Some Restrictions on Periodicity in Strings ⋆

Simon J. Puglisi[1], W. F. Smyth[1,2], and Andrew Turpin[3]

[1] Department of Computing, Curtin University, GPO Box U1987
Perth WA 6845, Australia
puglissj@computing.edu.au

[2] Algorithms Research Group, Department of Computing & Software
McMaster University, Hamilton ON L8S 4K1, Canada
smyth@mcmaster.ca
www.cas.mcmaster.ca/cas/research/groups.shtml

[3] School of Computer Science & Information Technology
RMIT University, GPO Box 2476V
Melbourne V 3001, Australia
aht@cs.rmit.edu.au www.seg.rmit.edu.au

Abstract. Given a string $\boldsymbol{x} = \boldsymbol{x}[1..n]$, a ***repetition*** of period p in $\boldsymbol{x}$ is a substring $\boldsymbol{u}^r = \boldsymbol{x}[i..i{+}rp{-}1]$, $p = |\boldsymbol{u}|$, $r \geq 2$, where neither $\boldsymbol{u} = \boldsymbol{x}[i..i{+}p{-}1]$ nor $\boldsymbol{x}[i..i{+}(r{+}1)p{-}1]$ is a repetition. The maximum number of repetitions in any string $\boldsymbol{x}$ is well known to be $\Theta(n \log n)$. A ***run*** or ***maximal periodicity*** of period p in $\boldsymbol{x}$ is a substring $\boldsymbol{u}^r\boldsymbol{t} = \boldsymbol{x}[i..i+rp+|\boldsymbol{t}|-1]$ of $\boldsymbol{x}$, where $\boldsymbol{u}^r$ is a repetition, $\boldsymbol{t}$ a proper prefix of $\boldsymbol{u}$, and no repetition of period p begins at position $i-1$ of $\boldsymbol{x}$ or ends at position $i+rp+|\boldsymbol{t}|$. In 2000 Kolpakov & Kucherov showed that the maximum number $\rho(n)$ of runs in any string $\boldsymbol{x}$ is $O(n)$, but their proof was nonconstructive and provided no specific constant of proportionality. At the same time, they presented experimental data strongly suggesting that $\rho(n) < n$. that the maximum any string $\boldsymbol{x}$ again encourages the belief that in fact $\sigma(n) < n$. Recently, Fan et al.("A new periodicity lemma", *Sixteenth Annual Symp. Combin. Pattern Matching*, 2005) took a first step toward proving these conjectures, by presenting results that establish limitations on the number of squares of a specified range of periods that can occur over a specified range of positions in $\boldsymbol{x}$. In this paper, we further tighten these restrictions by showing how the existence of two squares $\boldsymbol{u}$ and $\boldsymbol{v}$ ($\boldsymbol{v}$ longer than $\boldsymbol{u}$) at the same position i in $\boldsymbol{x}$ limits the occurrence of smaller squares with period $w \in (|\boldsymbol{v}| - |\boldsymbol{u}|, |\boldsymbol{u}|)$ in the neighborhood around i.

1 Introduction

Repetitions and other forms of periodicity have long been considered fundamental characteristics of strings. In fact, the work often cited as having

⋆ Supported in part by grants from the Natural Sciences & Engineering Research Council of Canada.

founded stringology [21], is an investigation of the periodicity properties of infinite strings. Today, the detection of repetitions has become of practical interest, primarily in the field of bioinformatics, with algorithms for the task a standard part of any software for whole genome analysis.

In this paper we extend recent results of Fan et al. [4] that specify restrictions on the nature and extent of periodic behaviour in strings. It is our hope that these theoretical results will eventually lead to more straightforward algorithms for detecting repetitions than those currently available.

Throughout we use boldface to represent strings, and italics to specify their lengths. For instance, the string under consideration is denoted $\boldsymbol{x} = \boldsymbol{x}[1..n]$, and its length is $x = |\boldsymbol{x}|$. We will also use n to refer to the length of $\boldsymbol{x}$ as is customary. We write $\boldsymbol{u}^k$ to represent a concatenation of k occurrences of the string $\boldsymbol{u}$.

A ***repetition*** in $\boldsymbol{x}$ is a substring $\boldsymbol{u}^r = \boldsymbol{x}[i..i+ru-1]$, $r \geq 2$, where neither $\boldsymbol{x}[i..i+u-1]$ nor $\boldsymbol{x}[i..i+(r+1)u-1]$ is a repetition. We call $\boldsymbol{u}$ the ***generator***, u the ***period*** of the repetition, and r the ***exponent***. We refer to a repetition where $k = 2$, $\boldsymbol{u}^2$, as a ***square***. A repetition can be encoded as an integer triple (i, u, r). In order to compute all repetitions efficiently Main [16] introduced a ***run*** or ***maximal periodicity*** of period u in $\boldsymbol{x}$ is a substring $\boldsymbol{u}^r\boldsymbol{t} = \boldsymbol{x}[i..i+ru+t-1]$, where $\boldsymbol{u}^r$ is a repetition, $\boldsymbol{t}$ a proper prefix of $\boldsymbol{u}$, and no repetition of period u begins at position $i-1$ of $\boldsymbol{x}$ or ends at position $i+ru+t$. $\boldsymbol{u}$ is called the ***generator*** of the run, $\boldsymbol{t}$ its ***tail***, and a run is economically represented by a 4-tuple (i, u, r, t). The critical observation that a run encapsulates t adjacent repetitions all having the same period implies that there are at most as many runs as repetitions. Further, by computing all runs we are implicitly computing all repetitions.

Kolpakov & Kucherov [13] describe an algorithm to compute all the runs (hence all the repetitions) in $\boldsymbol{x}$ in $\Theta(n)$ time. Their algorithm is essentially an extension of an earlier algorithm by Main [16] which guaranteed only computation of the "leftmost" runs. The complexity of Kolpakov & Kucherov's algorithm hinges on a lengthy and technical proof [13] that the maximum number $\rho(n)$ of runs that could exist in any string $\boldsymbol{x}$ satisfies:

$$k_1 n - k_2 \log_2 n \sqrt{n}, \tag{1}$$

where k_1 and k_2 are positive constants.

Remarkable though it is, there is a problem with (1): the proof is nonconstructive, providing no information about the magnitude of the constants k_1 and k_2. Nevertheless Kolpakov & Kucherov provide experimental evidence to prompt the following conjectures [20]:

* $\rho(n) < n$;
* $\rho(n)$ is achieved by a cube-free string $\boldsymbol{x}$ on alphabet $\{a, b\}$;
* $\rho(n+1) \leq \rho(n)+2$.

These questions of periodicity seem fundamental yet, so far, progress toward answering them has been scant. Franek et al. [8] bolster the first conjecture by the construction of an infinite family of strings which is very "run-rich" but always has $\rho(n) < n$.

In order to show that in general $\rho(n) < n$, it seems to be necessary to establish restrictions on the squares (with which runs must begin) that can occur in the neighbourhood of positions in a string at which one or two squares already appear. Very recently, Fan et al [4] proved several results in this direction, culminating in the following Lemma.

Definition 1 *A square $\boldsymbol{u}^2$ is said to be **regular** if no prefix of $\boldsymbol{u}$ is a square.*

Definition 2 *A square $\boldsymbol{v}^2$ is said to be **irreducible** if $\boldsymbol{v}$ is not a repetition.*

Lemma 3 [4, New Periodicity Lemma] *If $\boldsymbol{x}$ has regular prefix $\boldsymbol{u}^2$ and irreducible prefix $\boldsymbol{v}^2$, $u < v < 2u$, then for every $w \in (u, v)$ and for every $k \in [0, v-u)$, $\boldsymbol{x}[k+1..k+2w]$ is not a square.*

The lemma essentially restricts the occurrence of squares (or runs) having period between u and v. The results we present in Section 2 are an extension of this result for periods between $v - u$ and u.

We make use of two further lemmas from Fan et al. [4].

Lemma 4 [4, Lemma 8] *If $\boldsymbol{v}^2$ is irreducible with regular proper prefix $\boldsymbol{u}^2$, then*

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

Lemma 5 [4, Lemma 9] *If $\boldsymbol{x} = \boldsymbol{v}^2$ is irreducible with regular proper prefix $\boldsymbol{u}^2$, $v < 2u$, then*

$$\boldsymbol{x} = \boldsymbol{u_1 u_2 u_1 u_1 u_2 u_1 u_2 u_1 u_1 u_2},$$

where $u_1 = 2u-v, u_2 = 2v-3u$ (depicted in Figure 1).

The following terminology is also helpful. A substring of a given string $\boldsymbol{x}$ is said to be ***internal*** if and only if it is neither a prefix nor a suffix of $\boldsymbol{x}$. And if $\boldsymbol{x} = \boldsymbol{x_1 x_2}$, $\boldsymbol{x_2}$ nonempty, then $\boldsymbol{x_2 x_1} = R_{x_1}(\boldsymbol{x})$ is said to be the x_1^{th} ***rotation*** of $\boldsymbol{x}$.

2 Restricting Occurrence of Smaller Periods

As in Fan et al. [4] we consider the situation in which a regular square $\boldsymbol{u}^2$ and an irreducible square $\boldsymbol{v}^2$ occur at the same position. Our main result restricts squares with period $w \in (v-u, u)$ from occurring in a range about the center of the first occurrence of $\boldsymbol{u}$.

Lemma 6 *If $\boldsymbol{x}$ has a regular prefix of $\mathbf{u}^2$ and an irreducible prefix of $\boldsymbol{v}^2$, $u < v < 2u$, then for every period $w \in (v-u, u)$ and for every starting position $k \in [u_1, v-u)$, $\boldsymbol{x}[k+1..k+2w]$ is not a square.*

Proof. The proof is by contradiction. Suppose that for $u_1 \le k < v-u$ and $v-u < w < u$, the square $\boldsymbol{w}^2$ occurs at $\boldsymbol{x}[k+1..k+2w]$. Making use of the notation of Lemma 5, we consider two main cases, when k is small and when k is large, and show that in both cases the suffix of $\boldsymbol{w}$ that is also a prefix of $\boldsymbol{u}$ contains a square, violating the restriction that $\boldsymbol{u}$ is regular.

Case I, $k+w$ is small: $k+w < u+u_1$

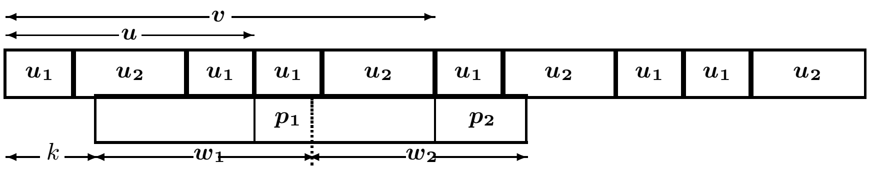

Fig. 1. Case I: when $k+w$ is small

Figure 1 shows the string $\boldsymbol{x}$ split into $\boldsymbol{u_1}$ and $\boldsymbol{u_2}$ substrings as stated in Lemma 5. Also shown are the first and second copy of $\boldsymbol{w}$, labeled $\boldsymbol{w_1}$ and $\boldsymbol{w_2}$ respectively. As $u_1 \le k$, the first copy of $\boldsymbol{w}$ must begin somewhere in the first copy of $\boldsymbol{u_2}$. As $w < u = 2u_1 + u_2$, $\boldsymbol{w_1}$ must finish somewhere in the third copy of $\boldsymbol{u_1}$. This is drawn in Figure 1. Also shown is the suffix of $\boldsymbol{w_1}$ that begins in the third copy of $\boldsymbol{u_1}$, which is labeled $\boldsymbol{p_1}$.

As a result, the second copy of $\boldsymbol{w}$, $\boldsymbol{w_2}$, must begin in the third $\boldsymbol{u_1}$ and finish somewhere in the third $\boldsymbol{u_1 u_2}$ substring. Let $\boldsymbol{p_2}$ be the prefix of $\boldsymbol{u}$ that is occupied by the suffix of $\boldsymbol{w_2}$ as shown in Figure 1. From the restrictions mentioned, we can see that the length of $\boldsymbol{w_2}$ is such that $w_2 = (u_1 - p_1) + u_2 + p_2$. Seeing as $u_1 + u_2 < w$, then $u_1 + u_2 < (u_1 - p_1) + u_2 + p_2$, hence $p_1 < p_2$.

This implies $\boldsymbol{w}$ ends with two *distinct* prefixes of $\boldsymbol{u_1 u_2}$ — $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$. Now, $p_1 = k{+}w - u$ and $p_2 = k{+}2w - v$ and so

$$\frac{p_2}{p_1} = \frac{k{+}2w{-}v}{k{+}w{-}u} \geq \frac{k{+}2w{-}(2u{-}k)}{k{+}w{-}u} = 2$$

so $p_1 \geq p_2/2$.

Case II, $k{+}w$ is large: $u{+}u_1 < k{+}w < u{+}u_1{+}u_2$

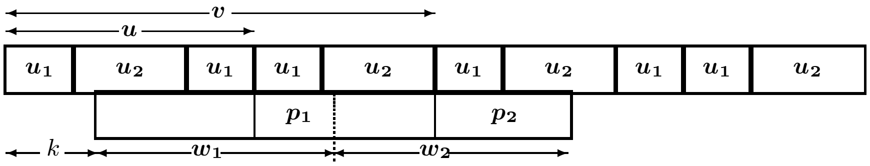

Fig. 2. Case II: when $k{+}w$ is large

Similar to Case I, we split $\boldsymbol{w}^2$ into $\boldsymbol{w_1}$ and $\boldsymbol{w_2}$, defining $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$ as their respective suffixes that are also prefixes of $\boldsymbol{u_1 u_2}$; see Figure 2. As before, $p_1 = k{+}w - u$ and $p_2 = k{+}2w - v$ and so

$$p_2 = k + 2w - 2u + u_1 = 2p_1 + u_1 - k < 2p_1.$$

In both cases $\boldsymbol{p_1}$ is both a prefix and suffix of $\boldsymbol{p_2}$, and $p_2/2 \leq p_1 < p_2$. If $p_1 = p_2/2$, then $\boldsymbol{p_2} = \boldsymbol{p_1 p_1}$, and $\boldsymbol{u}$ begins with a square, contradicting the assumption that $\boldsymbol{u}$ is regular. If $p_1 > p_2/2$ then a suffix of $\boldsymbol{p_1}$ is also a prefix of $\boldsymbol{p_1}$ (the end of the first $\boldsymbol{p_1}$ must overlap the second $\boldsymbol{p_1}$ in $\boldsymbol{p_2}$), say $\boldsymbol{p_3}$, and so $\boldsymbol{p_2} = \boldsymbol{p_3 y p_3 y p_3}$, for some substring $\boldsymbol{y}$. Again, $\boldsymbol{p_2}$ commences with a square, $(\boldsymbol{p_3 y})^2$, contradicting the assumption that $\boldsymbol{u}$ is regular. □

Lemma 7 *If $\boldsymbol{x}$ has a regular prefix of $\boldsymbol{u}^2$ and an irreducible prefix of $\boldsymbol{v}^2$, $u < v < 2u$, then for every $w \in (3u_1/2{+}u_2, u)$ and for every $k \in [0, u_1/2]$, $\boldsymbol{x}[k{+}1..k{+}2w]$ is not a square.*

Proof. Suppose that for $k \leq u_1/2$ and $w \in (3u_1/2 + u_2, u)$, the square $\boldsymbol{w}^2$ occurs at $\boldsymbol{x}[k{+}1..k{+}2w]$.

See Figure 3. Let $s = k + w - u < k$. Observe that

$$\begin{aligned}\boldsymbol{w_2} &= \boldsymbol{x}[w{+}k..2w + k] \\ &= \boldsymbol{x}[u + s..u + s + w] \\ &= \boldsymbol{x}[u + s..2u]\boldsymbol{u_2}[1..K] \\ &= \boldsymbol{x}[s..u]\boldsymbol{u_2}[1..K]\end{aligned}$$

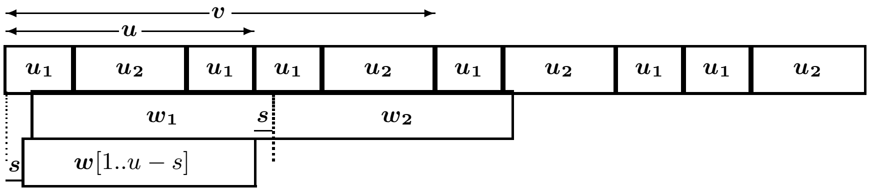

Fig. 3. Position of $\boldsymbol{w}^2$ in Lemma 7

So $\boldsymbol{x}[s..u]$ consists of two overlapping prefixes of $\boldsymbol{w}$, namely

$$\boldsymbol{x}[s..u] = \boldsymbol{w2}[1..u-s],$$
$$\boldsymbol{x}[k..u] = \boldsymbol{w1}[1..w-s]$$

Therefore $\boldsymbol{x}[s..u]$ has period $z = k - s < u_1/2$. Because $\boldsymbol{u_1}$ is an internal substring of x[k..] and $u_1 \geq 2z$, $\boldsymbol{u_1} = \boldsymbol{R(z)}^r \boldsymbol{R(z)}^*$, $r \geq 2$ and so $\boldsymbol{u}$ begins with a square, a contradiction to $\boldsymbol{u}$ being regular. We therefore conclude that such a $\boldsymbol{w}^2$ cannot exist. □

Note that the above results are directly applicable to runs. Observe first that by definition every run is irreducible. Observe also that if a run of period u and tail $\boldsymbol{t}$ occurs at position i in $\boldsymbol{x}$, no run of the same period can occur at any position $j \in [i, i{+}u{+}t]$. Thus, if we define a ***regular run*** to be a run of generator $\boldsymbol{u}$ where $\boldsymbol{u}^2$ is a regular square, we can state an equivalent of Lemma 6 and Lemma 7 for runs.

3 Discussion

We have proved two lemmata (6 and 7) that extend the results of Fan et al. [4] and restrict the periods w of squares that can occur at positions $i{+}k$ in $\boldsymbol{x}$ when at position i two squares are known to occur. It is our hope that these results will be of some help in making progress with the three conjectures arising out of Kolpakov & Kucherov's work [13].

The Main/Kolpakov-Kucherov algorithm [13, 16] is the only known linear-time algorithm for computing all the runs in a given string $\boldsymbol{x}$. It is complex and, until recently, depended for its worst-case linear behaviour on the use of Farach's algorithm [5], also complex and not space-efficient, for linear-time computation of suffix trees. Since 2003 three worst-case linear-time suffix array construction algorithms [10–12] have been available for use in the computation of the LZ factorization [1], but even after the substitution of suffix arrays for suffix trees in the all-runs algorithm, significant complications remain. For instance, the algorithm still requires at least $13n$ bytes of space. Further, it appears that due to their recursive nature the linear-time algorithms are not in practice the fastest suffix array construction algorithms available [19]. We expect that, with a more precise understanding of the periodicity of runs, it will become possible to design simpler algorithms that will compute all the runs in a string in a more direct and more efficient manner.

References

1. M. ABOUELHODA, S. KURTZ & E. OHLEBUSCH, Replacing Suffix Trees with Enhanced Suffix Arrays, J. Discrete Algorithms 2–1 (2004) pp. 53–86.
2. A. APOSTOLICO & F. P. PREPARATA, Optimal off-line detection of repetitions in a string, Theoret. Comput. Sci. 22 (1983) pp. 297–315.
3. M. CROCHEMORE, An optimal algorithm for computing the repetitions in a word, Inform. Process. Lett. 12–5 (1981) pp. 244–250.
4. K. FAN, W. F. SMYTH & R. J. SIMPSON, A new periodicity lemma, *Sixteenth Annual Symp. Combin. Pattern Matching* (2005) to appear.
5. M. FARACH, Optimal suffix tree construction with large alphabets, Proc. 38^{th} IEEE Symp. Found. Comput. Sci. (1997) pp. 137–143.
6. N. J. FINE & H. S. WILF, Uniqueness theorems for periodic functions, Proc. Amer. Math. Soc. 16 (1965) pp. 109–114.
7. A. S. FRAENKEL & R. J. SIMPSON, How many squares can a string contain?, J. Combin. Theory Ser. A 82 (1998) pp. 112–120.
8. F. FRANEK, R. J. SIMPSON & W. F. SMYTH, The maximum number of runs in a string, Proc. 14^{th} Australasian Workshop on Combin. Algorithms, Mirka Miller & Kunsoo Park (eds.) (2003) pp. 26–35.
9. L. ILIE, A simple proof that a word of length n has at most $2n$ distinct squares, J. Combin. Theory Ser. A (2005) to appear.
10. J. KÄRKKÄINEN & P. SANDERS, Simple linear work suffix array construction, Proc. 30^{th} Internat. Colloq. Automata, Languages & Programming (2003) pp. 943–955.
11. D. K. KIM, J. S. SIM, H. PARK & K. PARK, Linear-time construction of suffix arrays, Proc. 14^{th} Annual Symp. Combin. Pattern Matching, R. Baeza-Yates, E. Chávez & M. Crochemore (eds.), LNCS 2676, Springer-Verlag (2003) pp. 186–199.
12. P. KO & S. ALURU, Space efficient linear time construction of suffix arrays, Proc. 14^{th} Annual Symp. Combin. Pattern Matching, R. Baeza-Yates, E. Chávez & M. Crochemore (eds.), LNCS 2676, Springer-Verlag (2003) pp. 200–210.
13. R. KOLPAKOV & G. KUCHEROV, On maximal repetitions in words, J. Discrete Algorithms 1 (2000) pp. 159–186.
14. A. LEMPEL & J. ZIV, On the complexity of finite sequences, IEEE Trans. Information Theory 22 (1976) pp. 75–81.
15. M. LOTHAIRE, Algebraic Combinatorics on Words, Cambridge University Press (2002) 504 pp.
16. M. G. MAIN, Detecting leftmost maximal periodicities, Discrete Applied Maths. 25 (1989) pp. 145–153.
17. M. G. MAIN & R. J. LORENTZ, An $O(n \log n)$ algorithm for finding all repetitions in a string, J. Algs. 5 (1984) pp. 422–432.
18. E. M. MCCREIGHT, A space-economical suffix tree construction algorithm, J. Assoc. Comput. Mach. 32–2 (1976) pp. 262–272.

19. S. J. PUGLISI, W. F. SMYTH & A. TURPIN, The performance of linear time suffix sorting algorithms, Proc. Data Compression Conf., J. Storer & M. Cohn (eds.) (2005) pp. 358–367.
20. B. SMYTH, Computing Patterns in Strings, Pearson Addison-Wesley (2003) 423 pp.
21. A. THUE, Über unendliche zeichenreihen, Norske Vid. Selsk. Skr. I. Mat. Nat. Kl. Christiana 7 (1906) pp. 1-22.
22. P. WEINER, Linear pattern matching algorithms, Proc. 14th Annual IEEE Symp. Switching & Automata Theory (1973) pp. 1–11.