A note on Thue games

Robert Mercaş[a,1], Dirk Nowotka[b]

[a] *Department of Computer Science, Loughborough University, UK*
[b] *Department of Computer Science, Kiel University, Germany*

Abstract

In this work we improve on a result from [1]. In particular, we investigate the situation where a word is constructed jointly by two players who alternately append letters to the same end of the construction. One of the players (Ann) tries to avoid (non-trivial) repetitions, while the other one (Ben) tries to enforce them. We show a construction that is closer to the lower bound of 6 showed in [2] using entropy compression while building on the probabilistic arguments based on a version of the Lovász Local Lemma from [4]. Our result provides an explicit strategy for Ann to avoid (non-trivial) repetitions over a 7-letter alphabet.

Keywords: avoidability, Thue games, nonrepetitive games

1. Preliminaries

The investigation of this paper restricts to the "nonrepetitive game" played by two players. For more details see [2]. The game is played between two players which add letters one after the other (starting from scratch) to the same end of the existing sequence. One player, called Ann, tries to avoid creation of sequences of a certain length that repeat themselves consecutively, while the other player, called Ben, has as goal their creation.

There have been quite a number of results regarding the winning strategies that Ann should employ depending on the possible length of the repeating sequence [2, 4, 1], most of which proved only the existence of such

Email addresses: `R.G.Mercas@lboro.ac.uk` (Robert Mercaş), `dn@informatik.uni-kiel.de` (Dirk Nowotka)

[1]Work supported by the P.R.I.M.E. programme of DAAD co-funded by BMBF and EU's 7th Framework Programme (grant 605728)

a strategy. However, when considering the case of sequences of length 2 or longer (nontrivial repetitions), which is the basic case given the fact that repetitions of length 2 are already created by Ben whenever he copies the symbol that Ann has previously added, deterministic approaches of such an algorithm have been explicitly provided for an alphabet of size 37 in [4] and later on improved to an alphabet of size 9 in [1].

Our strategy is to join together the two approaches used in [1], to further refine the upper bound on the alphabet size to only 7 symbols.

In this work, we deal with sequences of characters from an alphabet which we will denote by Σ. The length of a sequence is represented by the number of (non-unique) characters it contains. For a sequence x of length n, we denote by x_i the symbol on position i, where $0 < i \leq n$. A square represents a consecutive repetition of the same sequence of symbols, $w = xx$, for any $x \in \Sigma^+$. In the previous example, if x has even length, then w is the square of an even length word, while for x of odd length, w becomes the square of a word of odd length. A square-free word is any word that does not have a factor which is a square. Every square of length 2 is considered to be trivial. In other words, every consecutive occurrence of the same symbol, e.g., 00, 11, 22, for $\{0, 1, 2\} \subseteq \Sigma$, is a trivial square. All squares of length at least 4 are called non-trivial. When referring to the components of an array x, we use the notation $x[i]$ for the value stored in its ith component.

2. Results

In [1] the authors give an explicit strategy for Ann to win the game by considering two cases separately. Namely the avoidability of non-trivial squares of words with odd length and the avoidability of squares of words with even length. Since both these strategies imply the use of a ternary alphabet, the result is just a cartesian product of the two disjoint alphabets.

For the avoidability of squares of even length, the authors propose a strategy, valid using any square-free infinite word, that involves Ann using the letters of the well known square-free word attributed to Hall [3] (Hall word), in the order of their appearance. We commit to the same choice as it seems the most viable.

For the other case, they show that if Ann choses a favourite letter which she continues to repeat for as long as Ben does not make the same choice of a letter, this avoids non-trivial squares of words of odd length. If Ben choses as his next letter Ann's favourite, then Ann changes her choice of a favourite to the letter that Ben has not used in his previous two choices (the letter he just chose and the one before that). This is the part of the proposed strategy that we will slightly tweak.

Our strategy follows the same lines as that from [1], but with a refined counting. In particular, we now already consider the pairs of letters, and not only a partial alphabet for each of the situations. That is, instead of the previous approach that considered Ann playing simultaneously on both coordinates such that she avoids squares of words of odd length on the first coordinate and squares of words of even length on the second, Ann will now combine the two strategies playing the avoidability game simultaneously. Therefore, we consider pairs of letters from $\{\mathbf{0}, \mathbf{1}\} \times \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$, where the second positions in the pair will render for Ann the Hall word. Furthermore, we will consider an extra letter $(\mathbf{2}, \mathbf{d})$ which has a unique first and second component among the others. Ann will still chose a favourite letter, but only for the first position of the pair, while she will adhere to the previous restrictions imposed by the Hall words, for the second component. She will repeat the letter for the first component as long as Ben has not made the same choice for the first component, previously.

First observe that according to the previous explanations, our considered alphabet is formed of seven pairs of letters, namely:

$$\Sigma = \{(\mathbf{0}, \mathbf{a}), (\mathbf{0}, \mathbf{b}), (\mathbf{0}, \mathbf{c}), (\mathbf{1}, \mathbf{a}), (\mathbf{1}, \mathbf{b}), (\mathbf{1}, \mathbf{c}), (\mathbf{2}, \mathbf{d})\}.$$

Algorithm 1 describes the game based on Ann's strategy of choosing a favourite, when she starts the game (the other situation is similar).

Let us first explain the algorithm. Ann starts by choosing $(\mathbf{0}, \mathbf{a})$ as her favourite, and appends it. If the letter that Ben choses as first component is different from $\mathbf{0}$, then Ann only updates its second component and we append both letters. Otherwise, Ann will also update the first component to $\mathbf{1}$ (line 5). Both letters are appended. Obviously, the next update of the second component for Ann must consider the third position of the Hall word, thus we set in line 9 our counter to 3.

Now let us look at the big loop of the game, and consider all situations.

If Ann has $(\mathbf{2}, \mathbf{d})$ as favourite, then the lines 11-12 say that the next choice that she has to make for the first component must be different, while the second component represents the next considered position of the Hall word. Since the current favourite of Ann was $(\mathbf{2}, \mathbf{d})$ it implies that the previous favourite was matched by Ben in the first component. Therefore, the new

Algorithm 1 Construction of the word using Ann's strategy of favourite letter

1: Let τ be the Hall word.
2: $Ann_{\mathtt{F}} = (\mathbf{0}, \mathbf{a})$. `//`$Ann_{\mathtt{F}}$ `is the current favourite letter of Ann`
3: $Ben_{\mathtt{M}} = \mathbf{read.in}()$. `//`$Ben_{\mathtt{M}}$ `is the current choice of Ben`
4: $\mathbf{Append}(Ann_{\mathtt{F}})$. `//we add the first character of the word`
5: $Ann_{\mathtt{F}} = (1 - \lceil \frac{Ben_{\mathtt{M}}}{2} \rceil, \mathbf{b})$. `//next` $Ann_{\mathtt{F}}$ `must be different from current` $Ben_{\mathtt{M}}$
6: $\mathbf{Append}(Ben_{\mathtt{M}})$, $\mathbf{Append}(Ann_{\mathtt{F}})$. `//we add the current characters`
7: $Ben_{\mathtt{P}} = Ben_{\mathtt{M}}$. `//`$Ben_{\mathtt{P}}$ `is the previous move of Ben`
8: $Ben_{\mathtt{M}} = \mathbf{read.in}()$.
9: $count = 3$. `//recall the next character of` τ `to be used.`
10: **while** Game Played **do**
11:    **if** $Ann_{\mathtt{F}}[1] == \mathbf{2}$ **then**
12:       $Ann_{\mathtt{F}} = (1 - \lfloor \frac{Ben_{\mathtt{M}}[1]}{2} \rfloor Ben_{\mathtt{P}}[1] - Ben_{\mathtt{M}}[1](2 - Ben_{\mathtt{M}}[1]), \tau[count])$.
13:    **else if** $Ben_{\mathtt{M}} == Ann_{\mathtt{F}}$ **then**
14:       $Ann_{\mathtt{F}} = (3 - Ben_{\mathtt{P}} - Ben_{\mathtt{M}}, \lfloor \frac{Ben_{\mathtt{P}}[2]}{\mathbf{d}} \rfloor \mathbf{d} + (1 - \lfloor \frac{Ben_{\mathtt{P}}[2]}{\mathbf{d}} \rfloor) \tau[count])$.
15:    **else**
16:       $Ann_{\mathtt{F}}[2] = \tau[count]$ `//update the second component of Ann's favourite`
17:    **end if**
18:    $\mathbf{Append}(Ben_{\mathtt{M}})$, $\mathbf{Append}(Ann_{\mathtt{F}})$. `//we add the current characters`
19:    $Ben_{\mathtt{P}} = Ben_{\mathtt{M}}$, $Ben_{\mathtt{M}} = \mathbf{read.in}()$.
20:    **if** $Ann_{\mathtt{F}} \neq (\mathbf{2}, \mathbf{d})$ **then**
21:       $count = count{+}{+}$. `//increment counter only if Ann's favourite is not (2,d)`
22:    **end if**
23: **end while**

favourite will be one whose first component is different from the one of the current choice of Ben, and, moreover, different from the first component of his previous choice, whenever his current choice is $(\mathbf{2}, \mathbf{d})$.

If Ann has a favourite different from $(\mathbf{2}, \mathbf{d})$ and this is matched in the first component by the current choice of Ben, then Ann choses her current favourite such that its first component is different from the first component of the last choice of Ben that was different from $(\mathbf{2}, \mathbf{d})$ (lines 13-14). Observe that in line 14, we know that $0 \leq 3 - Ben_{\mathtt{P}} - Ben_{\mathtt{M}} < 3$ since the first component of the choice of Ben matches the one of Ann's favourite, which in turn, does not match the first component of the previous choice of Ben.

If the first component of Ann's favourite is different from $\mathbf{2}$ and it is not matched by Ben, then we only update its second component to the current considered position in the Hall word.

The loop ends by appending the current letters, updating the previous choice of Ben and reading the new one. The counter traversing the Hall word is updated only when the current favourite of Ann is different from $(\mathbf{2}, \mathbf{d})$.

We are now ready to state our result.

Theorem 2.1. *There exists a strategy with finite description that allows Ann to win the non-repetitive game of any length on an alphabet with* 7 *letters.*

Proof. Assume that following this strategy there exist two consecutive repetitions of a word with length greater than 1. Consider the following claim.

Claim. Any square-free word remains square-free after insertions of a new letter, as long as no unary squares are created.

Proof: The insertion process that is mentioned here involves expanding a word of length n to length $n+1$ such that, for an insertion between positions i and $i+1$ of the new letter, where $i \leq n$, every letter at position j in the previous word is shifted in the new word to position $j+1$, for every $j > i$, while position $i+1$ in the new sequence is occupied by the inserted symbol.

The result is trivial since if a square is created, this would remain a square after deleting all occurrences of a letter from both sides (that is, employing a projection on an alphabet of a smaller size) as this process would only remove unary squares. □

Since Ann does not have the letter $(\mathbf{2}, \mathbf{d})$ as a favourite for two consecutive rounds, the previous claim establishes that if such a word exists, then it cannot have even length. In particular, if such an even length word would exist, then this word would actually give rise to a square occurrence in the Hall word that we considered for the strategy of Ann for the second component, which would lead to a contradiction.

Let us now denote such an occurrence of word by $x_1 x_2 \cdots x_n$ for some odd integer $n > 2$. Moreover, let us denote by $i+1$ a position where such a repetition occurs in the word obtained while playing the game, namely w. Hence

$$w_{i+1} w_{i+2} \cdots w_{i+2n} = x_1 x_2 \cdots x_n x_1 x_2 \cdots x_n$$

Furthermore, observe that for every position k with $1 \leq k \leq n$ we have that $x_k[1] \in \{\mathbf{0}, \mathbf{1}, \mathbf{2}\}$ and $x_k[2] \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$.

Assume that this instance of the factor was initiated by a move of Ben. Then, following our strategy we have that

$$w_{i+2}[1] \neq w_{i+1}[1], w_{i+4}[1] \neq w_{i+3}[1], \ldots, w_{i+n-1}[1] \neq w_{i+n-2}[1],$$
$$w_{i+n+1}[1] \neq w_{i+n}[1], w_{i+n+3}[1] \neq w_{i+n+2}[1], \ldots, w_{i+2n}[1] \neq w_{i+2n-1}[1],$$

Since $x_k = w_{i+k} = w_{i+k+n}$, we conclude that $x_k[1] \neq x_{k+1}[1]$ for every integer k with $1 \leq k < n$ and $x_1[1] \neq x_n[1]$. This implies that Ann was not constrained in fact to ever change her first letter, and therefore, throughout the factor she had only one favourite. Furthermore, Ben has used only letters different from the one that Ann used. This easily leads to a contradiction as every $x_k[1]$ is once chosen by Ann, and once by Ben.

Assume now that Ann choses the letter w_{i+1}. We once more conclude that $x_k[1] \neq x_{k+1}[1]$ for every integer k with $1 \leq k < n$, but in this case $x_1[1]$ might be equal to $x_n[1]$. In particular, the letter on position $i+n+1$ is chosen by Ben, and this has to match w_{i+1}. Therefore, Ann will change once and only once in the game her choice of favourite letter. It easily follows that in this case either the length of the sequence that is repeated is at most 3, or the choice of letter for Ben, that preceded the enforced change of favourite of Ann, is $(\mathbf{2}, \mathbf{d})$. Indeed if $n > 3$ and $Ben_{\mathtt{P}} \neq (\mathbf{2}, \mathbf{d})$, then there is going to be another change of letters for Ann, which is a contradiction with the uniqueness of the change.

It is not difficult to check that there is no word of length 3 whose square appears in our construction. For the second case, observe that since the last letter that Ben chose in the first half of the square must be $(\mathbf{2}, \mathbf{d})$, namely $x_{n-1} = (\mathbf{2}, \mathbf{d})$, we conclude that the last letter that Ann choses in the second half of the repetition, must be the same. However, in order for this to happen there must exist a second change of favourite for Ann, which is again a contradiction with the uniqueness of the choice. □

We end this work with an example of how the algorithm runs, considering only the first component of the pairs of letters that the players use (the second component is the current pointer in the Hall word, which ensures that no square of an even length sequence occurs).

Example 1. *Assume that the current favourite of Ann is **0**. If Ben choses **1** or **2**, then the next letter that Ann will put down is **0** again:*

$$\cdots \boldsymbol{010}.$$

Assume now that Ben decides to repeat Ann's choice. Following lines $13-14$ *of the algorithm, Ann will chose* $\boldsymbol{2} = 3 - \boldsymbol{1} - \boldsymbol{0}$ *as a new favourite:*

$$\cdots \boldsymbol{01002}.$$

Next, regardless of the choice that Ben made in the next step, Ann will change her current favourite from $\boldsymbol{2}$ *according to line* 12*, rendering one of the following sequences:*

$$\cdots \boldsymbol{0100201},$$
$$\cdots \boldsymbol{0100210},$$
$$\cdots \boldsymbol{0100221}.$$

References

[1] J. Grytczuk, K. Kosiński, and M. Zmarz. How to play Thue games. *Theoretical Computer Science*, 582:83 – 88, 2015.

[2] J. Grytczuk, J. Kozik, and P. Micek. New approach to nonrepetitive sequences. *Random Structures & Algorithms*, 42(2):214–225, 2013.

[3] M. J. Hall. Generators and relations in groups – the Burnside problem. In *Lectures on Modern Math.*, volume 2, pages 42–92, 1964.

[4] W. Pegden. Highly nonrepetitive sequences: Winning strategies from the local lemma. *Random Structures & Algorithms*, 38(1-2):140–161, 2011.

Answer to referees’ comments:

Reviewer #1

- In the abstract, give the lower bound referenced from [2].

= Done

- The authors call Thue’s ternary squarefree word the ‘‘Thue--Morse word’’.
This usually refers to the binary overlap-free word $0110100110010110\cdots$.
I would just say ‘‘Thue’s squarefree word’’.

= Done. We have added a reference to Hall and used the syntagm ‘‘Hall word’’

- In a few places ‘‘Ben’’ is misspelled as ‘‘Benn’’.

= We fixed this

- p. 3, line -1: ‘‘match by’’ -> ‘‘matched by’’

= Done

- p. 4, after Claim: the assertion that ‘‘if such a word exists,
then it cannot have even length’’ could use some more details.
For instance, explain that if the repeated word had even length
then the subsequence of Ann’s letters within this square with
the $(2,d)$’s deleted would result in a square in the squarefree Thue word.

= We added a clarification regarding this.

- p. 4, line -10: ‘‘Assume that Ben starts’’ should be made more
precise; maybe, ‘‘Assume that Ben chooses w_{i+1}’’

= We fixed this

- p. 4, line -1: ‘‘position w_i’’ should be ‘‘position w_{i+1}’’.
In fact, here and throughout, don’t even say ‘‘position w_{i+1}’’
since w_{i+1} is a letter. The position is $i+1$. Best would be to say
‘‘Ann chooses w_{i+1}’’.

= We fixed this

- p. 5, line 2: again i -> $i+1$

= We fixed this

- p. 5, last sentence of first paragraph: Again more details. I guess

the point is that if $n > 3$ and $w_{i+n-1} \neq (2,d)$ then Ann is forced to choose $(2,d)$ for w_{i+n+2}, since the first component must be different from Ben's two previous choices, which are necessarily 01 or 10. And if $n > 3$, then Ann must switch favourites again when choosing w_{i+n+4}.
= We made some adjustments in our descriptions, and hope things are clear now.

Reviewer #3:

- The paper currently does not have a background section. It would be good to include a background section containing definitions of terms such as non-trivial repetitions, even/odd length of a square etc.
= We fixed this by introducing a new section

- Page 1, Second paragraph. For the avoidability of even length squares Grytczuk et. al propose Ann use the letters of any square-free word over a three letter alphabet (e.g. a word constructed using Thue's morphism) and not letters of the Thue Morse word. This should be clarified.
= We followed the referee's comments

- Page 2, Second paragraph. The authors need to clarify that the strategy proposed for Ann is to play simultaneously on both coordinates in a way that she avoids even repetitions on the second coordinate and odd repetitions on the first coordinate.
= We followed the referee's comments

- Page 4. Rephrase Claim statement as: ''Any square-free word remains square-free after appending a new letter, as long as no unary squares are created''.
= The referee's suggestion is not correct. We clarified our statement in the proof

- Page 5, last paragraph. Change ''Benn'' to ''Ben''.
= We fixed this

- Finally, in the interest of clarity, the authors are encouraged to illustrate the proposed algorithm with examples.
= A small example was provided

Reviewer #4:

- The main issue with the paper is that it is not self-contained. In order to even understand the introduction, one must be fairly familiar with the papers cited in the bibliography.
= We fixed this

- The second issue with the paper is that it is too short in terms of results; there is one theorems, but as it is builds on the work of others, it does not warrant a paper in IPL.
= This work solves an open problem. It is intended as a note, and this constitutes the reason it has been submitted to IPL.