


## AUTHOR QUERY FORM

 ELSEVIER	<b>Journal:</b> JDA  <b>Article Number:</b> 366	<b>Please e-mail or fax your responses and any corrections to:</b>  <b>E-mail:</b> <a href="mailto:corrections.essd@elsevier.vtex.lt">corrections.essd@elsevier.vtex.lt</a>  <b>Fax:</b> +1 61 9699 6735
---	---	--

Dear Author,

Please check your proof carefully and mark all corrections at the appropriate place in the proof (e.g., by using on-screen annotation in the PDF file) or compile them in a separate list. To ensure fast publication of your paper please return your corrections within 48 hours.

For correction or revision of any artwork, please consult <http://www.elsevier.com/artworkinstructions>

Any queries or remarks that have arisen during the processing of your manuscript are listed below and highlighted by flags in the proof. Click on the '[Q](#)' link to go to the location in the proof.

<b>Location in article</b>	<b>Query / Remark: <a href="#">click on the Q link to go</a> Please insert your reply or correction at the corresponding line in the proof</b>
<a href="#">Q1</a>	Please check updated vol. number of issue (74) in Ref. [4]. (p. 12/ line 37)



ELSEVIER

Contents lists available at ScienceDirect

## Journal of Discrete Algorithms

www.elsevier.com/locate/jda



## The three squares lemma revisited ☆

Evguenia Kopylova <sup>a,b</sup>, W.F. Smyth <sup>a,c,\*</sup><sup>a</sup> Algorithms Research Group, Department of Computing & Software, McMaster University, Hamilton, ON L8S 4K1, Canada<sup>b</sup> Bonsai, LIFL, Université Lille 1, France<sup>c</sup> Centre for Stringology & Applications, Digital Ecosystems & Business Intelligence Institute, Curtin University, GPO Box U1987, Perth WA 6845, Australia

## ARTICLE INFO

## Article history:

Available online xxxx

## Keywords:

Combinatorics on words

String algorithms

Maximal periodicities

Runs

Repetitions

Three squares lemma

## ABSTRACT

A recent paper Fan et al. (2006) [10] showed that the occurrence of two squares at the same position in a string, together with the occurrence of a third near by, is possible only in very special circumstances, represented by 14 well-defined cases. Similar results were published in Simpson (2007) [19]. In this paper we begin the process of extending this research in two ways: first, by proving a “two squares” lemma for a case not considered in Fan et al. (2006) [10]; second, by showing that in other cases, when three squares occur, more precise results – a breakdown into highly periodic substrings easily recognized in a left-to-right scan of the string – can be obtained with weaker assumptions. The motivation for this research is, first, to show that the maximum number of runs (maximal periodicities) in a string is at most  $n$ ; second, and more important, to provide a combinatorial basis for a new generation of algorithms that directly compute repetitions in strings without elaborate preprocessing. Based on extensive computation, we present conjectures that describe the combinatorial behavior in all 14 of the subcases that arise. We then prove the correctness of seven of these conjectures. Along the way we establish a new combinatorial lemma characterizing strings of which two rotations have the same period.

© 2011 Published by Elsevier B.V.

## 1. Introduction

The rationale for this paper arises out of research done over the last two decades on maximal periodicities (or “runs”) in strings and, before that, on the computation of repetitions in strings. In order to reduce proliferation of notation, we adopt throughout the convention that a string denoted  $\mathbf{x}$  (in mathbold) has length  $x$  (regular math mode).

In 2006 it was shown [8] that the occurrence of two squares at the same position in a string, together with the occurrence of a third near by, is possible only in very special circumstances, represented by 14 well-defined subcases. Similar results were published in [19]. In this paper we first extend these results to a case not previously considered, then go on to make the results of [8] more precise under weaker assumptions. We describe experiments conducted on strings that satisfy the “three squares” condition, then use the results of these experiments to formulate conjectures about the nature of  $\mathbf{x}$  and its alphabet in each of the 14 subcases. We prove the correctness of seven of these conjectures.

These complicated combinatorial studies are motivated primarily by the desire to compute repetitions in strings more efficiently and more directly, as we now explain.

☆ This work was supported in part by the Natural Sciences & Engineering Research Council of Canada. The authors express their gratitude to the unsung referees whose perceptive commentary has materially improved this paper.

\* Corresponding author at: Algorithms Research Group, Department of Computing & Software, McMaster University, Hamilton, ON L8S 4K1, Canada.

E-mail addresses: [evguenia.kopylova@inria.fr](mailto:evguenia.kopylova@inria.fr) (E. Kopylova), [smyth@mcmaster.ca](mailto:smyth@mcmaster.ca), [B.Smyth@curtin.edu.au](mailto:B.Smyth@curtin.edu.au) (W.F. Smyth).

Given a nonempty string  $\mathbf{x} = \mathbf{x}[1..n]$  of length  $x = n$  on a finite alphabet  $\Sigma$ , a **repetition** (or **power**) in  $\mathbf{x}$  is a substring  $\mathbf{u}^e$ ,  $\mathbf{u}$  nonempty, integer  $e \geq 2$ , where  $\mathbf{x} = \mathbf{v}\mathbf{u}^e\mathbf{w}$  for some (possibly empty) strings  $\mathbf{v}$ ,  $\mathbf{w}$ . We call  $e$  the **exponent** of the repetition and the length  $p = u$  its **period**. We are interested in **irreducible** repetitions; that is, we assume always that  $\mathbf{u}$  is **primitive** (not itself a repetition), and that  $e$  cannot be increased by left or right extension in  $\mathbf{x}$ . For  $e = 2, 3$ , we say that  $\mathbf{u}^e$  is a **square** or **cube**, respectively. There are well-known algorithms [3,1,15] that compute all the repetitions in  $\mathbf{x}$  in  $O(n \log n)$  time, asymptotically optimal since Fibonacci strings of length  $n$  contain  $\Omega(n \log n)$  repetitions [3]. A repetition in  $\mathbf{x}$  can be represented in constant space by a triple  $(i, p, e)$ , where  $\mathbf{u}^e$  is said to **occur** at position  $i$  in  $\mathbf{x}$  and  $p = u$ . (Standard stringological notation and terminology used in this paper follow [20].)

A **run** in  $\mathbf{x}$  (originally introduced in [14] as a **maximal periodicity**) is a substring  $\mathbf{w}$  of  $\mathbf{x}$  of minimum period  $p \leq w/2$  occurring at some position  $i$ , where neither  $\mathbf{x}[i-1..i+w-1]$  nor  $\mathbf{x}[i..i+w]$  (whenever these are well defined) has period  $p$ . Note that a run always has a prefix  $\mathbf{u}^e$ ,  $p = u$ ,  $e = \lfloor w/p \rfloor \geq 2$ , that is a repetition. A run can be specified by a four-tuple  $(i, p, e, t)$ , where  $i, p, e$  are defined as for a repetition, and the **tail**  $t = w \bmod p$ . The Fibonacci string

$$\begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \mathbf{f} & = & a & b & a & a & b & a & b & a \end{array}$$

contains three runs  $(1, 3, 2, 0) = (aba)^2$ ,  $(3, 1, 2, 0) = a^2$  and  $(4, 2, 2, 1) = (ab)^2a$ . Each of the first two runs is also a square, but the third, by virtue of its nonzero tail, actually identifies two squares,  $(ab)^2$  and  $(ba)^2$ . In general, each run  $(i, p, e, t)$  determines  $t + 1$  repetitions of exponent  $e$ , and computing all the runs in  $\mathbf{x}$  implicitly computes all the repetitions.

In [12] it was shown that the maximum number  $\rho(n)$  of runs in any string of length  $n$  is  $O(n)$ ; specifically, that there exist universal positive constants  $k_1$  and  $k_2$  such that

$$\rho(n) \leq k_1 n - k_2 \sqrt{n} \log n. \quad (1)$$

The methods used to establish the upper bound (1) were not constructive, so that no information was provided about the magnitude of  $k_1$  and  $k_2$ ; nevertheless, based on computational evidence for strings of lengths  $n \leq 60$ , the authors conjectured further that

$$\rho(n)/n \leq 1. \quad (2)$$

Over the last 10 years, constructive methods have been discovered that have successively reduced the bound on  $\rho(n)/n$  to 5.0 [18], 3.48 [17], 1.60 [4], 1.49 [11], and finally 1.029 [6], the last achieved with the aid of three man-years of CPU time on a network of high-performance computers.<sup>1</sup> Perhaps more important from an algorithmic point of view, it has been shown [16] that the expected value of  $\rho(n)/n$  ranges from about 0.4 down to 0.1 for long strings on alphabet sizes ranging from 2 to 10, decreasing to less than 0.05 for English text. In other words, the number of runs in a string of length  $n$  can normally be expected to be sparse.

Also in [12] an algorithm was proposed to compute all the runs, hence all the repetitions, in a given string  $\mathbf{x}$ , based on the following steps:

- \* compute the suffix tree  $ST_{\mathbf{x}}$  of  $\mathbf{x}$  using Farach's algorithm [9];
- \* compute the Lempel–Ziv factorization of  $\mathbf{x}$  [21] using  $ST_{\mathbf{x}}$ ;
- \* compute the leftmost occurrence of each distinct run in  $\mathbf{x}$  using Main's algorithm [14];
- \* compute all the runs in  $\mathbf{x}$  from the leftmost ones [12].

Although each of these steps requires  $O(n)$  time in theory, the first step was not practical for large strings; later approaches using a suffix array [2,5] rather than a suffix tree provided practical linear-time all-runs computation algorithms.

The theoretical and algorithmic advances outlined above are impressive, but there remain significant challenges:

- \* The existing methods used to compute all the runs in a string, even though linear-time, are nevertheless time-consuming. Indeed, the case could be made that these methods are not so far removed from brute force:
  - they make no use of the expected sparsity of runs [16];
  - as noted above, they depend on the prior computation of global data structures (suffix tree or suffix array, LZ factorization);
  - they make no use of combinatorial insights into the degeneration of a string into repetitions of small period, as studied in [8,19], when the constraint is imposed that runs (therefore squares) overlap.

We believe that replacing heavy preprocessing by effective use of combinatorial properties will result in simpler, faster algorithms.

- \* The efforts to establish the conjecture (2) have depended to some extent on combinatorial methods, notably the division of runs into those with “small” periods and those with “large” periods. But the combinatorial constraints imposed by having more than two squares begin in **neighboring** locations, again as studied in [8,19], have not been taken into account.

<sup>1</sup> SHARCNET, <https://www.sharcnet.ca/my/front/>.

In this paper we begin a more precise study of **neighboring** occurrences of three squares than has been conducted in the past. Both [8] and [19] examine the combinatorial consequences of having two distinct squares at the same position  $i$  with a third square occurring distance  $k \geq 0$  to the right of  $i$ . This research thus generalizes the “Three Squares Lemma” [7] for which  $k = 0$ :

**Lemma 1.** *Suppose  $u$  is not a repetition, and suppose  $v \neq u^j$  for any  $j \geq 1$ . If  $u^2$  is a prefix of  $v^2$ , in turn a proper prefix of  $w^2$ , then  $w \geq u + v$ .*

[8] and [19] classify the various subcases that arise, depending on the relative magnitude of  $k$  and other factors, but do so using approaches that are orthogonal and not easily reconcilable. Moreover, neither of the classifications is complete, and it seems also that more precise and comprehensive results can be established. This paper is the first in what we expect will be a series designed to correct these deficiencies and ultimately to enable generalized three squares theory to become useful in an algorithmic context.

Section 2 provides the basic results that are required in order to understand and analyze the generalized three squares problem. Also it establishes a framework for future work in this area. Section 3 deals exhaustively with the case in which squares  $u^2$  and  $v^2$  occur at the same position, where  $v$  is restricted to the range  $u + 1..3u/2$  (the “Two Squares” lemma). Section 4 describes the software used to generate conjectures for the occurrence of three squares with  $v \in 3u/2 + 1..2u - 1$ , and presents its main results. Section 5 then establishes seven of the conjectures (that is, seven of the subcases considered in [8]), showing that in fact the entire range covered by the three squares must break down into a repetition of small periodicity. Section 6 briefly outlines future work.

## 2. Preliminaries

A simple, perhaps naïve, approach to the conjecture (2) derives from the observation that if, somehow, it can be shown that the occurrence of two squares at the same, or **neighboring**, positions is locally incompatible with the existence of a third square nearby, then it might be possible to show that the number of squares that can occur locally is bounded above by the number of positions. The situation that has been partially analyzed so far, and that we consider further in this paper, constrains two squares  $u^2$  and  $v^2$ ,  $u < v < 2u$ , to occur at the same position (taken for simplicity to be position 1 of a string  $x$ ), while a third square  $w^2$  may occur at position  $k + 1$  in  $x$  for some  $k \in 0..v - u - 1$ . There are two main cases that arise:

(C1)  $u < v \leq 3u/2$  (consideration of  $w$  not required);

(C2)  $3u/2 < v < 2u$  and  $v - u < w < v$ ,  $w \neq u$ .

In Section 3 we give new results for (C1); in Section 5 we provide more precise results for seven of the 14 subcases of (C2) that were analyzed in [8].

Before continuing, let us observe that the cases arising from the alternative assumption that  $u^2$  occurs at position 1 and  $v^2$  at position 2 are probably of equal importance to the situation described above, just as likely to yield important combinatorial insights, and have to date not been investigated at all.

Recall that all squares are assumed to be irreducible. In order to describe previous work, we define a **regular** square  $u^2$  to be such that no prefix of  $u$  is a square.

The following lemma is easily proved (see [8]):

**Lemma 2.** *If  $v^2$  has regular proper prefix  $u^2$ , then*

$$v > \max\{u + 1, 3u/2\}.$$

Note that for  $u \leq 2$ ,  $u^2$  can only take the forms  $\lambda^2$  and  $(\lambda\mu)^2$  and so is necessarily regular; furthermore, it can only be that  $v > 2u$ , so that neither of the cases (C1), (C2) identified above apply. For  $u > 2$ , Lemma 2 tells us that (C1) can arise only if  $u^2$  is not regular.

**Lemma 3.** *Suppose  $x = v^2$  has proper prefix  $u^2$ ,  $v < 2u$ . If either*

(a)  $u^2$  is regular, or

(b)  $v > 3u/2$ ,

then

$$x = u_1 u_2 u_1 u_1 u_2 u_1 u_2 u_1 u_1 u_2, \tag{3}$$

where  $u_1 = 2u - v$ ,  $u_2 = 2v - 3u$ .

**Proof.** (a) is proved in [8]. For (b), since  $u \geq 3$ , we may let  $\mathbf{u}_1$  be the nonempty suffix of  $\mathbf{u}$  of length  $u_1 = 2u - v$  that is a prefix of  $\mathbf{v}$ , hence also a prefix of  $\mathbf{u}$ . Then  $u/2 - u_1 = v - 3u/2 > 0$ , and so  $u_1 < u/2$ ; thus  $u = u_1u_2u_1$  for some nonempty  $\mathbf{u}_2$ . Then  $\mathbf{v} = \mathbf{u}_1\mathbf{u}_2\mathbf{u}_1\mathbf{u}_1\mathbf{u}_2$  and  $u_2 = 2v - 3u$ , as required.  $\square$

Thus the weaker condition of Lemma 3(b) still provides the assurance that the factorization (3) holds in (C2).

In order to prove our new results, the following definitions will be helpful. If  $\mathbf{x} = \mathbf{u}\mathbf{v}$ , then  $\mathbf{v}\mathbf{u} = R_u(\mathbf{x})$  is said to be the  $u$ th **rotation** of  $\mathbf{x}$ ; note that if  $\mathbf{u}$  is empty,  $R_0(\mathbf{x}) = \mathbf{x}$ , while if  $\mathbf{v}$  is empty,  $R_x(\mathbf{x}) = \mathbf{x}$ . Also, we extend the use of the term “period”, defined in Section 1 in the context of repetitions: a string  $\mathbf{x}$  is said to have **period**  $u$  if and only if for every  $i = 1, 2, \dots, x - u$ ,  $\mathbf{x}[i] = \mathbf{x}[i + u]$ . Finally, if  $\mathbf{u}$  is both a proper prefix and a suffix of  $\mathbf{x}$ , then it is said to be a **border** of  $\mathbf{x}$ ;  $\mathbf{x}$  has border  $\mathbf{u}$  if and only if it has period  $x - u$ .

We note also the following three well-known results [13]:

**Lemma 4.** (“Periodicity Lemma”, see [10].) *Let  $p$  and  $q$  be two periods of  $\mathbf{x}$ , and let  $d = \gcd(p, q)$ . If  $p + q \leq x + d$ , then  $d$  is also a period of  $\mathbf{x}$ .*

**Lemma 5.** (See [20, p. 76].) *Let  $\mathbf{x}$  be a string of minimum period  $u$ , and let  $v \in 1..x - 1$  be an integer. Then  $R_v(\mathbf{x}) = \mathbf{x}$  if and only if  $\mathbf{x}$  is a repetition and  $u$  divides  $v$ .*

**Remark 6.** If, for nonempty  $\mathbf{x}$  and positive integer  $v \leq x - 1$ ,  $R_v(\mathbf{x}) = \mathbf{x}$ , then  $\gcd(v, x)$  is a period of  $\mathbf{x}$ .

**Lemma 7.** (See [20, p. 76].) *If a string  $\mathbf{x}$  is a repetition of period  $u$ , then so is every rotation of  $\mathbf{x}$ .*

Rather surprisingly, the following lemma appears to be new; it will be useful for analyzing various subcases of our results.

**Lemma 8.** *Suppose both  $\mathbf{x}$  and  $R_v(\mathbf{x})$ ,  $0 < v < x$ , have period  $u$ , where  $\ell = x \bmod u > 0$  and  $r = \lfloor x/u \rfloor$ . Let  $\mathbf{x}_v$  denote  $R_v(\mathbf{x})$ , and let  $d = \gcd(u, \ell)$ . Then*

- (a) if  $r = 1$  and  $v \geq \ell$ ,  $\mathbf{x}_{v-\ell}[1..2\ell]$  is a square of period  $\ell$ ;
- (b) if  $r = 1$  and  $v \leq \ell$ ,  $\mathbf{x}[1..v + \ell]$  has period  $\ell$ ;
- (c) if  $r > 1$  and  $v < u$ ,  $\mathbf{x}[1..v + \ell]$  has period  $\ell$ ; if moreover  $v + d \geq u$ , then  $\mathbf{x}$  is a repetition of period  $d$ ;
- (d) if  $r > 1$  and  $u \leq v \leq x - u$ ,  $\mathbf{x}[1..u + \ell]$ , hence  $\mathbf{x}$ , is a repetition of period  $d$ ;
- (e) if  $r > 1$  and  $x - u < v$ , where  $v' = v - (x - u)$ ,  $\mathbf{x}[v' + 1..u + \ell]$  has period  $\ell$ ; if moreover  $v' \leq d$ , then  $\mathbf{x}$  is a repetition of period  $d$ .

**Proof.**

(a) Since the rotation is by  $v \geq \ell$ , it follows that the suffix  $\mathbf{x}_v[u + 1..u + \ell]$  of  $\mathbf{x}_v$  must equal the suffix  $\mathbf{x}[v - \ell + 1..v]$  of  $\mathbf{x}[1..v]$ . Since  $\mathbf{x}_v$  has period  $u$ , therefore

$$\mathbf{x}_v[u + 1..u + \ell] = \mathbf{x}_v[1..\ell],$$

and so

$$\mathbf{x}[v - \ell + 1..v]\mathbf{x}_v[1..\ell] = (\mathbf{x}_v[1..\ell])^2.$$

But in addition

$$\begin{aligned} \mathbf{x}[v - \ell + 1..v]\mathbf{x}_v[1..\ell] &= \mathbf{x}_{v-\ell}[1..\ell]\mathbf{x}_{v-\ell}[\ell + 1, 2\ell] \\ &= \mathbf{x}_{v-\ell}[1..2\ell], \end{aligned}$$

thus establishing (a).

(b) Periodicity  $u$ , together with the requirement that  $u + v \leq u + \ell = x$ , implies

$$\mathbf{x}[1..v] = \mathbf{x}_v[x - v + 1..x] = \mathbf{x}[x - u + 1..x - u + v] = \mathbf{x}[\ell + 1..\ell + v],$$

and so  $\mathbf{x}[1..v + \ell]$  has period  $\ell$ , as required.

(c) Observe that a prefix

$$\mathbf{x}[1..v] = \mathbf{x}_v[x - v + 1..x]$$

of  $\mathbf{x}[1..v + \ell]$  of length  $v$  is constrained to match the corresponding suffix  $\mathbf{x}[\ell + 1..\ell + v]$ . Thus  $\mathbf{x}[1..v + \ell]$  has a border of length  $v$ , hence period  $\ell$ , as required.

If in addition  $v + d \geq u$ , then also  $v + \ell \geq u$ , so that  $\mathbf{x}[1..v + \ell]$  has two periods,  $u$  and  $\ell$ . Since  $(v + \ell) + d \geq u + \ell$ , Lemma 4 applies, implying that  $\mathbf{x}[1..v + \ell]$  has period  $d$ . It follows that  $d$  is also a period of  $\mathbf{u}$ , and since  $d \mid u$ , therefore

**Table 1**

The 14 subcases identified in [8], slightly modified, for three neighboring squares  $\mathbf{u}$ ,  $\mathbf{v}$ ,  $\mathbf{w}$  (with  $v - u < w < v$ ,  $w \neq u$ ).

Subcase S	$k$	$k + w$	$k + 2w$	Special conditions
1	$0 \leq k \leq u_1$	$k + w \leq u$	$k + 2w \leq u + u_1$	$k \geq u_2$
2	$0 \leq k \leq u_1$	$k + w \leq u$	$k + 2w \leq u + u_1$	$k < u_2$
3	$0 \leq k \leq u_1$	$k + w \leq u$	$k + 2w > u + u_1$	-
4	$0 \leq k \leq u_1$	$u < k + w \leq u + u_1$	-	-
5	$0 \leq k \leq u_1$	$u + u_1 < k + w \leq v$	-	-
6	$0 \leq k \leq u_1$	$v < k + w < 2u$	-	-
7	$u_1 < k < u_1 + u_2$	$k + w \leq u + u_1$	$k + 2w \leq 2u$	-
8	$u_1 < k < u_1 + u_2$	$k + w \leq u + u_1$	$k + 2w > 2u$	-
9	$u_1 < k < u_1 + u_2$	$u + u_1 < k + w \leq v$	-	$w < u$
10	$u_1 < k < u_1 + u_2$	$k + w \leq v$	$k + 2w \leq u + v$	$w > u$
11	$u_1 < k < u_1 + u_2$	$k + w \leq v$	$u + v < k + 2w \leq 2v - u_2$	-
12	$u_1 < k < u_1 + u_2$	$k + w \leq v$	$2v - u_2 < k + 2w$	-
13	$u_1 < k < u_1 + u_2$	$v < k + w \leq 2u$	-	-
14	$u_1 < k < u_1 + u_2$	$2u < k + w < 2u + u_2 - 1$	-	-

$\mathbf{u}$  is a repetition of period  $d$ ; since furthermore  $d \mid \ell$ ,  $\mathbf{u}[1..\ell]$  must be a repetition of period  $d$ , and thus so also is  $\mathbf{x}$ , as required.

(d) In this case we must have

$$\mathbf{x}[1..u] = \mathbf{x}[\ell + 1..\ell + u],$$

from which we conclude that  $\mathbf{x}[1..u + \ell]$  has period  $\ell$ . Since  $\mathbf{x}[1..u + \ell]$  also has period  $u$ , it must therefore by Lemma 4 have period  $d$ . Since  $d \mid u + \ell$ , the result follows.

(e) Here we require

$$\mathbf{x}[v' + 1..u] = \mathbf{x}[\ell + v' + 1..\ell + u],$$

so that  $\mathbf{z} = \mathbf{x}[v' + 1..u + \ell]$  of length  $z = u + \ell - v'$  has a border of length  $u - v'$ , hence period  $\ell$ . Observe that when  $v' \leq d$ ,  $z = u + \ell - v' \geq u + \ell - d \geq u$ , so that  $\mathbf{z}$  also has period  $u$ . Since in this case  $z + d \geq u + \ell$ , Lemma 4 again applies, implying that  $\mathbf{z}$  has period  $d$ . As in (c), it follows that  $\mathbf{x}$  is a repetition of period  $d$ .  $\square$

In [8] a “new periodicity lemma” was proved:

**Lemma 9 (NPL).** *If  $\mathbf{x}$  has prefixes  $\mathbf{u}^2$  and  $\mathbf{v}^2$ ,  $\mathbf{u}^2$  regular,  $u < v < 2u$ , then for every  $k \in 0..v - u - 1$  and every  $w \in v - u + 1..v - 1$ ,  $w \neq u$ ,  $\mathbf{x}[k + 1..k + 2w]$  is not a square.*

The proof of this result required the analysis of 14 subcases (see Table 1), each established by showing that its existence contradicts the regularity of  $\mathbf{u}^2$ . Recall that if  $\mathbf{u}^2$  is not regular, then it must have a prefix, say  $\mathbf{u}_{-1}^2$ ,  $u_{-1} < u$ ; similarly, if  $\mathbf{u}_{-1}^2$  is not regular, it must have a prefix  $\mathbf{u}_{-2}^2$ ,  $u_{-2} < u_{-1}$ ; thus, eventually, there exists some integer  $t > 0$  such that  $\mathbf{u}_{-t}^2$  is regular. However, NPL applies to  $\mathbf{u}_{-t}^2$  and  $\mathbf{u}_{-t+1}^2$  only if it should happen that  $3u_{-t}/2 < u_{-t+1} < 2u_{-t}$ . Therefore, dropping the regularity assumption, while using instead the weaker condition  $3u/2 < v < 2u$ , extends the applicability of NPL to some cases where  $\mathbf{u}^2$  is in fact not regular. (For example,  $\mathbf{u} = \mathbf{aaba}$  of length 5,  $\mathbf{v} = \mathbf{aabaaaab}$  of length 8.)

In [8] a table of the 14 subcases was presented, showing for each one the periodicity induced by the occurrence of the three squares  $\mathbf{u}^2$  and  $\mathbf{v}^2$  as prefixes of  $\mathbf{x}$ , and  $\mathbf{w}^2$  at some position  $k + 1$ , for  $k \in 0..v - u - 1$ ,  $w \in v - u + 1..v - 1$ ,  $w \neq u$ . In Section 5 we begin to refine these results, showing that in fact seven of the 14 subcases result in highly periodic behavior that is easily recognized – essentially this means that three squares can occur only in trivial circumstances.

### 3. Case (C1) – $v \in u + 1..3u/2$

The first result establishes the basic structure of  $\mathbf{x}$  in this case:

**Lemma 10.** *If  $\mathbf{x} = \mathbf{v}^2$  with prefix  $\mathbf{u}^2$ ,  $u < v \leq 3u/2$ , then*

$$\mathbf{x} = \mathbf{u}_1^m \mathbf{u}_2 \mathbf{u}_1^{m+1} \mathbf{u}_2 \mathbf{u}_1, \tag{4}$$

where  $u_1 = v - u \leq u/2$ ,  $u_2 = u \bmod u_1 \geq 0$ ,  $m = \lfloor u/u_1 \rfloor \geq 2$ , and  $\mathbf{u}_2$  is a proper prefix of  $\mathbf{u}_1$ .

**Proof.** Let  $\mathbf{u}_1$  be the suffix of  $\mathbf{v}$  of length  $v - u$ , and observe that  $2u_1 \leq u$ . The result clearly holds in the trivial case that  $u_1 \mid u$  ( $u_2 = 0$ ), where  $\mathbf{v} = \mathbf{u}_1^{m+1}$  and  $\mathbf{x} = \mathbf{u}_1^{2(m+1)}$ ; assume therefore that  $u_2 > 0$ . Observe that  $\mathbf{u}_1$  is a prefix of  $\mathbf{u}$  and



therefore also a prefix of  $\mathbf{v}$ . Observe further that if for some  $\ell > 0$ ,  $\mathbf{u}_1^\ell$  is a prefix of  $\mathbf{v}$  with  $(\ell + 1)u_1 < u$ , then  $\mathbf{u}_1^{\ell+1}$  is necessarily a prefix of  $\mathbf{u}$ , hence also a prefix of  $\mathbf{v}$ . Since in particular this statement is true for  $\ell = m - 1$ , we see that  $\mathbf{u}_1^m$  is a prefix of  $\mathbf{v}$ ; in other words,  $\mathbf{v} = \mathbf{u}_1^m \mathbf{u}_2 \mathbf{u}_1$ , and (4) follows. Since  $\mathbf{u}^2 = \mathbf{u} \mathbf{u}_1^m \mathbf{u}_2$  is a prefix of  $\mathbf{v}^2 = \mathbf{u} \mathbf{u}_1^{m+1} \mathbf{u}_2 \mathbf{u}_1$ , and since  $u_2 < u_1$ , we see also that  $\mathbf{u}_2$  must be a proper prefix of  $\mathbf{u}_1$ .  $\square$

Lemma 10 speaks of squares, but since every run begins with a square, it also describes runs (assuming of course that the runs corresponding to  $\mathbf{u}^2$  and  $\mathbf{v}^2$  cannot be left-extended). Consider the example

$$\mathbf{x} = aabaabaabaab aabaabaabaab. \tag{5}$$

Here  $\mathbf{u}_1 = aab$ ,  $\mathbf{u}_2 = a$ ,  $m = 3$ , and the square  $\mathbf{u}^2 = (\mathbf{u}_1^3 \mathbf{u}_2)^2$  gives rise to the run  $\mathbf{u}^2 a$ . In the related example

$$\mathbf{x} = abbabbabbaabb abbabbabbaabb, \tag{6}$$

we again have  $\mathbf{u}_2 = a$ ,  $m = 3$ , but now  $\mathbf{u}_1 = abb$  and the run  $\mathbf{u}^2$  cannot be extended. If more generally we address ourselves to the question of what runs exist in (4) in the nontrivial case that  $u_2 > 0$ , we easily identify the following:

- (R1)  $\mathbf{v}^2$  and  $\mathbf{u}^2 \mathbf{u}^*$  for some possibly empty proper prefix  $\mathbf{u}^*$  of  $\mathbf{u}_1$  such that both  $\mathbf{u}^*$  and  $\mathbf{u}_2 \mathbf{u}^*$  are prefixes of  $\mathbf{u}_1$ ; for example,  $\mathbf{u}^* = a$  in (5),  $\varepsilon$  in (6).
- (R2)  $\mathbf{u} \mathbf{u}^* = \mathbf{u}_1^m \mathbf{u}_2 \mathbf{u}^*$  and  $\mathbf{u}_1 \mathbf{u} \mathbf{u}^* = \mathbf{u}_1^{m+1} \mathbf{u}_2 \mathbf{u}^*$ , runs that may be adjacent as in (6) or overlap as in (5), and that together cover all of  $\mathbf{x}$  except for a suffix of the final copy of  $\mathbf{u}_1$ .
- (R3)  $m + 1$  runs

$$\mathbf{u}_2^2 \mathbf{u}^*, (\mathbf{u}_1 \mathbf{u}_2)^2 \mathbf{u}^*, \dots, (\mathbf{u}_1^m \mathbf{u}_2)^2 \mathbf{u}^* = \mathbf{u}^2 \mathbf{u}^*, \tag{7}$$

all centred at position  $u + 1$  of  $\mathbf{x}$ , with the first one  $\mathbf{u}_2^2 \mathbf{u}^*$  repeated at position  $(2m + 1)u_1 + u_2 + 1$ . The centred runs (7) arise in the analyses of [18] and [4].

- (R4) Miscellaneous runs of period strictly less than  $u_1$ . For example, the runs  $aa$  that occur as a substring of occurrences of  $\mathbf{u}_1$  in (5). Another example: in the case  $\mathbf{u}_1 = abaab$ ,  $\mathbf{u}_2 = a$ ,  $m = 2$ ,

$$\mathbf{x} = abaababaabaabaab abaababaabaabaab,$$

we identify, in addition to  $2m + 4$  runs  $aa$ , a sequence of four overlapping runs  $(aba)^2$ ,  $(aba)^4$ ,  $(aba)^2$ ,  $(aba)^3 ab$ , that cover  $\mathbf{x}$ .

We prove the following:

**Lemma 11.** *The string (4),  $u_2 > 0$ , contains no repetitions (runs) of period  $z \geq u_1$  other than those characterized in (R1)–(R3).*

**Proof.** We consider possible runs  $R = \mathbf{z}^t \mathbf{z}^*$ ,  $t > 1$ ,  $\mathbf{z}^*$  a possibly empty proper prefix of  $\mathbf{z}$ . Suppose first that  $R$  is a substring of one of the runs (R2). In this case we may assume that  $z$  is not a multiple of  $u_1$ , but that  $z > u_1$ . Notice that  $R$  has a prefix  $\mathbf{z}_1 \mathbf{z}_2$ , where  $\mathbf{z}_1 = \mathbf{z}_2 = \mathbf{z}$ . Then  $\mathbf{z}_1$  has a prefix that is a rotation  $R_s(\mathbf{u}_1)$  for some  $s \in 0..u_1 - 1$ . Since  $z > u_1$  and not a multiple of  $u_1$ , it follows that  $\mathbf{z}_2$  has a prefix that is a rotation  $R_{s'}(\mathbf{u}_1)$  for some  $s' \neq s$ ,  $s' \in 0..u_1 - 1$ . Since these two prefixes must be equal, we conclude from Lemma 5 that  $\mathbf{u}_1$  is a repetition, contradicting the assumption that  $\mathbf{u}$  is a run of minimum period  $u_1$ .

Suppose then that  $R$  is not a substring of either run (R2), with  $z \geq u_1$ . We may suppose also that  $R$  is not a run of (R3). Then  $R$  overlaps the join  $\mathbf{u}^*$  of the two runs  $\mathbf{u} \mathbf{u}^*$  and  $\mathbf{u}_1 \mathbf{u} \mathbf{u}^*$ , both of period  $u_1$ . If the first occurrence, say  $\mathbf{z}_1$ , of the repeating substring  $\mathbf{z}$  does not overlap the join, then it must be a substring of  $\mathbf{u} \mathbf{u}^*$  and so have period  $u_1$ ; but since some other occurrence of  $\mathbf{z}$  does overlap the join, it cannot therefore, by the nonextendibility of the run  $\mathbf{u} \mathbf{u}^*$ , both have period  $u_1$  and be equal to  $\mathbf{z}_1$ , a contradiction. We conclude that  $\mathbf{z}_1$  overlaps the join; however, since  $R$  is not in (R3),  $\mathbf{z}_1$  cannot have period  $u_1$ . Consequently no other occurrence of  $\mathbf{z}$  can have period  $u_1$ , and so  $R = \mathbf{z}_1 \mathbf{z}_2 \mathbf{z}^*$ , where  $\mathbf{z}_2$  overlaps the final occurrence of  $\mathbf{u}_1$  in  $\mathbf{x}$ ; thus  $\mathbf{z}_2 = \mathbf{z}_2^* \mathbf{z}_2'$ , where  $\mathbf{z}_2^*$  has period  $u_1$  and  $\mathbf{z}_2'$  is a nonempty substring of  $\mathbf{u}_1 = \mathbf{u}^* \mathbf{z}_2' \mathbf{u}_1'$  for some proper suffix  $\mathbf{u}_1'$  of  $\mathbf{u}_1$ .

We see then that  $\mathbf{z}_2$  has no suffix of length greater than  $u_1$  that has period  $u_1$ ; since this must also be true of  $\mathbf{z}_1$ , it follows that  $\mathbf{z}_2^* \geq m u_1 + u_2 + u^*$ . But then, setting  $\mathbf{z}_1 = \mathbf{z}_1^* \mathbf{z}_1'$  with  $\mathbf{z}_1^* = \mathbf{z}_2^*$ , we see that  $\mathbf{z}_1^* \leq m u_1 + u_2 + u^*$ , hence that  $\mathbf{z}_1^* = \mathbf{z}_2^* = m u_1 + u_2 + u^*$ . Therefore the run  $R$  is exactly  $\mathbf{v}^2$ , already included in (R1). This completes the proof.  $\square$

To summarize: in the situation identified by Lemma 10, for a string  $\mathbf{x}$  of length

$$\mathbf{x} = (2m + 2)u_1 + 2u_2 = (2u_1)m + 2(u_1 + u_2), \quad u_2 > 0,$$

there exist exactly  $m + 5$  runs (R1)–(R3), together with runs (R4) of period strictly less than  $u_1$ , and no others.

---

```

1  – For every subcase  $(u_1, u_2, k, w)$  determined by  $u_{1\_max}, u_{2\_max}$ ,
2  – compute subcase identifier  $S$ , maximum alphabet  $\sigma$  and  $\mathbf{u}_1, \mathbf{u}_2$ 
3
4  1. for  $u_1 \leftarrow 1$  to  $u_{1\_max}$  do
5  2.   for  $u_2 \leftarrow 1$  to  $u_{2\_max}$  do
6  3.     for  $k \leftarrow 0$  to  $u^* - 1$  do – Recall  $u^* \equiv u_1 + u_2$ .
7  4.       for  $w \leftarrow u^* + 1$  to  $v - 1$  do
8  5.         if  $w \neq 2u_1 + u_2$  then
9  6.            $\sigma \leftarrow u^*$ 
10  7.             $\mathbf{u}_1 = 12 \cdots u_1$ ;  $\mathbf{u}_2 = u_1 + 1u_1 + 2 \cdots u^*$ 
11  8.              $(\sigma, \mathbf{u}_1, \mathbf{u}_2) \leftarrow \text{force\_square}(\sigma, \mathbf{u}_1, \mathbf{u}_2, k, w)$ 
12  9.              $S \leftarrow \text{compute\_subcase}$  – from Table 1
13 10.            return  $(S, \sigma, \mathbf{u}_1, \mathbf{u}_2, k, w)$ 

```

---

Fig. 1. Algorithm **construct\_x**( $S, \sigma, \mathbf{u}_1, \mathbf{u}_2, k, w$ ):  $u_1 \in 1..u_{1\_max}$ ,  $u_2 \in 1..u_{2\_max}$ .

---

```

function force_square( $\sigma, \mathbf{u}_1, \mathbf{u}_2, k, w$ )
15  – For given values  $k$  and  $w$ , apply condition (8) to
16  – recompute  $\sigma, \mathbf{u}_1, \mathbf{u}_2$  in  $\mathbf{x} = \mathbf{u}_1 \mathbf{u}_2 \mathbf{u}_1 \mathbf{u}_1 \mathbf{u}_2 \mathbf{u}_1 \mathbf{u}_2 \mathbf{u}_1 \mathbf{u}_2$ 
17
18  1.  $wlim \leftarrow \min(k + w, x - w)$  – possibly  $k + 2w > x$ 
19  2. for  $i \leftarrow k + 1$  to  $wlim$  do
20  3.   if  $\mathbf{x}[i] \neq \mathbf{x}[i + w]$  then
21  4.      $\sigma \leftarrow \sigma - 1$ 
22  5.     replace all occurrences of  $\max(\mathbf{x}[i], \mathbf{x}[i + w])$  in  $\mathbf{x}$  with  $\min(\mathbf{x}[i], \mathbf{x}[i + w])$ 
23  6. return  $(\sigma, \mathbf{u}_1, \mathbf{u}_2)$ 

```

---

Fig. 2. Function **force\_square**( $\sigma, \mathbf{u}_1, \mathbf{u}_2, k, w$ ).

#### 4. Generating conjectures

In this section we first describe the algorithm used to generate conjectures about the periodicity induced by two squares  $\mathbf{u}^2$  and  $\mathbf{v}^2$  at the same position, with a third square  $\mathbf{w}^2$  occurring at distance  $k$ , as described in Lemma 9. We then go on to provide precise statements of the conjectures generated.

The algorithm **construct\_x** is a function that, given two positive integer values  $u_{1\_max}$  and  $u_{2\_max}$ , generates all subcases allowable under the conditions specified in Table 1 for every  $u_1 \in 1..u_{1\_max}$ ,  $u_2 \in 1..u_{2\_max}$ ,  $k \in 0..u_1 + u_2 - 1$ ,  $w \in u_1 + u_2 + 1..3u_1 + 2u_2 - 1$ ,  $w \neq u$ . We claim that these subcases are disjoint and together correspond exactly to the conditions on  $\mathbf{u}, \mathbf{v}, k$  and  $w$  stated in Lemma 9. For each allowable set of values  $(u_1, u_2, k, w)$  **construct\_x** determines the applicable subcase identifier  $S$ . But the algorithm does more: it also computes the maximum alphabet size  $\sigma$  for  $\mathbf{x}$  that is consistent with  $S$ , where  $x = \max(2v, k + 2w)$ .

Initially the maximum alphabet size of  $\mathbf{x}$  is  $\sigma_0 = u_1 + u_2$ , since by (3)  $\mathbf{w}$  and  $\mathbf{v}$  can contain only entries from substrings  $\mathbf{u}_1$  and  $\mathbf{u}_2$ . Let  $\mathbf{u}^* = \mathbf{u}_1 \mathbf{u}_2$  and let the initial alphabet  $\Sigma_0 = \{1, 2, \dots, u^*\}$  with

$$\mathbf{u}_1 = 12 \cdots u_1, \quad \mathbf{u}_2 = (u_1 + 1)(u_1 + 2) \cdots u^*.$$

The condition applied to determine alphabet size  $\sigma$  is

$$\mathbf{x}[k + 1..k + w] = \mathbf{x}[k + w + 1..k + 2w], \quad (8)$$

where at each position  $i \in 1..w$  in  $\mathbf{w}$  such that  $\mathbf{x}[k + i] = \mathbf{u}^*[j_1]$  for some  $j_1 \in 1..u^*$ , and such that  $\mathbf{x}[k + w + i] = \mathbf{u}^*[j_2]$  for some  $j_2 \in 1..u^*$ ,<sup>2</sup> we require that the letter  $\mathbf{u}^*[j_1]$  equal the letter  $\mathbf{u}^*[j_2]$ . If these letters are not already equal, then in every copy of  $\mathbf{u}_1$  or  $\mathbf{u}_2$  in  $\mathbf{x}$ , we replace the numerically larger of the two by the smaller, updating the alphabet at each step as follows:

$$\Sigma \leftarrow \Sigma - \{\max(\mathbf{u}^*[j_1], \mathbf{u}^*[j_2])\}, \quad (9)$$

where initially  $\Sigma = \Sigma_0$ . After all  $w$  such pairs of positions have been considered, the letters remaining in  $\Sigma$  are exactly those that occur in  $\mathbf{x}$ :  $\sigma = |\Sigma|$ . Figs. 1 and 2 outline these calculations.

Algorithm **construct\_x** was executed for  $u_{1\_max} = u_{2\_max} = 30$ , yielding a total of 1,415,925 strings spread over the 14 cases as shown in Table 2. In this table,

- \* column 2 gives the number of strings generated for Subcase  $S$ ;
- \*  $\sigma_{max}$  is the maximum over all maximum alphabet sizes  $\sigma$  computed for any string generated for Subcase  $S$ ;
- \*  $d = \gcd(u_1, u_2, w)$  and columns 4 and 5 count the number of generated strings for which  $\sigma$  equals or exceeds  $d$ , respectively;

<sup>2</sup> In subcases 13 and 14 it may happen that  $k + w + i > 2v$ ; for such values of  $i$ , therefore, no such  $j_2$  exists.



**Table 2**

Statistics for 1 415 925 strings generated using  $u_1_{max} = u_2_{max} = 30$ .

1	2	3	4	5	6	7
S	# strings	$\sigma_{max}$	# $\sigma = d$	# $\sigma > d$	# $\Sigma = \{1, 2, \dots, \sigma\}$	# gaps
1	7840	7	7840	0	7840	0
2	8960	10	8960	0	8960	0
3	131 100	29	118 305	12 795	131 100	0
4	283 620	30	276 799	6821	278 132	5488
5	227 505	30	227 505	0	227 505	0
6	121 800	15	121 800	0	121 800	0
7	47 250	27	44 548	2702	44 860	2390
8	51 640	15	51 640	0	51 640	0
9	90 335	15	90 335	0	90 335	0
10	64 050	10	64 050	0	64 050	0
11	54 000	15	51 707	2293	54 000	0
12	16 800	15	15 612	1188	16 800	0
13	201 405	30	197 860	3545	201 405	0
14	109 620	15	108 770	850	108 831	789

**Table 3**

Overview of conjectures.

Subcases S	Conditions	Breakdown of $\mathbf{x}/\mathbf{v}^2$
1, 2, 5, 6, 8-10	$(\forall \mathbf{x}, \sigma = d)$	$\mathbf{x} = \mathbf{d}^{(x/d)}$
3, 4, 7	$\sigma = d$ $\sigma > d$	$\mathbf{x} = \mathbf{d}^{(x/d)}$ $\mathbf{x} = \mathbf{s}^\alpha \mathbf{s}[1..u_1 \bmod s] \mathbf{s}^\gamma \mathbf{s}[1..u_1 \bmod s] \mathbf{s}^\epsilon$
11-14	$\sigma = d$ $\sigma > d$	$\mathbf{x} = \mathbf{d}^{(x/d)}$ $\mathbf{v}^2 = (\mathbf{r}^\beta \mathbf{r}[1..r \bmod u_1])^2 (???)$

\* columns 6 and 7 give the number of generated strings for which the alphabet resulting from function **force\_square** consists of consecutive integers  $1, 2, \dots, \sigma$ , or not, respectively.

A string-by-string computer-based analysis of the generated strings counted in Table 2 yields a collection of conjectures, summarized in Table 3. Essentially, it appears that whenever  $\sigma = d$ ,  $\mathbf{x}$  breaks down into a repetition of period  $d$ . If  $\sigma > d$ , however, then there is still a highly repetitive breakdown, but of a more complex kind. For Subcases 3, 4, 7, the breakdown depends on parameters

$$s = \gcd(u - w, w - u_1); \quad \alpha = \lfloor u/s \rfloor; \quad \gamma = \lfloor v/s \rfloor; \quad \epsilon = (u_1 + u_2)/s; \tag{10}$$

while for Subcases 11-14, even though there is always a highly periodic breakdown, the exact nature of it remains a puzzle – often it depends, as shown, on

$$r = v - w; \quad \beta = (2u/r) - 1. \tag{11}$$

The conjectures given in the first collection of seven subcases in Table 3 – 1, 2, 5, 6, 8-10 – will be proved correct in Section 5. Since it is known therefore for these cases that  $\mathbf{x}$  is a repetition of period  $d = \gcd(u_1, u_2, w)$ , it follows that  $\sigma \neq d$  – otherwise, such a repetition would not always be possible. Moreover, since a repetition of period  $d$  can always be represented using  $d$  distinct letters, it follows that  $\sigma \neq d$ . In other words,  $\sigma = d$  is a condition necessary for periodicity  $d$ , but we have as yet no proof that it is also sufficient, as it appears to be judging from the conjectures for Subcases 3, 4, 7, 11-14. Note that according to the experiments done so far, about 96% of the generated strings  $\mathbf{x}$  yield  $\sigma = d$  and so reduce to  $\mathbf{x} = \mathbf{d}^{(x/d)}$ ; even more striking is the fact that out of 1 415 925 strings only 8667 (about 0.6%) have a prefix of length  $\sigma$  in which some letter is necessarily duplicated. These are combinatorial mysteries that require explanation.

**5. Case (C2) – subcases yielding full periodicity**

For the first group of seven subcases identified in Table 3, we prove the stated conjecture; namely, that  $\mathbf{x} = \mathbf{v}^2$  is a repetition of period  $d = \gcd(u_1, u_2, w)$ . Note that the lemma mentions only six subcases because the first two (1 and 2) are handled in the same way.

**Lemma 12.** (Figs. 3-8, Subcases 1, 2, 5, 6, 8-10.) Suppose that  $\mathbf{x}$  has prefixes  $\mathbf{u}^2$  and  $\mathbf{v}^2$ ,  $3u/2 < v < 2u$  (so that (3) holds). Suppose further that  $\mathbf{w}^2$ ,  $v - u < w < v$ ,  $w \neq u$ , occurs at position  $k$  of  $\mathbf{x}$ , where

- (a)  $0 \leq k \leq u_1$ ,  $k + w \leq u$  and  $k + 2w \leq u + u_1$ , **or**

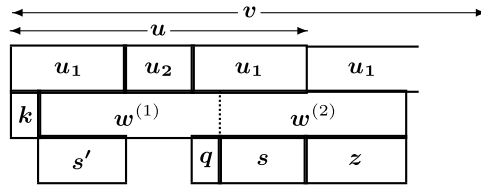


Fig. 3. (a) Subcases 1 and 2.

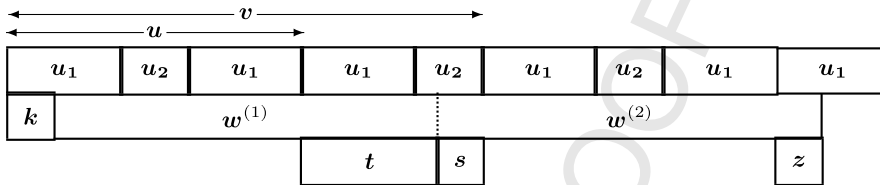


Fig. 4. (b) Subcase 5.

- (b)  $0 \leq k \leq u_1$  and  $u + u_1 < k + w \leq v$ , or
- (c)  $0 \leq k \leq u_1$  and  $v < k + w \leq 2u$ , or
- (d)  $u_1 < k < u_1 + u_2$  and  $u < k + w \leq u + u_1$  and  $2u < k + 2w$ , or
- (e)  $u_1 < k < u_1 + u_2$  and  $u + u_1 < k + w \leq v$  and  $w < u$ , or
- (f)  $u_1 < k < u_1 + u_2$  and  $k + w \leq v$  and  $k + 2w \leq u + v$  and  $w > u$ .

Then  $v^2$  is a repetition of period  $\gcd(u_1, u_2, w)$ .

**Proof.**

(a) Let  $q = w - (u_1 + u_2) + k$ ,  $z = w - u_1 + q$ , with corresponding  $q$  and  $z$ , so that  $z - q > q - k > 0$  with  $z > q > k$ . Define also  $s$  and  $s'$  with  $s = u_1 - q < u_1 - k = s'$ . Since  $w^{(2)}$ , the second copy of  $w$ , ends in  $u_1^{(3)}$ , the third copy of  $u_1$ , it follows that  $z \leq u_1$ ; hence  $q < u_1$  and  $s > 0$ . Since  $s$  and  $s'$  are both prefixes of  $w$  and suffixes of  $u_1$ , it follows that  $s$  is both a proper prefix and a proper suffix of  $s'$ , hence that  $s' - s = q - k$  is a period of  $s'$ . Similarly,  $q$  and  $z$  are both prefixes of  $u_1$  and suffixes of  $w$ , so that  $q$  is both a proper prefix and a proper suffix of  $z$ , implying that  $z - q$  is a period of  $z$ . The overlap of  $u_1$  and  $s'$  (suffix of  $u_1$ ) is a string  $f$  (actually a prefix of  $w$ ) of length  $z - k$  that must have both periods  $p_1 = q - k$  and  $p_2 = z - q$ . Since  $f = p_1 + p_2$ , it follows that  $p_1 + p_2 < f + d$ , where  $d = \gcd(p_1, p_2)$ ; thus the conditions of Lemma 4 are satisfied, and  $f$  has period  $d \leq p_1 < p_2$ , and in fact  $d \mid f$ , so that  $f = rd$ ,  $r \geq 3$ . Moreover, since  $f$  is a substring of  $u_1$ , it will be copied left by period  $p_1$ , right by period  $p_2$ , both divisible by  $d$ ; thus  $z$ ,  $s'$  and  $u_1$  all have period  $d$ . Note that  $d \mid u_2$ , since

$$d = \gcd(w - (u_1 + u_2), w - u_1) = \gcd(w - u_1, u_2). \tag{12}$$

Next observe that  $s'$  and  $z$  also overlap as prefix and suffix, respectively, of  $w$ , by a distance  $q - k$ . Since  $d \mid q - k$ , it follows that  $w$  also has period  $d$ . Considering  $w^{(2)}$ , we see that not only does  $u_1$  have period  $d$ , but so also does the nontrivial rotation  $R_q(u_1)$ .

Now we can apply Lemma 8 to  $u_1$  and  $R_q(u_1)$ . If  $u_1 = td$ ,  $u_1$  is a repetition of period  $d' = d$ . If however we suppose that  $u_1 = td + \ell$ ,  $0 < \ell < d$ , then since  $f \geq 3d$ ,  $t \geq 4 > 1$ . Since moreover  $d < q < u_1 - d$ , case (d) of Lemma 8 applies, and we conclude that  $u_1$  is necessarily a repetition of period  $d' = \gcd(d, \ell)$ . In both of these cases,  $d' \mid d$  and  $d' \mid u_1$ , so that from (12) we conclude that  $d' = \gcd(u_1, u_2, w)$ . Thus  $u_1$  and  $u_2$ , hence  $v^2$ , also are repetitions of period  $d'$ , completing the proof of (a).

(b) Let  $s = v - (w + k)$  with  $0 \leq s < u_2$ ; let  $t = u_1 + u_2 - s = k + (w - u)$  with  $u_1 < t \leq u_1 + u_2$ ; let  $z = (w - u) - s = (k + 2w) - (v + u)$ , and note that possibly  $z \leq 0$ . Fig. 4 shows the corresponding strings  $s$ ,  $t$ ,  $z$ , with  $z > 0$ .

As observed in [8],  $R_k(u_1 u_2)$  and  $R_t(u_1 u_2)$  are both prefixes of  $w$ , with  $k \leq u_1 < t$ . For  $t < u_1 + u_2$  ( $s > 0$ ), Lemma 5 and Remark 6 can be applied to conclude, since  $t - k = w - u$ , that  $u_1 u_2$  is a repetition of period  $d = \gcd(w - u, u_1 + u_2)$  (incorrectly stated in [8] to be period  $t - k$ ). However, for  $t = u_1 + u_2$  (that is,  $s = 0$ , a case missed in [8]), since  $k + s = v - w$  and  $v - w > 0$  by hypothesis, therefore  $R_k(u_1 u_2) = u_1 u_2$  is a repetition of period  $d' = \gcd(v - w, u_1 + u_2)$ . Because

$$w - u = (u_1 - k) + (u_2 - s) = (u_1 + u_2) - (k + s) = (u_1 + u_2) - (v - w), \tag{13}$$

we see that these cases are really the same:  $d' = d$ . Note further that by Lemma 7  $u_2 u_1$  is also a repetition of period  $d$ , and that  $u$  necessarily has period  $d$ .

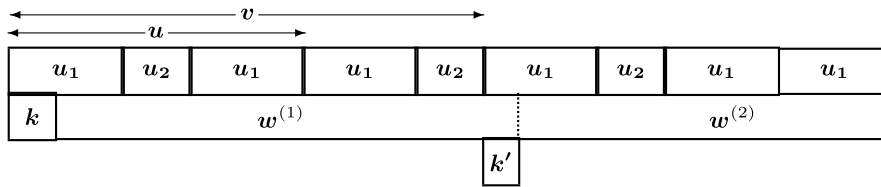


Fig. 5. (c) Subcase 6.

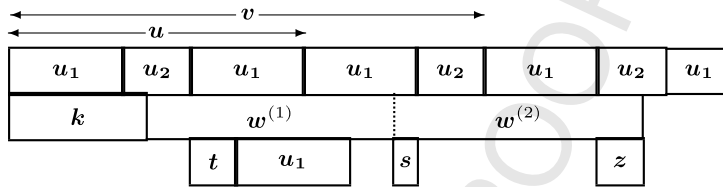


Fig. 6. (d) Subcase 8.

Suppose first that  $z \leq 0$ , so that  $\mathbf{w}^{(2)} = (R_{u_2-s}(\mathbf{u}_2\mathbf{u}_1))^{q/(u_1+u_2)}$  for some integer  $q > u_1 + u_2$ , and so  $\mathbf{w}$  has period  $d$ . Then of course  $\mathbf{w}^{(1)}$  and in particular its prefix  $R_k(\mathbf{u})$ , as well as its substring  $R_{u_1}(\mathbf{u})$  and its suffix  $R_t(\mathbf{u})$ , all have period  $d$ . Let  $\mathbf{y} = R_k(\mathbf{u})$ , so that  $R_{t-k}(\mathbf{y}) = R_{w-u}(\mathbf{y}) = R_t(\mathbf{u})$ . We now apply Lemma 8 to  $\mathbf{y}$  and  $R_{w-u}(\mathbf{y})$ , where  $y = u$ . Since  $d \mid w - u$ , therefore  $d \leq w - y$ . Also  $2w \leq v + u$  implies  $w \leq y + (u_1 + u_2)/2$ ; since  $d \mid u_1 + u_2$  and  $\mathbf{u}_1\mathbf{u}_2$  is a repetition, therefore  $w + d < 2y$  or  $w - y < y - d$ . We conclude that  $d \leq w - y < y - d$ , so that case (d) of Lemma 8 applies (with  $x \sim y, u \sim d, v \sim w - y$ ):  $\mathbf{y}$ , hence by Lemma 7  $\mathbf{u}$ , is a repetition of period  $d$ . Since  $d$  divides both  $u$  and  $u_1 + u_2$ , it therefore divides  $u_1$ , hence  $u_2$ , hence  $w$ , and so  $d = \gcd(u_1, u_2, w)$  is a period of  $\mathbf{v}^2$ , as required. On the other hand, if  $z > 0$ , we compare  $\mathbf{w}^{(1)}$  and  $\mathbf{w}^{(2)}$  to discover that  $\mathbf{u} = R_{k+s}(\mathbf{u})$ . Since from (13)  $k + s = 0$  implies  $w = v$ , a case excluded by hypothesis, we conclude, again using Lemma 5 and Remark 6, that  $\mathbf{u}$  is a repetition of period  $d'' = \gcd(k + s, u)$ . Thus  $\mathbf{u}$  has two periods  $d$  and  $d''$  such that  $d + d'' \leq (u + u_1 + u_2)/2 < u$ , and so Lemma 4 applies. We conclude that  $\mathbf{u}$  is a repetition of period  $d^* = \gcd(d, d'')$ , where as above  $d^*$  divides  $u, u_1, u_2$  and  $w$ . Thus  $\gcd(u_1, u_2, w)$  is again a period of  $\mathbf{v}^2$ .

- (c) Let  $k' = w + k - v$  with corresponding  $k'$  as shown in Fig. 5; since  $w < v$ , therefore  $k' < k$ . Comparing the prefixes of  $\mathbf{w}^{(1)}$  and  $\mathbf{w}^{(2)}$ , we see that  $R_k(\mathbf{u}) = R_{k'}(\mathbf{u})$ , so that  $\mathbf{u}$  is a repetition of period  $\gcd(k - k', u) = \gcd(v - w, u)$ . A similar argument shows that  $\mathbf{u}_1\mathbf{u}_2$  is a repetition of period  $\gcd(v - w, u_1 + u_2)$ , and the argument of (a) shows that  $\mathbf{v}^2$  is again a repetition of period  $\gcd(u_1, u_2, w)$ .
- (d) Let  $s = u + u_1 - w - k$  with corresponding string  $\mathbf{s}$  as shown in Fig. 6, a suffix of  $\mathbf{u}_1$  and a prefix of  $\mathbf{w}$ . As in [8] we observe that  $\mathbf{u}_1^{(4)}$  occurs at position  $s + u_2 + 1$  of  $\mathbf{w}^{(2)}$ , hence at the same position of  $\mathbf{w}^{(1)}$ , thus overlapping  $\mathbf{u}_1^2$  in  $\mathbf{w}^{(1)}$  by  $t = s + u_2 - (u_1 + u_2 - k) = s + k - u_1 > 0$ . Thus  $\mathbf{u}_1 = R_t(\mathbf{u}_1)$  and we conclude, again using Lemma 5 and Remark 6, that  $\mathbf{u}_1$  is a repetition whose period  $d$  divides  $t$ , where  $d = u_1/p$  for some integer  $p > 1$ . We see then that  $d \mid u_1$  and  $d \mid s + k$ . Note also that since  $u - w = s + k - u_1$ , therefore  $d \mid u - w$ ; furthermore, since  $u - w = 2u_1 - (w - u_2)$ , it follows that  $d \mid w - u_2$ .  
Now let  $\mathbf{z}$  be the nonempty proper prefix of  $\mathbf{u}_2\mathbf{u}_1$  and suffix of  $\mathbf{w}$  defined as shown in Fig. 6 by  $\mathbf{z} = k + 2w - 2u$ . Then  $\mathbf{sz}$  is a border of  $\mathbf{w}$ , which therefore has period  $w - (s + z) < w$ . Consider now the substring  $\mathbf{w}[s + k - u_1 + 1..s + u_2 + u_1]$  of  $\mathbf{w}^{(2)}$  that must equal the prefix  $\mathbf{w}[1..2u_1 + u_2 - k]$  of  $\mathbf{w}^{(1)}$ . It follows that  $\mathbf{w}' = \mathbf{w}[1..s + u_2 + u_1]$  has period  $s + k - u_1 = u - w$ . Note from  $\mathbf{w}^{(2)}$  that  $u + z = w + u_1 - s$ , hence that  $u - w = u_1 - (s + z) < u_1$ . Since  $d$  divides  $u - w$ , the period of  $\mathbf{w}'$ , and since the suffix  $\mathbf{u}_1$  of  $\mathbf{w}'$  also has period  $u - w$  in addition to being a repetition of period  $d$ , it follows that  $\mathbf{w}'$  also has period  $d$ . Note now that since  $\mathbf{w}^{(2)}$  has suffix  $\mathbf{sz}$ , so also does  $\mathbf{w}^{(1)}$ , and thus, provided  $2s + z \leq u_1$ ,  $\mathbf{u}_1$  has suffix  $\mathbf{szs}$ . (That  $2s + z < u_1$  follows from the identities noted above:  $u - w = s + k - u_1 = u_1 - (s + z)$ ; that is,  $2s + z = 2u_1 - k$ .) Recalling that  $d \mid s + z$ , we see that periodicity  $d$  is extended from  $\mathbf{w}'$  with suffix  $\mathbf{szs}$  to  $\mathbf{w} = \mathbf{w}'\mathbf{z}$ .  
The preceding paragraph establishes that  $\mathbf{w}$  has periods  $p_1 = w - (s + z)$  and  $p_2 = d$ , where  $d \mid s + z$ , so that  $p_1 + p_2 \leq w$  and Lemma 4 can be applied, yielding the conclusion that  $\mathbf{w}$  has period  $d' = \gcd(w - (s + z), d)$ . But then  $d'$  must divide both  $d$  and  $w$ , hence also  $u_2$  (since  $d \mid w - u_2$ ) as well as  $u_1$ . Thus  $\mathbf{v}^2$  is a repetition of period  $\gcd(u_1, u_2, w)$ , as required.
- (e) As shown in Fig. 7,  $\mathbf{r}$  and  $\mathbf{s}$  are both suffixes of  $\mathbf{u}_2$  and prefixes of  $\mathbf{w}$ , where, since  $w < u$ , therefore  $r < s$ . Then  $\mathbf{s}$  has border  $\mathbf{r}$ , hence period  $s - r = u - w < s$ . Note that the string  $\mathbf{su}_1$  is a prefix of  $\mathbf{w}^{(2)}$ , hence of  $\mathbf{w}^{(1)}$ , so that  $\mathbf{su}_1$  overlaps with itself by  $s - r$ ; thus  $\mathbf{su}_1$  also has period  $u - w$ .  
Since  $v - s = k + w$ , therefore  $k + s = v - w < u$  by hypothesis, and so  $k + s + u_1 < u + u_1 < k + w$ . Thus the copy of  $\mathbf{u}_1$  in  $\mathbf{w}^{(2)}$  must overlap with  $\mathbf{u}_1^2$  in  $\mathbf{w}^{(1)}$ , telling us that  $\mathbf{u}_1 = R_{u-w}(\mathbf{u}_1)$ , where  $u - w < u_1$ . From Lemma 5 we

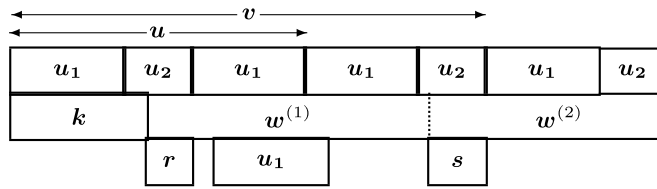


Fig. 7. (e) Subcase 9.

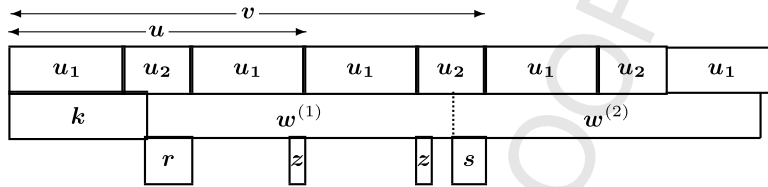


Fig. 8. (f) Subcase 10.

conclude that  $\mathbf{u}_1$  is a repetition of period  $u - w$ , and further that the prefix  $\mathbf{z} = \mathbf{r}\mathbf{u}_1^2$  of  $\mathbf{w}$  has period  $u - w$ . Observe that  $u - w \mid u_1$ , moreover since  $w - u_2 = 2u_1 - (u - w)$ , that  $u - w \mid w - u_2$ . Now consider the occurrence of  $\mathbf{u}^* = \mathbf{u}_2\mathbf{u}_1$  that has as its prefix a suffix of  $\mathbf{w}^{(2)}$ . Considering the corresponding suffix of  $\mathbf{w}^{(1)}$ , we see that

$$\mathbf{u}^*[1..u_1 - (s - r)] = \mathbf{u}_1[s - r + 1..u_1], \tag{14}$$

a suffix of  $\mathbf{u}_1$  of length  $u_1 - (s - r) = u_1 - (u - w) \geq u - w$ , since  $u_1 = q(u - w)$  for some  $q > 1$ . Therefore  $\mathbf{u}^*[1..u_1 - (s - r)]$  of length  $(q - 1)(u - w)$  has period  $u - w$ . Continuing the match, we find that

$$\mathbf{u}^*[u_1 - (s - r) + 1..w - (u_1 + s)] = \mathbf{u}^*[1..u_2 - s], \tag{15}$$

that by virtue of (14) continues the period  $u - w$ . Thus from (14) and (15) we conclude that  $\mathbf{w}$  has a suffix of length  $w - (u_1 + s)$  of period  $u - w$ , and as we saw earlier, it also has a prefix  $\mathbf{z}$  of length  $r + 2u_1$  of period  $u - w$ . The periodic suffix and prefix of  $\mathbf{w}$  have an overlap of

$$w - u_1 - s + r + 2u_1 - w = u_1 - (u - w),$$

as we have seen divisible by  $u - w$ . We conclude therefore that  $\mathbf{w}$  has period  $u - w$ .

Similarly, since (14)–(15) also tell us that  $\mathbf{u}_2$  has a prefix of length  $\min(u_2, w - (u_1 + s))$  of period  $u - w$ , and since as we have seen the suffix of  $\mathbf{u}_2$  of length  $s$  has the same period, we find the overlap

$$\begin{aligned} \min(u_2, w - (u_1 + s)) + s - u_2 &= \min(s, w - (u_1 + u_2)) \\ &= \min(s, u - (k + s)) \\ &= \min(s, u_1 - (s - t)), \end{aligned}$$

a quantity greater than  $u - w$ . We conclude therefore that  $\mathbf{u}_2$  also has period  $u - w$ . But the periodicity of  $\mathbf{w}$ , taken together with the facts that  $\mathbf{u}_1$  is a repetition of period  $u - w$  and that  $\mathbf{s}$  has period  $u - w$ , tell us that the string  $\mathbf{s}\mathbf{u}_2[1..u_2 - s] = R_{u_2-s}(\mathbf{u}_2)$  also has period  $u - w$ . By Lemma 5, therefore,  $\mathbf{u}_2$  is a repetition of period  $u - w$  and  $\gcd(u_2, u_2 - s)$ . Since as we have seen  $u - w$  also divides  $u_1$  and  $w - u_2$ , (d) holds.

(f) As shown in Fig. 8, let  $\mathbf{r}$  and  $\mathbf{s}$  be suffixes of  $\mathbf{u}_2$  that are also prefixes of  $\mathbf{w}$ ; since  $w > u$ , it follows that  $r > s$ , where we define  $z = r - s = w - u$ . Thus  $z < r < u_2$ , and so as shown we may let  $\mathbf{z}$  be a prefix of  $\mathbf{u}_2$ , so that  $\mathbf{w}^{(2)}$  has prefix  $\mathbf{s}\mathbf{u}_1\mathbf{z}$  that must equal prefix  $\mathbf{r}\mathbf{u}_1$  of  $\mathbf{w}^{(1)}$ . Thus  $\mathbf{w}^{(1)}$  must have prefix  $\mathbf{s}\mathbf{y} = \mathbf{s}(\mathbf{u}_1\mathbf{z})^2$ . Of course  $\mathbf{y}$  has period  $p_1 = u_1 + z$ , but since  $\mathbf{u}_1\mathbf{z}$  overlaps itself, therefore  $\mathbf{y}$  also has period  $p_2 = z$ . Since  $p_1 + p_2 = u_1 + 2z < y$ , we conclude by Lemma 4 that  $\mathbf{y}$  has period  $d = \gcd(u_1 + z, z)$ . Note that  $d$  divides  $z = w - u$  and  $u_1$ , thus  $w - u_2$  and  $y$ , where  $\mathbf{y}$  is a repetition of period  $d$ .

Next observe that since  $\mathbf{u}_2$  is a proper substring of  $\mathbf{w}^{(2)}$ , therefore for the largest integer  $j \geq 0$  such that  $\mathbf{z}(\mathbf{u}_1\mathbf{z})^j$  is a prefix of  $\mathbf{u}_2$ ,  $\mathbf{s}\mathbf{z}(\mathbf{u}_1\mathbf{z})^{j+1}$  must be a prefix of  $\mathbf{w}^{(1)}$ . This is because the  $j$ th occurrence of  $\mathbf{u}_1\mathbf{z}$  in  $\mathbf{u}_2$  within  $\mathbf{w}^{(2)}$  corresponds to the  $(j + 1)$ th occurrence of  $\mathbf{u}_1\mathbf{z}$  in  $\mathbf{w}^{(1)}$ . (Fig. 8 shows  $z < u_1$  and  $j = 0$ .)

Moreover  $R_s(\mathbf{w}) = \mathbf{z}\mathbf{u}_1^2\mathbf{u}_2$  must be a prefix of  $(\mathbf{u}_1\mathbf{z})^{j+2}$  and therefore has period  $d$ . Consequently  $\mathbf{u}_1\mathbf{u}_2$ , a prefix of  $\mathbf{v}$  that overlaps  $R_s(\mathbf{w})$  by  $\mathbf{z}$ , has period  $d$ , and since  $d \mid z$ , we find that  $\mathbf{v}$  has period  $d$ . Within  $\mathbf{v}$  both  $\mathbf{u}_2\mathbf{u}_1$  and  $\mathbf{u}_1\mathbf{u}_2$  occur, and so both have period  $d$ . If  $u_2 < u_1$ , let  $\mathbf{g} = \mathbf{u}_1\mathbf{u}_2$  and apply Lemma 8(d) to  $\mathbf{g}$  and  $R_{u_2}(\mathbf{g})$  to show that  $\mathbf{u}_2\mathbf{u}_1$ , thus  $\mathbf{v}^2$ , is a repetition of period  $d'$ , where  $d' \mid d$ , hence that  $d' \mid u_2$ , so that  $d' = \gcd(u_1, u_2, w)$ . Similarly for  $u_1 \leq u_2$ . This completes the proof.  $\square$

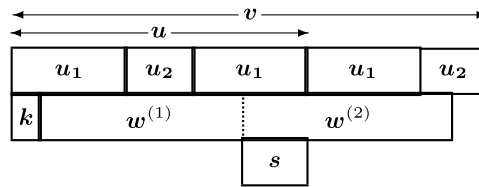


Fig. 9. Subcase 3.

## 6. Commentary and future research

An obvious direction for continuation of this research is to deal with the unproved conjectures of Table 3. We have already started to examine Subcase 3, illustrated in Fig. 9. A natural first angle of approach would also be to study the unproved subcases on the assumption that  $\sigma = d$ . Proofs for  $\sigma > d$  seem likely to be more difficult.

As noted in Section 4, there remain several other combinatorial puzzles related to (C2), that might well be relevant to future application of these results to theory and practice:

- \* It would be of interest to establish an upper bound on  $\sigma$  – it seems that  $\sigma \leq (u_1 + u_2)/2$ .
- \* Does  $\sigma = \gcd(u_1, u_2, w)$  imply that  $\mathbf{x}$  is a repetition of period  $d$ ? If so, can this fact be used to simplify proofs?
- \* If a string  $\mathbf{x}$  generated by function **force\_square** includes a maximum letter greater than alphabet size  $\sigma$ , does it follow that  $\sigma > \gcd(u_1, u_2, w)$ ?

Also for the case (C1),  $u < v < 3u/2$ , considered in Section 3, it seems that further analysis would yield a precise upper bound on the number of runs that takes into account the runs (R4).

We will maintain a website accessible from

<http://www.cas.mcmaster.ca/~bill/cv.shtml>

to monitor progress with three squares research. We remark that computer experiment has played a critical, perhaps indispensable, role in both the formulation and the proof of these fundamental combinatorial results.

## References

- [1] Alberto Apostolico, Franco P. Preparata, Optimal off-line detection of repetitions in a string, Theoret. Comput. Sci. 22 (1983) 297–315.
- [2] Gang Chen, Simon J. Puglisi, W.F. Smyth, Fast & practical algorithms for computing all the runs in a string, in: B. Ma, K. Zhang (Eds.), Proc. 18th Annual Symp. Combinatorial Pattern Matching, in: LNCS, vol. 4580, Springer-Verlag, 2007, pp. 307–315.
- [3] Maxime Crochemore, An optimal algorithm for computing all the repetitions in a word, Inform. Process. Lett. 12 (5) (1981) 244–248.
- [4] Maxime Crochemore, Lucian Ilie, Maximal repetitions in strings, J. Comput. System Sci. 74 (2008) 796–807.
- [5] Maxime Crochemore, Lucian Ilie, Computing longest previous factor in linear time and applications, Inform. Process. Lett. 106 (2008) 75–80.
- [6] Maxime Crochemore, Lucian Ilie, Liviu Tinta, Towards a solution to the “runs” conjecture, in: P. Ferragina, G. Landau (Eds.), Proc. 19th Annual Symp. Combinatorial Pattern Matching, in: LNCS, vol. 5029, Springer-Verlag, 2008, pp. 290–302.
- [7] Maxime Crochemore, Wojciech Rytter, Squares, cubes, and time-space efficient strings searching, Algorithmica 13 (1995) 405–425.
- [8] Kangmin Fan, Simon J. Puglisi, W.F. Smyth, Andrew Turpin, A new periodicity lemma, SIAM J. Discrete Math. 20 (3) (2006) 656–668.
- [9] Martin Farach, Optimal suffix tree construction with large alphabets, in: Proc. 38th IEEE Symp. Found. Computer Science, IEEE Computer Society, 1997, pp. 137–143.
- [10] N.J. Fine, H.S. Wilf, Uniqueness theorems for periodic functions, Proc. Amer. Math. Soc. 16 (1965) 109–114.
- [11] Mathieu Giraud, Not so many runs in strings, in: Carlos Martín-Vide, Friedrich Otto, Henning Fernau (Eds.), Proc. 2nd Int. Conference on Language and Automata Theory and Applications, in: LNCS, vol. 5196, Springer-Verlag, 2008, pp. 232–239.
- [12] Roman Kolpakov, Gregory Kucherov, On maximal repetitions in words, J. Discrete Algorithms 1 (2000) 159–186.
- [13] M. Lothaire, Combinatorics on Words, Cambridge University Press, 1997, 238 pp.
- [14] Michael G. Main, Detecting leftmost maximal periodicities, Discrete Appl. Math. 25 (1989) 145–153.
- [15] Michael G. Main, Richard J. Loretz, An  $O(n \log n)$  algorithm for finding all repetitions in a string, J. Algorithms 5 (1984) 422–432.
- [16] Simon J. Puglisi, R.J. Simpson, The expected number of runs in a word, Australas. J. Combin. 42 (2008) 45–54.
- [17] Simon J. Puglisi, R.J. Simpson, W.F. Smyth, How many runs can a string contain? Theoret. Comput. Sci. 401 (2008) 165–171.
- [18] Wojciech Rytter, The number of runs in a string: improved analysis of the linear upper bound, in: B. Durand, W. Thomas (Eds.), Proc. 23rd Symp. Theoretical Aspects of Computer Science, in: LNCS, vol. 2884, Springer-Verlag, 2006, pp. 184–195.
- [19] R.J. Simpson, Intersecting periodic words, Theoret. Comput. Sci. 374 (2007) 58–65.
- [20] Bill Smyth, Computing Patterns in Strings, Pearson Addison-Wesley, 2003, 423 pp.
- [21] Jacob Ziv, Abraham Lempel, A universal algorithm for sequential data compression, IEEE Trans. Inform. Theory 23 (1977) 337–343.