



**HAL**  
open science

## Clustering and Arnoux-Rauzy words

Sébastien Ferenczi, Luca Q Zamboni

► **To cite this version:**

Sébastien Ferenczi, Luca Q Zamboni. Clustering and Arnoux-Rauzy words. *Advances in Applied Mathematics*, 2023, 153, pp.102621. 10.1016/j.aam.2023.102621 . hal-04276825

**HAL Id: hal-04276825**

**<https://hal.science/hal-04276825v1>**

Submitted on 10 Nov 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

1 **CLUSTERING AND ARNOUX-RAUZY WORDS**

2 SÉBASTIEN FERENCZI AND LUCA Q. ZAMBONI

ABSTRACT. We characterize the clustering of a word under the Burrows-Wheeler transform in terms of the resolution of a bounded number of bispecial factors belonging to the language generated by all its powers. We use this criterion to compute, in every given Arnoux-Rauzy language on three letters, an explicit bound  $K$  such that each word of length at least  $K$  is not clustering; this bound is sharp for a set of Arnoux-Rauzy languages including the Tribonacci one. In the other direction, we characterize all standard Arnoux-Rauzy clustering words, and all perfectly clustering Arnoux-Rauzy words. We extend some results to episturmian languages, characterizing those which produce infinitely many clustering words, and to larger alphabets.

3 In [14], the authors give a characterization of the *clustering* phenomenon for the *Burrows-*  
4 *Wheeler transform*, using a class of dynamical systems, the *interval exchange* transformations.  
5 This gives a way to build examples of clustering words, but is not very operative in deciding  
6 whether a given word clusters. Here, inspired by [11] but independently and with purely combi-  
7 natorial methods, we give, in Theorem 1, a characterization of the clustering of a primitive word  
8  $w$  in terms of the resolution of a finite number of bispecial words of the language generated by all  
9 the  $w^n$ ,  $n > 0$ .

10 A very popular family of languages consists in the *Sturmian* languages; these are known since  
11 [17] to be good producers of clustering words, though not all of their factors are clustering. As  
12 is proved in [14], their natural generalizations, the interval exchange languages, can also produce  
13 infinitely many clustering words. In both cases, our Theorem 1 gives a new criterion to identify  
14 those factors which cluster. Another well-known generalization of the Sturmian languages consists  
15 in the *Arnoux-Rauzy languages* on three letters, and the question of clustering of their factors was  
16 asked by Francesco Dolce at the Journées Montoises 2022: in contrast with the previous cases, we  
17 are able to answer it by a broad negative, the Arnoux-Rauzy words are in general not clustering.

18 More precisely, for every given Arnoux-Rauzy language, we compute, in Theorem 10, an ex-  
19 plicit bound  $K$  such that each word of length at least  $K$  is not clustering; by Corollary 16 this  
20 bound is indeed sharp for a family of Arnoux-Rauzy languages including the Tribonacci language,  
21 but we can build counter-examples for which this bound is not optimal. In the other direction, in  
22 each Arnoux-Rauzy language we want to find primitive clustering words  $v$ : it turns out that this  
23 is easier for *standard* Arnoux-Rauzy words, as we prove in Proposition 7 that an Arnoux-Rauzy  
24 word  $v$  is cyclically conjugate to a standard one if and only if  $vv$  is an Arnoux-Rauzy word. in  
25 Proposition 12, and Corollary 13, we are able to characterize those which are clustering. As a  
26 consequence, there exist arbitrarily long primitive perfectly clustering Arnoux-Rauzy words, and  
27 every Arnoux-Rauzy language contains a primitive perfectly clustering word of length at least 22.  
28 As for clustering Arnoux-Rauzy words not conjugate to standard ones, in Propositions 17 and 18

---

*Date:* September 6, 2023.

*2010 Mathematics Subject Classification.* Primary 68R15.

This research was supported by project IZES, ANR 22-CE40-0011.

29 we characterize, using methods and results of [19], those which cluster perfectly, but there are also  
30 infinitely many of them which cluster but not perfectly.

31 Finally, we turn to generalizations of Arnoux-Rauzy languages: these include Arnoux-Rauzy  
32 languages on more than three letters, for which we give a (non-optimal) bound on the possi-  
33 ble length of a clustering word, and *episturmian* languages, which include Sturmian languages,  
34 Arnoux-Rauzy languages, some periodic languages, and some intermediate cases which behave  
35 essentially like Sturmian words. Among episturmian languages on three letters we give in Theo-  
36 rem 22 a full characterization of those which produce only finitely many clustering words: rather  
37 unexpectedly, these include not only Arnoux-Rauzy languages, but also some (not all) of the peri-  
38 odic and intermediate cases.

39

## 1. USUAL DEFINITIONS

40 Let  $\mathcal{A}$  be a finite set called the *alphabet*, its elements being *letters*. A *word*  $w$  of length  $n = |w|$   
41 is  $a_1 a_2 \cdots a_n$ , with  $a_i \in \mathcal{A}$ . The *concatenation* of two words  $w$  and  $w'$  is denoted by  $ww'$ .

42 A word is *primitive* if it is not a power of another word.

43 The *reverse* of a word  $w = w_1 \dots w_n$  is the word  $\bar{w} = w_n \dots w_1$ .

44

45 By a language  $\Lambda$  over  $\mathcal{A}$  we mean a *factorial extendable language*: a collection of sets  $(\Lambda_n)_{n \geq 0}$   
46 where the only element of  $\Lambda_0$  is the *empty word*, and where each  $\Lambda_n$  for  $n \geq 1$  consists of words of  
47 length  $n$ , such that for each  $v \in \Lambda_n$  there exists  $a, b \in \mathcal{A}$  with  $av, vb \in \Lambda_{n+1}$ , and each  $v \in \Lambda_{n+1}$   
48 can be written in the form  $v = au = u'b$  with  $a, b \in \mathcal{A}$  and  $u, u' \in \Lambda_n$ .

49 A word  $v = v_1 \dots v_r$  *occurs* at index  $i$  in a word  $w = w_1 \dots w_s$  if  $v_1 = w_i, \dots, v_r = w_{i+r-1}$ , we say also  
50 that  $w$  contains  $v$  and  $v$  is a *factor* of  $w$ .

51

52 The *complexity* function of a language  $\Lambda$  is  $p(n) = \#\Lambda_n, n \geq 0$ .

53 The *Rauzy graph* of length  $n$  of a language  $\Lambda$  is a directed graph whose vertex set consists of all  
54 words of length  $n$  of  $\Lambda$ , with an edge from  $w$  to  $w'$  whenever  $w = av, w' = vb$  for letters  $a$  and  $b$ ,  
55 and the word  $avb$  is in  $\Lambda$ ; this edge is then labelled by  $b$ .

56

57 A word  $w$  in  $\Lambda$  is *right special* (resp. *left special*) if it has more than one *right extension*  $wx$   
58 (resp. *left extension*  $xw$ ) in  $\Lambda$ , with  $x$  in  $\mathcal{A}$ . If  $w$  is both right special and left special, then  $w$  is  
59 *bispecial*. If  $\#\Lambda_1 > 1$ , the empty word  $\varepsilon$  is bispecial. To *resolve* a bispecial word  $w$  is to find all  
60 words in  $\Lambda$  of the form  $xwy$  for letters  $x$  and  $y$ .

61 A *singular word* is  $w = xvy$  for letters  $x, y$ , such that some  $x'vy, x' \neq x$ , and  $xvy', y' \neq y$ , exist  
62 in  $\Lambda$ .

63

64 For a word  $w$ , we denote by  $w^\omega$  the one-sided infinite word  $www\dots$ , and by  $\Lambda_w$  the language  
65 consisting of all the factors of  $w^\omega$ . A language  $\Lambda$  is *closed under reversal* if  $w \in \Lambda \Leftrightarrow \bar{w} \in \Lambda$ .

66 A language  $\Lambda$  is *uniformly recurrent* if for every word  $w$  in  $\Lambda$ , there exists a constant  $K$  such that  
67  $w$  occurs in every word in  $\Lambda$  of length at least  $K$ .

68

## 2. BURROWS-WHEELER AND CLUSTERING

69 Let  $\mathcal{A} = \{a_1 < a_2 < \cdots < a_r\}$  be an ordered alphabet.

70 **Definition 1.** *The (cyclic) conjugates of  $w$  are the words  $w_i \cdots w_n w_1 \cdots w_{i-1}, 1 \leq i \leq n$ . If  $w$  is*  
71 *primitive,  $w$  has precisely  $n$  conjugates. Let  $w_{i,1} \cdots w_{i,n}$  denote the  $i$ -th conjugate of  $w$  where the*

72  $n$  conjugates of  $w$  are ordered by ascending lexicographical order.

73 Then the Burrows-Wheeler transform of  $w$ , defined in [5] and denoted by  $B(w)$ , is the word  
74  $w_{1,n}w_{2,n}\cdots w_{n,n}$ . In other words,  $B(w)$  is obtained from  $w$  by first ordering its conjugates in  
75 ascending order in a rectangular array, and then reading off the last column.

76 We say  $w$  is clustering for the permutation  $\pi$  if  $B(w) = (\pi a_1)^{n\pi a_1} \cdots (\pi a_r)^{n\pi a_r}$ , where  $\pi$  is a per-  
77 mutation on  $\mathcal{A}$  and  $n_a$  is the number of occurrences of  $a$  in  $w$  (we allow some of the  $n_a$  to be 0,  
78 thus, given the order and  $w$ , there may be several possible  $\pi$ ). We say  $w$  is perfectly clustering if it  
79 is clustering for the symmetric permutation  $\pi a_i = a_{r+1-i}$ ,  $1 \leq i \leq r$ .

80 **Non-primitive words.** As remarked in [17], the Burrows-Wheeler transform can be extended  
81 to a non-primitive word  $w_1 \cdots w_n$ , by ordering its  $n$  (non necessarily distinct) cyclic conjugates by  
82 non-strictly increasing lexicographical order and taking the word made by their last letters. Then  
83  $B(v^m)$  is deduced from  $B(v)$  by replacing each of its letters  $x_i$  by  $x_i^m$ , and  $v^m$  is clustering for  $\pi$   
84 iff  $v$  is clustering for  $\pi$ .

85

86 We shall now relate clustering to an *order condition*. This condition can be traced to [13], but  
87 was first mentioned explicitly in [8] in the particular case of symmetric permutations, and [11] in  
88 the general case, where it is studied extensively.

89 **Theorem 1.** For a given order  $<$  on the alphabet  $\mathcal{A}$ , a primitive word  $w$  over  $\mathcal{A}$  is clustering for the  
90 permutation  $\pi$  if and only if every bispecial word  $v$  in the language  $\Lambda_w$  satisfies the following order  
91 condition: whenever  $xvy$  and  $x'vy'$  are in  $\Lambda_w$  with letters  $x \neq x'$  and  $y \neq y'$ , then  $\pi^{-1}x < \pi^{-1}x'$   
92 if and only if  $y < y'$ .

93 Any bispecial word in  $\Lambda_w$  is a factor of  $ww$  and is of length at most  $|w| - 2$ .

94 **Proof**

95 We begin by proving the last assertion. Suppose  $v$  is a bispecial of  $\Lambda_w$ . Then  $v$  must occur at  
96 two different positions in some word  $w^k$ . If  $|w| = n$  and  $|v| \geq |w| - 1$ , this implies in particular  
97  $w_i \dots w_n w_1 \dots w_{i-2} = w_j \dots w_n w_1 \dots w_{j-2}$  for  $1 < j - i < n$ , and we notice that each  $w_l$  is in at least  
98 one member of the equality, thus we get that  $w$  is a power of a word whose length is the GCD of  $n$   
99 and  $j - i$ , which contradicts the primitivity. Thus the length of  $v$  is at most  $|w| - 2$ , and it occurs  
100 in  $ww$ .

101

102 We prove now that our order condition is equivalent to the following *modified order condition*:  
103 whenever  $z = z_1 \dots z_n$  and  $z' = z'_1 \dots z'_n$  are two different cyclic conjugates of  $w$ ,  $z < z'$  (lexico-  
104 graphically) if and only if  $\pi^{-1}z_k < \pi^{-1}z'_k$  for the largest  $k \leq n$  such that  $z_k \neq z'_k$ .

105 Indeed, by definition  $z < z'$  if and only if  $z_j < z'_j$  for the smallest  $j \geq 1$  such that  $z_j \neq z'_j$ . If  
106  $w$  satisfies the order condition, we apply it to the bispecial word  $z_{k+1} \dots z_n z_1 \dots z_{j-1}$ , with  $k$  and  $j$  as  
107 defined, and get the modified order condition.

108

109 Let  $v$  be a bispecial word in  $\Lambda_w$ ; by the first paragraph of this proof it can be written as  $z_1 \dots z_{k-1}$   
110 for some  $1 \leq k \leq n$ , with the convention that  $k = 1$  whenever  $v$  is empty, for at least two different  
111 cyclic conjugates  $z$  of  $w$ .

112 Then its possible extensions are the corresponding  $z_n z_1 \dots z_k$ , thus, if the modified order condi-  
113 tion is satisfied,  $v$  does satisfy the requirement of the order condition.

114

115 The modified order condition implies clustering, as then if two cyclic conjugates of  $w$  satisfy  
116  $z < z'$ , their last letters  $z_n$  and  $z'_n$  satisfy either  $z_n = z'_n$  or  $\pi^{-1}z_n < \pi^{-1}z'_n$ .

117 Suppose  $w = w_1 \cdots w_n$  is clustering for  $\pi$ . Suppose two cyclic conjugates of  $w$  are such that  
 118  $z_k \neq z'_k$ ,  $z_j = z'_j$  for  $k+1 \leq j \leq n$ . Then  $z < z'$  is (by definition of the lexicographical order)  
 119 equivalent to  $z_{k+1} \dots z_n z_1 \dots z_k < z'_{k+1} \dots z'_n z'_1 \dots z'_k$ , and, as these two words have different last letters,  
 120 because of the clustering this is equivalent to  $\pi^{-1}z_k < \pi^{-1}z'_k$ , thus the modified order condition is  
 121 satisfied.  $\square$

122

123 Theorem 1 remains valid if  $w = v^m$  is non-primitive (it can be slightly improved as there are  
 124 less bispecial words to be considered, it is enough to look at factors of  $vv$  of length at most  $|v| - 2$ ).

125

126 The following consequences of Theorem 1 or of [14] seem to be new.

127 **Proposition 2.** *If  $w$  clusters for the order  $<$  and the permutation  $\pi$ , its reverse clusters for the*  
 128  *$\pi$ -order, defined by  $x <_{\pi} y$  whenever  $\pi^{-1}x < \pi^{-1}y$ , and the permutation  $\pi^{-1}$ .*

129 **Proof**

130 This follows immediately from Theorem 1.  $\square$

131

132 **Proposition 3.** *Let  $w$  be a word on  $\mathcal{A}$ , ordered by  $<$ .*

133 *If  $w$  is perfectly clustering,  $\Lambda_w$  is closed under reversal.*

134 **Proof**

135 By Theorem 4 of [14], every perfectly clustering word  $w$  is such that  $ww$  is in the language  $\Lambda$   
 136 generated by a minimal discrete interval exchange with the symmetric permutation (we refer the  
 137 reader to [14] for the definitions), and  $\Lambda_w = \Lambda$ . It is known from [13] that such a  $\Lambda$  is stable under  
 138 reversal, thus we get our first assertion. This could also be deduced from Corollary 4.4 of [19].  $\square$

139

140 **Proposition 4.** *Let  $w$  be a word on  $\mathcal{A}$ , ordered by  $<$ .*

141 *If  $\Lambda_w$  is closed under reversal, the following conditions are equivalent*

142 (1)  *$w$  is clustering.*

143 (2)  *$w$  is perfectly clustering.*

144 (3) *For all words  $u$  and  $v$  with  $u \neq \bar{u}$  and  $v \neq \bar{v}$ , if  $uv$  is conjugate to  $w$ , then  $u < \bar{u}$  if and*  
 145 *only if  $v < \bar{v}$ .*

146 **Proof**

147 We begin by showing the equivalence between (1) and (2) then we show that (2)  $\Leftrightarrow$  (3). Clearly  
 148 (2)  $\Rightarrow$  (1). To see that (1)  $\Rightarrow$  (2), assume that  $w$  is clustering for some permutation  $\pi$  on  $\mathcal{A}$ . Let  $\mathcal{A}'$   
 149 be the set of all letters  $a \in \mathcal{A}$  which occur in  $w$ . To show that  $w$  is perfectly clustering, it suffices  
 150 to show that  $\pi^{-1}a < \pi^{-1}b \Leftrightarrow b < a$  for each pair of distinct letters  $a, b \in \mathcal{A}'$ . To this end, we will  
 151 show that the following set

$$\mathcal{E} = \{(a, b) \in \mathcal{A}' \times \mathcal{A}' : a \neq b \text{ and } a < b \Leftrightarrow \pi^{-1}a < \pi^{-1}b\}$$

152 is empty. We begin by establishing two claims:

153 **Claim 1 :** Assume  $xvy, x'vy' \in \Lambda_w$  with  $v$  a word, letters  $x \neq x'$  and  $y \neq y'$ . Then  $(x, x') \in \mathcal{E}$   
 154 if and only if  $(y, y') \in \mathcal{E}$ .

155 **Proof :** As  $\Lambda_w$  is closed under reversal, we also have  $y\bar{v}x, y'\bar{v}x' \in \Lambda_w$ . By Theorem 1 we obtain  
 156  $y < y' \Leftrightarrow \pi^{-1}x < \pi^{-1}x' \Leftrightarrow x < x' \Leftrightarrow \pi^{-1}y < \pi^{-1}y'$ .

157 **Claim 2 :** Assume  $xvy, x'vy' \in \Lambda_w$  with  $v$  a word, letters  $x \neq x'$  and  $y \neq y'$ . If  $(x, x') \in \mathcal{E}$ ,  
 158 then  $x < x' \Leftrightarrow y < y'$ .

159 **Proof :** Again by Theorem 1 we have  $x < x' \Leftrightarrow \pi^{-1}x < \pi^{-1}x' \Leftrightarrow y < y'$ .

160

161 Now assume to the contrary that  $\mathcal{E} \neq \emptyset$  and let  $(x_1, y_1) \in \mathcal{E}$ . Without loss of generality we may  
 162 assume that  $x_1 < y_1$ . Let  $u$  and  $v$  be conjugates of  $w$  with  $u$  beginning in  $x_1$  and  $v$  beginning in  
 163  $y_1$ . Then we may write  $u = x_1v_1x_2v_2 \cdots x_nv_n$  and  $v = y_1v_1y_2v_2 \cdots y_nv_n$  for some  $n \geq 2$  with  
 164 words  $v_i$  and letters  $x_i \neq y_i$  for each  $i = 1, 2, \dots, n$ . By application of Claims 1 and 2 we have that  
 165  $(x_i, y_i) \in \mathcal{E}$  and  $x_i < y_i$  for  $i = 1, 2, \dots, n$ . As  $u$  and  $v$  are conjugate to one another, in particular  
 166 they have the same number of occurrences of each letter, and hence the same is true of the words  
 167  $x = x_1x_2 \cdots x_n$  and  $y = y_1y_2 \cdots y_n$ . Pick a permutation  $\sigma$  of  $\{1, 2, \dots, n\}$  such that  $y_i = x_{\sigma(i)}$  for  
 168 each  $i = 1, 2, \dots, n$ . It follows that  $x_i < x_{\sigma(i)}$  for each  $i = 1, 2, \dots, n$ . Putting  $i$  equal to  $\sigma^j(1)$  we  
 169 obtain  $x_{\sigma^j(1)} < x_{\sigma^{j+1}(1)}$  for each  $j \geq 0$ . Thus  $x_1 < x_{\sigma(1)} < x_{\sigma^2(1)} < \cdots$ . Since  $\sigma^{n!}(1) = 1$  we  
 170 eventually get  $x_1 < x_1$ , a contradiction.

171

172 We next show that (2)  $\Rightarrow$  (3). So assume that  $w$  is perfectly clustering. Then by Theorem 1  
 173 we have that  $\Lambda_w$  satisfies the following order condition : whenever  $xzy$  and  $x'zy'$  are in  $\Lambda_w$  with  
 174  $x \neq x'$  and  $y \neq y'$ , we have  $y < y' \Leftrightarrow x' < x$ . Assume  $uv$  is conjugate to  $w$  with  $u \neq \bar{u}$  and  $v \neq \bar{v}$ .  
 175 Write  $u = rxtx'\bar{r}$  and  $v = sy't'y\bar{s}$  with words  $r, s, t, t'$ , letters  $x, x', y, y', x \neq x'$  and  $y \neq y'$ . Thus  
 176  $x'\bar{r}sy', y\bar{s}rx \in \Lambda_w$  and hence also  $x\bar{r}sy \in \Lambda_w$ . Applying the order condition to the words  $x\bar{r}sy$   
 177 and  $x'\bar{r}sy'$  we obtain  $y' < y \Leftrightarrow x < x'$  or equivalently  $u < \bar{u} \Leftrightarrow v < \bar{v}$  as required.

178

179 Finally we show that (3)  $\Rightarrow$  (2). Again by application of Theorem 1 it suffices to show that  
 180  $\Lambda_w$  satisfies the following order condition : whenever  $xzy$  and  $x'zy'$  are in  $\Lambda_w$  with  $|xzy| \leq |w|, z$   
 181 a word,  $x, x', y, y' \in \mathcal{A}$ ,  $x \neq x'$  and  $y \neq y'$ , we have  $y < y' \Leftrightarrow x' < x$ . So assume that  
 182  $xzy, x'zy' \in \Lambda_w$  with  $|xzy| \leq |w|, x \neq x'$  and  $y \neq y'$ . Then  $y'\bar{z}x' \in \Lambda_w$  since  $\Lambda_w$  is closed under  
 183 reversal. Let  $w'$  be a conjugate of  $w$  beginning in  $xzy$ . If the words  $xzy$  and  $y'\bar{z}x'$  are equal, then  
 184 in particular  $x = y'$  and  $y = x'$ , and therefore  $y < y'$  iff  $x' < x$ , which is what we want.

185 If the words  $xzy$  and  $y'\bar{z}x'$  are not equal, then we claim that these two words cannot overlap  
 186 one another, i.e., no non-empty prefix of one is equal to a suffix of the other. In fact, let  $u$  be a  
 187 non-empty prefix of  $xzy$  and let  $u'$  be a suffix of  $y'\bar{z}x'$ , then we will show  $u \neq u'$ . This is clear if  
 188  $|u| \neq |u'|$ . So let's suppose  $|u| = |u'|$ . Now if  $u = xzy$  then  $u' = y'\bar{z}x'$  and hence  $u \neq u'$ . On the  
 189 other hand if  $u$  is a proper prefix of  $xzy$ , then we can write  $u = xv$  and  $u' = \bar{v}x'$  for some prefix  $v$   
 190 (possibly empty) of  $z$ . As  $x \neq x'$ , it follows that  $u$  and  $u'$  are not abelian equivalent ( $u$  having one  
 191 additional occurrence of  $x$  than  $u'$ ) and hence in particular  $u \neq u'$ . A similar argument holds if  $u$   
 192 is a non-empty suffix of  $xzy$  and  $u'$  a prefix of  $y'\bar{z}x'$ .

193 Thus, if  $xzy \neq y'\bar{z}x'$ , as these words do not overlap one another we can write  $w' = xzyry'y'\bar{z}x's$   
 194 for some choice of words  $r, s$ . Put  $u = zyr'y'\bar{z}$  and  $v = x'sx$ . Then as  $uv$  is conjugate to  $w$  and  
 195  $u \neq \bar{u}$  and  $v \neq \bar{v}$ , we deduce  $y < y' \Leftrightarrow u < \bar{u} \Leftrightarrow v < \bar{v} \Leftrightarrow x' < x$  as required.  $\square$

196

197 We use now Theorem 1 to give a simple criterion which will be useful to avoid clustering.

198 **Lemma 5.** *Let  $x, y, z$  be three different letters in an alphabet  $\mathcal{A}$ , and  $w$  be a word on  $\mathcal{A}$ . Suppose*  
 199  *$w$  is clustering for the order  $<$  and the permutation  $\pi$ . Let  $v$  be a bispecial word in  $\Lambda_w$ :*

- 200  $\bullet$  *if the four words  $xvy, xvz, yvx, zvx$  are in  $\Lambda_w$ , then  $x$  is not between  $y$  and  $z$  (or  $z$  and  $y$ )*  
 201 *for the order  $<$ ,  $x$  is not between  $y$  and  $z$  (or  $z$  and  $y$ ) for the order  $<_{\pi}$ , and  $x$  is not on the*  
 202 *same side of  $y$  and  $z$  for the orders  $<$  and  $<_{\pi}$  ;*

- 203 • if three of the four words  $xvy$ ,  $xvz$ ,  $yvx$ ,  $zvx$  are in  $\Lambda_w$ , then  $x$  is not between  $y$  and  $z$  (or  
204  $z$  and  $y$ ) for the order  $<$ , or  $x$  is not between  $y$  and  $z$  (or  $z$  and  $y$ ) for the order  $<_\pi$ .

205 **Proof**

206 By Theorem 1, we have to check the order condition, for any fixed  $<$  and  $\pi$ . To check the require-  
207 ment of the order condition for the bispecial word  $v$ , we write the *extension graph* of  $v$ , with  $x$   
208  $y z$  in the order  $<$  on a line,  $x y z$  in the order  $<_\pi$  on a line below, and an edge from  $x'$  below  
209 to  $y'$  above whenever  $x'vy'$  is in  $\Lambda_w$ . If two of these edges have an intersection not reduced to an  
210 endpoint, the order condition is not satisfied.

211 In the first case, suppose  $y < x < z$ . if the  $\pi$  order is  $x, y, z$ , or  $x, z, y$ , or  $y, x, z$ , the edges  $xz$   
212 and  $zx$  intersect; if it is  $y, z, x$  or  $z, y, x$ ,  $xy$  and  $yx$  intersect; if it is  $z, x, y$ ,  $xy$  and  $zx$  intersect.  
213 This takes all possible  $\pi$ -orders into account, and no clustering is possible. The same is true if  
214  $z < x < y$  by left/right symmetry. Thus  $x$  cannot be in the middle for the order  $<$ , nor for the  
215  $\pi$ -order by up/down symmetry.

216 If  $x < y$  and  $x <_\pi y$ , or  $y < x$  and  $y <_\pi x$ ,  $xy$  and  $yx$  intersect whatever the position of  $z$ , thus  
217 we get the remaining assertions of the first case.

218 Suppose now for example  $xvy$ ,  $yvx$  and  $xvz$  are in  $\Lambda_w$ , we have to test all orders where  $x$  is  
219 twice in the middle. As above, if  $x < y$  and  $x <_\pi y$ , or  $y < x$  and  $y <_\pi x$ ,  $xy$  and  $yx$  intersect.  
220 There remain  $z < x < y$  and  $y <_\pi x <_\pi z$ , and  $y < x < z$  and  $z <_\pi x <_\pi y$ , and in both cases  $xz$   
221 intersects  $yx$ . And similarly for other sets of three words.  $\square$   
222

223

### 3. ARNOUX-RAUZY

224 **3.1. Definitions.** Throughout Section 3, we use the alphabet  $\{a, b, c\}$ , which can be equipped with  
225 any one of the six possible orders.

226 **Definition 2.** An AR language is a language on  $\{a, b, c\}$  generated by three families of words  $A_k$ ,  
227  $B_k$ ,  $C_k$ , build recursively from  $A_0 = a$ ,  $B_0 = b$ ,  $C_0 = c$ , by using a sequence of combinatorial  
228 rules (a), (b), (c), such that each one of the three rules is used infinitely many times, where

- 229 • by rule (a) at stage  $k$ ,  $A_{k+1} = A_k$ ,  $B_{k+1} = B_k A_k$ ,  $C_{k+1} = C_k A_k$ ;  
230 • by rule (b) at stage  $k$ ,  $A_{k+1} = A_k B_k$ ,  $B_{k+1} = B_k$ ,  $C_{k+1} = C_k B_k$ ;  
231 • by rule (c) at stage  $k$ ,  $A_{k+1} = A_k C_k$ ,  $B_{k+1} = B_k C_k$ ,  $C_{k+1} = C_k$ .

232 By an AR word we shall mean a factor of an AR language.

233 A standard AR word is an  $A_k$ ,  $B_k$ , or  $C_k$ , in an AR language

234 If the rules at stage  $k$  is  $(x_k)$ ,  $k \geq 0$ , the word  $D = x_0 x_1 \dots$  is called the directive word of  $\Lambda$ .

235 The Tribonacci language is the AR language defined by the directive word  $(abc)^\omega$ .

236 Every AR language is uniformly recurrent and closed under reversal, and has one right special  
237 and one left special of each length [1], thus AR languages are in the slightly more general class of  
238 episturmian languages, see Section 4.2 below.

239 An equivalent way to define an AR language is through AR morphisms. For  $x, y$  in  $\{a, b, c\}$ , we  
240 define  $\sigma_x x = x$ ,  $\sigma_x y = yx$  if  $x \neq y$ . For a word  $w$ ,  $\sigma_x w$  is defined by making  $\sigma_x$  a morphism for  
241 the concatenation, and the morphism  $\sigma_w$  is defined to be  $\sigma_{w_1} \circ \dots \circ \sigma_{w_n}$  if  $w = w_1 \dots w_n$ . We do  
242 the same with the morphisms  $\tau_x$  defined by  $\tau_x y = \overline{\sigma_x y}$  for each  $x, y \in \mathcal{A}$ . If the directive word  
243 of  $\Lambda$  is  $D = x_0 x_1 \dots$ , we put  $D_k = x_0 \dots x_{k-1}$ , and we have  $A_k = \sigma_{D_k} a$ ,  $B_k = \sigma_{D_k} b$ ,  $C_k = \sigma_{D_k} c$ ,  
244  $\bar{A}_k = \tau_{D_k} a$ ,  $\bar{B}_k = \tau_{D_k} b$ ,  $\bar{C}_k = \tau_{D_k} c$ . Being closed under reversal,  $\Lambda$  can be generated either by

245  $\sigma_{D_k}x, x \in \mathcal{A}, k \geq 0$ , or by  $\tau_{D_k}x, x \in \mathcal{A}, k \geq 0$ .

246

247 For an AR language  $\Lambda$ , note first that for all  $k$   $A_k$  begins with  $a$ ,  $B_k$  with  $b$ ,  $C_k$  with  $c$ . As  
 248 explained in [1], the three rules correspond to the building of the successive bispecials  $w_k$  in  $\Lambda$ ,  
 249 with rules (a), (b), (c) corresponding respectively to  $w_{k+1} = w_k A_k, w_{k+1} = w_k B_k, w_{k+1} = w_k C_k$ ,  
 250 starting with  $w_0$  being the empty word. From this and the closure under reversal, we deduce  
 251 that AR rule (x),  $x = a, b, c$ , is used at stage  $k$  if and only if the bispecial  $w_k$  is resolved by  
 252  $\{aw_kx, bw_kx, cw_kx, xw_ka, xw_kb, xw_kc\}$ , two of these six words being equal. Moreover, the words  
 253  $A_k, B_k, C_k$  are the (suffix) return words of  $w_k$  i.e.  $w_k Z_k$  contains  $w_k$  as a prefix and suffix and at  
 254 no other place, for  $Z = A, B, C$ .

255

256 An AR word  $w$  belonging to an AR language whose first rule is (x) will be such that each letter  
 257 of  $w$  which is not  $x$  is preceded (except if it is the first letter of  $w$ ), and followed (except if it is the  
 258 last letter of  $w$ ) by  $x$ , and  $x$  is the only letter with this property. We call  $x$  the *separating letter* of  
 259  $w$ .

260 We recall the description of the Rauzy graphs for AR languages from [1]: there are a left special  
 261 factor  $G$  with three left extensions, a right special factor  $R$  with three right extensions, a central  
 262 branch (with at least one vertex) from  $G$  to  $R$ , and three branches from  $R$  to  $G$ . The three *elemen-*  
 263 *tary circuits* in the Rauzy graphs of length  $n$  begin at  $R$  and follow one of the three branches from  
 264  $R$  to  $G$  then the central branch. Their labels are  $A_k, B_k, C_k$  for  $|w_{k-1}| + 1 \leq n \leq |w_k|$ .

265

266 We shall always use the obvious notation that if  $x, y$  or  $z$  is  $a, b$  or  $c$ ,  $X, Y$  and  $Z$  are the corre-  
 267 sponding  $A, B$  or  $C$ .

268

269 Let (Rabc) be the following assumption: *the rule at stage 0 is (a), and the first rule different*  
 270 *from (a) is (b)*. If it is not satisfied, we can make a permutation on the letters. When (Rabc) holds,  
 271 we define  $\lambda_1 > 0$  as the stage of the first rule (b),  $\lambda_2 > \lambda_1$  as the stage of the first rule (c).

272

273 The following *LMS notation* is defined in [6]; it is equivalent to the ABC notation, and will be  
 274 useful to express and show some of our results.

275 **Lemma 6.** *Assuming (Rabc), for all  $k > \lambda_1$ ,  $A_k, B_k$ , and  $C_k$  have three different lengths, and*  
 276 *we rename them such that  $|S_k| < |M_k| < |L_k|$ . We put  $S_k = A_k, M_k = C_k, L_k = B_k$  for all*  
 277  *$1 \leq k \leq \lambda_1, S_0 = c, M_0 = b, L_0 = a$ . The AR rules can be written as*

- 278 • if  $w_{k+1} = w_k S_k, S_{k+1} = S_k, M_{k+1} = M_k S_k, L_{k+1} = M_k L_k$ ;
- 279 • if  $w_{k+1} = w_k M_k, S_{k+1} = M_k, M_{k+1} = S_k M_k, L_{k+1} = L_k M_k$ ;
- 280 • if  $w_{k+1} = w_k L_k, S_{k+1} = L_k, M_{k+1} = S_k L_k, L_{k+1} = M_k L_k$ .

281 We have  $w_{p+1} = w_p L_p$  whenever  $p = 0, p = \lambda_1, p = \lambda_2$ , or at stage  $p$  we have a rule (x<sub>1</sub>)  
 282 preceded by a string of rules (x<sub>2</sub>) and a string of rules (x<sub>3</sub>), for  $\{x_1, x_2, x_3\} = \{a, b, c\}$ , thus this  
 283 happens for infinitely many  $p$ . We have  $w_{p+1} = w_p S_p$  whenever the rules at stages  $p - 1$  and  $p$  are  
 284 the same, and  $w_{p+1} = w_p M_p$  for the remaining  $p$ .

### 285 Proof

286 We have  $w_k = a^k, A_k = a, B_k = ba^k, C_k = ca^k$  for all  $1 \leq k \leq \lambda_1$ , then  $w_{\lambda_1+1} = a^{\lambda_1} ba^{\lambda_1}$ ,  
 287 and  $A_{\lambda_1+1} = aba^{\lambda_1} = M_{\lambda_1+1}, B_{\lambda_1+1} = ba^{\lambda_1} = S_{\lambda_1+1}, C_{\lambda_1+1} = ca^{\lambda_1} ba^{\lambda_1} = L_{\lambda_1+1}$  have three  
 288 different lengths; this is preserved by further rules. The other assertions are straightforward, see



289 [6] for more details. □

290

### 291 3.2. Lengths, squares, conjugates.

292 **Lemma 7.** *Assuming (Rabc),*

293  $A_p$  is a suffix of  $w_p$  iff  $|A_p| \leq |w_p|$  or equivalently iff  $p \geq 1$ ,

294  $B_p$  is a suffix of  $w_p$  iff  $|B_p| \leq |w_p|$  or equivalently iff  $p \geq \lambda_1 + 1$ ,

295  $C_p$  is a suffix of  $w_p$  iff  $|C_p| \leq |w_p|$  or equivalently iff  $p \geq \lambda_2 + 1$ ,

296  $|M_p| + |S_p| > |L_p|$  for all  $p > 0$ ,

297  $|L_p| < |M_{p+1}|$  for all  $p > 0$ .

#### 298 **Proof**

299 By the analysis of Lemma 5,  $A_p$  is a suffix of  $w_p$  for  $p = 1$  and strictly longer than  $w_p$  for  $p = 0$ ,  
 300  $B_p$  is a suffix of  $w_p$  for  $p = \lambda_1 + 1$  and strictly longer than  $B_p$  for  $p \leq \lambda_1$ .  $C_p$  is strictly longer  
 301 than  $w_p$  for  $p = \lambda_1 + 1$ . If  $C_p$  is strictly longer than  $w_p$  for some  $p \leq \lambda_2 - 1$ , then  $w_{p+1} = w_p Y_p$   
 302 and  $C_{p+1} = C_p Y_p$  for  $Y = A, B$ , thus  $C_{p+1}$  is strictly longer than  $w_{p+1}$ . Then  $w_{\lambda_2+1} = w_{\lambda_2} C_{\lambda_2}$   
 303 and  $C_{\lambda_2+1} = C_{\lambda_2}$  is a suffix of  $w_{\lambda_2+1}$ . Suppose now  $Z_p$  is a suffix of  $w_p$ , for  $Z = A, B, C$ . Then  
 304 either  $w_{p+1} = w_p Z_p$  and  $Z_{p+1} = Z_p$ , or  $w_{p+1} = w_p Y_p$  and  $Z_{p+1} = Z_p Y_p$ , thus in both cases  $Z_{p+1}$  is  
 305 a suffix of  $w_{p+1}$ , and thus shorter.

306 The fourth assertion is proved in [7], although with non-strict inequalities, but the proof, using  
 307 the AR rules, does give the strict ones.

308 The last one comes from the fourth one and the fact that  $|M_{p+1}|$  is either  $|M_p| + |S_p|$  or  
 309  $|M_p| + |L_p|$ . □

310

311 **Proposition 8.** *In an AR language  $\Lambda$  satisfying (Rabc), the primitive words  $v$  such that  $vv$  is in  $\Lambda$   
 312 are all the  $A_p$  for  $p \geq 0$ , the  $B_p$  for  $p \geq \lambda_1$ , the  $C_p$  for  $p \geq \lambda_2$ , and some (possibly none) of their  
 313 (cyclic) conjugates.*

#### 314 **Proof**

315 Let  $Z_p = A_p$  for  $p \geq 1$ ,  $B_p$  for  $p \geq \lambda_1 + 1$ , or  $C_p$  for  $p \geq \lambda_2 + 1$ . Then  $w_p Z_p$  is in  $\Lambda$  while by  
 316 Lemma 6  $Z_p$  is a suffix of  $w_p$ , thus  $Z_p^2$  is in  $\Lambda$ . This is true also for  $A_0 = a$  as  $aa$  is a suffix of  
 317  $w_1 A_1$ , for  $B_{\lambda_1}$  as  $w_{\lambda_1+1} B_{\lambda_1+1} = w_{\lambda_1} B_{\lambda_1}^2$  is in  $\Lambda$ , and for  $C_{\lambda_2}$  as  $w_{\lambda_2+1} C_{\lambda_2+1} = w_{\lambda_2} C_{\lambda_2}^2$  is in  $\Lambda$ .  
 318 Note that some conjugates of the  $Z_p$  may have the same property, but for the Tribonacci language  
 319 and  $v = A_2 = aba$ ,  $vv$  is in  $\Lambda$  but no  $v'v'$  for  $v' = baa$  or  $v' = aab$ .

320 As remarked in [4], for a primitive  $v$ , if  $vv$  is in  $\Lambda$ , then  $v$  is the label of a circuit in the Rauzy  
 321 graph of length  $|v|$  where no vertex is used more than once; thus  $v$  can only be some conjugate of  
 322 some  $A_p$ ,  $B_p$ , or  $C_p$ , for a  $p$  such that  $|w_{p-1}| + 1 \leq |v| \leq |w_p|$ . This will not be the case for the  
 323 conjugates of  $B_p$ ,  $p < \lambda_1$ , as then  $B_p$  is strictly longer than  $w_p$  and  $B_{p+1} \neq B_p$  because the rule at  
 324 stage  $p$  is not (b), nor for the conjugates of  $C_p$ ,  $p < \lambda_2$ , as then  $C_p$  is strictly longer than  $w_p$  and  
 325  $C_{p+1} \neq C_p$  because the rule at stage  $p$  is not (c). □

326

327 For non-primitive words Proposition ?? fails: in an AR language  $\Lambda$  where there are five consec-  
 328 utive rules (a) at stages  $p$  to  $p + 4$ ,  $A_p^4$  is in  $\Lambda$  while  $A_p^2$  is not any  $A_k$ ,  $B_k$  or  $C_k$  in  $\Lambda$ . If we take  
 329  $p = 0$ , we see that  $A_0^2 = aa$  is not an  $A_k$ ,  $B_k$  or  $C_k$  in any AR language.

330 **Proposition 9.** *For a primitive word  $v$ , the following assertions are equivalent*

331 (1)  $v$  is conjugate to a standard AR word,

- 332 (2)  $vv$  is an AR word,  
 333 (3)  $v'v'$  is an AR word for some conjugate  $v'$  of  $v$ ,  
 334 (4) all the conjugates of  $v$  are in some AR language  $\Lambda$ ,  
 335 (5)  $x$  can be de-substituted down (using the six AR morphisms  $\tau_x$  and  $\sigma_x$ ,  $x \in \mathcal{A}$ ) to a single  
 336 letter.

### 337 Proof

338 Suppose  $v$  satisfies (1); to show that  $vv$  is always an AR word, by Proposition ?? what remains  
 339 to prove is that when  $v$  is one of the initial  $A_p, B_p, C_p$ , or is conjugate to any  $A_p, B_p, C_p$  in an  
 340 AR language  $\Lambda$  satisfying  $(Rabc)$ , then  $vv$  is in an AR language  $\Lambda'$ . This is true for  $B_0, C_0$  by  
 341 exchanging  $a$  with  $b$  or  $c$ . For  $B_p$ ,  $1 \leq p \leq \lambda_1$ , this will be true for  $\Lambda'$  defined by the same rules as  
 342  $\Lambda$  up to stage  $p - 1$ , then rule  $(b)$  at stage  $p$ , and any admissible sequence of rules beyond: in this  
 343 language our  $B_p$  is  $B'_{\lambda_1}$ . For  $C_p$ ,  $1 \leq p \leq \lambda_1$ , this will be true for  $\Lambda'$  deduced from the previous  $\Lambda'$   
 344 by exchanging  $b$  and  $c$ . For  $C_p$ ,  $\lambda_1 + 1 \leq p \leq \lambda_2 - 1$ , this will be true for  $\Lambda'$  defined by the same  
 345 rules as  $\Lambda$  up to stage  $p - 1$ , then rule  $(c)$  at stage  $p$ , and any admissible sequence of rule beyond:  
 346 in this language our  $C_p$  is  $C'_{\lambda_1}$ . Let now  $u$  be a conjugate to some  $Z_p$  which is a suffix of  $w_p$ ; then  
 347 we define  $\Lambda'$  by the same rules as  $\Lambda$  up to stage  $p - 1$ , then rule  $(z)$  at stage  $p$ , and any admissible  
 348 sequence of rules beyond: then  $w_p Z_p^2$ , thus  $Z_p^3$ , thus  $uu$  is in  $\Lambda$ . This will still be true for  $Z_p = B_{\lambda_1}$   
 349 or  $Z_p = C_{\lambda_2}$  if we define  $\Lambda'$  by the same rules as  $\Lambda$  up to stage  $p$ , then rule  $(z)$  at stage  $p + 1$ , and  
 350 any admissible sequence of rules beyond. Thus we get (2).

351 If  $v$  satisfies (2) with  $vv$  in  $\Lambda$ , then  $v'v'$  is in  $\Lambda$  for the conjugate  $v' = v$ , which gives (3), and if  
 352  $v$  satisfies (3) with  $v'v'$  in  $\Lambda$ , all the conjugates of  $v'$ , thus of  $v$ , are in  $\Lambda$ , thus we get (4).

353 Suppose  $v$  satisfies (4), with all the conjugates of  $v$  in an AR language  $\Lambda$ . In the Rauzy graph  
 354 of length  $|v|$  of  $\Lambda$ , we see each conjugate  $v^{(i)}$ ; there is an edge between  $v^{(i)}$  and the next one in  
 355 the circular order, because either  $v^{(i)}$  has only one right extension or  $v^{(i)}$  has all the possible right  
 356 extensions. Thus there is a circular path whose vertices are all the  $v^{(i)}$ , each one occurring only  
 357 once as  $v$  is primitive; at least one vertex  $v'$  in this path is on the central branch, and if we start  
 358 from this point the circular path is an allowed path. Thus  $v'v'$  is in  $\Lambda$ , as remarked in [4], thus  $v'$   
 359 is conjugate to a standard AR word by the reasoning above, thus  $v$  is conjugate to a standard AR  
 360 word. Thus we have proved the equivalence of (1), (2), (3) and (4).

361

362 To deal with (5), let us show first that a standard AR word  $Z$  is conjugate to its reverse. This  
 363 is true if  $Z$  has one letter. Other  $Z$  are of the form  $Z = \sigma_x Z'$ , for a shorter standard AR word  $Z'$ .  
 364 Then, if we suppose  $Z'$  is conjugate to  $\bar{Z}'$ , we get that  $Z = \sigma_x Z'$  is conjugate to  $\sigma_x \bar{Z}'$ , and the  
 365 latter is conjugate to  $\bar{Z} = \tau_x \bar{Z}'$  as these two words are of the form  $yw$  and  $wy$  for some letter  $y$ .

366 Let  $v$  be a standard AR word, with separating letter  $a$ , and  $v'$  a conjugate of  $v$ . Then  $v'$  must  
 367 either begin or end in  $a$ , otherwise  $v' = xv''y$  with each of  $x$  and  $y$  different from  $a$ ; then every  
 368 conjugate of  $v'$  which begins in  $a$  must contain  $yx$  as a factor, and in particular  $v$  contains  $yx$   
 369 which contradicts the fact that  $a$  is the separating letter of  $w$ . If  $v'$  starts in  $a$ , then  $v' = \tau_a u'$ , by  
 370 the properties of the separating letter, and if  $v'$  ends in  $a$  then  $v' = \sigma_a u'$ . If  $v' = \sigma_a u'$ , then  $u'$  is  
 371 conjugate to the standard AR word  $u$  such that  $v = \sigma_a u$ . If  $v' = \tau_a u'$ , then  $v'$  is also conjugate to  
 372  $\bar{v}$ , where  $\bar{v} = \tau_a \bar{u}$  for a standard AR word  $u$ , and  $u'$  is conjugate to  $u$ . Then we apply the same  
 373 process to  $u'$  as long as  $u'$  has at least two letters, and end when we get to a single letter.

374 Finally, to get that (5) implies (1), it is enough to prove that if  $u'$  is conjugate to a standard AR  
 375 word  $u$ , then, for any letter  $x$ ,  $\sigma_x u'$  or  $\tau_x u'$  is also conjugate to a standard AR word, and this is true

376 as both these words are conjugate to  $\sigma_x u$ . □

377

378 For  $v$  to be conjugate to  $v'$  such that  $v'v'$  is an AR word, it is not enough that all the conjugates of  
 379  $v$  are AR words, take  $abc$  for example. Also, a conjugate of a standard AR word is not necessarily  
 380 a standard AR word, for example  $caba$  is standard in the Tribonacci language, but  $abac$  is not  
 381 standard in any AR language, otherwise it could be written as the concatenation of two words with  
 382 the same last letter.

383 To emphasize the part played by the cyclic conjugates, we notice the following fact.

384 **Proposition 10.** *Let  $\Lambda$  be any language on an alphabet  $\mathcal{A}$  such that for every  $w$  in  $\Lambda$  and  $a$  in  $\mathcal{A}$ ,  
 385 there exists  $v$  in  $\Lambda$  such that  $wva$  is in  $\Lambda$ . Then the closure of  $\Lambda$  for the cyclic conjugacy is made of  
 386 all the possible words on  $\mathcal{A}$ .*

387 **Proof**

388 Let  $x_1x_2\dots x_n$  be any word. We will show  $x_1x_2\dots x_n$  is in this closure, denoted by  $\Lambda'$ . Let  
 389  $x_nwx_{n-1}$  be in  $\Lambda$  such that  $w$  contains each letter sufficiently many times. So  $x_{n-1}x_nw$  is in  $\Lambda'$ .  
 390 Now write  $w = ux_{n-2}v$  where  $u$  contains each letter sufficiently many times. So  $x_{n-1}x_nux_{n-2}$   
 391 is in  $\Lambda'$  (factorial property of languages), hence  $x_{n-2}x_{n-1}x_nu$  is in  $\Lambda'$  (closed under cyclic conju-  
 392 gates). And so on for  $x_{n-3}, x_{n-4}, \dots$  □

393

### 394 3.3. Non-clustering AR words.

395 **Lemma 11.** *Let  $\Lambda$  be an AR language satisfying (Rabc). If  $u \in \Lambda$  contains at least three of the  
 396 non-singular words  $xw_{\lambda_2}y$ ,  $x, y$  in  $\{a, b, c\}$ ,  $(x, y) \neq (c, c)$ ,  $u$  cannot cluster for any permutation  
 397  $\pi$  and any order on  $\{a, b, c\}$ , nor can  $v$  if  $u = vv$ .*

398 **Proof**

399 Suppose  $u$  contains at least three of  $cw_{\lambda_2}a$ ,  $cw_{\lambda_2}b$ ,  $bw_{\lambda_2}c$ ,  $aw_{\lambda_2}c$ . As  $w_{\lambda_1}$  is both a prefix and a  
 400 suffix of  $w_{\lambda_2}$ ,  $u$  contains also  $cw_{\lambda_1}$ ,  $w_{\lambda_1}c$  and at least three of  $aw_{\lambda_1}$ ,  $w_{\lambda_1}a$ ,  $bw_{\lambda_1}$ ,  $w_{\lambda_1}b$ . As  $u$  is in  $\Lambda$   
 401 and the rule at stage  $\lambda_1$  is  $(b)$ ,  $u$  contains  $cw_{\lambda_1}b$ ,  $bw_{\lambda_1}c$  and at least either  $aw_{\lambda_1}b$  or  $bw_{\lambda_1}a$ . As  $w_{\lambda_1}$   
 402 begins and ends with  $a$ ,  $u$  contains  $ca$ ,  $ac$ ,  $ab$ ,  $ba$ ,  $aa$ .

403 Suppose  $u$  clusters, then we apply Lemma 4. By its first assertion applied to the empty bispe-  
 404 cial, we must assign  $a$  to an end of the  $<$  order and the opposite one of  $<_{\pi}$ . Then, by the second  
 405 assertion applied to  $w_{\lambda_1}$  and  $w_{\lambda_2}$ , as we cannot give two ends to both  $b$  and  $c$ , we must assign  $b$  to  
 406 the middle of one order, and  $c$  to the middle of the other one. Thus, up to left/right and up/down  
 407 symmetries in the pictures in the proof of Lemma 4, we have  $a < b < c$  and  $b <_{\pi} c <_{\pi} a$ . But  
 408 we know that  $cw_{\lambda_1}b$  and  $bw_{\lambda_1}c$  are factors of  $u$ , which gives two intersecting lines  $bc$  and  $cb$  in the  
 409 picture for  $w_{\lambda_1}$ , and this contradicts Theorem 1. □

410

411 Together with uniform recurrence, Lemma 9 provides our main answer to Dolce's question: *in*  
 412 *a given AR language, there are only finitely many clustering words.* This is noticed in [3] in the  
 413 particular case of an infinite sequence of words in the Tribonacci language called the Tribonacci  
 414 standard words, for which the Burrows-Wheeler transform is explicitly computed, and this is gen-  
 415 eralized to  $r$ -Bonacci, see Section 4.3 below. Note also that both AR words and clustering are  
 416 mentioned in [19], but no relation between these notions is established.

417 The following theorem gives an estimate for the maximal length of a clustering word, for which  
 418 some claims to optimality will be given In Corollary 16 below. It relies on a method used in [18]  
 419 for Sturmian languages and [6] for AR languages.

420 **Theorem 12.** *We recall that, assuming  $(Rabc)$ ,  $\lambda_2$  is the stage of the first rule  $(c)$ ; Let  $\lambda_a$  be the*  
 421 *stage of the last rule  $(a)$  before  $\lambda_2$ ,  $\lambda_b$  the stage of the last rule  $(b)$  before  $\lambda_2$ ,  $\mu_a$  the stage of the*  
 422 *first rule  $(a)$  after  $\lambda_2$ ,  $\mu_b$  the stage of the first rule  $(b)$  after  $\lambda_2$ . Let  $x$  and  $y$  be the elements of  $\{a, b\}$*   
 423 *such that  $\mu_x < \mu_y$ .*  
 424 *Then no word of length at least  $|w_{\lambda_y}| + \max(|C_{\mu_y+1}|, |X_{\mu_y+1}|) + 1$  can cluster for any permutation*  
 425 *and any order.*

426 **Proof**

427 We want to estimate the minimal length of a word containing at least three  $xw_{\lambda_2}y$ ,  $(x, y) \neq (c, c)$ ,  
 428 so that we can apply Lemma 9. As is noticed in [18], a word  $w$  occurs in any word of  $\Lambda$  whose  
 429 length is at least  $|w| - 1 + t(w)$ , where  $t(w)$  is the maximal return time of  $w$ , i.e. the maximal  
 430 possible difference between the indexes of two consecutive occurrences of  $w$ . And  $t(w)$  is the same  
 431 as  $t(u)$ , where  $u$  is the longest singular word contained in  $w$ , or  $u$  is a single letter if  $w$  contains no  
 432 singular word.

433 We reprove, with other notations, Lemma 2.3 of [6]. Let  $v = zw_{p-1}z$  be a singular word. We  
 434 define four assertions:

435  $(OA_q)$   $v$  occurs once in  $w_qA_q$ ,  $v$  does not occur in  $w_qB_q$  or  $w_qC_q$ ,  
 436  $(OB_q)$   $v$  occurs once in  $w_qB_q$ ,  $v$  does not occur in  $w_qA_q$  or  $w_qC_q$ ,  
 437  $(OC_q)$   $v$  occurs once in  $w_qC_q$ ,  $v$  does not occur in  $w_qA_q$  or  $w_qB_q$ ,  
 438  $(OT_q)$   $v$  occurs at least once in  $w_qA_q$ ,  $w_qB_q$  and  $w_qC_q$ , the maximal return time of  $v$  is  $|L_q| =$   
 439  $\max(|A_q|, |B_q|, |C_q|)$ .

440 If  $w_p = w_{p-1}Z_{p-1}$ ,  $z$  is the first letter of  $Z_{p-1}$  and  $Z_p = Z_{p-1}$ , thus  $(OZ_p)$  holds. Then the  
 441 AR rules imply that if the rule at stage  $q$  is  $(a)$ ,  $(OA_q)$  implies  $(OT_{q+1})$ ,  $(OB_q)$  implies  $(OB_{q+1})$ ,  
 442  $(OC_q)$  implies  $(OC_{q+1})$ , and mutatis mutandis for rules  $(b)$  and  $(c)$ .

443  
 444 We need to know the maximal return times of  $u = yw_{\lambda_y}y$  and  $u' = xw_{\lambda_x}x$ . The analysis above  
 445 implies that  $t(u)$  is known as soon as we see a rule  $(y)$  after  $\lambda_2$ , which happens at stage  $\mu_y$ , and  
 446 that  $t(u) = |L_{\mu_y+1}|$ , and similarly  $t(u') = |L_{\mu_x+1}|$ .

447  
 448 The rule at stage  $\lambda_2$  is  $(c)$ . There are only rules  $(c)$  (or none) (strictly) between  $\lambda_2$  and  $\mu_x$ , there  
 449 are only rules  $(x)$  or  $(c)$  (or none) (strictly) between  $\mu_x$  and  $\mu_y$ . By Lemma 5, for  $p = \lambda_2$ , and  
 450  $p = \mu_y$ , we have  $w_{p+1} = w_pL_p$ . This is true also for  $p = \mu_x$  if  $\lambda_x < \lambda_y$ , and in any case this  
 451 happens for no other  $\lambda_2 \leq p \leq \mu_y$ . In particular, we get that  $L_{\mu_y} = Y_{\mu_y}$  and  $L_{\mu_y+1}$  is the longest  
 452 of  $X_{\mu_y+1}$  and  $C_{\mu_y+1}$ .

453

454 **First case,**  $\lambda_x < \lambda_y$ .

455 Then  $\lambda_y = \lambda_2 - 1$ . We know that  $xw_{\lambda_2}c$  and  $cw_{\lambda_2}x$ , which contain  $u'$  as maximal singular word,  
 456 occur in any word in  $\Lambda$  of length at least  $|w_{\lambda_2}| + |L_{\mu_x+1}| + 1$ , and this is smaller than the required  
 457 bound as  $|w_{\lambda_2}| - |w_{\lambda_2-1}| = |Y_{\lambda_2-1}| < |L_{\lambda_2}| < |M_{\mu_y}| \leq |L_{\mu_y+1}| - |L_{\mu_x+1}|$ .

458 We know also that  $yw_{\lambda_2}c$  and  $cw_{\lambda_2}y$ , which contain  $u$  as maximal singular word, occur in any  
 459 word in  $\Lambda$  of length at least  $|w_{\lambda_2}| + |L_{\mu_y+1}| + 1$ . But we can improve this bound a little if we  
 460 want only to see  $yw_{\lambda_2}c$  or  $cw_{\lambda_2}y$ . Indeed,  $u$  occurs in any word  $Z$  in  $\Lambda$  of length  $|L_{\mu_y}| + |u| - 1$ .  
 461 Also,  $u$  is a prefix of  $yw_{\lambda_2}c$  and there is only one way to extend  $u$  to the right to length  $|w_{\lambda_2}| + 2$ ,  
 462 giving  $yw_{\lambda_2}c$ ,  $u$  is a suffix of  $cw_{\lambda_2}y$ , and there is only one way to extend  $u$  to the left to length  
 463  $|w_{\lambda_2}| + 2$ , giving  $cw_{\lambda_2}y$ . Thus  $yw_{\lambda_2}c$  or  $cw_{\lambda_2}y$  is in  $Z$ , provided  $Z$  is long enough to ensure that  
 464 we can extend  $u$  to the right or left as far as that length while remaining in  $Z$ ; in the worst case, we

465 can extend it by a length  $\frac{|Z|-|u|}{2}$ , so we have to check this is at least  $|w_{\lambda_2}| + 2 - |u|$ . Thus we have  
 466 to prove that  $|L_{\mu_y+1}| > 2(|w_{\lambda_2}| - |w_{\lambda_2-1}|)$ : the right side is  $2|Y_{\lambda_2-1}|$  while the left side is at least  
 467  $|L_{\lambda_2+1}| = |X_{\lambda_2-1}| + |C_{\lambda_2-1}| + 2|Y_{\lambda_2-1}|$ . And we conclude by Lemma 9.

468

469 **Second case,  $\lambda_y < \lambda_x$ .**

470 Then  $\lambda_x = \lambda_2 - 1$ . Again, we have to check that  $xw_{\lambda_2}c$  and  $cw_{\lambda_2}x$  occur in any word of the required  
 471 length, and  $yw_{\lambda_2}c$  or  $cw_{\lambda_2}y$  occur in any word of the required length. Using the same methods as  
 472 in the previous paragraph, this is done by checking that  $|w_{\lambda_2}| - |w_{\lambda_y}| < |L_{\mu_y+1}| - |L_{\mu_x+1}|$  and  
 473  $|L_{\mu_x+1}| > 2(|w_{\lambda_2}| - |w_{\lambda_y}|)$ . We have  $|w_{\lambda_2}| - |w_{\lambda_y}| = t(|Y_{\lambda_y}| + |X_{\lambda_y}|)$ , for some positive in-  
 474 teger  $t$ . Knowing the rules between stages  $\lambda_2$  and  $\mu_x$ , we get that both  $|L_{\mu_x+1}|$  and  $|M_{\mu_x+1}|$  are  
 475 at least  $2t(|X_{\lambda_y}| + |Y_{\lambda_y}|)$ . Then we can conclude, using also that  $|L_{\mu_y+1}| - |L_{\mu_x+1}|$  is at least  
 476  $|M_{\mu_y}| \geq |M_{\mu_x+1}|$ .  $\square$

477

478 The sharpness of the bound in Theorem 10 will be studied in Corollary 16 and Examples 1, 2  
 479 and 3 below.

#### 480 3.4. AR words conjugate to standard.

481 **Lemma 13.** *If a bispecial  $v$  in a language  $\Lambda_w$  is resolved by a subset of  $\{avb, avc, ava, bva, cva\}$ ,  
 482  $v$  satisfies the requirement of the order condition for any order  $<$  such that  $a$  is at an end, and the  
 483 symmetric permutation. If in  $\Lambda_w$  a bispecial  $v'$  is resolved by a subset of  $\{av'b, cv'b, bv'b, bv'a, bv'c\}$ ,  
 484 both  $v$  and  $v'$  satisfy the requirement of the order condition for the orders  $a < c < b$  or  $b < c < a$ ,  
 485 and the symmetric permutation.*

#### 486 **Proof**

487 We draw the extension graphs as in the proof of Lemma 4 and check that any two edges do not  
 488 intersect except at their endpoints.  $\square$

489

490 Lemma 11 provides a partial converse to Lemma 9, as it allows us to build clustering AR words  
 491 in the absence of the obstructions in its hypothesis, but it does not give a necessary condition for  
 492 clustering, as we shall see in Section 3.5 below.

493 **Proposition 14.** *With the assumption  $(Rabc)$  and the notations of Theorem 10 above,  $Y_p$  clusters  
 494 if and only if  $p \leq \mu_x$ ,  $C_p$  and  $X_p$  cluster if and only if  $p \leq \mu_y$ .*

#### 495 **Proof**

496 We begin by the negative direction. Let  $Z_p$  be  $A_p$ ,  $B_p$  or  $C_p$  for the values in the hypotheses. Then,  
 497 using the rules between  $\lambda_2$  and  $\mu_y$  as determined in the proof of Theorem 10, we check that  $A_{\lambda_2}$ ,  
 498  $B_{\lambda_2}$  and  $C_{\lambda_2}$  all appear in the decomposition of  $Z_p$  by the AR rules. As each  $A_{\lambda_2}$  and  $B_{\lambda_2}$  in this  
 499 decomposition is preceded by  $w_{\lambda_2}$ , if  $|Z_p| \geq |w_{\lambda_2}|$  then  $w_{\lambda_2}A_{\lambda_2}$  and  $w_{\lambda_2}B_{\lambda_2}$ , thus  $w_{\lambda_2}a$  and  $w_{\lambda_2}b$ ,  
 500 occur in  $Z_p^2$ . Indeed, for these values of  $p$ , we have  $|Z_p| \geq |w_{\lambda_2}| + 1$  by Lemma 6, thus  $cw_{\lambda_2}a$   
 501 and  $cw_{\lambda_2}b$  occur in  $Z_p^2$ . A symmetric reasoning holds for  $aw_{\lambda_2}c$  and  $bw_{\lambda_2}c$ , as  $w_{\lambda_2}A_{\lambda_2} = A'_{\lambda_2}w_{\lambda_2}$ ,  
 502  $w_{\lambda_2}B_{\lambda_2} = B'_{\lambda_2}w_{\lambda_2}$ , where  $A'_{\lambda_2}$  ends with  $a$  and  $B'_{\lambda_2}$  ends with  $b$ . This contradicts the clustering by  
 503 Lemma 9.

504

505 In the positive direction, let  $Z_p$  be an  $A_p$  for  $p \geq 0$ , or a  $B_p$  for  $p \geq \lambda_1$ , or a  $C_p$  for  $p \geq \lambda_2$ . By  
 506 the reasoning of Proposition ??,  $Z_p^2$  is a suffix of  $w_pZ_p$ , or  $w_{p+1}Z_{p+1}$  if  $Z_p = A_0$ ,  $Z_p = B_{\lambda_1}$  or  
 507  $Z_p = C_{\lambda_2}$ , and the bispecials in the language  $\Lambda_{Z_p}$  are resolved by AR rules. By Lemma 11, those

508 which are resolved by rule (a) or (b) satisfy the order condition ; as for bispecials  $w_t$  resolved  
 509 by rule (c), they do satisfy the order condition if they are resolved in  $\Lambda_{Z_p}$  by  $\{cwc, awc, cwa\}$  or  
 510  $\{cwc, bwc, cwb\}$ . This will happen if the longest singular word  $aw_qa$  occurring in  $awc$  and  $cwa$ ,  
 511 or the longest singular word  $bw_qb$  occurring in  $cwb$  and  $bwc$ , does not occur in  $Z_p^2$ .

512 Using the rules between  $\lambda_2$  and  $\mu_y$  as determined in the proof of Theorem 10, we track  $bw_{\lambda_b}b$   
 513 and  $aw_{\lambda_a}a$  as in Theorem 10, and get that one of them, namely  $u = yw_{\lambda_y}y$ , does not occur in  
 514  $w_pZ_p$ , nor in  $w_{p+1}Z_{p+1}$  when needed, hence in  $Z_p^2$ , and thus  $w_{\lambda_2}$  satisfies the order condition in  
 515  $\Lambda_{Z_p}$ . We look now at any longer bispecial  $w_t$  resolved by rule (c): as there are only rules (c) and  
 516 (x) (strictly) between  $\lambda_2$  and  $\mu_y$ , the  $yw_qy$  defined above is  $u$  as long as  $\lambda_2 \leq t \leq \mu_y$ , and we  
 517 know that  $u$  does not occur in  $Z_p^2$ . Thus all these bispecials satisfy the order condition in  $\Lambda_{Z_p}$ ; as  
 518 for still longer bispecial words of  $\Lambda$ , they are too long to occur in  $Z_p^2$ , as  $w_{\mu_y+1} = w_{\mu_y}L_{\mu_y}$  has a  
 519 length greater than  $2|L_{\mu_y}|$ .

520 There remain to consider the  $A_p$ ,  $B_p$  or  $C_p$  for initial values of  $p$ . It is immediate that those of  
 521 the form  $ca^k$  or  $ba^k$  do cluster, while the  $C_p$ ,  $\lambda_1 + 1 \leq p \leq \lambda_2 - 1$ , are dealt with as in the proof  
 522 of Proposition 7, by changing the language and checking that  $p \leq \mu'_y$  in the new language, and all  
 523 these cluster.  $\square$

524

525 The following statements give an equivalent criterion for  $A_p$ ,  $B_p$  and  $C_p$  to cluster, which gives  
 526 more information and does not particularize any order of apparition of the rules.

527 **Corollary 15.** *With or without the assumption (Rabc),  $Z_p$  clusters if and only if at least one of the*  
 528 *three following assertions holds:*

- 529 (1) *neither the letters  $a, b, c$  nor the letters  $a, c, b$  occur in the word  $D_pz$  at any increasing*  
 530 *sequence of indices,*  
 531 (2) *neither the letters  $b, a, c$  nor the letters  $b, c, a$  occur in the word  $D_pz$  at any increasing*  
 532 *sequence of indices,*  
 533 (3) *neither the letters  $c, a, b$  nor the letters  $c, b, a$  occur in the word  $D_pz$  at any increasing*  
 534 *sequence of indices.*

535 *Moreover, in the cases where  $Z_p$  clusters, when assertion (1), resp. (2), resp. (3) is satisfied, it*  
 536 *does cluster perfectly for any order for which  $a$ , resp.  $b$ , resp.  $c$ , is in the middle of  $\{a, b, c\}$ , and*  
 537 *does not cluster for any order and permutation other than those mentioned.*

538 **Proof**

539 Suppose first (Rabc) holds. Then the first result is deduced directly from Proposition 12. As for  
 540 the second one, it is a consequence of Proposition 12 in the case of assertion (3), with orders dic-  
 541 tated by Lemmas 9 and 11, and is proved in the same way in the case of assertion (1) or (2). The  
 542 other cases for the order of apparition of the rules are deduced by a permutation of the letters, after  
 543 which we get the same conclusions.  $\square$

544

545 Thus, as clustering is invariant by conjugacy, we know all the clustering AR words satisfying  
 546 the assertions of Proposition 7. We know also all the clustering AR words which are conjugate to  
 547 a power of a standard AR word, or equivalently can be de-substituted to a power of one letter by  
 548 way of the six AR morphisms, they are the powers of the standard clustering words of Proposition  
 549 12 or Corollary 13 and all their conjugates. But this does not tell which ones are in a fixed language.

550

551 In a given AR language  $\Lambda$ , we find now a clustering word, conjugate to a standard AR word,  
 552 which is longer than all the ones in Proposition 12.

553 **Proposition 16.** *Let  $\Lambda$  be an AR language satisfying (Rabc). With the notations of Theorem 10,*  
 554 *let also  $\mu$  be the stage of the first rule in the string of rules  $(z)$ ,  $z = x$  or  $z = c$ , just before stage*  
 555  *$\mu_y$ . The word  $v = S_{\mu_y}^{\mu_y - \mu + 1} M_{\mu_y}$  is a primitive perfectly clustering (for the order  $a < c < b$  or its*  
 556 *symmetric) word of  $\Lambda$  conjugate to a standard AR word. .*

557 **Proof**

558 We have  $S_{\mu_y} = Z_{\mu_y}$  and, by Lemma 5,  $L_{\mu_y} = Y_{\mu_y}$ . We define another AR language  $\hat{\Lambda}$  by the  
 559 same rules as  $\Lambda$  up to (and including) stage  $\mu_y - 1$ , then  $\mu_y - \mu + 1$  rules  $(z)$ , and any acceptable  
 560 sequence of rules beyond. Then  $\hat{\Lambda}$  has the same  $x$  and  $y$  as  $\Lambda$ , and  $v$  is conjugate to  $\hat{X}_{2\mu_y + \mu - 1}$  (of  
 561  $\hat{\Lambda}$ ) if  $z = c$ , to  $\hat{C}_{2\mu_y + \mu - 1}$  if  $z = x$ , and  $2\mu_y + \mu - 1 < \hat{\mu}_y$  (of  $\hat{\Lambda}$ ), thus by Proposition 12  $v$  is a  
 562 primitive perfectly clustering word.

563

564 It remains to prove that  $v$  is in  $\Lambda$ . We know that  $w_{\mu_y + 1} L_{\mu_y + 1} = w_{\mu_y} L_{\mu_y} M_{\mu_y} L_{\mu_y}$  is in  $\Lambda$ . We have  
 565  $\mu \geq \mu_x$ ; suppose first that either  $\mu > \mu_x$ , or  $\mu = \mu_x$  and  $\lambda_y < \lambda_x$ . Then by Lemma 5  $Z_\mu = M_\mu$ .  
 566 Thus  $S_{\mu_y} = Z_\mu$  while  $L_{\mu_y} = L_\mu Z_\mu^{\mu_y - \mu}$ ,  $M_{\mu_y} = M_\mu Z_\mu^{\mu_y - \mu}$ . Then  $v = M_\mu^{\mu_y - \mu + 1} S_\mu M_\mu^{\mu_y - \mu}$ , while  
 567  $L_{\mu_y} M_{\mu_y} = L_\mu M_\mu^{\mu_y - \mu} S_\mu M_\mu^{\mu_y - \mu}$  is in  $\Lambda$ , and all we have to prove is that  $M_\mu$  is a suffix of  $L_\mu$ , which  
 568 is true by Lemma 6 as  $\mu \geq \lambda_2 + 1$ .

569

570 Suppose now that  $\mu = \mu_x$  and  $\lambda_x < \lambda_y$ . Then  $z = x$ ,  $Z_\mu = X_\mu = L_\mu$ ,  $v = L_\mu^{\mu_y - \mu + 1} S_\mu L_\mu^{\mu_y - \mu}$ ,  
 571 while  $w_{\mu_y} L_{\mu_y} M_{\mu_y} = w_{\mu_y} M_\mu L_\mu^{\mu_y - \mu} S_\mu L_\mu^{\mu_y - \mu}$  is in  $\Lambda$ . Thus what we have to prove is that  $L_\mu u = X_\mu$   
 572 is a suffix of  $w_{\mu_y} M_\mu = w_{\mu_y} Y_\mu$ , which will be true if  $X_\mu$  is a suffix of  $w_\mu Y_\mu$ , as  $w_{\mu_y}$  ends with  $w_\mu$ .

573 Going backward through rules  $(c)$ , what we have to prove is that  $X_{\lambda_2}$  is a suffix of  $w_\mu Y_{\lambda_2}$ . Then  
 574 the rule at stage  $\lambda_2 - 1$  is  $(y)$ , thus we have to prove that  $X_{\lambda_2 - 1}$  is a suffix of  $w_\mu$ , and this is true as  
 575  $w_{\lambda_2 - 1}$  is a suffix of  $w_\mu$ , and by Lemma 6  $X_{\lambda_2 - 1}$  is a suffix of  $w_{\lambda_2 - 1}$ , except possibly if  $x = b$  and  
 576  $\lambda_2 - 1 = \lambda_1$ , which cannot happen as then the rule at stage  $\lambda_1 = \lambda_2 - 1$  should be both  $(b)$  and  
 577  $(y) = (a)$ .  $\square$

578

579 **Corollary 17.** *For every  $n$ , there exist arbitrarily long primitive (perfectly) clustering Arnoux-*  
 580 *Rauzy words with at least  $n$  occurrences of each letter.*

581 *Every Arnoux-Rauzy language contains a primitive (perfectly) clustering word of length at least*  
 582 *22.*

583 **Proof**

584 Take an AR language where  $D$  begins with  $abc^n$ . Then we get  $B_p = ba(caba)^n$  for some  $p \leq \mu_x$ .

585 For a general AR language, the smallest possible value of the length of the word in Proposition  
 586 14 is  $|S_{\mu_a} S_{\mu_a} M_{\mu_a}|$  where  $D$  begins with  $abcba$ , which gives 22.  $\square$

587

588 **Corollary 18.** *When the directive word  $D$  begins with  $ab^{n_1} c^{n_2} a^{n_3} b$  for any integers  $n_i \geq 1$ ,  $i =$   
 589  $1, 2, 3$ , the word of Proposition 14 has maximal length among clustering words of  $\Lambda$ , and the bound  
 590 in Theorem 10 is optimal. Assuming (Rabc), in all other cases, there is a gap between the length  
 591 of the word in Proposition 14 and the bound in Theorem 10.*

592 **Proof**

593 We look at the proof of Proposition 14. In the case where either  $\mu > \mu_x$ , or  $\mu = \mu_x$  and  $\lambda_y < \lambda_x$ ,  
 594 there is always a difference of at least 2 between the best bound in Theorem 10 and the length of  
 595 the word in Proposition 14. In the case where  $\mu = \mu_x$  and  $\lambda_x < \lambda_y$ , this difference is reduced to 1

596 whenever  $X_{\lambda_2} = w_{\lambda_2-1}Y_{\lambda_2}$ , which is equivalent to  $D$  being as in the assertion above.  $\square$   
 597

598 **Example 1.** Take the Tribonacci language or any AR language where the directive word  $D$  begins  
 599 with  $abcab$ . The  $A_p, B_p$  or  $C_p$  which cluster (perfectly) for the order  $a < c < b$  (or its symmetric)  
 600 are  $A_0$  to  $A_4, B_0$  to  $B_3, C_0$  to  $C_4 = cabaabacaba$ , of length 11, the longest standard word which  
 601 clusters, while  $B_4 = bacabaabacaba$ , of length 13, is the shortest standard one which does not  
 602 cluster. Also,  $A_0$  to  $A_2, B_0$  to  $B_3, C_0$  to  $C_3$  cluster (perfectly) for the order  $a < b < c$  (or its  
 603 symmetric),  $A_0$  to  $A_2, B_0$  to  $B_2, C_0$  and  $C_1$  cluster (perfectly) for the order  $b < a < c$  (or its  
 604 symmetric).

605 In the notations of Theorem 10  $\lambda_a < \lambda_b, x = a, y = b$ , and the bound is 26. The word in  
 606 Proposition 14 is  $v_1 = aB_4C_4 = aL_4M_4 = (abacaba)^2cabaabacaba$ , of length 25, which thus is  
 607 conjugate to a standard AR word. It is a palindrome, and we check that it is the only clustering  
 608 word of maximal length in  $\Lambda$ . We are in one of the cases where the bound is sharp.

609 Another clustering word is the non-primitive  $v_2 = A_3^3 = (abacaba)^3$ , of length 21, which is  
 610 conjugate to a power of standard. For  $v = v_1$ , or  $v = v_2, vv$  is not in the Tribonacci language.

611 **Example 2.** Take any AR language  $\Lambda$  where  $D$  begins with  $abacba$ . In the notations of Theorem 10  
 612 we have  $\lambda_b < \lambda_a, x = b, y = a$  and we are in the first case of the proof.

613 The bound in Theorem 10 is 45, and we look at words of length 44. By the reasoning of  
 614 Theorem 10  $bw_{\lambda_2}c$  and  $cw_{\lambda_2}b$  must occur in all words of this length; the assertion  $OT_{\mu_a+1}$  above  
 615 holds for the singular word  $aw_{\lambda_2-1}a$ , and, by looking precisely at its occurrences, we check that  
 616 the only word in  $\Lambda$  of length 44 without  $aw_{\lambda_2}c$  and without  $cw_{\lambda_2}a$  is  $v = w_{\lambda_a}L_{\mu_a}M_{\mu_a}$ , namely  $v =$   
 617  $abaaba(cabaabababacabaaba)^2$ . We check that  $v$  cannot cluster for any order and permutation,  
 618 by hand or by noticing that the four extensions  $aw_{\lambda_2}c, bw_{\lambda_2}c, cw_{\lambda_2}a, cw_{\lambda_2}b$  appear in  $vv$  as do  
 619 indeed  $cw_{\lambda_2}c$  and  $aw_{\lambda_2}a$ , thus  $v$  does not cluster and  $v$  is not conjugate to a standard AR word.  
 620 Thus no word of length 44 can cluster and *the bound in Theorem 10 is not optimal in the first case*  
 621 *of the proof.*

622 The word in Proposition 14 has length 43. It is indeed  $v$  deprived of its first letter, and is a  
 623 clustering word of maximal length in  $\Lambda$ ; it is not the only one, as its reverse is also clustering.

624 **Example 3.** Take any AR language  $\Lambda$  where  $D$  begins with  $abcba$ . The bound in Theorem 10 is  
 625 24, and we are in the second case of the proof. By the same reasoning as in Example 2 the only  
 626 word of length 23 which might cluster is  $v' = a(bacaba)^2cababacaba$ , and we check that  $v'$  cannot  
 627 cluster for any order and permutation, nor can any word of length 23, and *the bound in Theorem*  
 628 *10 is not optimal in the second case of the proof.*

629 Indeed  $v'$  is  $L_{\mu_a}M_{\mu_a}$  thus is conjugate to  $L_{\mu_a+1}$ , hence by Proposition 7  $v'$  is conjugate to a  
 630 standard AR word, but does not cluster. Thus the word in Proposition 14, which is  $v$  deprived of  
 631 its first letter and has length 22, is a clustering word of maximal length in  $\Lambda$ ; it is not the only one,  
 632 as its reverse is also clustering.

633 **Conjecture 1.** *In a given AR language  $\Lambda$  satisfying  $(Rabc)$ , the word in Proposition 14 is the*  
 634 *longest clustering word, or, if this fails, the longest clustering word conjugate to a standard AR*  
 635 *word.*

636 **3.5. Clustering AR words non conjugate to standard.** We turn now to words which are not  
 637 conjugate to standard AR words.

638 **Example 4.** For all  $n, ba(ca)^nb$  is an AR word not conjugate to a standard AR word, and it is  
 639 perfectly clustering.



640 The following propositions characterize, in two steps, all the words having this property, by  
 641 identifying the particular way they are generated in the general construction of [19].

642 **Proposition 19.** *Let  $w$  on the alphabet  $\mathcal{A} = \{a, b, c\}$  be a perfectly clustering AR word which  
 643 is not in the range of any of the six AR morphisms  $\tau_x$  and  $\sigma_x$  for  $x \in \{a, b, c\}$ . Then, up to a  
 644 permutation of the letters  $a, b$  and  $c$ , there exists a word  $v$  on the alphabet  $\{a, c\}$  containing both  
 645  $a$  and  $c$  such that the conjugate  $w' = b^{-1}wb$  is obtained from  $\tau_v(b)$  by inserting a single  $b$  between  
 646 each pair of consecutive occurrences of  $a$  or between each pair of consecutive occurrences of  $c$  in  
 647  $\tau_v(b)$  (where at most one of  $aa$  and  $cc$  can occur) plus a  $b$  at the very beginning. Furthermore,  $w$  is  
 648 a palindrome beginning and ending in  $b$  containing both  $a$  and  $c$  but no  $a^2$  nor  $c^2$ , is primitive, and  
 649 any order for which  $w$  is perfectly clustering has  $b$  as the middle letter.*

650 *Conversely, any word  $w$  built as above is a perfectly clustering AR word for the order  $a < b < c$ ,  
 651 not in the range of any of the six AR morphisms.*

652 **Proof**

653 Let  $w$  be as in the first sentence. Then  $w$  is not a power of a single letter and  $|w| \geq 5$ . Let  $a$  denote  
 654 the separating letter of  $w$ . Since  $w$  is not in the range of  $\tau_a$  nor  $\sigma_a$ ,  $w$  begins and ends in some letter  
 655 different from  $a$ . Let  $b$  denote the first letter of  $w$ . Then  $w$  also ends in  $b$ , because otherwise  $cb$  is in  
 656  $\Lambda_w$  but not  $bc$ , while  $\Lambda_w$  is closed under reversal by Proposition ???. Also, since  $a$  is the separating  
 657 letter of  $w$ ,  $bb$  does not occur in  $w$ , although it occurs in  $\Lambda_w$ . Thus  $w$  is a palindrome, as  $\bar{w}$  must  
 658 be conjugate to  $w$ , but the only conjugate of  $w$  which does not contain  $bb$  is  $w$  itself. Also,  $w$  is  
 659 primitive: if  $w = v^n$  for some  $n \geq 2$ , then as  $v$  must begin and end in  $b$ ,  $bb$  is a factor of  $w$ , a  
 660 contradiction.

661

662 **Claim 1 :** Each letter of  $\mathcal{A}$  must occur in  $w$ .

663 By assumption each of  $a$  and  $b$  occurs in  $w$ . If  $c$  does not occur in  $w$ , then  $w$  is a perfectly clus-  
 664 tering binary palindrome of the form  $w = bub$  where  $u$  begins and ends in the letter  $a$ . Furthermore  
 665  $w$  cannot contain a factor of the form  $ba^n b$  with  $n > 1$ , for otherwise  $\Lambda_w$  contains both  $aa$  and  
 666  $bb$ , which contradicts Theorem 1. Moreover, as  $bb$  does not occur in  $w$ , it follows that any two  
 667 consecutive occurrences of  $b$  in  $w$  must be separated by a single  $a$ . Thus  $w = ba(ba)^n b$  for some  
 668  $n \geq 0$  and hence  $w = \tau_b(a^{n+1}b)$  contradicting that  $w$  is not in the range of  $\tau_b$ .

669

670 As  $w$  is perfectly clustering and  $bb \in \Lambda_w$ , by Theorem 1  $b$  must be the middle letter under any  
 671 (perfectly) clustering order on  $A$ , and furthermore  $aa$  does not occur in  $w$ . Let us consider the  
 672 conjugate  $w' = b^{-1}wb$ . Note that  $w'$  begins in  $a$  and ends in  $abb$ .

673

674 **Claim 2 :**  $w' = \psi(\tau_v(b))$  for some word  $v$  on the alphabet  $\{a, c\}$  beginning in  $a$  and containing  
 675  $c$  and where the mapping  $\psi$ , defined in Section 3 of [19], amounts to inserting a single  $b$  in the  
 676 middle of each occurrence of  $aa$  and  $ab$  in  $\tau_v(b)$ .  $\tau_u(w)$  is a perfectly clustering AR word for  
 677 the order  $a < b < c$ . Furthermore, every conjugate of  $\tau_u(w)$  different from  $\tau_u(w)$  is not an AR  
 678 word and hence in particular,  $\tau_u(w)$  is not conjugate to a standard AR word. We note that  $w'$  is  
 679 a perfectly clustering word (for the order  $a < b < c$ ) and no conjugate of  $w'$  is in the range of  
 680  $\tau_a$  nor  $\tau_c$ . In fact, every conjugate of  $w'$  (other than  $w$ ) contains  $bb$  as a factor, hence is not in the  
 681 range of  $\tau_a$  nor  $\tau_c$ , and by assumption the same is true for  $w$ . By application of Lemmas 3.7 and  
 682 3.8 of [19],  $w' = \psi(ub)$  where  $u$  is a word on the alphabet  $\{a, c\}$  and  $ub$  is also perfectly clustering  
 683 relative to the order  $a < b < c$ . Note that if  $b$  occurred in  $u$ , then  $w$  would admit an occurrence  
 684 of  $bb$  contrary to our assumption that  $a$  is the separating letter of  $w$ . Thus  $ub$  contains both  $a$  and  $c$

685 and  $\Lambda_{ub}$  contains each of  $ab, ba, ac, ca$ . It follows from Lemma 3.1 of [19] that  $cc$  is not in  $\Lambda_{ub}$  and  
 686 hence each occurrence of  $c$  in  $ub$  must be directly preceded and followed by the letter  $a$ . In other  
 687 words,  $a$  is a separating letter of  $ub$  and  $w'$  is obtained from  $ub$  by inserting a single  $b$  in the middle  
 688 of each  $aa$  occurring in  $u$ , plus an additional  $b$  at the end. Also  $u$  is a palindrome.

689 By an iterated application of Lemma 3.4 in [19], we can write  $ub = \tau_a^r(u'b)$  for some  $r \geq 1$  and  
 690  $u'$  on the alphabet  $\{a, c\}$  beginning in  $c$  and furthermore  $u'b$  is perfectly clustering for  $a < b < c$ .  
 691 Thus  $u'$  is also a palindrome and hence  $bc$  and  $cb$  are both in  $\Lambda_{u'b}$ . If the letter  $a$  does not occur in  
 692  $u'$ , then we can write  $u'b = c^s b = \tau_{c^s}(b)$  for some  $s \geq 1$  and hence

$$w' = \psi(ub) = \psi(\tau_{a^r}(u'b)) = \psi(\tau_{a^r} \circ \tau_{c^s}(b)) = \psi(\tau_{a^r c^s}(b))$$

693 as required. On the other hand, if  $a$  occurs in  $u'$ , then it follows from Lemma 3.1 in [19] that each  
 694  $a$  in  $u'$  must be preceded and followed by the letter  $c$  and thus  $c$  is a separating letter of  $u'b$ . Thus  
 695 similarly we can write  $u'b = \tau_c(u''b)$  where  $u''b$  is perfectly clustering for the order  $a < b < c$  and  
 696  $u''$  is a palindrome beginning and ending in  $a$  or  $c$ . Continuing in this way, we eventually obtain  
 697 that  $ub = \tau_v(b)$  for some word  $v$  on the alphabet  $\{a, c\}$  containing each of  $a$  and  $c$ .

698  
 699 In the other direction, let  $v$  be a word on the alphabet  $\{a, c\}$  containing each of  $a$  and  $c$ . Without  
 700 loss of generality, we may assume that  $v$  begins in the letter  $a$ . It follows from Lemmas 3.3 and 3.4  
 701 of [19] that  $\tau_v(b)$  is perfectly clustering for the order  $a < b < c$ . Also clearly  $\tau_v(b)$  is an AR word.  
 702 By definition,  $w' = \psi(\tau_v(b))$  where  $\psi$  is the mapping defined in Section 3 of [19]. It follows from  
 703 Lemma 3.7 of [19] that  $w'$ , and hence  $w$ , is perfectly clustering for the order  $a < b < c$ . It remains  
 704 to show that  $w$  is an AR word which is not in the range of any of the six AR morphisms. We first  
 705 note that by definition  $w$  begins and ends in  $b$  and hence can only be in the range of  $\tau_b$  or  $\sigma_b$ . But  
 706 as  $w$  contains  $ac$  neither is possible. Finally it is easily verified that  $b^{-1}wb^{-1} = \tau_{av'}(a)$  where  $v'$  is  
 707 obtained from  $a^{-1}v$  by exchanging the letters  $a$  and  $b$  and keeping  $c$  fixed. It follows that  $b^{-1}wb^{-1}$   
 708 is a bispecial AR word from which it follows that  $w$  is an AR word as required.  $\square$

709

710 We note that the shortest  $w$  verifying the assumptions of Proposition 17 is (up to a permutation  
 711 of the letters)  $bacab$ , built from  $\tau_{ac}b$ .

712 **Proposition 20.** *Assume  $v$  is a perfectly clustering AR word which is not conjugate to a standard*  
 713 *AR word, nor to any power of a standard AR word. Then up to a permutation of the letters,  $v$*   
 714 *is conjugate to a word of the form  $\tau_u(w)$  where  $u$  (possibly empty) is on the alphabet  $\{a, c\}$  and*  
 715 *where  $w$  is as in Proposition 17.*

716 *Conversely, let  $v$  be as in the previous sentence; then it is an AR word perfectly clustering for*  
 717 *the order  $a < b < c$ . Furthermore, every conjugate of  $v$  different from  $v$  is not an AR word and*  
 718 *hence  $v$  is not conjugate to any power of a standard AR word.*

719 **Proof**

720 By Proposition 7, an AR word is conjugate to a power of a standard if and only if it can be de-  
 721 substituted to a power of one letter by way of the six AR morphisms. This means that otherwise  
 722 we can write  $v = f(w)$  where  $f$  (possibly the identity) is some concatenation of AR morphisms,  
 723 and where  $v$  is not in the range of any of the six AR morphisms. Assume  $w$  is perfectly clustering  
 724 for the order  $a < b < c$ , then  $f$  cannot involve  $\tau_b$  nor  $\sigma_b$  and hence is a concatenation of  $\{\tau_a, \tau_c,$   
 725  $\sigma_a, \sigma_c\}$ ; replacing  $v$  by a conjugate, we can get that  $f$  is a concatenation of  $\tau_a, \tau_c$ . By Lemmas 3.3  
 726 and 3.4 in [19], we have that  $v$  is also perfectly clustering for  $a < b < c$ .

727

728 In the other direction, we begin by considering the case when  $u$  is empty. It follows from  
 729 Proposition 17 that  $v$  is perfectly clustering AR word for the order  $a < b < c$ . Furthermore  $v$   
 730 contains each of  $a$ ,  $b$  and  $c$ , begins and ends in  $b$  and has either  $a$  or  $c$  as a separating letter. Now let  
 731  $v'$  be a conjugate of  $v$  with  $v' \neq v$ . Then  $v'$  contains  $bb$  and either  $ac$  or  $ca$  or both and hence  $v'$  is  
 732 not an AR word.

733 Now assume that  $u$  is a non-empty word on the alphabet  $\{a, c\}$ . The properties of  $v$  being an  
 734 AR word containing each letter, and no conjugate  $v' \neq v$  being an AR word, are clearly stable  
 735 under application of  $\tau_u$ . By Lemmas 3.3 and 3.4 in [19], the clustering property is also stable by  
 736 application of  $\tau_u$ .  $\square$

737

738 However, there are also infinitely many primitive AR words which cluster but not perfectly, and  
 739 hence are not conjugate to a standard AR word.

740 **Example 5.** The word  $w = abaca$ , which belongs to the Tribonacci language, does cluster for  
 741 the order  $a < c < b$  (and no other one), for the permutation  $\pi a = c$ ,  $\pi b = a$ ,  $\pi c = b$  (thus not  
 742 perfectly), and in the language  $\Lambda_w$  the bispecial  $a$  is resolved by  $caa$ ,  $aab$ ,  $bac$ , thus the bispecials  
 743 of  $\Lambda_w$  are not in any AR language, though they all satisfy the order condition. Thus  $abaca$  is  
 744 not conjugate to a standard AR word, and it is clustering but not perfectly clustering. The same  
 745 properties are shared by  $aba^n ca^n$  for all  $n$ .

746 **Question 1.** *What are the primitive AR words which cluster but not perfectly?*

747 As for general clustering words on three letters, they are characterized in [19] for the symmetric  
 748 permutation, and in [14] for all permutations, and are not always AR words.

749 **Proposition 21.** *For any order and any permutation  $\pi$  different from the identity, there are infinitely  
 750 many  $\pi$ -clustering words on three letters which are not AR words.*

751 **Proof**

752 We fix an order and a permutation. By [14], any word  $w$  such that  $ww$  is in the language  $\Lambda$  cor-  
 753 responding to an interval exchange transformation built with this order and this permutation, and  
 754 satisfying the *i.d.o.c. condition*, is clustering for this order and this permutation. Such a  $\Lambda$  is uni-  
 755 formly recurrent, and contains infinitely many squares, by [12] if  $\pi$  is the symmetric permutation,  
 756 [10] in general, thus there are infinitely many such clustering words  $w$ . If a word  $w$  in  $\Lambda$  contains  
 757 all the extensions  $xvy$  of all bispecial words  $v$  in  $\Lambda$  longer than some constant,  $w$  cannot be an AR  
 758 word. And this will be true for any long enough clustering word  $w$  in  $\Lambda$ .  $\square$

759

#### 4. RELATED LANGUAGES

760 4.1. **Sturmians.** On two letters  $a$  and  $b$ , as is shown in [1], the Sturmian languages of [18] can be  
 761 generated by words  $A_k$  and  $B_k$ , which are called *standard* Sturmian words, using AR-type rules  
 762 on two letters. Each Sturmian language contains infinitely many clustering words, and all these are  
 763 known since [17] and [16]. For sake of completeness, we reprove this result by the methods of the  
 764 present paper.

765 **Proposition 22.** *The primitive clustering Sturmian words, as well as the primitive clustering words  
 766 on  $\{a, b\}$ , are all the standard Sturmian words and all their conjugates.*

767 **Proof**

768 Note first that the only clustering words with the identity as permutation are the  $a^m$  and  $b^m$ , thus

769 we can restrict ourself to perfectly clustering words, for the order  $a < b$ . The Sturmian languages  
 770 are identified in [18] with 2-interval exchange languages, thus we deduce from [14] that a primi-  
 771 tive Sturmian word  $v$ , or a primitive word  $v$  on  $\{a, b\}$ , is clustering iff  $vv$  is a factor of a Sturmian  
 772 language. This, by the same proof as Proposition ??, is equivalent to  $v$  conjugate to some  $A_p$  or  
 773  $B_p$  in some Sturmian language.  $\square$

774

775 Note that, for Sturmian languages or more generally for interval exchange languages, the neces-  
 776 sary and sufficient condition for  $v$  to cluster in Theorem 1 is, by [11], equivalent to the one given  
 777 in [14], namely that  $vv$  be a factor of such a language.

778 However, to determine if a given word clusters, our Theorem 1 is more explicit. Take for ex-  
 779 ample the two Sturmian words  $v = abaa$  and  $v' = baab$ ,  $v$  and  $v'$  are factors of the *Fibonacci*  
 780 language, while  $vv$  and  $v'v'$  are not in this language. It is easy to check by hand that  $v$  clusters (for  
 781  $a < b$  and the symmetric permutation) and  $v'$  does not cluster for any order and permutation; thus  
 782 we know that  $vv$  must be in some Sturmian language and  $v'v'$  cannot be in any Sturmian language,  
 783 but it is easier, and quicker in general than computing the Burrows-Wheeler transform, to check  
 784 directly that the bispecials in  $vv$  satisfy the order condition and those in  $v'v'$  do not; in this last  
 785 case, it is immediate that no order condition can be satisfied by the empty bispecial, as  $aa$ ,  $bb$ ,  $ab$ ,  
 786  $ba$  occur in  $v'v'$ .

787 **4.2. Episturmians on three letters.** In the literature, for which we refer the reader to the two  
 788 surveys [2] and [15], we found only the definition of episturmian infinite words, one-sided in  
 789 general though two-sided words are briefly considered in [15]. To make the present paper coherent,  
 790 we define here the corresponding languages, our definition having been chosen to correspond to  
 791 what is used in practice.

792 **Definition 3.** *A language is episturmian if it is uniformly recurrent, closed under reversal, and*  
 793 *admits at most one right special factor of each length.*

794 An episturmian language on three letters can be generated by AR rules or by AR morphisms,  
 795 as is proved in the founding paper [9]. Indeed, these episturmian languages can be defined by  
 796 a modification of Definition 2 above, where the assumption “each one of the three rules is used  
 797 infinitely many times” is replaced by “each one of the three rules is used at least once”.

798 The description of the bispecial words is deduced from the one given after Definition 2 by  
 799 the following modifications: the possible bispecials are among the  $w_k$ , and  $w_k$  has at most three  
 800 suffix return words which are among  $A_k$ ,  $B_k$  and  $C_k$ . More precisely,  $A_k$  is a return word of  $w_k$ ,  
 801 or equivalently the label of an elementary circuit in the Rauzy graphs, if and only if  $w_k A_k$ , or  
 802 equivalently  $w_k a$ , is in  $\Lambda$ , and similarly for  $B_k$  and  $C_k$ .

803 **Lemma 23.** *The word  $w_p A_p$  is in an episturmian language  $\Lambda$  on three letters if and only if the*  
 804 *directive word of  $\Lambda$  is such that there exist rules (a) at or after stage  $p$ , and similarly for  $B_p$  and*  
 805 *rules (b),  $C_p$  and rules (c).*

806 **Proof**

807 Suppose for example there is a rule (a) at or after stage  $p$ . Then, for some  $q \geq p$ ,  $w_{q+1} = w_q A_q$ ,  
 808 thus  $w_q A_q$  is in  $\Lambda$ , and so is  $w_p A_p$  as  $w_p$  is a suffix of  $w_q$  and  $A_p$  is a prefix of  $A_q$ .

809 Suppose there is no rule (a) at or after stage  $p$ . Then  $w_p A_p$  cannot be in  $B_r$  or  $C_r$  as  $B_r$  and  $C_r$   
 810 do not have  $A_p$  in their decomposition by AR rules. As there are infinitely many rules (b) or (c),  
 811 the length of  $B_r$  or  $C_r$  tends to infinity with  $r$ , thus this contradicts uniform recurrence.  $\square$

812

813 The assertion  $(Rabc)$  is defined as for AR languages.

814 **Theorem 24.** *An episturmian language  $\Lambda$  on three letters, satisfying  $(Rabc)$ , contains infinitely*  
 815 *many clustering words if and only if its directive word is  $D'D''$ , where  $D'$  is a finite word on the*  
 816 *alphabet  $\{a, b\}$  and  $D''$  is a one-sided infinite word on the alphabet  $\{a, c\}$  or  $\{b, c\}$ .*

817 *When this is not the case, with the notations of Theorem 10, no word of length at least*

818  $|w_{\lambda_2}| + \max_{Z \in \{A, B, C\}, w_{\mu_y+1} Z_{\mu_y+1} \in \Lambda} |Z_{\mu_y+1}| + 1$

819 *can cluster for any permutation and any order.*

820 **Proof**

821 Suppose  $D$  is  $D'D''$  as in the hypothesis. Let  $v$  be any word such that  $vv$  is in  $\Lambda$ . The bispecials  
 822 in the language  $\Lambda_v$  are some of the  $w_p$  of  $\Lambda$ , and are resolved as in the AR rules, possibly with  
 823 some extensions missing. By Lemma 11, those which are resolved as in rule (a) or (b) satisfy the  
 824 order condition for  $a < c < b$ . For those  $w_p$  which are resolved as in rule (c), there is no rule (z)  
 825 on or after stage  $p$  for  $z = a$  or  $z = b$ , thus by Lemma 21  $w_p z$  is not in  $\Lambda$ , and  $w_p$  is resolved by  
 826 a subset of  $\{aw_p c, cw_p a, cw_p c\}$  or  $\{bw_p c, cw_p b, cw_p c\}$ , thus satisfies the order condition. Thus by  
 827 Lemma 9  $v$  clusters perfectly. To find such  $v$ , we follow the reasoning of Proposition ???: any  $Z_p$  in  
 828  $\{A_p, B_p, C_p\}$  will have its square in  $\Lambda$  provided  $p$  is large enough and  $w_p Z_p$  is in  $\Lambda$ . If each letter  
 829 in  $D''$  occurs infinitely many times, we get arbitrarily long primitive clustering words; otherwise  
 830 all the  $Z_p^n$  will be clustering for one value of  $Z$ .

831 Suppose  $D$  is not  $D'D''$  as in the hypothesis. Then after the first rule (c) there is at least one  
 832 rule (a) and one rule (a). Thus all the quantities in Theorem 10 can be defined, and we can follow  
 833 the reasoning of this theorem, with the following modifications: the assertion  $(OA_q)$  is now that  $v$   
 834 occurs once in  $w_q A_q$ , does not occur in  $w_q C_q$  or  $w_q B_q$ , and  $w_q A_q$  is in  $\Lambda$ , and similarly for  $(OB_q)$   
 835 and  $(OC_q)$ ; the assertion  $(OT_q)$  is now that  $v$  occurs at least once in each  $w_q Z_q$  which is in  $\Lambda$ ,  $Z$   
 836 in  $\{A, B, C\}$ , and the maximal return time of  $v$  is the maximal length of these  $Z_q$ . Then, we get  
 837 the maximal return times of the two special words, and conclude immediately as, in contrast with  
 838 Theorem 10, we keep the quantity  $|w_{\lambda_2}|$  in the bound.  $\square$

839

840 **Example 6.** Let  $D = abc(ab)^\omega$ . This gives an episturmian language which contains only finitely  
 841 many clustering words, but is not an AR language. Its complexity function is  $p(n) = 2n + 1$  for  
 842  $1 \leq n \leq 4$ ,  $p(n) = n + 5$  for  $n \geq 5$ .

843 Note that these properties are shared by all episturmian languages where  $D = abcD''$  where  
 844  $D''$  is a one-sided infinite word on  $\{a, b\}$  in which both  $a$  and  $b$  occur infinitely many times. By  
 845 Theorem 22, any episturmian language whose complexity is at least  $n + 1$  for all  $n$  but is strictly  
 846 smaller than the  $p(n)$  of Example 6 produces infinitely many clustering words. One can wonder  
 847 whether this is still true for any language, or at least for any uniformly recurrent language, of  
 848 complexity at least  $n + 1$  for all  $n$  but strictly smaller than this  $p(n)$ . For sake of completeness, we  
 849 give a (non episturmian) counter-example.

850 **Example 7.** We build a language  $\Lambda$  on  $\{a, b, c\}$  in the following way: the empty bispecial word is  
 851 resolved by  $\{ab, ac, ba, ca\}$ ; the bispecial  $a$  is resolved by  $\{bab, bac, cab, cac\}$ ; the bispecial  $aba$   
 852 is resolved by  $\{babab, babac, cabab\}$ ; the bispecial  $aca$  is resolved by  $\{bacac, cacab\}$ ; every further  
 853 bispecial  $w$  is resolved either by  $\{bwb, bwc, cw b\}$  or by  $\{bwc, cw b, cwc\}$ , each possibility being  
 854 used infinitely many times. Its complexity function is  $p(1) = 3$ ,  $p(2) = 4$ ,  $p(3) = 6$ ,  $p(n) = n + 4$   
 855 for  $n \geq 4$ . Its Rauzy graphs of length 4 and more have the same shape as the Rauzy graphs of  
 856 Sturmian languages, thus the alternating of resolution rules ensures that  $\Lambda$  is uniformly recurrent.

857 But the bispecial word  $a$  does not satisfy the requirement of the order condition, for any order and  
 858 permutation, and its four extensions  $bab, bac, cab, cac$  occur in every long enough factor of  $\Lambda$ . Thus  
 859 by Theorem 1  $\Lambda$  contains only finitely many clustering words.

860 **4.3. Larger alphabets.** AR languages can be generalized to any alphabet  $\mathcal{A} = \{a_1, a_2, \dots, a_r\}$   
 861 (note that here the order will not necessarily be  $a_1 < a_2 < \dots < a_r$ ).

862 **Definition 4.** An AR language is generated by words  $A_k^{(i)}$ ,  $1 \leq i \leq r$ , starting from  $A_0^{(i)} = a_i$ ,  
 863  $1 \leq i \leq r$ , and by rule  $(a_i)$  at stage  $k$ ,  $A_{k+1}^{(i)} = A_k^{(i)}$ ,  $A_{k+1}^{(j)} = A_k^{(j)} A_k^{(i)}$  for all  $i \neq j$ .

864 Each rule is used infinitely many times. The directive word is defined in the usual way. The  
 865  $A_k^{(i)}$  are again the labels of the elementary circuits in the Rauzy graphs. The  $r$ -Bonacci language,  
 866  $r \geq 3$ , is defined by  $D = (a_1 \dots a_r)^\omega$ .

867 There the methods of Section 3 apply mutatis mutandis, but the number of cases to be consid-  
 868 ered grows very quickly, and a lot of space would be required to generalize all the above study.  
 869 Thus we shall just generalize Theorem 10, with some loss of optimality.

870

871 **Proposition 25.** We denote by  $(a)$ ,  $(b)$ ,  $(c)$  the first three rules by order of appearance, and define  
 872 all quantities in Theorem 10 above in the same way. Let  $v$  be an AR word on an  $r$ -letter alphabet  
 873 of length at least  $|w_{\lambda_2}| + \max_{1 \leq i \leq r} (|A_{\mu_y+1}^{(i)}|) + 1$ . Then  $v$  cannot cluster for any permutation and  
 874 any order on the alphabet.

875 For 4-Bonacci, this bound is not optimal; the better bound  $|w_{\lambda_y}| + \max_{1 \leq i \leq r} (|A_{\mu_y+1}^{(i)}|) + 1$  holds  
 876 but is not optimal either.

877 **Proof**

878 In this case Lemma 9 is still valid, by restricting the orders on  $\mathcal{A}$  to the set  $\{a, b, c\}$ , and again we  
 879 need to know the maximal return times of  $u' = xw_{\lambda_x}x$  and  $u = yw_{\lambda_y}y$ . These are computed exactly  
 880 as in Theorem 10, mutatis mutandis: the assertions are now  $(O_q^{(j)})$ , that  $v$  occurs once in  $w_q A_q^{(j)}$  and  
 881 does not occur in any  $w_q A_q^{(i)}$ ,  $i \neq j$  ( $a, b, c$  being denoted also by  $a_1, a_2, a_3$ ), and  $OT_q$ , that  $v$  occurs  
 882 at least once in each  $w_q A_q^{(i)}$  and the maximal return time of  $v$  is  $\max_{1 \leq i \leq r} (|A_q^{(i)}|)$ . These evolve  
 883 like in the proof of Theorem 10, and thus the maximal return time of  $u$  is  $\max_{1 \leq i \leq r} (|A_{\mu_y}^{(i)}|)$ , the  
 884 maximal return time of  $u'$  is  $\max_{1 \leq i \leq r} (|A_{\mu_x}^{(i)}|)$ . We conclude immediately as we keep the quantity  
 885  $|w_{\lambda_2}|$  in the bound.

886 For 4-Bonacci, if we denote the letters by  $a, b, c, d$ , we have  $\lambda_2 = 2$ ,  $w_3 = abacaba$ ,  $w_2 = aba$ ,  
 887  $w_1 = a$ ,  $w_0$  is the empty word. The bound in the conclusion is  $|C_6| + 4 = 60$ , but in this sim-  
 888 ple case we can mimic the end of the proof of Theorem 10 and replace the  $|w_{\lambda_2}|$  in the bound  
 889 by  $|w_{\lambda_y}|$ , thus getting  $|C_6| + 2 = 58$ . As  $C_6 = C_5 B_5$ , by the usual reasoning the only word of  
 890 length  $|C_6| + 1$  which does not contain  $bw_{\lambda_2}c$  nor  $cw_{\lambda_2}b$  is, up to cyclic conjugacy,  $u = aB_5C_5$ .  
 891 In the language  $\Lambda_u$ , we check that  $w_0$  is resolved by  $\{aa, ab, ac, ad, ba, ca, da\}$ ,  $w_2$  is resolved by  
 892  $\{cw_2c, aw_2c, cw_2a, dw_2c, cw_2d\}$ ,  $w_3$  is resolved by  $\{aw_3d, dw_3a, dw_3c, cw_3d\}$ , thus, if  $u$  clusters,  
 893 by Lemma 4 each one of  $a, c, d$  must be at an end of the order between them, thus no word of  
 894 length 57 can cluster for any order and permutation.  $\square$

895

896 In the general case, we do not try to replace the  $|w_{\lambda_2}|$  in the bound by  $|w_{\lambda_y}|$  as the proof would  
 897 be complicated by the presence of rules  $(a_i)$ ,  $i \geq 4$ , between stages  $\lambda_2$  and  $\mu_y$ , and the improved  
 898 bound is not optimal even for 4-Bonacci.

899

900 Similarly, the main result of Section 4.3 can be generalized to episturmian languages on larger  
 901 alphabets: an episturmian language  $\Lambda$  on  $r$  letters contains infinitely many clustering words if and  
 902 only if, up to a permutation of letters, its directive word is  $D^{(1)}D^{(2)} \dots D^{(r-1)}$ , where  $D^{(1)}$  is a  
 903 finite word on the alphabet  $\{a_1, a_2\}$ ,  $D^{(2)}$  is a finite word on the alphabet  $\{a_3, x_3\}$  with  $x_3 = a_1$  or  
 904  $x_3 = a_2$ ,  $D^{(3)}$  is a finite word on the alphabet  $\{a_4, x_4\}$  with  $x_4 = a_3$  or  $x_4 = x_3$ , ...,  $D^{(r-2)}$  is a  
 905 finite word on the alphabet  $\{a_{r-1}, x_{r-1}\}$  with  $x_{r-1} = a_{r-2}$  or  $x_{r-1} = x_{r-2}$ ,  $D^{(r-1)}$  is a one-sided  
 906 infinite word on the alphabet  $\{a_r, x_r\}$  with  $x_r = a_{r-1}$  or  $x_r = x_{r-1}$ . This can be proved in the  
 907 same way as Theorem 22.

908

## REFERENCES

- 909 [1] P. ARNOUX, G. RAUZY: Représentation géométrique de suites de complexité  $2n + 1$  (French), *Bull. Soc. Math.*  
 910 *France* 119 (1991), p. 199–215.
- 911 [2] J. BERSTEL: Sturmian and episturmian words (a survey of some recent results), in: Algebraic informatics, p.  
 912 23–47, *Lecture Notes in Comput. Sci.* 4728, Springer, Berlin, 2007.
- 913 [3] S. BRLEK, A. FROSINI, I. MANCINI, E. PERGOLA, S. RINALDI: Burrows-Wheeler transform of words  
 914 defined by morphisms, in: Combinatorial Algorithms, IWOCA 2019, p. 393–404, *Lecture Notes in Computer*  
 915 *Science* 11638, Springer, Cham, 2019.
- 916 [4] S. BRLEK, S. LI: On the number of squares in a finite word, arXiv:2204.10204.
- 917 [5] M. BURROWS, D.J. WHEELER: A block-sorting lossless data compression algorithm, *Technical Report 124*  
 918 (1994), Digital Equipment Corporation.
- 919 [6] J. CASSAIGNE, N. CHEKHOVA: Fonctions de récurrence des suites d’Arnoux-Rauzy et réponse à une question  
 920 de Morse et Hedlund, (French) [Recurrence functions of Arnoux-Rauzy sequences and answer to a question  
 921 posed by Morse and Hedlund], in: Numération, pavages, substitutions, *Ann. Inst. Fourier (Grenoble)* 56 (2006),  
 922 p. 2249–2270.
- 923 [7] J. CASSAIGNE, S. FERENCZI, A. MESSAOUDI: Weak mixing and eigenvalues for Arnoux-Rauzy sequences,  
 924 *Ann. Inst. Fourier (Grenoble)* 58 (2008), p. 1983–2005.
- 925 [8] A. DE LUCA, M. EDSON, L.Q. ZAMBONI: Extremal values of semi-regular continuants and codings of interval  
 926 exchange transformations, *Mathematika* 69 (2023), p. 432–457.
- 927 [9] X. DROUBAY, J. JUSTIN, G. PIRILLO, Episturmian words and some constructions of de Luca and Rauzy,  
 928 *Theoret. Comput. Sci.* 255 (2001), p. 539–553.
- 929 [10] S. FERENCZI: A generalization of the self-dual induction to every interval exchange transformation, *Ann. Inst.*  
 930 *Fourier (Grenoble)* 64 (2014), p. 1947–2002.
- 931 [11] S. FERENCZI, P. HUBERT, L.Q. ZAMBONI: Languages of general interval exchange transformations, arXiv:  
 932 2212.01024.
- 933 [12] S. FERENCZI, L.Q. ZAMBONI: Structure of  $K$ -interval exchange transformations: induction, trajectories, and  
 934 distance theorems, *J. Anal. Math.* 112 (2010), p. 289–328.
- 935 [13] S. FERENCZI, L.Q. ZAMBONI: Languages of  $k$ -interval exchange transformations, *Bull. Lond. Math. Soc.* 40  
 936 (2008), p. 705–714.
- 937 [14] S. FERENCZI, L.Q. ZAMBONI: Clustering words and interval exchanges, *J. Integer Seq.* 16 (2013), Article  
 938 13.2.1, 9 pp.
- 939 [15] A. GLEN, J. JUSTIN: Episturmian words: a survey, *Theor. Inform. Appl.* 43 (2009), p. 403–442.
- 940 [16] O. JENKINSON, L. Q. ZAMBONI: Characterisations of balanced words via orderings, *Theoret. Comput. Sci.*  
 941 310 (2004), p. 247–271.
- 942 [17] S. MANTACI, A. RESTIVO, M. SCIORTINO: Burrows-Wheeler transform and Sturmian words, *Inform. Pro-*  
 943 *cess. Lett.* 86 (2003), p.241–246.
- 944 [18] M. MORSE, G.A. HEDLUND: Symbolic dynamics II. Sturmian trajectories, *Amer. J. Math.* 62 (1940), p. 1–42.
- 945 [19] J. SIMPSON, S. J. PUGLISI: Words with simple Burrows-Wheeler Transforms, *The Electronic Journal of Com-*  
 946 *binatorics* 15 (2008), Research Paper 83, 17pp.

947 (CORRESPONDING AUTHOR), AIX MARSEILLE UNIVERSITÉ, CNRS, CENTRALE MARSEILLE, INSTITUT DE  
948 MATHÉMATIQUES DE MARSEILLE, I2M - UMR 7373, 13453 MARSEILLE, FRANCE.  
949 *Email address:* `sebastien-simon.ferenczi@univ-amu.fr`

950 INSTITUT CAMILLE JORDAN, UNIVERSITÉ CLAUDE BERNARD LYON 1, 43 BOULEVARD DU 11 NOVEMBRE  
951 1918, 69622 VILLEURBANNE CEDEX, FRANCE  
952 *Email address:* `zamboni@math.univ-lyon1.fr`