The maximal number of cubic runs in a string

Maxime Crochemore, Costas S. Iliopoulos, Marcin Kubica, Jakub Radoszewski, Wojciech Rytter, Tomasz Walen

The Maximal Number of Cubic Runs in a String

M. Crochemore[a,b], C. Iliopoulos[a,c], M. Kubica[d], J. Radoszewski[d,*], W. Rytter[d,e], T. Waleń[d]

[a] *King's College London, London WC2R 2LS, UK*
[b] *Université Paris-Est, France*
[c] *Digital Ecosystems & Business Intelligence Institute, Curtin University of Technology, Perth WA 6845, Australia*
[d] *Dept. of Mathematics, Computer Science and Mechanics, University of Warsaw, ul. Banacha 2, 02-097 Warsaw, Poland*
[e] *Dept. of Math. and Informatics, Copernicus University, ul. Chopina 12/18, 87-100 Toruń, Poland*

Abstract

A run is an inclusion maximal occurrence in a string (as a subinterval) of a factor in which the period repeats at least twice. The maximal number of runs in a string of length n has been thoroughly studied, and is known to be between $0.944\,n$ and $1.029\,n$. The proofs are very technical. In this paper we investigate cubic runs, in which the period repeats at least three times. We show the upper bound on their maximal number, $\mathsf{cubic\text{-}runs}(n)$, in a string of length n: $\mathsf{cubic\text{-}runs}(n) < 0.5\,n$. The proof of linearity of $\mathsf{cubic\text{-}runs}(n)$ utilizes only simple properties of Lyndon words and is considerably simpler than the corresponding proof for general runs. For binary strings, we provide a better upper bound $\mathsf{cubic\text{-}runs}_2(n) < 0.48\,n$ which requires computer-assisted verification of a large number of cases. We also construct an infinite sequence of words over binary alphabet for which the lower bound is $0.41\,n$. [1]

Keywords: run in a string, Lyndon word, Fibonacci string.

*Corresponding author. Some parts of this paper were written during the corresponding author's Erasmus exchange at King's College London

[1] In particular, we improve both the lower and the (binary) upper bound from the conference version of the paper [7].

1. Introduction

Repetitions and periodicities in strings are one of the fundamental topics in combinatorics on words [2, 14]. They are also important in other areas: lossless compression, word representation, computational biology etc. Repetitions are studied from different points of view: classification of words not containing repetitions of a given exponent, efficient identification of factors being repetitions of different types and, finally, computing the bounds on the number of repetitions of a given exponent that a string may contain, which we consider in this paper. Both the known results in the topic and a deeper description of the motivation can be found in a survey by Crochemore et al. [5].

The concept of runs (also called maximal repetitions) has been introduced to represent all repetitions in a string in a succinct manner. The crucial property of runs is that their maximal number in a string of length n (denoted as $\mathsf{runs}(n)$) is $O(n)$, see Kolpakov & Kucherov [11]. This fact is the cornerstone of any algorithm computing all repetitions in strings of length n in $O(n)$ time. Due to the work of many people, much better bounds on $\mathsf{runs}(n)$ have been obtained. The lower bound $0.927\,n$ was first proved by Franek & Yang [9]. Afterwards, it was improved by Kusano et al. [13] to $0.944565\,n$ employing computer experiments, and recently by Simpson [20] to $0.944575712\,n$. On the other hand, the first explicit upper bound $5\,n$ was settled by Rytter [17], afterwards it was systematically improved to $3.48\,n$ by Puglisi et al. [16], $3.44\,n$ by Rytter [19], $1.6\,n$ by Crochemore & Ilie [3, 4] and $1.52\,n$ by Giraud [10]. The best known result $\mathsf{runs}(n) \leq 1.029\,n$ is due to Crochemore et al. [6], but it is conjectured [11] that $\mathsf{runs}(n) < n$. The maximal number of runs was also studied for special types of strings and tight bounds were established for Fibonacci strings [11, 18] and more generally Sturmian strings [1].

The combinatorial analysis of runs is strongly related to the problem of estimation of the maximal number of squares in a string. In the latter problem the gap between the upper and lower bound is much larger than for runs [5, 8]. However, a recent paper [12] by some of the authors shows that introduction

of integer exponents larger than 2 may lead to obtaining tighter bounds for the number of corresponding repetitions.

In this paper we introduce and study the concept of cubic runs in which the period is at least three times shorter than the run itself. We show the following bounds on their maximal number, $\mathsf{cubic\text{-}runs}(n)$, in a string of length n:

$$0.41\,n < \mathsf{cubic\text{-}runs}(n) < 0.5\,n\ .$$

The upper bound is achieved by analysis of Lyndon words (i.e. words that are primitive and minimal/maximal in the class of their cyclic equivalents) that appear as periods of cubic runs (Section 4). In Section 6 we improve this bound for *binary* words to $0.48\,n$ by examining short factors of the string.

As for the lower bound, we describe an infinite family of binary words that contain more than $0.41\,n$ cubic runs (Section 5). The proof of this property utilizes results obtained by analyzing the structure of cubic runs in Fibonacci strings, described in Section 3.

2. Preliminaries

We consider *words* (*strings*) u over a finite alphabet Σ, $u \in \Sigma^*$; the empty word is denoted by ε; the positions in u are numbered from 1 to $|u|$. By Σ^n we denote the set of all words of length n from Σ^*. By u^R we denote the reversed word u. By $\mathsf{Alph}(u)$ we denote the set of all letters of u. For $u = u_1u_2\ldots u_n$, let us denote by $u[i\,..\,j]$ a *factor* of u equal to $u_i\ldots u_j$ (in particular $u[i] = u[i\,..\,i]$). Words $u[1\,..\,i]$ are called prefixes of u, and words $u[i\,..\,n]$ — suffixes of u.

We say that a positive integer q is the (shortest) *period* of a word $u = u_1\ldots u_n$ (notation: $q = \mathsf{per}(u)$) if q is the smallest positive number, such that $u_i = u_{i+q}$ holds for all $1 \le i \le n-q$.

If $u = w^k$ (k is a non-negative integer), that is $u = ww\ldots w$ (k times), then we say that u is the k^{th} power of the word w. A *square* is the 2^{nd} power of some non-empty word. The *primitive root* of a word u, denoted $\mathsf{root}(u)$, is the shortest word w such that $w^k = u$ for some positive integer k. We call a word u *primitive* if $\mathsf{root}(u) = u$, otherwise it is called *non-primitive*. We say that words

n	3	4	5	6	7	8	9	10	11
$\mathsf{cubic\text{-}runs}_2(n)$	1	1	1	2	2	2	3	3	3
n	12	13	14	15	16	17	18	19	20
$\mathsf{cubic\text{-}runs}_2(n)$	4	4	5	5	5	6	7	7	7
n	21	22	23	24	25	26	27	28	29
$\mathsf{cubic\text{-}runs}_2(n)$	8	8	8	9	9	10	10	10	11

Table 1: The maximum number $\mathsf{cubic\text{-}runs}_2(n)$ of cubic runs in a binary string of length n for $n = 3, \ldots, 29$. Example binary words for which the maximal number of cubic runs is attained are shown in the following Table 2.

u and v are cyclically equivalent (or that one of them is a cyclic rotation of the other) if $u = xy$ and $v = yx$ for some $x, y \in \Sigma^*$. It is a simple and well-known observation, that if u and v are cyclically equivalent then $|\mathsf{root}(u)| = |\mathsf{root}(v)|$.

A *run* (also called a maximal repetition) in a string u is an interval $[i\,..\,j]$ such that:

- the period q of the associated factor $u[i\,..\,j]$ satisfies $2q \leq j - i + 1$,
- the interval cannot be extended to the left nor to the right, without violating the above property, that is, $u[i-1] \neq u[i+q-1]$ and $u[j-q+1] \neq u[j+1]$, provided that the respective letters exist.

By $\mathcal{R}(u)$ we denote the set of runs in u.

A *cubic run* is a run $[i\,..\,j]$ for which the shortest period q satisfies $3q \leq j - i + 1$. By $\mathcal{CR}(u)$ we denote the set of cubic runs in u, additionally denote $\mathsf{cubic\text{-}runs}(u) = |\mathcal{CR}(u)|$. For positive integer n, by $\mathsf{cubic\text{-}runs}(n)$ we denote the maximum of $\mathsf{cubic\text{-}runs}(u)$ for all $u \in \Sigma^*$ of length n, and by $\mathsf{cubic\text{-}runs}_2(n)$ we denote the maximum over all such binary strings.

For simplicity, in the rest of the text we sometimes refer to runs or cubic runs as to occurrences of corresponding factors of u.

Example. All cubic runs for an example Fibonacci word are shown in Figure 1.

n	cubic-runs$_2(n)$	u
3	1	000
6	2	000111
9	3	000111000
12	4	000100010001
14	5	00010001000111
17	6	00010001000111000
18	7	000111000111000111
21	8	000111000111000111000
24	9	000111000111000111000111
26	10	00010001000111000111000111
29	11	00010001000111000111000111000

Table 2: Lexicographically smallest binary words $u \in \{0,1\}^n$, for which cubic-runs(u) = cubic-runs$_2(n)$ (see also Table 1).

3. Fibonacci Strings

Let us start by analyzing the behavior of function cubic-runs for a very common benchmark in text algorithms, i.e. the Fibonacci strings, defined recursively as:

$$F_0 = a, \quad F_1 = ab, \quad F_n = F_{n-1}F_{n-2} \quad \text{for} \quad n \geq 2 \ .$$

Denote by $\Phi_n = |F_n|$, the n^{th} Fibonacci number (we assume that for $n < 0$, $\Phi_n = 1$) and by g_n the word F_n with the last two letters removed.

Lemma 1. *[15, 18] Each run in F_n is of the form $F_k \cdot F_k \cdot g_{k-1}$ (short runs) or $F_k \cdot F_k \cdot F_k \cdot g_{k-1}$ (long runs), each of the runs of period Φ_k.*

Obviously, in Lemma 1 only runs of the form $F_k^3 \cdot g_{k-1}$ are cubic runs.

Denote by $\#occ(u, v)$ the number of occurrences (as a factor) of a word u in a word v.

Lemma 2. *For every $k, n \geq 0$,*

$$\#occ(F_k^3 \cdot g_{k-1},\ F_n) \ = \ \#occ(F_k^3,\ F_n) \ .$$

Proof. Each occurrence of F_k^3 within F_n must be followed by g_{k-1}, since otherwise it would form a run different from those specified in Lemma 1. □

Lemma 3. *For every $k \geq 2$ and $m \geq 0$,*

a) $\#occ(F_k^3,\ F_{m+k}) = \#occ(aaba,\ F_m)$,

b) $\#occ(aaba,\ F_m) = \Phi_{m-3} - 1$.

PROOF. Recall the Fibonacci morphism φ:

$$\varphi(a) = ab, \quad \varphi(b) = a\ .$$

Recall that $F_n = \varphi^n(a)$. The following claim provides a useful tool for the proof of items (a) and (b).

Claim 4. *Assume $F_n = uvw$, where $u, v, w \in \{a, b\}^*$, $v[1] = a$ and either $w[1] = a$ or $w = \varepsilon$. Then there exist unique words u', v', w' such that*

$$u = \varphi(u'),\ v = \varphi(v'),\ w = \varphi(w'),\ F_{n-1} = u'v'w'\ .$$

And conversely, if v' is a factor of some F_{n-1} and $v = \varphi(v')$ then v is a factor of F_n.

PROOF. It is a straightforward consequence of the definition of φ and the fact that $F_n = \varphi(F_{n-1})$. □

Now we proceed to the actual proof of the lemma.

We prove item (a) by induction on k. For $k = 2$ we show the following equalities:

$$\#occ(abaabaaba, F_{m+2}) \ = \ \#occ(ababaa, F_{m+1}) \ = \ \#occ(aaba, F_m)\ . \quad (1)$$

As for the first of the equalities (1), the occurrence of F_2^3 within F_{m+2} cannot be followed by the letter a (since this would imply a larger run, contradicting Lemma 1) and cannot be a suffix of F_{m+2} (since either F_4 or F_5 is a suffix of F_{m+2}). Thus,

$$\#occ(abaabaaba, F_{m+2}) \ = \ \#occ(abaabaabab, F_{m+2}) \ = \ \#occ(ababaa, F_{m+1})\ .$$

The latter of the above equalities holds due to Claim 4, which applies here since no occurrence of $abaabaabab$ in F_{m+2} can be followed by the letter b (bb is not a factor of any Fibonacci string).

To prove the second equality (1), we apply a very similar approach: $ababaa$ is not a suffix of F_{m+1} and its occurrence cannot be followed by the letter a, since no Fibonacci string contains the factor aaa. Hence, by Claim 4,

$$\#occ(ababaa, F_{m+1}) = \#occ(ababaab, F_{m+1}) = \#occ(aaba, F_m) \ .$$

Finally, the inductive step for $k \geq 3$ also follows from Claim 4. Indeed, F_k^3 starts with the letter a and any of its occurrences in F_{m+k} is followed by the letter a, since, by Lemma 1, it is a part of a larger run $F_k^3 \cdot g_{k-1}$. Thus,

$$\#occ(F_k^3,\ F_{m+k}) = \#occ(F_{k-1}^3,\ F_{m+k-1}) \ .$$

The proof of item (b) goes by induction on m. For $m \leq 3$ one can easily check that $\#occ(aaba,\ F_m) = 0$, and there is exactly one occurrence of $aaba$ in F_4. The inductive step is a conclusion of the fact that for $m \geq 5$ the word F_m contains all occurrences of $aaba$ from F_{m-1} and F_{m-2} and one additional occurrence overlapping their concatenation:

$$\ldots ab\underbrace{a \mid aba}ab\ldots \qquad\qquad \ldots ab\underbrace{aab \mid a}ba\ldots$$

The case of $2 \nmid m$. The case of $2 \mid m$.

This concludes the proof of the lemma. □

Lemma 5. *For $n > 5$, the word F_n contains (see Fig. 1):*

- $\Phi_{n-5} - 1$ *cubic runs* $F_2^3 \cdot g_1$
- $\Phi_{n-6} - 1$ *cubic runs* $F_3^3 \cdot g_2$
- $\ldots$
- $\Phi_1 - 1$ *cubic runs* $F_{n-4}^3 \cdot g_{n-5}$.

Words $F_0, F_1, \ldots, F_5$ do not contain any cubic runs.

PROOF. It is easy to check that words F_n for $n \leq 5$ do not contain any cubic runs. Let $n > 5$ and $k \in \{2, 3, \ldots, n-4\}$. Denote $m = n - k$. Combining the formulas from Lemmas 2 and 3, we obtain that:

$$\begin{aligned} \#occ(F_k^3 \cdot g_{k-1},\ F_n) &= \#occ(F_k^3 \cdot g_{k-1},\ F_{m+k}) = \#occ(F_k^3,\ F_{m+k}) = \\ &= \#occ(aaba,\ F_m) = \Phi_{m-3} - 1 = \\ &= \Phi_{n-k-3} - 1\ . \end{aligned}$$

□

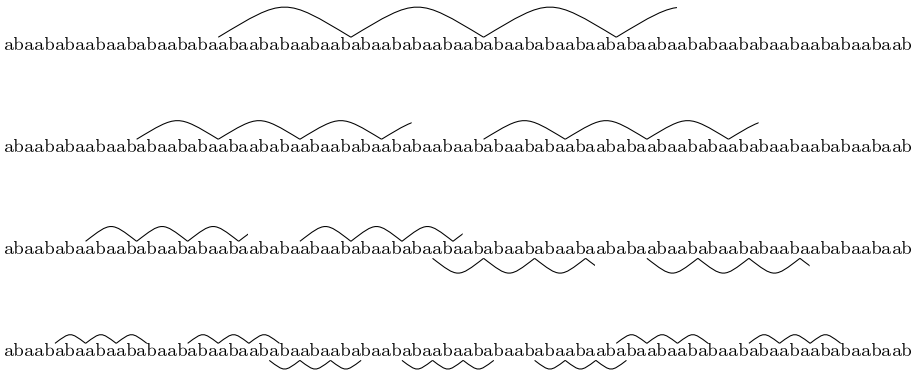

Figure 1: The structure of cubic runs in the Fibonacci word F_9. The cubic runs are distributed as follows: 1 run $F_5^3 \cdot g_4$, 2 runs $F_4^3 \cdot g_3$, 4 runs $F_3^3 \cdot g_2$, and 7 runs F_2^3.

We are now ready to describe the behaviour of the function cubic-runs(F_n). The following theorem not only provides an exact formula for it, but also shows a relationship between the number of cubic runs and the number of distinct cubes in Fibonacci words. This relationship is even more elegant than the corresponding relationship between the number of (ordinary) runs and the number of (distinct) squares in Fibonacci words, which always differ exactly by 1, see [15, 18].

Theorem 6.

a) cubic-runs$(F_n) = \Phi_{n-3} - n + 2$.

b) $\lim_{n\to\infty} \frac{\textit{cubic-runs}(F_n)}{|F_n|} = \frac{1}{\phi^3} \approx 0.2361$, *where* $\phi = \frac{1+\sqrt{5}}{2}$ *is the golden ratio.*

c) The total number of cubic runs in F_n *equals the number of distinct cubes in* F_n.

Proof. a) From Lemma 5 we obtain:

$$\mathsf{cubic\text{-}runs}(F_n) \;=\; \sum_{i=1}^{n-5}(\Phi_i - 1) \;=\; \Phi_{n-3} - 3 - (n-5) \;=\; \Phi_{n-3} - n + 2\ .$$

b) It is a straightforward application of the formula from (a):

$$\lim_{n\to\infty} \frac{\mathsf{cubic\text{-}runs}(F_n)}{|F_n|} = \lim_{n\to\infty} \frac{\Phi_{n-3} - n + 2}{\Phi_n} = \frac{1}{\phi^3}\ .$$

c) It suffices to note that the number of distinct cubes of length $3\Phi_{k+1}$in $F_{k+1}^3 \cdot g_k$ is $|g_k|+1 = \Phi_k - 1$, and thus the total number of distinct cubes in F_n equals:

$$\sum_{k=1}^{n-5} (\Phi_k - 1) = \Phi_{n-3} - n + 2 = \mathsf{cubic\text{-}runs}(F_n).$$

□

4. Upper Bound of 0.5 n

Assume that Σ is totally ordered by $\leq$, what induces a lexicographical order on Σ^*, also denoted by $\leq$. We say that $\lambda \in \Sigma^*$ is a *Lyndon word* if it is primitive and minimal or maximal in the class of words that are cyclically equivalent to it. It is known (see [14]) that a Lyndon word has no non-trivial prefix that is also its suffix.

Let $u \in \Sigma^n$. Let us denote by $\mathcal{I} = \{p_1, p_2, \ldots, p_{n-1}\}$ the set of inter-positions in u that are located *between* pairs of consecutive letters of u.

Definition 7. *We say that* $F : \mathcal{R}(u) \to \textit{subsets}(\mathcal{I})$ *is a* handle function *for the runs in word* u *if the following conditions hold:*

$$F(v_1) \cap F(v_2) = \emptyset \quad \textit{for any } v_1 \neq v_2. \tag{2}$$

$$|F(v)| \geq 2 \quad \textit{for any } v \in \mathcal{CR}(u). \tag{3}$$

We say that $F(v)$ *is the set of* handles *of the run* v.

Clearly, if a word $u \in \Sigma^n$ admits a handle function then $\mathsf{cubic\text{-}runs}(u) \leq \frac{n-1}{2}$.

We define a function $H : \mathcal{R}(u) \to \mathit{subsets}(\mathcal{I})$ as follows. Let v be a run with period q and let w be the prefix of v of length q. Let $w_{\min}$ and $w_{\max}$ be the minimal and maximal words (in lexicographical order) cyclically equivalent to w. $H(v)$ is defined as follows:

a) if $w_{\min} \neq w_{\max}$ then $H(v)$ contains all inter-positions in the middle of any occurrence of $w_{\min}^2$ in v, and in the middle of any occurrence of $w_{\max}^2$ in v,

b) if $w_{\min} = w_{\max}$ then $H(v)$ contains all inter-positions within v.

Example. Let us consider a word $(aabab)^3aab^4$, see Fig. 2. It contains two cubic runs: $v_1 = (aabab)^3aab$ and $v_2 = b^4$. For v_1 we have $\mathsf{per}(v_1) = 5$, $w_1 = v_1[1\,..\,5] = aabab = w_{\min\,1}$ and $w_{\max\,1} = babaa$. For v_2 we have $\mathsf{per}(v_2) = 1$, $w_2 = v_2[1] = b = w_{\min\,2} = w_{\max\,2}$.

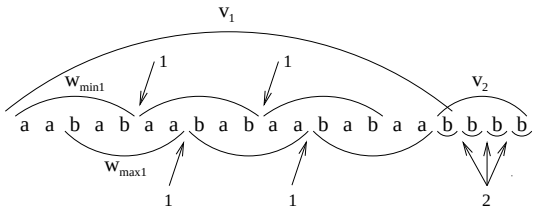

Figure 2: An example of a word with two cubic runs v_1 and v_2. For v_1 we have $w_{\min\,1} \neq w_{\max\,1}$ and for v_2 the corresponding words are equal to b (a single-letter word). The inter-positions belonging to the sets $H(v_1)$ and $H(v_2)$ are pointed by arrows.

Lemma 8. *For any word $u \in \Sigma^*$, H is a handle function.*

Proof. Let us start by showing two simple properties of $w_{\min}$ and $w_{\max}$.

(P1) $w_{\min}$ and $w_{\max}$ are Lyndon words.

(P2) If $w_{\min} = w_{\max}$ (case (b) of the definition of $H(v)$), then $|w_{\min}| = 1$ and consequently each $p_i \in H(v)$ is located in the middle of $w_{\min}^2$.

As for the property (P1), by the definition of $w_{\min}$ and $w_{\max}$ we know that these words are lexicographically minimal and maximal respectively, hence it suffices to show that both words are primitive. This follows from the fact that, due to the minimality of q, w is primitive and that $w_{\min}$ and $w_{\max}$ are cyclically equivalent to w.

We show property (P2) by contradiction. Assume that $|w_{\min}| \geq 2$. By property (P1), $w_{\min} = w_{\max}$ is a Lyndon word. Therefore it contains at least two distinct letters, let us say: $a = w_{\min}[1]$ and $b = w_{\min}[i] \neq a$. If $b < a$ ($b > a$) then the cyclic rotation of $w_{\min} = w_{\max}$ by $i - 1$ letters is lexicographically smaller than $w_{\min}$ (greater than $w_{\max}$) and $w_{\min} \neq w_{\max}$ — a contradiction. Hence, the above assumption is false and $|w_{\min}| = 1$.

Using properties (P1) and (P2), in the following two claims we show that H satisfies conditions (2) and (3).

Claim 9. *$H(v_1) \cap H(v_2) = \emptyset$ for any two different runs v_1 and v_2 in u.*

Proof. Assume, to the contrary, that $p_i \in H(v_1) \cap H(v_2)$ is a handle of two different runs v_1 and v_2. By the definition of H and properties (P1) and (P2), p_i is located in the middle of two squares of Lyndon words: w_1^2 and w_2^2, where $|w_1| = \mathsf{per}(v_1)$ and $|w_2| = \mathsf{per}(v_2)$. Note that $w_1 \neq w_2$, since otherwise runs v_1 and v_2 would be the same. Without the loss of generality, we can assume that $|w_1| < |w_2|$. Thus the word w_1 is both a prefix and a suffix of w_2 (see Fig. 3), what contradicts the fact that w_2 is a Lyndon word. □

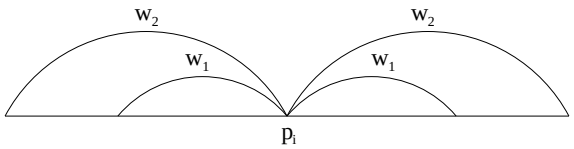

Figure 3: A situation where p_i is in the middle of two different squares w_1^2 and w_2^2.

Claim 10. *For any $v \in \mathcal{CR}(u)$, we have $|H(v)| \geq 2$.*

PROOF. Let v be a cubic run. Recall that $3q \leq |v|$, where $q = \mathsf{per}(v)$.

If $w_{\max} = w_{\min}$, then, by property (P2), $|w_{\min}| = 1$ and $|H(v)| = |v| - 1 \geq 2$.

If $w_{\max} \neq w_{\min}$, then it suffices to note that the first occurrences of each of the words $w_{\min}$ and $w_{\max}$ within v start no further than q positions from the beginning of v. Of course, they start at different positions. Hence, $w_{\min}^2$ and $w_{\max}^2$ are both factors of v and contribute different handles to $H(v)$. □

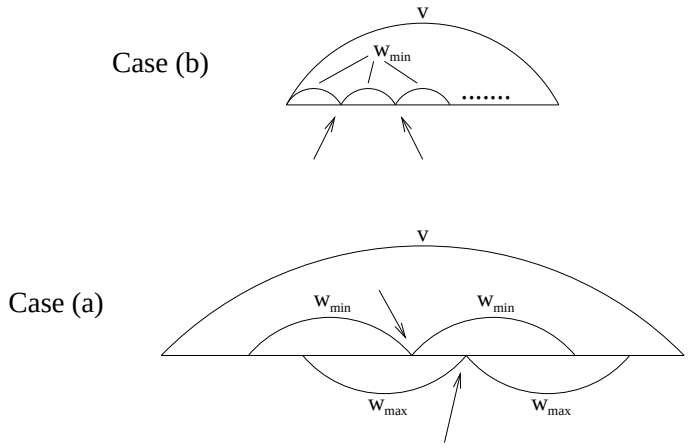

Figure 4: Illustration of the definition of H and Claim 10. The arrows in the figure point to the elements of $H(v)$ for cubic runs.

Thus we have showed that H satisfies both conditions of a handle function, what concludes the proof of the lemma. □

Theorem 11 (Weak Bounds for cubic-runs).

1. *cubic-runs*$(n) < 0.5\,n$.
2. *For infinitely many* n *we have:* $0.4\,n \leq$ *cubic-runs*(n).

PROOF. The upper bound is a corollary of Lemma 8.

As for the lower bound, define:

$$u = 0^3 1^3,\ v = 1^3 2^3,\ w = 2^3 0^3,\ x_k\ =\ (\ u^2\ 0^3\ v^2\ 1^3\ w^2\ 2^3\ )^k\ .$$

Observe that for any $k \geq 1$, the word x_k contains at least $18k - 1$ cubic runs. Indeed, we have $15k$ cubic runs with period 1, of the form 0^3, 1^3 or 2^3. Moreover, there are $3k - 1$ cubic runs with period 6: $2k$ cubic runs of the form $\left(0^3 1^3\right)^3$ or $\left(1^3 2^3\right)^3$, fully contained within each occurrence of x_1 in $x_k = (x_1)^k$, and $k - 1$ cubic runs of the form $\left(2^3 0^3\right)^3$, overlapping the concatenations of consecutive x_1's.

Note that for $k \geq 3$, the whole word x_k forms an additional cubic run. Hence, in this case the word x_k has length $45k$ and contains at least $18k$ cubic runs. Thus:

$$\mathsf{cubic\text{-}runs}(x_k) \;\geq\; 0.4\,|x_k| = 0.4n \qquad \text{for } k \geq 3.$$

□

The lower bound can be improved in two ways: restricting strings to be over binary alphabet and improving 0.4 to 0.41. The coefficient in the upper bound will be also slightly improved, for the case of binary alphabet (decreased by $\frac{1}{50}$). However even such small improvements require quite technical proofs.

5. Improving the Lower Bound

In this section we show an example sequence of *binary* words which gives the bound of $0.41\,n$. For this, we use the following morphism, which was found experimentally using a genetic algorithm:

$$\psi(a) = 001110, \quad \psi(b) = 0001110\ .$$

Recall that F_n is the n-th Fibonacci word.

Theorem 12 (Improving Lower Bound). *There are infinitely many binary strings $\psi(F_n)$ such that*

$$\frac{r_n}{\ell_n} > 0.41\ ,$$

where $r_n = \mathsf{cubic\text{-}runs}(\psi(F_n))$, $\ell_n = |\psi(F_n)|$.

n	r_n	ℓ_n	r_n/ℓ_n	w_n
0	1	6	0.16667	$0^2 1^3 0$
1	3	13	0.23077	$0^2 1^3 0^4 1^3 0$
2	5	19	0.26316	$0^2 1^3 0^4 1^3 0^3 1^3 0$
3	10	32	0.31250	$0^2 1^3 0^4 1^3 0^3 1^3 0^3 1^3 0^4 1^3 0$
4	17	51	0.33333	$0^2 1^3 0^4 1^3 0^3 1^3 0^3 1^3 0^4 1^3 0^3 1^3 0^4 1^3 0^3 1^3 0$
5	30	83	0.36145	...
6	49	134	0.36567	
7	83	217	0.38249	

Table 3: Characteristics of a first few elements of the sequence (w_n).

PROOF. Denote $w_n = \psi(F_n)$, see Table 3. We will show, that for sufficiently large n we have $\frac{r_n}{\ell_n} > 0.41$. Note that

$$\ell_n = \ell_{n-1} + \ell_{n-2} \ . \tag{4}$$

Additionally, we have:

$$w_n = \psi(F_n) = \psi(F_{n-1}F_{n-2}) = \psi(F_{n-1})\psi(F_{n-2}) = w_{n-1}w_{n-2} \ .$$

Let us start the analysis of cubic runs in w_n with the following corner case.

Claim 13. *Each word w_n contains exactly $2\Phi_n - 1$ cubic runs with period 1.*

PROOF. Each letter of F_n contributes two cubic runs with period 1 to $w_n = \psi(F_n)$, except the first letter, which contributes just one such run. Each $\psi(b)$ contributes runs: 0^3 and 1^3. $\psi(a)$ contributes one or two such runs, depending on whether the considered occurrence of a is the first letter of F_n, or not. If not, then it is either preceded by a or b. In both cases it contributes two cubic runs (conf. $\psi(aa)$ and $\psi(ba)$). Finally, recall that each F_n starts with the letter a. □

Now, let us define recursively a sequence:

$$\begin{aligned} t_n &= r_n && \text{for} \quad n \leq 5 \\ t_n &= t_{n-1} + t_{n-2} + n - 4 && \text{for} \quad 2 \mid n \text{ and } n \geq 6 \\ t_n &= t_{n-1} + t_{n-2} + n - 3 && \text{for} \quad 2 \nmid n \text{ and } n \geq 7 \ . \end{aligned} \tag{5}$$

Claim 14. $r_n \geq t_n$.

Proof. For each word w_n we will identify t_n cubic runs appearing in it. First, we will show that the runs identified in w_{n-1} and w_{n-2} do not merge in $w_n = w_{n-1}w_{n-2}$. Hence, we obtain the recursive part $(t_{n-1} + t_{n-2})$ of the equations defining t_n. Then, we will identify a number of new cubic runs overlapping the concatenation $w_n = w_{n-1} \cdot w_{n-2}$. We start the analysis by considering several small and corner cases.

Let us first consider cubic runs with period 1. It is straightforward to check that they are the only cubic runs in w_0, w_1 and w_2. Thus, using Claim 13, we obtain the values of $t_n = r_n$ for $n \leq 2$. Additionally, by Claim 13, for $n \geq 3$ the cubic runs with period 1 from w_{n-1} and w_{n-2} do not merge and for each n we obtain one new cubic run overlapping the concatenation $w_{n-1} \cdot w_{n-2}$ (due to the first letter, a, of F_{n-2}).

For $w_3 = \psi(abaab)$ we obtain one additional cubic run $\psi(baa)00 = (0^3 1^3)^3 0^3$ overlapping the concatenation $w_2 \cdot w_1$. Note, that it is not extendable, either to the left or to the right, therefore it does not merge with any other runs in any w_n. Thus $t_3 = r_3 = t_2 + t_1 + 2 = 10$.

In $w_4 = \psi(abaababa)$ we obtain one additional cubic run with period 13 overlapping the concatenation, that is: $0\psi(aababa) = (0^3 1^3 0^3 1^3 0)^3$. It cannot be extended to the left, but we have to show that when w_4 is used to build longer words w_n, the considered cubic run does not merge with any other cubic run. Let us note that in such a case it is always followed by w_3 which starts with $\psi(a)$. Hence, this cubic run can extend to the right, but only for 2 characters 00, and does not merge with any other cubic runs. In conclusion, $t_4 = r_4 = 17$.

Now, let us consider words w_n for $n \geq 5$. We have shown that cubic runs contributed by $w_0, \ldots, w_4$ used to build w_n do not merge and can be counted

separately.

A new type of cubic runs that appears in w_n for $n \geq 5$ are runs present in the words F_n — each cubic run v in F_n corresponds to a cubic run $\psi(v)$ in w_n. Due to Theorem 6, we obtain

$$\begin{aligned}&\mathsf{cubic\text{-}runs}(F_n) - \mathsf{cubic\text{-}runs}(F_{n-1}) - \mathsf{cubic\text{-}runs}(F_{n-2}) = \\ &= \Phi_{n-3} - n + 2 - (\Phi_{n-4} - n + 3) - (\Phi_{n-5} - n + 4) = n - 5\end{aligned}$$

such cubic runs overlapping the concatenation of F_{n-1} and F_{n-2}, and consequently new cubic runs overlapping the concatenation of w_{n-1} and w_{n-2}. Obviously, these runs do not merge with the ones that were considered previously. Moreover, they do not merge with each other. Indeed, if v is a run in F_n that ends before the last letter of F_n then the corresponding occurrence of $\psi(v)$ in w_n extends to the right exactly by the longest common prefix of $\psi(a)$ and $\psi(b)$, that is by two letters 00 only.

We group all the remaining cases into even and odd values of n. For odd n we have $n-2$ new cubic runs: $n-5$ from F_n, one with period 1, and two additional cubic runs: the first one is $\psi(baa)00$ which we have already considered in the case of w_3 (it is not extendable to either side) and the second one is a cubic run with period 19: $\psi(babaabaab) = (0^3 1^3 0^3 1^3 0^4 1^3)^3 0$, not extendable to the left, but possibly extendable to the right:

$$\ldots abaa \underbrace{baba \,|\, abaab}\ldots$$

For the latter cubic run we need to be more careful: in the special case of w_5 it is a suffix of the whole word, but for w_n and $n \geq 7$ it forms the same cubic run as the following run from F_n:

$$\ldots aab \underbrace{aba \,|\, abaaba}\, ba \ldots$$

Thus, $t_5 = r_5 = 30$, and $t_n = t_{n-1} + t_{n-2} + n - 3$ for $n \geq 7$, as declared in (5).

For even n we have $n-3$ new cubic runs: $n-5$ from F_n, one with period 1 and one, already mentioned for w_4, $0\psi(aababa)$ with period 13. However, in

this case we also need to consider the troublesome run $\psi(babaabaab)$ from the special case of w_5 separately, since w_5 is a suffix of w_{n-1} (as F_5 is a suffix of F_{n-1}). Indeed, this cubic run is merged with the following cubic run from F_n:

$$\ldots ab\underbrace{abaabaab\,|\,a}\,ba\ldots$$

Thus, $t_n = t_{n-1} + t_{n-2} + n - 4$, what concludes the proof of Claim 14. □

Completing the proof of Theorem 12. We prove by induction, that for $n \geq 20$, $r_n \geq 0.41 \cdot \ell_n$. The following inequalities:

$$\frac{r_{20}}{\ell_{20}} \geq \frac{46\,348}{113\,031} > 0.41\ ,$$

$$\frac{r_{21}}{\ell_{21}} \geq \frac{75\,005}{182\,888} > 0.41\ ,$$

are consequences (obtained by heavily using a calculator) of the formulas (4), (5) and Claim 14. The inductive step (for $n \geq 22$) follows from:

$$r_n - 0.41 \cdot \ell_n \ \geq\ t_n - 0.41 \cdot \ell_n \ \geq\ t_{n-1} + t_{n-2} - 0.41(\ell_{n-1} + \ell_{n-2}) \ >\ 0\ .$$

This concludes the inductive proof and also the proof of the whole theorem. □

Remark. A naive approach to obtain arbitrarily long binary words with large number of cubic runs would be to concatenate many copies of the same word $\psi(F_{20})$. However, it would not work, since some boundary runs can be glued together. Hence, a more advanced machinery was needed to prove Theorem 12.

6. Improving the Upper Bound in the Case of Binary Alphabet

Let $u \in \{0,1\}^n$. Recall that $\mathcal{I} = \{p_1, p_2, \ldots, p_{n-1}\}$ is the set of all inter-positions of u. These are all candidates for handles of cubic runs from $\mathcal{CR}(u)$.

Recall also the definition of the handle function H. We have observed that the maximal number of cubic runs would be obtained when there are $\frac{n-1}{2}$ cubic runs, and H assigns to each of them exactly two handles.

Some cubic runs can have more than two handles and some inter-positions can be not a handle of any cubic runs. Such inter-positions are called here *free* inter-positions. The key to the improvement of the upper bound is the localizations of free inter-positions and cubic runs with more than two handles.

Denote:

$$Y \;=\; \{\, 0,\; 01,\; 0001,\; 0111,\; 000111,\; 1,\; 10,\; 1000,\; 1110,\; 111000 \,\}\;.$$

By an *internal factor* of a word w we mean any factor of w having an occurrence which is neither a prefix nor a suffix of w. An internal factor can also have an occurrence at the beginning or at the end of w. For example, ab is an internal factor of $ababa$, but not of $abab$.

Let X be the set of binary words w which satisfy at least one of the properties:

(1) w has an internal factor which is a non-cubic run containing a square of a word from Y.

(2) w has a factor which is a cube of a word in $Y \setminus \{0,\; 1\}$.

(3) w has a factor 0000 or 1111.

The words $x \in X$ have several useful properties. For example, if $x = 110001000101$ then the center of the square 00010001 is a free inter-position in x, since it could only be a handle of a cubic run with period 4, but the run with period 4 containing this square is not cubic. The word 1000100010 is a non-cubic run which is an internal factor of x.

On the other hand, if x contains a factor 000100010001 then it implies a cubic run with 3 handles — the centers of the squares 00010001 and 10001000 (0001 is the minimal rotation and 1000 is the maximal rotation of the period of the run).

The words in X can be checked to satisfy the following simple fact.

Observation 15. *Let* $u \in \{0,1\}^n$.

(a) *If a factor $u[i\,..\,j]$ contains any factor satisfying point (1) of the definition of X then there is at least one free inter-position in u amongst $p_i, p_{i+1}, \ldots, p_{j-1}$.*

(b) *If a factor $u[i\,..\,j]$ contains any factor satisfying point (2) or (3) then there are at least 3 inter-positions in u amongst $p_i, p_{i+1}, \ldots, p_{j-1}$ which are handles of the same cubic run.*

This implies directly the following fact.

Theorem 16 (Improving Upper Bound).

$$\mathsf{cubic\text{-}runs}_2(n) \leq 0.48\, n\ .$$

Proof. Each binary word of length 25 contains a factor from X. It has been shown experimentally by checking all binary words of size 25.

Let $u \in \{0,1\}^n$. Let us partition the word u into factors of length 25: $u[1\,..\,25], u[26\,..\,50], \ldots$ (possibly discarding at most 24 last letters of u). By Observation 15, it is possible to remove one inter-position from every one of these factors so that each cubic run in u has at least two handles in the set of remaining inter-positions.

The total number of inter-positions in u is $n-1$ and we have shown that at least $\left\lfloor \frac{n-1}{25} \right\rfloor$ of them can be removed and each cubic run will have at least two handles among remaining inter-positions. Hence:

$$\begin{aligned}
\mathsf{cubic\text{-}runs}(u) &\leq \frac{1}{2}\cdot\left(n-1-\left\lfloor\frac{n-1}{25}\right\rfloor\right) = \\
&= \frac{1}{2}\cdot\left(\frac{24\cdot(n-1)}{25}+\frac{n-1}{25}-\left\lfloor\frac{n-1}{25}\right\rfloor\right) \leq \\
&\leq \frac{1}{2}\cdot\left(\frac{24\cdot(n-1)}{25}+\frac{24}{25}\right) = 0.48\,n\ .
\end{aligned}$$

This completes the proof. □

References

[1] P. Baturo, M. Piatkowski, and W. Rytter. The number of runs in Sturmian words. In O. H. Ibarra and B. Ravikumar, editors, *CIAA*, volume 5148 of *Lecture Notes in Computer Science*, pages 252–261. Springer, 2008.

[2] J. Berstel and J. Karhumäki. Combinatorics on words: a tutorial. *Bulletin of the EATCS*, 79:178–228, 2003.

[3] M. Crochemore and L. Ilie. Analysis of maximal repetitions in strings. In L. Kucera and A. Kucera, editors, *MFCS*, volume 4708 of *Lecture Notes in Computer Science*, pages 465–476. Springer, 2007.

[4] M. Crochemore and L. Ilie. Maximal repetitions in strings. *J. Comput. Syst. Sci.*, 74(5):796–807, 2008.

[5] M. Crochemore, L. Ilie, and W. Rytter. Repetitions in strings: Algorithms and combinatorics. *Theor. Comput. Sci.*, 410(50):5227–5235, 2009.

[6] M. Crochemore, L. Ilie, and L. Tinta. Towards a solution to the "runs" conjecture. In P. Ferragina and G. M. Landau, editors, *CPM*, volume 5029 of *Lecture Notes in Computer Science*, pages 290–302. Springer, 2008.

[7] M. Crochemore, C. S. Iliopoulos, M. Kubica, J. Radoszewski, W. Rytter, and T. Walen. On the maximal number of cubic runs in a string. In A. H. Dediu, H. Fernau, and C. Martín-Vide, editors, *LATA*, volume 6031 of *Lecture Notes in Computer Science*, pages 227–238. Springer, 2010.

[8] M. Crochemore and W. Rytter. Squares, cubes, and time-space efficient string searching. *Algorithmica*, 13(5):405–425, 1995.

[9] F. Franek and Q. Yang. An asymptotic lower bound for the maximal number of runs in a string. *Int. J. Found. Comput. Sci.*, 19(1):195–203, 2008.

[10] M. Giraud. Not so many runs in strings. In C. Martín-Vide, F. Otto, and H. Fernau, editors, *LATA*, volume 5196 of *Lecture Notes in Computer Science*, pages 232–239. Springer, 2008.

[11] R. M. Kolpakov and G. Kucherov. Finding maximal repetitions in a word in linear time. In *Proceedings of the 40th Symposium on Foundations of Computer Science*, pages 596–604, 1999.

[12] M. Kubica, J. Radoszewski, W. Rytter, and T. Walen. On the maximal number of cubic subwords in a string. In J. Fiala, J. Kratochvíl, and M. Miller, editors, *IWOCA*, volume 5874 of *Lecture Notes in Computer Science*, pages 345–355. Springer, 2009.

[13] K. Kusano, W. Matsubara, A. Ishino, H. Bannai, and A. Shinohara. New lower bounds for the maximum number of runs in a string. *CoRR*, abs/0804.1214, 2008.

[14] M. Lothaire. *Combinatorics on Words*. Addison-Wesley, Reading, MA., U.S.A., 1983.

[15] F. Mignosi and G. Pirillo. Repetitions in the Fibonacci infinite word. *ITA*, 26:199–204, 1992.

[16] S. J. Puglisi, J. Simpson, and W. F. Smyth. How many runs can a string contain? *Theor. Comput. Sci.*, 401(1-3):165–171, 2008.

[17] W. Rytter. The number of runs in a string: Improved analysis of the linear upper bound. In B. Durand and W. Thomas, editors, *STACS*, volume 3884 of *Lecture Notes in Computer Science*, pages 184–195. Springer, 2006.

[18] W. Rytter. The structure of subword graphs and suffix trees in Fibonacci words. *Theor. Comput. Sci.*, 363(2):211–223, 2006.

[19] W. Rytter. The number of runs in a string. *Inf. Comput.*, 205(9):1459–1469, 2007.

[20] J. Simpson. Modified Padovan words and the maximum number of runs in a word. *Australasian J. of Comb.*, 46:129–145, 2010.