ab^h ab^l a$ is a fixed point if and only if it is defined by the triplets:*

1. $(3l + 1, 4l + 2, l) \quad \forall l \in \mathbb{N}$ *if* $l < k < h$;
2. $(3l + 2, 2l + 2, l) \quad \forall l \in \mathbb{N}$ *if* $l < h \leq k$;
3. $(6p + 1, 2p, 3p) \quad \forall p \in \mathbb{N}$ *if* $h \leq l < k$.

Proof. We give a detailed proof only for the case $l < k < h$. Then w can be factorized as follows:

$$\underbrace{b\cdots b}_{k-l}\underbrace{b\cdots b}_{l}a\underbrace{b\cdots b}_{h-k}\underbrace{b\cdots b}_{k-l}\underbrace{b\cdots b}_{l}a\underbrace{b\cdots b}_{l}a.$$

Let us compute $\mathtt{bwt}(w)$. The first three conjugates are the ones beginning with a, all preceded by a b, then we get b^3. Then we alternate l times conjugates from the second, third and first blocks of b's, respectively, from right to left. The first $l - 1$ iterations produce $b^{3(l-1)}$. In the l-th iteration we have, in order, the conjugate starting at position $h + k - l + 2$ preceded by a b, the conjugate starting at position $k + h + 3$, preceded by a a and the one starting at position $k - l + 1$, preceded by a b, yielding the new factor bab. Then we keep on alternating $k-l-1$ times from right to left conjugates from the second and the first block, all preceded by a b, giving the factor $b^{2(k-l-1)}$. Then we have the conjugate starting at position $h + 2$, preceded by a b and the one in position 1, circularly preceded by a . We get the factor ba. Finally there are the remaining conjugates from the second block of b's, all preceded by a b, except the last one, starting at position $k + 2$, preceded by a a. All together $\mathtt{bwt}(w) = b^{3l+1}ab^{2k-2l}ab^{h-k-1}a$ and $\mathtt{bwt}(w) = w$ if and only if:

$$\begin{cases} k = 3l + 1 \\ h = 2k - 2l \\ l = h - k - 1 \end{cases} \quad \Rightarrow \quad \begin{cases} k = 3l + 1 \\ h = 4l + 2 \\ l = l \end{cases}$$

The proofs of all the other cases are similar. □

In order to characterize fixed points with four occurrences of a and four occurrences of b we consider the words $ba^kba^hba^lba^m$ and $b^kab^hab^lab^ma$. In the following proposition we show that there exist eleven 4-tuple (k, h, l, m) that give fixed points with four occurrences of b and twelve with four occurrences of a. The proof uses techniques analogous to the one of previous theorem.

Proposition 4.4. *$w = ba^kba^hba^lba^m$ is a fixed point if and only if one of the following cases holds:*

1. $(k, 2k, 3k, 4k + 1) \quad \forall k \in \mathbb{N}$ *if* $k \leq h \leq l < m$;
2. $(3p, 8p, 15p + 1, 12p + 1) \quad \forall p \in \mathbb{N}$ *if* $k \leq h < m \leq l$;
3. $(k, 10k + 6, 9k + 6, 4k + 3) \quad \forall k \in \mathbb{N}$ *if* $k < m < l \leq h$;
4. $(2p, 6p + 1, 3p, 8p + 2) \quad \forall p \in \mathbb{N}$ *if* $k \leq l < h < m$;
5. $(k, 8k + 3, 9k + 4, 4k + 2) \quad \forall k \in \mathbb{N}$ *if* $k < m < h < l$;
6. $(2p + 1, 10p + 8, 3p + 2, 8p + 7) \quad \forall p \in \mathbb{N}$ *if* $k < l < m < h$;
7. $(7p + 6, 4p + 3, 9p + 7, 16p + 15) \quad \forall p \in \mathbb{N}$ *if* $h < k < l < m$;
8. $(7p + 3, 2p, 15p + 6, 8p + 3) \quad \forall p \in \mathbb{N}$ *if* $h < k \leq m < l$;
9. $(4p + 3, 8p + 8, 3p + 2, 12p + 11) \quad \forall p \in \mathbb{N}$ *if* $l < k < h < m$;
10. $(4p, 16p + 3, 3p, 12p + 2) \quad \forall p \in \mathbb{N}$ *if* $l \leq k < m < h$;
11. $(9p + 4, 2p, 18p + 8, 8p + 3) \quad \forall p \in \mathbb{N}$ *if* $h < m < k < l$.

Proposition 4.5. *$w = b^kab^hab^lab^ma$ is a fixed point if and only if one of the following cases holds:*

1. $(8p + 7, 3p + 2, 6p + 6, 2p + 1) \quad \forall p \in \mathbb{N}$ *if* $m < h < l < k$;
2. $(4m + 3, 3m + 3, 2m + 2, m) \quad \forall m \in \mathbb{N}$ *if* $m \leq l < h < k$;
3. $(4m + 1, 9m + 4, 8m + 4, m) \quad \forall m \in \mathbb{N}$ *if* $m < k < l \leq h$;
4. $(12p + 10, 15p + 13, 8p + 7, 3p + 2) \quad \forall p \in \mathbb{N}$ *if* $m < l < k < h$;
5. $(4m + 1, 9m + 3, 10m + 4, m) \quad \forall m \in \mathbb{N}$ *if* $m < k < h < l$;
6. $(8p + 2, 3p, 10p + 3, 2p) \quad \forall p \in \mathbb{N}$ *if* $m < h < k < l$;

7. $(12p+2, 3p, 4p, 6p) \quad \forall p \in \mathbb{N}$ *if* $h \leq l \leq m < k$;

8. $(12p+2, 3p, 8p+1, 4p) \quad \forall p \in \mathbb{N}$ *if* $h < m < l < k$;

9. $(12p+11, 3p+2, 16p+14, 4p+3) \quad \forall p \in \mathbb{N}$ *if* $h < m < k < l$;

10. $(16p+1, 9p+1, 4p, 7p) \quad \forall p \in \mathbb{N}$ *if* $l \leq m < h < k$.

11. $(8p+1, 15p+3, 2p, 7p+1) \quad \forall p \in \mathbb{N}$ *if* $l < m \leq k < h$;

12. $(8p+6, 18p+15, 2p+1, 9p+7) \quad \forall p \in \mathbb{N}$ *if* $l < k < m < h$.

As one can see, the number of cases to consider grows substantially as the number of b's and the number of a's grows. The case with five b's (a's, respectively) would need to verify $5! = 120$ different possible sorting of the exponents of the blocks of consecutive a's (b's, respectively). We omit the proofs due to space constraints.

5 Conclusions

In this paper we have discussed fixed points of the BWT. Besides its theoretical interest, the existence of an infinite family of fixed points highlights that there are many cases where the use of BWT is totally useless for data compression aims.

The method that we used for finding the explicit form of fixed points with a limited number of a's or b's can hardly be applied for fixed points with a great number of a's and b's, since it would need to consider $k!$ different cases, where k is the minimum between the number of a's and b's.

We leave as an open problem to give an explicit characterization for fixed points in the general case, possibly using the characterization of the permutations τ_w that define fixed points.

Acknowledgements

Thanks to Published source. Partially supported by the project MIUR-SIR CMACBioSeq ("Combinatorial methods for analysis and compression of biological sequences") grant n. RBSI146R5L and by the Gruppo Nazionale per il Calcolo Scientifico (GNCS-INDAM).

References

[1] M. Burrows and D. J. Wheeler. A block sorting data compression algorithm. Technical report, DIGITAL System Research Center, 1994.

[2] M. Crochemore, J. Désarménien, and D. Perrin. A note on the Burrows-Wheeler transformation. *Theoret. Comput. Sci.*, 332:567–572, 2005.

[3] S. Ferenczi and L. Q. Zamboni. Clustering Words and Interval Exchanges. *Journal of Integer Sequences*, 16(2):Article 13.2.1, 2013.

[4] P. Ferragina and G. Manzini. Indexing compressed text. *J. ACM*, 52:552–581, 2005.

[5] I. M. Gessel and C. Reutenauer. Counting permutations with given cycle structure and descent set. *J. Combin. Theory Ser. A*, 64(2):189–215, 1993.

[6] P. M. Higgins. Burrows-Wheeler transformations and de Bruijn words. *Theor. Comput. Sci.*, 457:128–136, 2012.

[7] K. M. Likhomanov and A. M. Shur. Two Combinatorial Criteria for BWT Images. volume 6651 of *LNCS*, pages 385–396. Springer, 2011.

[8] M. Lothaire. *Algebraic Combinatorics on Words*. Cambridge University Press, 2002.

[9] S. Mantaci, A. Restivo, G. Rosone, and M. Sciortino. An extension of the Burrows-Wheeler Transform. *Theoret. Comput. Sci.*, 387(3):298–312, 2007.

[10] S. Mantaci, A. Restivo, G. Rosone, and M. Sciortino. A new combinatorial approach to sequence comparison. *Theory Comput. Syst.*, 42(3):411–429, 2008.

[11] S. Mantaci, A. Restivo, and M. Sciortino. Burrows-Wheeler transform and Sturmian words. *Information Processing Letters*, 86:241–246, 2003.

[12] D. Perrin and A. Restivo. Words. In *Handbook of Enumerative Combinatorics*. CRC Press, 2015.

[13] A. Restivo and G. Rosone. Burrows-Wheeler transform and palindromic richness. *Theoret. Comput. Sci.*, 410(30-32):3018 – 3026, 2009.

[14] A. Restivo and G. Rosone. Balancing and clustering of words in the Burrows-Wheeler transform. *Theoret. Comput. Sci.*, 412(27):3019 – 3032, 2011.

[15] Michael Schindler. A fast block-sorting algorithm for lossless data compression. *Data Compression Conference*, 0:469, 1997.

[16] Julian Seward. The BZIP2 home page. `http://www.bzip.org`.

[17] J. Simpson and S. J. Puglisi. Words with simple Burrows-Wheeler transforms. *Electronic Journal of Combinatorics*, 15, article R83, 2008.