

Searching of gapped repeats and subrepetitions in a word

Roman Kolpakov

Lomonosov Moscow State University
Leninskie Gory, Moscow, 119992 Russia
foroman@mail.ru

Mikhail Podolskiy

Lomonosov Moscow State University
Leninskie Gory, Moscow, 119992 Russia
mpodolskii@inbox.ru

Mikhail Posypkin

Institute for Information Transmission Problems
Bolshoy Karetny per., Moscow, 127994 Russia
mposypkin@gmail.com

Nickolay Khrapov

Institute for Information Transmission Problems
Bolshoy Karetny per., Moscow, 127994 Russia
nkhrapov@gmail.com

Abstract

A gapped repeat is a factor of the form uvu where u and v are nonempty words. The period of the gapped repeat is defined as $|u| + |v|$. The gapped repeat is maximal if it cannot be extended to the left or to the right by at least one letter with preserving its period. The gapped repeat is called α -gapped if its period is not greater than $\alpha|v|$. A δ -subrepetition is a factor which exponent is less than 2 but is not less than $1 + \delta$ (the exponent of the factor is the quotient of the length and the minimal period of the factor). The δ -subrepetition is maximal if it cannot be extended to the left or to the right by at least one letter with preserving its minimal period. We reveal a close relation between maximal gapped repeats and maximal subrepetitions. Moreover, we show that in a word of length n the number of maximal α -gapped repeats is bounded by $O(\alpha^2 n)$ and the number of maximal δ -subrepetitions is bounded by $O(n/\delta^2)$. Using the obtained upper bounds, we propose algorithms for finding all maximal α -gapped repeats and all maximal δ -subrepetitions in a word of length n . The algorithm for finding all maximal α -gapped repeats has $O(\alpha^2 n)$ time complexity for the case of constant alphabet size and $O(n \log n + \alpha^2 n)$ time complexity for the general case. For finding

all maximal δ -subrepetitions we propose two algorithms. The first algorithm has $O(\frac{n \log \log n}{\delta^2})$ time complexity for the case of constant alphabet size and $O(n \log n + \frac{n \log \log n}{\delta^2})$ time complexity for the general case. The second algorithm has $O(n \log n + \frac{n}{\delta^2} \log \frac{1}{\delta})$ expected time complexity.

1 Introduction

Let $w = w[1]w[2] \dots w[n]$ be an arbitrary word. The length of w is denoted by $|w|$. A fragment $w[i] \dots w[j]$ of w , where $1 \leq i \leq j \leq n$, is called a *factor* of w and is denoted by $w[i..j]$. Note that for factors we have two different notions of equality: factors can be equal as the same fragment of the original word or as the same word. To avoid this ambiguity, we will use two different notations: if two factors u and v are the same word (the same fragment of the original word) we will write $u = v$ ($u \equiv v$). For any $i = 1, \dots, n$ the factor $w[1..i]$ ($w[i..n]$) is called a *prefix* (a *suffix*) of w . By positions in w we mean the order numbers $1, 2, \dots, n$ of letters of the word w . For any factor $v \equiv w[i..j]$ of w the positions i and j are called *start position* of v and *end position* of v and denoted by $\text{beg}(v)$ and $\text{end}(v)$ respectively. The factor v covers a letter $w[k]$ if $\text{beg}(v) \leq k \leq \text{end}(v)$. For any two factors u, v of w the factor u is *contained* (*is strictly contained*) in v if $\text{beg}(v) \leq \text{beg}(u)$ and $\text{end}(u) \leq \text{end}(v)$ (if $\text{beg}(v) < \text{beg}(u)$ and $\text{end}(u) < \text{end}(v)$). Let u, v be two factors of w such that $\text{beg}(v) = \text{end}(u) + 1$. In this case we say that v *follows* u . The number $\text{end}(u)$ is called *the frontier* between the factors u and v . A factor v *contains* a frontier j if $\text{beg}(v) - 1 \leq j \leq \text{end}(v)$. If some word u is equal to a factor v of w then v is called *an occurrence* of u in w .

A positive integer p is called a *period* of w if $w[i] = w[i + p]$ for each $i = 1, \dots, n - p$. We denote by $p(w)$ the minimal period of w and by $e(w)$ the ratio $|w|/p(w)$ which is called the *exponent* of w . A word is called *primitive* if its exponent is not an integer greater than 1. By repetition in a word we mean any factor of exponent greater than or equal to 2. Repetitions are fundamental objects, due to their primary importance in word combinatorics [18] as well as in various applications, such as string matching algorithms [9, 3], molecular biology [10], or text compression [19]. The simplest and best known example of repetitions is factors of the form uu , where u is a nonempty word. Such repetitions are called *squares*. We call the first (second) factor u of the square uu *the left (right) root* of this square. Avoiding ambiguity¹, by *the period* of a square we mean the length of its roots. A square is called *primitive* if its roots are primitive. The questions concerned to squares are well studied in the literature. In particular, it is known (see, e.g., [3]) that a word of length n contains no more than $\log_{\varphi} n$ primitive squares. In [2] an $O(n \log n)$ -time algorithm for finding of all primitive squares in a word of length n is proposed. In [11] an algorithm for finding of all primitive squares in a word of length n with time complexity $O(n + S)$ where S is the size of output is proposed for the case of constant alphabet size.

¹Note that the period of a square is not necessarily the minimal period of this word.

A repetition in a word is called *maximal* if this repetition cannot be extended to the left or to the right in the word by at least one letter with preserving its minimal period. More precisely, a repetition $r \equiv w[i..j]$ in w is called *maximal* if it satisfies the following conditions:

1. if $i > 1$, then $w[i - 1] \neq w[i - 1 + p(r)]$,
2. if $j < n$, then $w[j + 1 - p(r)] \neq w[j + 1]$.

Maximal repetitions are usually called *runs* in the literature. Since runs contain all the other repetitions in a word, the set of all runs can be considered as a compact encoding of all repetitions in the word which has many useful applications (see, for example, [6]). For any word w we will denote by $\mathcal{R}(w)$ the set of all maximal repetitions in w and by $E(w)$ the sum of exponents of all maximal repetitions in w . The following facts are proved in [13].

Theorem 1 $E(w) = O(n)$ for any w .

Corollary 1 $|\mathcal{R}(w)| = O(n)$ for any w .

Moreover, in [13] an $O(n)$ time algorithm for finding of all runs in a word of length n is proposed for the case of constant alphabet size (in the case of arbitrary alphabet size all runs in a word of length n can be found in $O(n \log n)$ time). Further many papers were devoted to obtaining more precise upper bounds on $E(w)$ and $|\mathcal{R}(w)|$. In our knowledge, at present time the best upper bounds for these values are obtained in [5] and [7].

A natural generalization of squares is factors of the form uvu where u and v are nonempty words. We call such factors *gapped repeats*. In the gapped repeat uvu the first (second) factor u is called *the left (right) copy*, and v is called *the gap*. By *the period* of this gapped repeat we will mean the value $|u| + |v|$. For a gapped repeat σ we denote the length of copies of σ by $c(\sigma)$ and the period of σ by $p(\sigma)$. By (u', u'') we will denote the gapped repeat with the left copy u' and the right copy u'' . Note that gapped repeats with distinct periods can be the same factor, i.e. can have the same both start and end positions in the word. In this case, for convenience, we will consider this repeats as different ones, i.e. a gapped repeat is not determined uniquely by its start and end positions in the word because this information is not sufficient for determining the both copies and the gap of the repeat. For any real $\alpha > 1$ a gapped repeat σ is called *α -gapped* if $p(\sigma) \leq \alpha c(\sigma)$. Analogously to repetitions, we can introduce the notion of maximality for gapped repeats. A gapped repeat $(w[i'..j'], w[i''..j''])$ in w is called *maximal* if it satisfies the following conditions:

1. if $i' > 1$, then $w[i' - 1] \neq w[i'' - 1]$,
2. if $j'' < n$, then $w[j' + 1] \neq w[j'' + 1]$.

In other words, a gapped repeat in a word is maximal if its copies cannot be extended to the left or to the right in the word by at least one letter with preserving its period. Note that any α -gapped repeat is contained either in a

determined uniquely maximal α -gapped repeat with the same period or, otherwise, in a determined uniquely maximal repetition which minimal period is a divisor of the period of the repeat. Therefore, for computing all α -gapped repeats in a given word it is enough to find all maximal α -gapped repeats and all maximal repetitions in this word. Thus, taking into account the existence effective algorithms for finding of all runs in a word, we can conclude that the problem of computing all α -gapped repeats in a word is reduced to the problem of finding all maximal α -gapped repeats in a word. The set of all maximal α -gapped repeats in w will be denoted by $\mathcal{GR}_\alpha(w)$. The problem of finding gapped repeats in a word was investigated before. In particular, it is shown in [1] that all maximal gapped repeats with a gap length belonging to a specified interval can be found in a word of length n with time complexity $O(n \log n + S)$ where S is the size of output. An algorithm for finding in a word all gapped repeats with a fixed gap length is proposed in [14]. The proposed algorithm has time complexity $O(n \log d + S)$ where d is the gap length, n is the word length, and S is the size of output.

Another natural generalization of repetitions is factors with exponents strictly less than 2. We will call such factors *subrepetitions*. More precisely, for any δ such that $0 < \delta < 1$ by δ -subrepetition we mean a factor v such that $1 + \delta \leq e(v) < 2$. Note that the notion of maximal repetition is directly generalized to the case of subrepetitions: maximal subrepetitions are defined exactly in the same way as maximal repetitions. Further we reveal a close relation between maximal subrepetitions and maximal gapped repeats. Some results concerning the possible number of maximal subrepetitions in words were obtained in [16]. In particular, it was proved that the number of maximal δ -subrepetitions in a word of length n is bounded by $O(\frac{n}{\delta} \log n)$.

The aim of our research is to develop effective algorithms of finding maximal gapped repeats and maximal subrepetitions in a given word. Firstly we estimate the number of maximal α -gapped repeats in a word of length n . In the paper we prove $O(\alpha^2 n)$ upper bound on this number. From this bound we derive $O(n/\delta^2)$ upper bound on the number of maximal δ -subrepetitions in a word of length n . Using the obtained bound on the number of maximal gapped repeats in a word, we show that in the case of constant alphabet size all maximal α -gapped repeats in a word of length n can be found in $O(\alpha^2 n)$ time. For finding all maximal δ -subrepetitions in the word we propose two algorithms. The first algorithm has time complexity $O(\frac{n \log \log n}{\delta^2})$ in the case of constant alphabet size and $O(n \log n + \frac{n \log \log n}{\delta^2})$ in the general case. The second algorithm has $O(n \log n + \frac{n}{\delta^2} \log \frac{1}{\delta})$ expected time complexity.

2 Auxiliary definitions and results

Further we will consider an arbitrary word $w = w[1]w[2] \dots w[n]$ of length n . Recall that any repetition r in w is extended to just one maximal repetition r' with the same minimal period. We will call the repetition r' *the extension* of r . We will use the following quite evident fact on maximal repetitions (see,

e.g., [15][Lemma 8.1.3]).

Lemma 1 *Two distinct maximal repetitions with the same minimal period p can not have an overlap of length greater than or equal to p .*

For primitive words the following well-known fact takes place (see, e.g., [4]).

Lemma 2 (primitivity lemma) *If u is a primitive word, then u can not be strictly contained in the square uu .*

Using Lemma 2, it is easy to prove

Proposition 1 *If a square uu is primitive, for any two distinct occurrences v' and v'' of uu in w the inequality $|\text{beg}(v') - \text{beg}(v'')| \geq |u|$ holds.*

Corollary 2 *If a square uu is primitive, any factor v contains no more than $|v|/|u|$ occurrences of uu .*

Let r be a repetition in the word w . We call any factor of w which has the length $p(r)$ and is contained in r a *cyclic root* of r . The cyclic root which is the prefix (suffix) of r is called *prefix (suffix) cyclic root* of r . Note that for any cyclic root u of r the word r is a factor of the word u^k for some big enough k . So it follows from the minimality of the period $p(r)$ that any cyclic root of r has to be a primitive word. Hence any two adjacent cyclic roots of r form a primitive square with the period $p(r)$ which is called a *cyclic square* of r . The cyclic square which is the prefix (suffix) of r is called *prefix (suffix) cyclic square* of r . The following proposition can be easily obtained from Lemma 2.

Proposition 2 *Two cyclic root u' , u'' of a repetition r are equal if and only if $\text{beg}(u') \equiv \text{beg}(u'') \pmod{p}$.*

Thus we have

Corollary 3 *Any repetition r contains no more than $|r|/p(r)$ equal cyclic roots.*

For obtaining our results, we introduce the following classification of maximal gapped repeats. We say that a maximal gapped repeat is *periodic* if the copies of this repeat are repetitions. The set of all periodic maximal α -gapped repeats in the word w is denoted by \mathcal{PP}_α . A gapped maximal repeat is called *prefix (suffix) semiperiodic* if the copies of this repeat are not repetitions, but these copies have a prefix (suffix) satisfying the following conditions:

1. this prefix (suffix) is a repetition;
2. the length of this prefix (suffix) is not less than the half of the copies length.

In a copy of a prefix semiperiodic repeat the longest prefix satisfying the above conditions is called *periodic prefix* of this copy. The periodic prefixes of the copies of a prefix semiperiodic repeat are also called periodic prefixes of this repeat. The set of all prefix (suffix) semiperiodic α -gapped maximal repeats in the word w is denoted by \mathcal{PSP}_α (\mathcal{SSP}_α). A gapped maximal repeat is called *semiperiodic* if it is either prefix or suffix semiperiodic. The set of all semiperiodic α -gapped maximal repeats in the word w is denoted by \mathcal{SP}_α . Gapped maximal repeats which are neither periodic nor semiperiodic are called *ordinary*. The set of all ordinary α -gapped maximal repeats in the word w is denoted by \mathcal{OP}_α .

Let $\delta < 1$ and r be a maximal δ -subrepetition in w . Then we can consider in w the repeat $\sigma \equiv (w[\text{beg}(r)..end(r) - p(r)], w[\text{beg}(r) + p(r)..end(r)])$. It follows from $e(r) < 2$ that σ is gapped. Moreover, $p(\sigma) = p(r)$ and, since r is maximal, it is obvious that σ is maximal. Since r is a δ -subrepetition, we have also that $|r| - p(r) \geq \delta p(r)$, so $c(\sigma) = |r| - p(r) \geq \delta p(r) = \delta p(\sigma)$, i.e. $p(\sigma) \leq \frac{1}{\delta} c(\sigma)$. Thus, σ is a maximal $\frac{1}{\delta}$ -gapped repeat in w . We will call the subrepetition r and the repeat σ *respective* to each other. Note that for each maximal δ -subrepetition r there exists a maximal $\frac{1}{\delta}$ -gapped repeat σ respective to r . Moreover, the subrepetition r is determined uniquely by the repeat σ , so the same repeat can not be respective to different subrepetitions. Thus we have

Proposition 3 *Let $0 < \delta < 1$. Then in any word the number of maximal δ -subrepetitions is no more than the number of maximal $1/\delta$ -gapped repeats.*

On the other hand, it is easy to see that a maximal gapped repeat can have no a respective maximal subrepetition. Maximal gapped repeats which have respective maximal subrepetitions will be called *principal*. Thus we have the one-to-one correspondence between maximal δ -subrepetitions and principal $\frac{1}{\delta}$ -gapped repeats in a word. It is easy to check the following fact.

Proposition 4 *A maximal gapped repeat σ in w is principal if and only if $p(w[\text{beg}(\sigma)..end(\sigma)]) = p(\sigma)$.*

Let σ be a maximal gapped repeat, and r be a maximal repetition or subrepetition. We will say that σ is *stretched* by r if σ is contained in r and $p(r) < p(\sigma)$ and call σ *stretchable* if σ is stretched by some maximal repetition or subrepetition. It follows from Proposition 4 that σ is not principal if and only if $p(w[\text{beg}(\sigma)..end(\sigma)]) < p(\sigma)$, i.e. σ is contained in some maximal repetition or subrepetition with minimal period less than $p(\sigma)$. So we obtain

Proposition 5 *A maximal gapped repeat is principal if and only if it is not stretchable.*

We will say that a gapped repeat σ is *stretched* by a gapped repeat σ' if σ is contained in σ' and $p(\sigma') < p(\sigma)$. It is easy to see that a gapped repeat is stretched by a subrepetition if and only if this repeat is stretched by the gapped repeat respective to this subrepetition. Using this observation, we can derive the following

Proposition 6 *A maximal δ -gapped repeat is stretchable if and only if it is stretched by either a maximal repetition or a maximal δ -gapped repeat.*

3 Estimation of the number of maximal repeats and repetitions

In this section we estimate the number of maximal α -gapped repeats in a word. For convenience sake we assume that α is integer although our proof can be easily generalized to the case of any α . More precisely, we prove that for any integer $k \geq 2$ the number of maximal k -gapped repeats in the word w is $O(nk^2)$. To obtain this bound, we estimate separately the numbers of periodic, semiperiodic and ordinary maximal k -gapped repeats in w .

First we estimate the number of periodic maximal k -gapped repeats in w . Let $\sigma = (v', v'')$ be an arbitrary repeat from \mathcal{PP}_k . Then the both copies v' , v'' of σ are repetitions in w which are extended respectively to some maximal repetitions r' , r'' with the same minimal period in w . If r' and r'' are the same repetition r then we call σ *private* repeat and we say that σ is *generated* by r . Otherwise σ is called *non-private*. To estimate the number of private maximal k -gapped repeats in w , we use

Lemma 3 *Any maximal repetition r generates no more than $e(r)/2$ different private gapped maximal repeats.*

Proof. Let r be a maximal repetition in w with the minimal period p , and $\sigma \equiv (v', v'')$ be a private maximal gapped repeat generated by r . Denote by u' and u'' the prefixes of length p in v' and v'' respectively. Note that u' and u'' are equal cyclic roots of r , so by Proposition 2 we have $\text{beg}(u') \equiv \text{beg}(u'') \pmod{p}$. Thus $\text{beg}(v') \equiv \text{beg}(v'') \pmod{p}$. Therefore, if $\text{beg}(v') > \text{beg}(v'')$ then $w[\text{beg}(v') - 1] = w[\text{beg}(v'') - 1]$ which contradicts that σ is maximal. Thus $\text{beg}(v') = \text{beg}(v'')$, i.e. v' is a prefix of r and u' is the prefix cyclic root of r . Similarly we can prove that v'' is a suffix of r . Thus σ is determined uniquely by the cyclic root u'' which is equal to the prefix cyclic root of r . Moreover, since σ is gapped, u'' has to be contained in the suffix of length $\lfloor |r|/2 \rfloor$ in r . By Corollary 3 there exist no more than $|r|/2p = e(r)/2$ cyclic roots satisfying the above conditions for u'' . Thus there exist no more than $e(r)/2$ private maximal gapped repeats generated by r .

Lemma 3 implies immediately that the number of private maximal gapped repeats in w is not greater than $E(w)/2$. Thus, taking into account Theorem 1, we obtain

Corollary 4 *The number of private maximal gapped repeats in w is $O(n)$.*

Now let σ be non-private, i.e. r' and r'' be different maximal repetitions. Then we choose from the repetitions r' and r'' the shortest repetition (if $|r'| = |r''|$ we choose any of these repetitions) and say that σ is *generated* by the chosen repetition. More precisely, if the chosen repetition is r' (r'') we will say

that σ is generated *from the left (from the right)* by the repetition r' (r''). We prove the following fact.

Lemma 4 *For any maximal repetition r the number of non-private maximal k -gapped repeats generated by r is $O(ke(r))$.*

Proof. Let r be an arbitrary maximal repetition with the minimal period p in w . We will prove that the number of non-private maximal k -gapped repeats generated from the left by r is $O(ke(r))$. Since the number of non-private maximal k -gapped repeats generated from the right by r can be estimated similarly, it will imply the statement of the lemma. Denote by $P(r)$ the set of all non-private maximal k -gapped repeats generated from the left by r . Let $\sigma \equiv (v, v')$ be an arbitrary repeat from $P(r)$. Denote by r' the extension of v' which is the maximal repetition with the same minimal period p . If $\text{beg}(v) > \text{beg}(r)$ and $\text{beg}(v') > \text{beg}(r')$ then

$$w[\text{beg}(v)-1] = w[p+\text{beg}(v)-1] = v[p] = v'[p] = w[p+\text{beg}(v')-1] = w[\text{beg}(v')-1]$$

which contradicts that σ is maximal. Thus we have either $\text{beg}(v) = \text{beg}(r)$ or $\text{beg}(v') = \text{beg}(r')$. We can prove similarly that either $\text{end}(v) = \text{end}(r)$ or $\text{end}(v') = \text{end}(r')$. Thus we can consider the following four possible cases.

1. $\text{beg}(v) = \text{beg}(r)$ and $\text{end}(v) = \text{end}(r)$;
2. $\text{beg}(v) = \text{beg}(r)$ and $\text{end}(v') = \text{end}(r')$;
3. $\text{beg}(v') = \text{beg}(r')$ and $\text{end}(v) = \text{end}(r)$;
4. $\text{beg}(v') = \text{beg}(r')$ and $\text{end}(v') = \text{end}(r')$.

Note that in the case 4 we have $|r'| = |v'| = |v| \leq |r|$. Therefore, since $|r| \leq |r'|$ by the definition of generated repeat, in this case we obtain that $|r| = |r'| = |v|$, i.e. $\text{beg}(v) = \text{beg}(r)$ and $\text{end}(v) = \text{end}(r)$. So the case 4 is actually a subcase of the case 1. Thus $P(r) = P_1(r) \cup P_2(r) \cup P_3(r)$ where $P_i(r)$ the set of all repeats from $P(r)$ which satisfy the case i . We will estimate separately $|P_1(r)|$, $|P_2(r)|$, and $|P_3(r)|$.

Let $\sigma \in P_1(r)$, i.e. $v \equiv r$. Denote by u and u' the prefixes of length $2p$ in v and v' respectively. Note that in this case σ is determined uniquely by the factor u' . Note also that $u = u'$ and u is the prefix cyclic square of r . Thus u' is a primitive square with period p which is equal to the prefix cyclic square of r . Moreover, since σ is k -gapped, u' is contained in $w[\text{end}(v) + 1.. \text{end}(v) + k|v|]$. Therefore, by Corollary 2 the number of different factors satisfying the above conditions required for u' is not greater than

$$\frac{1}{p}|w[\text{end}(v) + 1.. \text{end}(v) + k|v|]| = \frac{1}{p}k|v| = \frac{1}{p}k|r| = ke(r).$$

Thus $|P_1(r)| \leq ke(r)$.

Now let $\sigma \in P_2(r)$. Denote again by u (u') the prefix of length $2p$ in v (v'). Note that in this case v' is determined as $w[\text{beg}(u').. \text{end}(r')]$ where r' is determined as the extension of u' . Thus σ is determined uniquely by the factor u' . As in the case 1, we have that u' is a primitive square with period p

which is equal to the prefix cyclic square of r . Moreover, since σ is k -gapped and, according to Lemma 1, r' can not overlap with r by at least p letters, u' is contained in the factor $w[\text{end}(r) + 1 - p.. \text{end}(r) + k|v|]$ which is contained in $w[\text{end}(r) + 1 - p.. \text{end}(r) + k|r|]$. Therefore, by Corollary 2 the number of different factors satisfying the conditions required for u' is not greater than

$$\frac{1}{p}|w[\text{end}(r) + 1 - p.. \text{end}(r) + k|r|]| = \frac{1}{p}(k|r| + p) = ke(r) + 1.$$

Thus $|P_2(r)| \leq ke(r) + 1$.

Finally let $\sigma \in P_3(r)$. Denote by u and u' the suffixes of length $2p$ in v and v' respectively. Note that in this case v' is determined as $w[\text{beg}(r').. \text{end}(u')]$ where r' is determined as the extension of u' . Thus σ is determined uniquely by the factor u' . Since $u = u'$ and u is the suffix cyclic square of r , the factor u' is a primitive square with period p which is equal to the suffix cyclic square of r . Moreover, since σ is k -gapped, u' is contained in the factor $w[\text{end}(r) + 1.. \text{end}(r) + k|v|]$ which is contained in $w[\text{end}(r) + 1.. \text{end}(r) + k|r|]$. Therefore, as in the case 1, we obtain that the number of different factors satisfying the conditions required for u' is not greater than $ke(r)$. Thus $|P_3(r)| \leq ke(r)$.

Summing up the obtained bounds for $|P_1(r)|$, $|P_2(r)|$, and $|P_3(r)|$, we conclude that $|P(r)| \leq 3ke(r) + 1$.

Since any non-private maximal gapped repeat is generated by some maximal repetition, Lemma 4 implies immediately that the number of non-private maximal k -gapped repeats in w is $O(kE(w))$. Therefore, from Theorem 1 we derive

Corollary 5 *The number of non-private maximal k -gapped repeats in w is $O(kn)$.*

From Corollaries 4 and 5 we have

Corollary 6 $|\mathcal{P}\mathcal{P}_k| = O(kn)$.

To estimate the number of semiperiodic maximal k -gapped repeats in w , we estimate separately the numbers of prefix semiperiodic and suffix semiperiodic maximal k -gapped repeats in w . Let $\sigma \equiv (v', v'')$ be an arbitrary maximal repeat from $\mathcal{P}\mathcal{S}\mathcal{P}_k$, and p be the minimal period of periodic prefixes of σ . Denote by u' (u'') the periodic prefix of v' (v''), and by r' (r'') the extension of r' (r'') in w . Note that r' and r'' are maximal repetitions of w with the minimal period p . From $v'[[u'| + 1] = v''[[u''| + 1] \neq v'[[u'| + 1 - p] = v''[[u''| + 1 - p]$ we have $w[\text{end}(u') + 1] \neq w[\text{end}(u') + 1 - p]$ and $w[\text{end}(u'') + 1] \neq w[\text{end}(u'') + 1 - p]$, so

$$\text{end}(r') = \text{end}(u'), \quad \text{end}(r'') = \text{end}(u''). \quad (1)$$

Thus r' and r'' are different maximal repetitions in w . If $|r'| \leq |r''|$, we will say that σ is *generated from the left* by the repetition r' per the repetition r'' . Otherwise we will say that σ is *generated from the right* by the repetition r'' per the repetition r' . A maximal repeat σ from $\mathcal{P}\mathcal{S}\mathcal{P}_k$ is *generated* by a repetition r if σ is generated from the left or from the right by r .

Proposition 7 *If a maximal repeat from \mathcal{PSP}_k is generated by a repetition r then r coincides with the periodic prefix of this repeat contained in r .*

Proof. Assume that the maximal repeat σ from \mathcal{PSP}_k is generated from the left by the repetition r' (the case when σ is generated from the right by the repetition r'' is considered analogously). According to (1), we have that $\text{end}(r') = \text{end}(u')$. Let $\text{beg}(r') < \text{beg}(u')$. Then from relations (1), $u' = u''$, and $|r'| \leq |r''|$ we obtain that $\text{beg}(r'') < \text{beg}(u'')$. So $w[\text{beg}(u') - 1] = w[\text{beg}(u') - 1 + p]$ and $w[\text{beg}(u'') - 1] = w[\text{beg}(u'') - 1 + p]$. Since $u' = u''$ we have also that $w[\text{beg}(u') - 1 + p] = u'[p] = u''[p] = w[\text{beg}(u'') - 1 + p]$. Thus $w[\text{beg}(u') - 1] = w[\text{beg}(u'') - 1]$, which contradicts that the repeat σ is maximal. Hence $\text{beg}(r') = \text{beg}(u')$, i.e. $r' \equiv u'$.

Proposition 8 *For any maximal repetitions r', r'' in w , at most one maximal repeat from \mathcal{PSP}_k can be generated from the left by r' per r'' .*

Proof. Let $\sigma \equiv (v', v'')$ be an arbitrary maximal repeat from \mathcal{PSP}_k generated from the left by a repetition r' per a repetition r'' . Then, using relations (1) and Proposition 7, we obtain that $\text{beg}(v') = \text{beg}(u') = \text{beg}(r')$ and $\text{beg}(v'') = \text{end}(r'') - |u''| + 1 = \text{end}(r'') - |r'| + 1$. Denote by x the suffix of v' and v'' such that $v' = u'x = u''x = v''$. Using relations (1) and taking into account that the repeat σ is maximal, it is easy to see that $\text{end}(v') = \text{end}(r') + |x|$, $\text{end}(v'') = \text{end}(r'') + |x|$, and x is the greatest common prefix of $w[\text{end}(r') + 1..n]$ and $w[\text{end}(r'') + 1..n]$. Thus the copies v' and v'' of the repeat σ are uniquely defined by the repetitions r' and r'' which implies Proposition 8.

If some maximal repeat from \mathcal{PSP}_k is generated from the left by a repetition r' per a repetition r'' , we call the repetition r'' *left associated* with the repetition r' .

Proposition 9 *If a repetition r'' is left associated with a repetition r' then $\text{end}(r') < \text{end}(r'') \leq \text{end}(r') + 2k|r'|$.*

Proof. Let some maximal repeat $\sigma \equiv (v', v'')$ from \mathcal{PSP}_k be generated from the left by the repetition r' per the repetition r'' . It follows from relations (1) that $\text{end}(r'') - \text{end}(r')$ is the period of σ . Therefore, since σ is k -gapped,

$$0 < \text{end}(r'') - \text{end}(r') \leq k|v'| \leq 2k|u'| \leq 2k|r'|.$$

These inequalities imply Proposition 9.

Lemma 5 *For any maximal repetition r in w there exist no more than $4k$ repetitions left associated with r .*

Proof. Let p be the minimal period of r , and r_1, r_2, \dots, r_s be all repetitions in w which are left associated with r and sorted in non-decreasing order of their end positions, i.e. $\text{end}(r_1) \leq \text{end}(r_2) \leq \dots \leq \text{end}(r_s)$. Recall that according to the definition of left associated repetitions all the repetitions r_1, r_2, \dots, r_s are maximal repetitions with the minimal period p which are not shorter than r .

So, by Lemma 1, the overlap of each adjacent repetitions r_{i-1} and r_i is less than p . Therefore

$$\text{end}(r_i) - \text{end}(r_{i-1}) > |r_i| - p \geq |r| - p \geq |r|/2.$$

Thus, taking into account Proposition 9, we have

$$\text{end}(r) < \text{end}(r_1) < \text{end}(r_2) < \dots < \text{end}(r_s) \leq \text{end}(r) + 2k|r|$$

where $\text{end}(r_i) > \text{end}(r_{i-1}) + |r|/2$. These inequalities imply that $s < \frac{2k|r|}{|r|/2} + 1 = 4k + 1$, i.e. $s \leq 4k$.

From Lemma *reflemonPSP* and Proposition 8 we immediately obtain that for any maximal repetition r in w there exist no more than $4k$ repeats from \mathcal{PSP}_k which are generated from the left by r . In the symmetrical way we can prove that for any maximal repetition r in w there exist no more than $4k$ repeats from \mathcal{PSP}_k which are generated from the right by r . Thus, any maximal repetition in w can generate no more than $8k$ repeats from \mathcal{PSP}_k . Therefore, since any repeat from \mathcal{PSP}_k is generated by some maximal repetition in w , from Corollary 1 we obtain

Corollary 7 $|\mathcal{PSP}_k| = O(kn)$.

In an analogous way we can prove that $|\mathcal{SSP}_k| = O(kn)$. Thus we have

Corollary 8 $|\mathcal{SP}_k| = O(kn)$.

For estimating the number of ordinary maximal k -gapped repeats in w we use the idea which was used before in [17]. Namely, we consider pairs of positive integers (j, p) . We call such pairs *points*. For any two points (j, p) , (j'', p'') we say that the point (j', p') *covers* the point (j'', p'') if $p' \leq p'' \leq p' - \frac{p'}{4k}$ and $j' - \frac{p'}{3k} \leq j'' \leq j'$. Let \mathcal{Q} be the set of all points (j, p) such that $1 \leq j, p \leq n$. We represent any maximal repeat σ from \mathcal{OP}_k by the point (j, p) in \mathcal{Q} where j is the end position of the left copy of σ and p is the period of σ . It is obvious that σ is uniquely defined by the values j and p , so two different repeats from \mathcal{OP}_k can not be represented by the same point. A point is *covered* by σ if the point is covered by the point representing σ . By $V[\sigma]$ we denote the set of all points covered by the repeat σ . We show that any point from \mathcal{Q} can not be covered by two different repeats from \mathcal{OP}_k .

Lemma 6 *Two different repeats from \mathcal{OP}_k can not cover the same point.*

Proof. Let σ' , σ'' be two different repeats from \mathcal{OP}_k covering the same point (j, p) . Let $v' \equiv w[i'..j']$ ($v'' \equiv w[i''..j'']$) be the left copy of σ' (σ''), and p' (p'') be the period of σ' (σ''). Thus σ' and σ'' are represented respectively in \mathcal{Q} by the points (j', p') and (j'', p'') .

First we consider the case $p' = p''$. In this case we obviously have $j' \neq j''$, and without loss of generality we assume that $j' < j''$. From inequalities

$j'' - \frac{p''}{3k} \leq j \leq j' < j''$ and $|v''| \geq \frac{p''}{k}$ we obtain that the letter $w[j' + 1]$ is contained in v'' . Hence $w[j' + 1] = w[j' + 1 + p''] = w[j' + 1 + p']$ which contradicts the maximality of σ' . Thus the case $p' = p''$ is impossible.

Now consider the case $p' \neq p''$. Without loss of generality we assume that $p' > p''$. Define $\delta = p' - p'' > 0$. From the inequalities $p' - \frac{p'}{4k} \leq p \leq p'' < p'$ we obtain that $\delta < \frac{p'}{4k}$. To prove that this case is also impossible, we show that in this case either σ' or σ'' has to be periodic or semi-periodic which contradicts that both σ' and σ'' are ordinary. We consider separately the following four subcases.

Subcase 1. Let $j'' \geq j'$ and $i'' \geq i'$. Denote by u the overlap $w[i''..j']$ of the factors v' and v'' . From the inequalities $j'' - \frac{p''}{3k} \leq j \leq j' \leq j''$ we have $j'' - j' \leq \frac{p''}{3k} \leq \frac{|v''|}{3}$, so

$$|u| = |v''| - (j'' - j') \geq \frac{2|v''|}{3} \geq \frac{2p''}{3k}.$$

Since $p' - \frac{p'}{4k} \leq p \leq p''$, we have also $p'' \geq p' \frac{4k-1}{4k}$. Thus

$$|u| \geq p' \frac{4k-1}{4k} \cdot \frac{2}{3k} > \frac{p'}{2k} > 2\delta.$$

Since u is contained in the both left copies v' and v'' , we obtain that $u = w[i'' + p''..j' + p''] = w[i'' + p'..j' + p']$. Thus, δ is a period of u and $|u| > 2\delta$, i.e. u is a repetition. Therefore, since $|u| \geq \frac{2|v''|}{3} > \frac{|v''|}{2}$, we conclude that in this case σ'' has to be semi-periodic or periodic.

Subcase 2. Let $j'' \geq j'$ and $i'' < i'$, i.e. v' is contained in v'' . Therefore, $v' = w[i' + p''..j' + p''] = w[i' + p'..j' + p']$, so δ is a period of v' . Note also that $|v'| \geq \frac{p'}{k} > 4\delta$. Thus v' is a repetition, so σ' is periodic in this case.

Subcase 3. Let $j'' < j'$ and $i'' \geq i'$, i.e. v'' is contained in v' . Therefore, $v' = w[i'' + p''..j'' + p''] = w[i'' + p'..j'' + p']$, so δ is a period of v'' . Note also that $p'' \geq p' \frac{4k-1}{4k} > \frac{3}{4}p'$, so $v'' \geq \frac{p''}{k} > \frac{3}{4k}p' > 3\delta$. Thus v'' is a repetition, so σ'' is periodic in this case.

Subcase 4. Let $j'' < j'$ and $i'' < i'$. Denote by u the overlap $w[i'..j'']$ of the factors v' and v'' . From the inequalities $j' - \frac{p'}{3} \leq j \leq j'' < j'$ we have $j' - j'' \leq \frac{p'}{3k} \leq \frac{|v'|}{3}$, so

$$|u| = |v'| - (j' - j'') \geq \frac{2|v'|}{3} \geq \frac{2p'}{3k} > \frac{8}{3}\delta.$$

Since u is contained in the both left copies v' and v'' , we obtain that $u = w[i' + p''..j'' + p''] = w[i' + p'..j'' + p']$. Thus, δ is a period of u and $|u| > 2\delta$, i.e. u is a repetition. Therefore, since $|u| \geq \frac{2|v'|}{3} > \frac{|v'|}{2}$, we conclude that in this case σ' has to be semi-periodic or periodic.

From Lemma 6 we obtain

Lemma 7 $|\mathcal{OP}_k| = O(nk^2)$.

Proof. To prove the lemma, we assign to each point (j, p) the weight $\rho(j, p) = 1/p^2$, and for any finite set A of points we define

$$\rho(A) = \sum_{(j,p) \in A} \rho(j, p) = \sum_{(j,p) \in A} \frac{1}{p^2}.$$

Let σ be an arbitrary repeat from \mathcal{OP}_k . Then

$$\rho(V[\sigma]) = \sum_{j - \frac{p}{3k} \leq i \leq j} \left(\sum_{p - \frac{p}{4k} \leq q \leq p} \frac{1}{q^2} \right) > \frac{p}{3k} \sum_{p - \frac{p}{4k} \leq q \leq p} \frac{1}{q^2}$$

where (j, p) is the point representing σ . For further estimating of $\rho(V[\sigma])$ we consider separately the cases $p < 4k$ and $p \geq 4k$. Let $p < 4k$. Then

$$\frac{p}{3k} \sum_{p - \frac{p}{4k} \leq q \leq p} \frac{1}{q^2} = \frac{p}{3k} \cdot \frac{1}{p^2} = \frac{1}{3kp} > \frac{1}{12k^2}.$$

Now let $p \geq 4k$. Then

$$\begin{aligned} \sum_{p - \frac{p}{4k} \leq q \leq p} \frac{1}{q^2} &= \sum_{q=p - \lfloor \frac{p}{4k} \rfloor}^p \frac{1}{q^2} > \int_{p - \lfloor \frac{p}{4k} \rfloor}^p \frac{1}{x^2} dx \\ &= \frac{1}{p - \lfloor \frac{p}{4k} \rfloor} - \frac{1}{p+1} = \frac{1 + \lfloor \frac{p}{4k} \rfloor}{(p - \lfloor \frac{p}{4k} \rfloor)(p+1)} \\ &> \frac{p/4k}{(p-1)(p+1)} > \frac{p/4k}{p^2} = \frac{1}{4kp}. \end{aligned}$$

Therefore,

$$\frac{p}{3k} \sum_{p - \frac{p}{4k} \leq q \leq p} \frac{1}{q^2} > \frac{p}{3k} \cdot \frac{1}{4kp} = \frac{1}{12k^2}.$$

Thus, for any repeat σ from \mathcal{OP}_k we have $\rho(V[\sigma]) > \frac{1}{12k^2}$. Using this estimation, we obtain that

$$\sum_{\sigma \in \mathcal{OP}_k} \rho(V[\sigma]) > \frac{|\mathcal{OP}_k|}{12k^2}. \quad (2)$$

Note that any point covered by repeats from \mathcal{OP}_k belongs to \mathcal{Q} . On the other hand, by Lemma 6, each point of \mathcal{Q} can not be covered by two repeats from \mathcal{OP}_k . Therefore,

$$\sum_{\sigma \in \mathcal{OP}_k} \rho(V[\sigma]) \leq \rho(\mathcal{Q}) = \sum_{j=1}^n \sum_{p=1}^n \frac{1}{p^2} = n \sum_{p=1}^n \frac{1}{p^2} < n \sum_{p=1}^{\infty} \frac{1}{p^2} = \frac{n\pi^2}{6}.$$

Thus, using inequality (2), we can conclude that $|\mathcal{OP}_k| < 2\pi^2 nk^2$.

Summing up Corollaries 6 and 8 and Lemma 7, we obtain that for any integer $k \geq 2$ the number of maximal k -gapped repeats in w is $O(k^2 n)$. This upper bound is obviously generalized to the case of maximal α -gapped repeats for any real $\alpha > 1$. Thus we can conclude

Lemma 8 For any $\alpha > 1$ the number of maximal α -gapped repeats in w is $O(\alpha^2 n)$.

From Lemma 8, using the Proposition 3, one can easily derive the following upper bound for maximal δ -subrepetitions.

Corollary 9 Let $0 < \delta < 1$. Then the number of maximal δ -subrepetitions in w is $O(n/\delta^2)$.

4 Computing of maximal gapped repeats

In this section we propose an algorithm for finding of all maximal α -gapped repeats in the given word w for a fixed value of α . The proposed algorithm is actually a modification of the algorithm described in [15] for finding all repeats with a fixed gap in a given word. In particular, the two following basic tools are used in this modification.

The first tool is special functions which are defined as follows. Let u, v be two arbitrary words. For each $i = 2, 3, \dots, |u|$ we define $LP_u(i)$ as the length of the longest common prefix of u and $u[i..|u|]$. For each $i = 1, 2, \dots, |u| - 1$ we define $LS_u(i)$ as the length of the longest common suffix of u and $u[1..|u| - i]$. For each $i = 0, 1, \dots, |u| - 1$ we define $LP_{u|v}(i)$ as the length of the longest common prefix of $u[|u| - i..|u|]v$ and v . For each $i = 1, 2, \dots, |v|$ we define $LS_{u|v}(i)$ as the length of the longest common suffix of u and $uv[1..i]$. The functions LP_u and LS_u can be computed in $O(|u|)$ time and the functions $LP_{u|v}$ and $LS_{u|v}$ can be computed in $O(|u| + |v|)$ time (see, e.g., [15]).

The second tool is a factorization $f \equiv f_1 f_2 \dots f_t$ of the word w which is called *non-overlapping s-factorization* and defined inductively as follows:

- $f_1 \equiv w[1]$.
- Let for $i > 1$ the factors f_1, \dots, f_{i-1} are already computed, and $w[j]$ be the letter which follows the factor f_{i-1} in w . Then $f_i \equiv w[j]$ if the letter $w[j]$ has no occurrences in $f_1 f_2 \dots f_{i-1}$; otherwise f_i is the longest factor in w which follows f_{i-1} and has an occurrence in $f_1 f_2 \dots f_{i-1}$.

The factorization f can be computed in $O(n)$ time for the case of constant alphabet size and in $O(n \log n)$ time for the general case (see, e.g., [15]). By a_i (b_i) we denote the start (end) position of the factor f_i . The length of f_i is denoted by l_i . For $i = 1, 2, \dots, t - 1$ we will consider also the factor $w[a_i..b_i + 1]$ which is denoted by f'_i .

For convenience sake we consider the case when α is integer, i.e. for any integer $k \geq 2$ we describe the algorithm of finding in w all repeats from $\mathcal{GR}_k(w)$. To this purpose we divide the set $\mathcal{GR}_k(w)$ into the following two nonoverlapping subsets: \mathcal{FGR} is the set of all repeats from $\mathcal{GR}_k(w)$ which are not strictly contained in any factor f_i of the factorization f , and \mathcal{SGR} is the set of all repeats from $\mathcal{GR}_k(w)$ which are strictly contained in factors of the factorization f . To compute the set $\mathcal{GR}_k(w)$, we compute separately the sets \mathcal{FGR} and \mathcal{SGR} . For

each $i = 2, 3, \dots, t$ we define in the set \mathcal{FGR} the following subsets: \mathcal{FGR}'_i is the set of all repeats σ from \mathcal{FGR} such that

1. $b_{i-1} < \text{end}(\sigma) \leq b_i$;
2. $\text{beg}(\sigma) \leq a_i$;

and \mathcal{FGR}''_i is the set of all repeats σ from \mathcal{FGR} such that

1. $\text{end}(\sigma) = b_i$;
2. $\text{beg}(\sigma) > a_i$.

It is easy to see that all the subsets \mathcal{FGR}'_i and \mathcal{FGR}''_i are nonoverlapping. Moreover, taking into account that the factor f_1 consists of only one letter, we have that $\mathcal{FGR} = \bigcup_{i=2}^t \mathcal{FGR}'_i \cup \bigcup_{i=2}^t \mathcal{FGR}''_i$. To compute the set \mathcal{FGR} , we compute separately the sets \mathcal{FGR}'_i and \mathcal{FGR}''_i for $i = 2, 3, \dots, t$.

To compute \mathcal{FGR}'_i , we consider in this set the following nonoverlapping subsets: $\mathcal{FGR}_i^{\text{lrt}}$ is the set of all repeats from \mathcal{FGR}'_i which left copies contain the frontier between the factors f_{i-1} and f_i , $\mathcal{FGR}_i^{\text{rrt}}$ is the set of all repeats from \mathcal{FGR}'_i which right copies contain the frontier between the factors f_{i-1} and f_i , and $\mathcal{FGR}_i^{\text{mid}}$ is the set of all repeats σ from \mathcal{FGR}'_i such that neither left nor right copies of σ contain the frontier between the factors f_{i-1} and f_i . It is obvious that $\mathcal{FGR}'_i = \mathcal{FGR}_i^{\text{lrt}} \cup \mathcal{FGR}_i^{\text{rrt}} \cup \mathcal{FGR}_i^{\text{mid}}$. We compute separately the considered subsets of \mathcal{FGR}'_i .

1. **Computing the set $\mathcal{FGR}_i^{\text{lrt}}$.** Let $\sigma \equiv (w[i'..j'], w[i''..j''])$ be a repeat from $\mathcal{FGR}_i^{\text{lrt}}$ with a period p . Note that in this case $p \leq l_i$, so $c(\sigma) < l_i$. Thus, σ is strictly contained in the factor $w[a_i - l_i..a_{i+1}] \equiv g_i f'_i$ where $g_i \equiv w[a_i - l_i..b_{i-1}]$. Since $w[i'..j']$ contains the frontier between f_{i-1} and f_i , we have $i' - 1 \leq b_{i-1} \leq j'$. Thus, we can consider the factors $w[a_i..j']$ and $w[i'..b_{i-1}]$. Note that $w[a_i..j'] = w[a_i + p..j'']$ and $w[j' + 1] \neq w[j'' + 1]$. Moreover, from the condition $\text{end}(\sigma) \leq b_i$ we have $j'' \leq b_i$. Therefore, $w[a_i..j']$ is the longest common prefix of f'_i and $f'_i[p+1..l'_i]$, i.e. $|w[a_i..j']| = |w[a_i + p..j'']| = \text{LP}_{f'_i}(p+1)$. Note also that $w[i'..b_{i-1}] = w[i''..b_{i-1} + p]$ and $w[i' - 1] \neq w[i'' - 1]$. Moreover, $|w[i'..b_{i-1}]| \leq c(\sigma) < l_i$. Therefore, $w[i'..b_{i-1}]$ is the longest common suffix of the words g_i and $g_i f'_i[1..p]$, i.e. $|w[i'..b_{i-1}]| = |w[i''..b_{i-1} + p]| = \text{LS}_{g_i|f'_i}(p)$. Thus,

$$\sigma \equiv (w[a_i - \hat{\text{LS}}(p)..b_{i-1} + \hat{\text{LP}}(p)], w[a_i + p - \hat{\text{LS}}(p)..b_{i-1} + p + \hat{\text{LP}}(p)]) \quad (3)$$

where $\hat{\text{LP}}(p) = \text{LP}_{f'_i}(p+1)$ and $\hat{\text{LS}}(p) = \text{LS}_{g_i|f'_i}(p)$, i.e. σ is defined uniquely by the period p . Since $\hat{\text{LP}}(p) + \hat{\text{LS}}(p) = c(\sigma)$ and σ is a k -gapped repeat, we have the following restrictions for $\hat{\text{LP}}(p)$ and $\hat{\text{LS}}(p)$:

$$p/k \leq \hat{\text{LP}}(p) + \hat{\text{LS}}(p) < p. \quad (4)$$

Moreover, from the condition $\text{end}(\sigma) \leq b_i$ we have the restriction

$$\hat{\text{LP}}(p) \leq l_i - p. \quad (5)$$

On the other hand, if for some p such that $p \leq l_i$ the conditions (4) and (5) hold, in the set $\mathcal{FGR}_i^{\text{rt}}$ there exists the maximal k -gapped repeat (3) with the period p . Thus, to compute $\mathcal{FGR}_i^{\text{rt}}$, for each $p = 1, 2, \dots, l_i$ we compute the values $\hat{\text{LP}}(p)$ and $\hat{\text{LS}}(p)$ and check the conditions (4) and (5). If these conditions are valid we add the corresponding repeat (3) to $\mathcal{FGR}_i^{\text{rt}}$. As noted above, all the values $\hat{\text{LP}}(p)$ and $\hat{\text{LS}}(p)$ can be computed in $O(|g_i| + |f'_i|) = O(l_i)$ time, and all the conditions (4) and (5) can be checked in $O(l_i)$ time. Thus, the set $\mathcal{FGR}_i^{\text{rt}}$ can be computed in $O(l_i)$ time.

2. **Computing the set $\mathcal{FGR}_i^{\text{rt}}$.** Let $\sigma \equiv (w[i'..j'], w[i''..j''])$ be a repeat from $\mathcal{FGR}_i^{\text{rt}}$ with a period p . Then for σ we have the following

Proposition 10 *The right copy of σ doesn't contain the frontier between the factors f_{i-2} and f_{i-1} .*

Proof. Assume that the right copy $w[i''..j'']$ contains the frontier between f_{i-2} and f_{i-1} . Then we can consider the factor $u' \equiv w[a_{i-1}..j'']$ which is a suffix of the right copy. Since u' is also a suffix of the right copy of σ , in $f_1 f_2 \dots f_{i-2}$ there is an occurrence of u' . Moreover, the factor u' immediately follows $f_1 f_2 \dots f_{i-2}$ and $|u'| > |f_{i-1}|$ because of $j'' = \text{end}(\sigma) > b_{i-1}$. This contradicts the definition of the factor f_{i-1} .

From Proposition 10 and the condition $\text{end}(\sigma) \leq b_i$ we immediately obtain

Corollary 10 $c(\sigma) < l_{i-1} + l_i$.

Thus, $p \leq kc(\sigma) < k(l_{i-1} + l_i)$ and $i'' > a_{i-1}$ by Proposition 10. Therefore, $i' = i'' - p > a_{i-1} - k(l_{i-1} + l_i)$, i.e. σ is strictly contained in the factor $w[a_{i-1} - k(l_{i-1} + l_i)..a_{i+1}] \equiv g'_i f'_i$ where $g'_i \equiv w[a_{i-1} - k(l_{i-1} + l_i)..b_{i-1}]$. Since $w[i''..j'']$ contains the frontier between f_{i-1} and f_i , we can consider the factors $w[a_i..j'']$ and $w[i''..b_{i-1}]$. Note that $w[a_i..j''] = w[a_i - p..j']$ and $w[j' + 1] \neq w[j'' + 1]$. Moreover, $j'' \leq b_i$. Therefore, $|w[a_i..j'']| = |w[a_i - p..j']| = \text{LP}_{g'_i | f'_i}(p - 1)$. Note also that $w[i''..b_{i-1}] = w[i'..b_{i-1} - p]$ and $w[i' - 1] \neq w[i'' - 1]$. Thus $|w[i''..b_{i-1}]| = |w[i'..b_{i-1} - p]| = \text{LS}_{g'_i}(p)$. Hence

$$\sigma \equiv (w[a_i - p - \hat{\text{LS}}(p)..b_{i-1} - p + \hat{\text{LP}}(p)], w[a_i - \hat{\text{LS}}(p)..b_{i-1} + \hat{\text{LP}}(p)]) \quad (6)$$

where $\hat{\text{LP}}(p) = \text{LP}_{g'_i | f'_i}(p - 1)$ and $\hat{\text{LS}}(p) = \text{LS}_{g'_i}(p)$, i.e. σ is defined uniquely by the period p . As in the the case of computing $\mathcal{FGR}_i^{\text{rt}}$, we have for the period p the restrictions (4). Moreover, since $b_{i-1} < j'' \leq b_i$, we have the following additional restriction:

$$0 < \hat{\text{LP}}(p) \leq l_i. \quad (7)$$

On the other hand, if for some p such that $p < k(l_{i-1} + l_i)$ the conditions (4) and (7) hold, in the set $\mathcal{FGR}_i^{\text{rt}}$ there exists the k -gapped repeat (6) with the period p . Thus, to compute $\mathcal{FGR}_i^{\text{rt}}$, for each $p < k(l_{i-1} + l_i)$ we check the conditions (4) and (7) for the values $\hat{\text{LP}}(p)$ and $\hat{\text{LS}}(p)$. If these conditions hold we add the corresponding repeat (6) to $\mathcal{FGR}_i^{\text{rt}}$. Note that all the values $\hat{\text{LP}}(p)$ and $\hat{\text{LS}}(p)$ can be computed in $O(|g'_i| + |f'_i|) = O(k(l_{i-1} + l_i))$ time, and all the

conditions (4) and (7) can be checked in $O(k(l_{i-1} + l_i))$ time. Thus, the set $\mathcal{FGR}_i^{\text{rrt}}$ can be computed in $O(k(l_{i-1} + l_i))$ time.

3. Computing the set $\mathcal{FGR}_i^{\text{mid}}$. Note that the right copies of all repeats from $\mathcal{FGR}_i^{\text{mid}}$ are strictly contained in f'_i . Let $q = \lfloor \log k/(k-1)l_i \rfloor$. We denote by d_s the position $\lfloor ((k-1)/k)^s l_i \rfloor + 1$ for $s = 0, 1, \dots, q$ and divide the set $\mathcal{FGR}_i^{\text{mid}}$ into nonoverlapping subsets MP_1, MP_2, \dots, MP_q where MP_s is the set of all repeats from $\mathcal{FGR}_i^{\text{mid}}$ which right copies cover the letter $f'_i[d_s]$ but don't cover the letter $f'_i[d_{s-1}]$.

Proposition 11 $\mathcal{FGR}_i^{\text{mid}} = \bigcup_{s=1}^q MP_s$.

Proof. Let $\sigma \equiv (w[i'..j'], w[i''..j''])$ be a repeat from $\mathcal{FGR}_i^{\text{mid}}$. Since the right copy $w[i''..j'']$ doesn't cover the letter $f'_i[d_0] \equiv w[a_{i+1}]$, for proving the proposition we have to show that $w[i''..j'']$ covers at least one of the letters $f'_i[d_1], f'_i[d_2], \dots, f'_i[d_q]$. Let $w[i''..j'']$ do not cover any of these letters. It is easy to check that $d_q \leq 2$, so $w[i''..j'']$ can not be to the left of the letter $f'_i[d_q]$. Thus, $w[i''..j'']$ has to be situated between some letters $f'_i[d_s]$ and $f'_i[d_{s-1}]$. Then

$$\begin{aligned} c(\sigma) &= |w[i''..j'']| \leq d_{s-1} - 1 - d_s = \left\lfloor \left(\frac{k-1}{k}\right)^{s-1} l_i \right\rfloor - \left\lfloor \left(\frac{k-1}{k}\right)^s l_i \right\rfloor - 1 \\ &< \left(\frac{k-1}{k}\right)^{s-1} l_i - \left(\frac{k-1}{k}\right)^s l_i = \left(\frac{k-1}{k}\right)^{s-1} \frac{l_i}{k}. \end{aligned}$$

Moreover, since the left copy $w[i'..j']$ is to the left of the letter $w[a_i]$, we have

$$p(\sigma) - c(\sigma) \geq d_s > \left(\frac{k-1}{k}\right)^s l_i > (k-1)c(\sigma).$$

Thus $p(\sigma) > kc(\sigma)$, which contradicts the assumption that σ is k -gapped.

Using Proposition 11, for computing $\mathcal{FGR}_i^{\text{mid}}$ we compute separately the sets MP_1, MP_2, \dots, MP_q . In order to compute the set MP_s , consider an arbitrary repeat $\sigma \equiv (w[i'..j'], w[i''..j''])$ with a period p in this set. Note that in this case the right copy of σ is strictly contained in $f'_i[1..d_{s-1}]$, so $c(\sigma) < d_{s-1}$. Thus, $j < kd_{s-1}$ and σ is strictly contained in

$$w[a_i - kd_{s-1}..b_{i-1}]f'_i[1..d_{s-1}] \equiv h_{i_s}h'_{i_s}$$

where $h_{i_s} \equiv w[a_i - kd_{s-1}..b_{i-1}]f'_i[1..d_{s-1}]$ and $h'_{i_s} \equiv f'_i[d_s..d_{s-1}]$. Since $w[i''..j'']$ covers the letter $f'_i[d_s]$ we can consider the factors $w[i'..b_{i-1} + d_s - p]$, $w[a_{i-1} + d_s - p..j']$, $w[i''..b_{i-1} + d_s]$, $w[a_{i-1} + d_s..j'']$ and note that

$$|w[i'..b_{i-1} + d_s - p]| = |w[i''..b_{i-1} + d_s]| = \text{LS}_{h_{i_s}}(p),$$

$$|w[a_{i-1} + d_s - p..j']| = |w[a_{i-1} + d_s..j'']| = \text{LP}_{h_{i_s}|h'_{i_s}}(p-1).$$

Thus, σ is defined uniquely by the period p as

$$(w[a_i + d_s - p - \hat{\text{L}}\text{S}(p)..b_{i-1} + d_s - p + \hat{\text{L}}\text{P}(p)], w[a_i + d_s - \hat{\text{L}}\text{S}(p)..b_{i-1} + d_s + \hat{\text{L}}\text{P}(p)]) \quad (8)$$

where $\hat{L}\hat{S}(p) = \text{LS}_{h_{i_s}}(p)$ and $\hat{L}\hat{P}(p) = \text{LP}_{h_{i_s}|h'_{i_s}}(p-1)$. Since the repeat σ is k -gapped, the conditions (4) have to be valid for the period p . Moreover, p has to satisfy the additional restrictions

$$\hat{L}\hat{P}(p) \leq p - d_s, \quad (9)$$

$$0 < \hat{L}\hat{P}(p) \leq d_{s-1} - d_s, \quad (10)$$

$$\hat{L}\hat{S}(p) < d_s - 1, \quad (11)$$

following from the definition of the set MP_s . On the other hand, for each p satisfying the inequality $p < kd_{s-1}$ and the conditions (4), (9), (10), and (11), there exists the k -gapped repeat (8) with the period p in the set MP_s . Thus, to compute MP_s , we check the conditions (4), (9), (10), and (11) for each p such that $p < kd_{s-1}$. If for some p these conditions hold we add the corresponding repeat (8) to MP_s . Note that the time required for computing the involved values $\text{LP}(p)$ and $\text{LS}(p)$ is bounded by $O(|h_{i_s}| + |h'_{i_s}|) = O(kd_{s-1})$ and the total time required for checking these conditions is bounded by $O(kd_{s-1})$. Thus, MP_s can be computed in $O(kd_{s-1}) = O((k-1)/k)^{s-1}kl_i$ time. Hence, taking into account that

$$\sum_{s=1}^q \left(\frac{k-1}{k}\right)^{s-1} kl_i < kl_i \sum_{s=0}^{\infty} \left(\frac{k-1}{k}\right)^s = k^2l_i,$$

by Proposition 11 we obtain that $\mathcal{FGR}_i^{\text{mid}}$ can be computed in $O(k^2l_i)$ time.

Summing up the obtained time bounds for computing the sets $\mathcal{FGR}_i^{\text{lrt}}$, $\mathcal{FGR}_i^{\text{rrt}}$ and $\mathcal{FGR}_i^{\text{mid}}$, we conclude that \mathcal{FGR}'_i can be computed in $O(kl_{i-1} + k^2l_i)$ time.

It is easy to note that the set \mathcal{FGR}''_i can also be computed in $O(l_i)$ time by a simplified version of the described above algorithm for computing $\mathcal{FGR}_i^{\text{rrt}}$.

The set \mathcal{SGR} is also divided into nonoverlapping subsets $\mathcal{SGR}_2, \mathcal{SGR}_3, \dots, \mathcal{SGR}_t$ where \mathcal{SGR}_i is the set of all repeats from \mathcal{SGR} which are strictly contained in f_i . These subsets are computed separately by the procedure described below.

Now we give a general description of the algorithm for computing $\mathcal{GR}_k(w)$. Initially we compute the factorization f for w . During the computation of f , for each factor f_i such that $|f_i| > 1$ we store a pointer to an occurrence of f_i in $f_1f_2 \dots f_{i-1}$ (such occurrence exists by the definition of non-overlapping s -factorization and will be denoted by v_i). More exactly, we store the difference Δ_i between the start positions of f_i and v_i . Computation of values Δ_i does not affect the time complexity of computing the factorization f . Further we execute the following procedure of finding all repeats from $\mathcal{GR}_k(w)$. During this procedure all found repeats are stored in lists $\text{start}[j]$ for $j = 1, 2, \dots, n$ where $\text{start}[j]$ is a list of all found repeats with the start position j sorted in non-decreasing order of their end positions. The procedure consists of $t-1$ consecutive steps. At the $i-1$ -th step we find all repeats σ from $\mathcal{GR}_k(w)$ such that $a_i \leq \text{end}(\sigma) \leq b_i$, i.e. after $i-1$ -th step we have found all repeats σ

from $\mathcal{GR}_k(w)$ such that $\text{end}(\sigma) \leq b_i$. Thus, after the last step all repeats from $\mathcal{GR}_k(w)$ are found. The $i - 1$ -th step is executed as follows. First we compute the set \mathcal{FGR}'_i as described above. During the computation all found repeats from \mathcal{FGR}'_i are initially stored in auxiliary lists $\text{fin}[j]$ for $j = a_i, a_i + 1, \dots, b_i$ where in the list $\text{fin}[j]$ we store all found repeats with the end position j . After the computation we process consecutively all lists $\text{fin}[j]$ in the increasing order of j by replacing all found repeats from these lists into the lists $\text{start}[j]$ according to their start positions. The auxiliary sorting through the lists $\text{fin}[j]$ guarantees that all found repeats will be placed into the lists $\text{start}[j]$ in the required order. Further, if $|f_i| > 1$, we compute the set \mathcal{SGR}_i . For computing this set consider an arbitrary repeat $\sigma \equiv (u, v)$ from \mathcal{SGR}_i . Since σ is strictly contained in f_i , there exists the occurrence

$$\sigma' \equiv (w[\text{beg}(u) - \Delta_i .. \text{end}(u) - \Delta_i], w[\text{beg}(v) - \Delta_i .. \text{end}(v) - \Delta_i])$$

of σ which is strictly contained in v_i , i.e. $\text{beg}(\sigma') > \text{beg}(v')$ and $\text{end}(\sigma') < \text{end}(v') \leq \text{end}f_{i-1}$. It is obvious that σ' is also a maximal k -gapped repeat from $\mathcal{GR}_k(w)$, so it has to be found before the $i - 1$ step. Thus σ' is contained in the list $\text{start}[\text{beg}(\sigma) - \Delta_i]$. On the other hand, for each repeat $\sigma' \equiv (u', v')$ which is contained in a list $\text{start}[j]$ where $\text{beg}(v') < \text{beg}(\sigma') < \text{end}(v')$ and satisfies the condition $\text{end}(\sigma') < \text{end}(v') = b_i - \Delta_i$ there exists the repeat

$$\sigma \equiv (w[\text{beg}(u') + \Delta_i .. \text{end}(u') + \Delta_i], w[\text{beg}(v') + \Delta_i .. \text{end}(v') + \Delta_i]) \quad (12)$$

in the set \mathcal{SGR}_i . Thus, to compute \mathcal{SGR}_i it is enough for any j such that $a_i < j < b_i$ to copy each repeat $\sigma' \equiv (u', v')$ from the list $\text{start}[j - \Delta_i]$ such that $\text{end}(\sigma') < b_i - \Delta_i$ into the new list $\text{start}[j]$ as the repeat σ defined in (12) with preserving the order of repeats in the lists. It can be done in $O(l_i + |\mathcal{SGR}_i|)$ time, so \mathcal{SGR}_i can be computed in this time. Finally we compute the set \mathcal{FGR}''_i in $O(l_i)$ time. During the computation of this set each found repeat σ is placed into the respective list $\text{start}[\text{beg}(\sigma)]$. It is easy to see that at the $i - 1$ -th step all repeats σ from $\mathcal{GR}_k(w)$ such that $a_i \leq \text{end}(\sigma) \leq b_i$ will be found and placed into the lists $\text{start}[j]$ in the required order. The time complexity bound for the $i - 1$ -th step is $O(kl_{i-1} + k^2l_i + |\mathcal{SGR}_i|)$. It easily implies $O(k^2n + |\mathcal{SGR}|)$ total time complexity bound for all steps. Since $|\mathcal{SGR}| < |\mathcal{GR}_k(w)| = O(k^2n)$ by Lemma 8, we obtain $O(k^2n)$ time complexity bound for the described procedure of finding all repeats from $\mathcal{GR}_k(w)$. Taking into account the time for constructing the factorization f , we conclude that $\mathcal{GR}_k(w)$ can be computed in $O(k^2n)$ time for the case of constant alphabet size and in $O(n \log n + k^2n)$ time for the general case.

For convenience sake we have considered the case of maximal k -gapped repeats where k is integer but it is easy to see that the proposed algorithm can be directly generalized to the case of maximal α -gapped repeats for real $\alpha > 1$ with preserving the upper bound for time complexity. Thus we have

Theorem 2 *For any real $\alpha > 1$ all maximal α -gapped repeats in w can be computed in $O(\alpha^2n)$ time for the case of constant alphabet size and in $O(n \log n + \alpha^2n)$ time for the general case.*

Now consider the problem of finding all maximal δ -subrepetitions in a word for a fixed δ . Because of the established above one-to-one correspondence between maximal δ -subrepetitions and principal $\frac{1}{\delta}$ -gapped repeats, this problem is reduced to computing all principal $\frac{1}{\delta}$ -gapped repeats in a word. We propose the following algorithm for computing all principal $\frac{1}{\delta}$ -gapped repeats in the word w . Further, for convenience, by the period of a repetition we will mean its minimal period. First we compute the ordered set \mathcal{OSR}_δ of all maximal repetitions and all maximal $\frac{1}{\delta}$ -gapped repeats in w such that all elements of \mathcal{OSR}_δ are ordered in non-decreasing order of their start positions and, furthermore, elements of \mathcal{OSR}_δ with the same start position are ordered in increasing order of their periods (it is easy to note that any element of \mathcal{OSR}_δ determined uniquely by its start position and its period, so the introduced order in \mathcal{OSR}_δ is uniquely defined). To compute \mathcal{OSR}_δ , we find in w all maximal repetitions and all maximal $\frac{1}{\delta}$ -gapped repeats. Using Theorem 2 and the algorithm for finding maximal repetitions proposed in [13], it can be done in $O(n/\delta^2)$ time for the case of constant alphabet size and in $O(n \log n + n/\delta^2)$ time for the general case. Then we arrange the found repetitions and repeats in the order required for \mathcal{OSR}_δ . By Lemma 8 the number of the maximal $\frac{1}{\delta}$ -gapped repeats is $O(n/\delta^2)$ and by Corollary 1 the number of the maximal repetitions is $O(n)$, so $|\mathcal{OSR}_\delta| = O(n/\delta^2)$. Therefore, using bucket sort, the required arrangement can be done in $O(n + |\mathcal{OSR}_\delta|)$ time which is bounded by $O(n/\delta^2)$. Thus, \mathcal{OSR}_δ can be computed in $O(n/\delta^2)$ time for the case of constant alphabet size and in $O(n \log n + n/\delta^2)$ time for the general case. Note that by Proposition 5 for discovering all principal repeats from the maximal $\frac{1}{\delta}$ -gapped repeats it is enough to compute all stretchable $\frac{1}{\delta}$ -gapped repeats in w . To compute stretchable $\frac{1}{\delta}$ -gapped repeats, we maintain an auxiliary two-way queue SRQ consisting of elements from \mathcal{OSR}_δ . Elements from \mathcal{OSR}_δ are presented by pairs (p, q) where p and q are respectively the period and the end position of the presented element (it is easy to note that any element of \mathcal{OSR}_δ determined uniquely by its period and its start position, so two different elements can not be presented by the same pair in SRQ). At any time the queue SRQ has a form:

$$(p_1, q_1), (p_2, q_2), \dots, (p_s, q_s) \tag{13}$$

where $p_1 < p_2 < \dots < p_s$ and $q_1 < q_2 < \dots < q_s$. Starting from empty SRQ, we try to insert in SRQ each element of \mathcal{OSR}_δ in the prescribed order by the following way. The first element of \mathcal{OSR}_δ is simply inserted in empty SRQ. Let an element τ with period p and end position q be the next candidate for insertion in the queue SRQ presented in (13). Firstly we find the periods p_i and p_{i+1} such that $p_i \leq p < p_{i+1}$ and² compare q with q_i . If $q \leq q_i$ we establish that τ is a stretchable repeat³ and don't insert τ in SRQ. Otherwise we insert τ in SRQ and remove from SRQ all pairs (p_j, q_j) such that $j > i$ and $q_j \leq q$ in order to preserve SRQ in the proper form. Using Proposition 6, one can check that the

²We describe our algorithm for the general case when both p_i and p_{i+1} are exist. The cases when either p_i or p_{i+1} does not exist are easily derived from this general case.

³It is easy to check that in this case τ can not be a repetition.

described procedure compute correctly all stretchable repeats from \mathcal{OSR}_δ which allows to compute all principal $\frac{1}{\delta}$ -gapped repeats in w . For effective execution of operations required in this procedure we use the data structure proposed in [8]. This data structure can be constructed in $O(n \log \log n)$ time and allows to execute the operations of finding p_i , inserting an element to SRQ and removing an element from SRQ in $O(\log \log n)$ time. Note that in the described procedure no more than one of each of these three operations is required for treating any element from \mathcal{OSR}_δ . Thus, the time required for computing all stretchable repeats in \mathcal{OSR}_δ is $O(n \log \log n + |\mathcal{OSR}_\delta| \log \log n)$, so can be bounded by $O(n \log \log n / \delta^2)$. Summing up this time bound with the time bound for computing the set \mathcal{OSR}_δ , we obtain

Theorem 3 *Let $0 < \delta < 1$. Then all maximal δ -subrepetitions in w can be computed in $O(\frac{n \log \log n}{\delta^2})$ time for the case of constant alphabet size and in $O(n \log n + \frac{n \log \log n}{\delta^2})$ time for the general case.*

Another algorithm for computing all principal $\frac{1}{\delta}$ -gapped repeats in a word is based on Proposition 4. By this proposition, in order to check if a maximal gapped repeat σ in w is principal we can compute the minimal period of $w[\text{beg}(\sigma)..\text{end}(\sigma)]$ and compare this period with $p(\sigma)$: if these periods are equal then σ is principal; otherwise σ is not principal. The problem of effective answering to queries related to minimal periods of factors in a word is studied in [12]. In particular, in [12] a hash table data structure is proposed for resolving this problem. This data structure can be constructed in $O(n \log n)$ expected time and allows to compute the minimal period p of a required factor u in $O(\log(1 + \frac{|u|}{|u|-p}))$ time. Note that for any $\frac{1}{\delta}$ -gapped repeat σ in w we have $p(w[\text{beg}(\sigma)..\text{end}(\sigma)]) \leq p(\sigma) \leq \frac{|p|}{1+\delta}$, so $p(w[\text{beg}(\sigma)..\text{end}(\sigma)])$ can be computed in $O(\log(1 + \frac{1}{\delta}))$ time. Therefore, using the data structure from [12], for any maximal $\frac{1}{\delta}$ -gapped repeat σ in w we can check if σ is principal in $O(\log(1 + \frac{1}{\delta}))$ time. Thus, in our second algorithm we compute the set $\mathcal{GR}_{1/\delta}(w)$ and for each repeat σ from $\mathcal{GR}_{1/\delta}(w)$ check, as described above, if σ is principal. By Theorem 2 the set $\mathcal{GR}_{1/\delta}(w)$ can be computed in $O(n \log n + \frac{n}{\delta^2})$ time. The expected total time for checking all repeats from $\mathcal{GR}_{1/\delta}(w)$ is $O(n \log n + |\mathcal{GR}_{1/\delta}(w)| \log(1 + \frac{1}{\delta}))$, so this time can be bounded by $O(n \log n + \frac{n}{\delta^2} \log \frac{1}{\delta})$ since $|\mathcal{GR}_{1/\delta}(w)| = O(\frac{n}{\delta^2})$ by Lemma 8. Thus we have

Theorem 4 *Let $0 < \delta < 1$. Then all maximal δ -subrepetitions in w can be computed in $O(n \log n + \frac{n}{\delta^2} \log \frac{1}{\delta})$ expected time.*

5 Conclusion

One of our results is the $O(\alpha^2 n)$ upper bound on the number of maximal α -gapped repeats in a word of length n . On the other hand, it is easy to see that this number can be at least⁴ $\Omega(\alpha n)$, so we have a gap between upper and lower

⁴We will naturally assume that $\alpha \leq n$ and $\delta \geq 1/n$.

bounds on this number. Thus we have an open question on the optimality of the obtained upper bound. The performed computer experiments show that the order of growth for the maximal number of maximal α -gapped repeats in a word of length n is αn . It would imply that the order of growth for the maximal number of maximal δ -subrepetitions in a word of length n is $O(n/\delta)$. Checking this conjecture would be of interest to us. We assume also that the proposed algorithms are not time optimal, so improving these algorithms is another direction for further research.

Acknowledgments

This work is partially supported by Russian Foundation for Fundamental Research (Grant 11-01-00508).

References

- [1] G. Brodal, R. Lyngso, C. Pedersen, J. Stoye, Finding Maximal Pairs with Bounded Gap, *Journal of Discrete Algorithms* **1(1)** (2000), 77–104.
- [2] M. Crochemore, An optimal algorithm for computing the repetitions in a word, *Information Processing Letters* **12** (1981), 244–250.
- [3] M. Crochemore, W. Rytter, Squares, cubes, and time-space efficient string searching, *Algorithmica* **13** (1995), 405–425.
- [4] M. Crochemore, C. Hancart, T. Lecroq, Algorithms on Strings, Cambridge University Press, 2007.
- [5] M. Crochemore, L. Ilie, and L. Tinta, Towards a solution to the "runs" conjecture, *Lecture Notes in Comput. Sci.* **5029** (2008), 290–302.
- [6] M. Crochemore, C. Iliopoulos, M. Kubica, J. Radoszewski, W. Rytter, T. Walen, Extracting powers and periods in a string from its runs structure, *Lecture Notes in Comput. Sci.* **6393** (2010), 258–269.
- [7] M. Crochemore, M. Kubica, J. Radoszewski, W. Rytter, T. Walen, On the maximal sum of exponents of runs in a string, *Lecture Notes in Comput. Sci.* **6460** (2011), 10–19.
- [8] P. van Emde Boas, R. Kaas, E. Zulstra, Design and Implementation of an Efficient Priority Queue, *Mathematical Systems Theory* **10** (1977), 99–127.
- [9] Z. Galil, J. Seiferas, Time-space optimal string matching, *Journal of Computer and System Sciences* **26(3)** (1983), 280–294.
- [10] D. Gusfield, Algorithms on Strings, Trees, and Sequences, Cambridge University Press, 1997.

- [11] D. Gusfield, J. Stoye, Linear time algorithms for finding and representing all the tandem repeats in a string, *Journal of Computer and System Sciences* **69(4)** (2004), 525–546.
- [12] T. Kociumaka, J. Radoszewski, W. Rytter, T. Walen, Efficient Data Structures for the Factor Periodicity Problem, *Lecture Notes in Comput. Sci.* **7608** (2012), 284–294.
- [13] R. Kolpakov, G. Kucherov, On Maximal Repetitions in Words, *Journal of Discrete Algorithms* **1(1)** (2000), 159–186.
- [14] R. Kolpakov, G. Kucherov, Finding Repeats with Fixed Gap, *Proceedings of 7th International Symposium on String Processing and Information Retrieval (SPIRE'00)* (2000), 162–168.
- [15] R. Kolpakov, G. Kucherov, Periodic structures in words, chapter for *the 3rd Lothaire volume Applied Combinatorics on Words*, Cambridge University Press, 2005.
- [16] R. Kolpakov, G. Kucherov, P. Ochem, On maximal repetitions of arbitrary exponent, *Information Processing Letters*, **110(7)** (2010), 252–256.
- [17] R. Kolpakov, On primary and secondary repetitions in words, *Theoretical Computer Science*, **418** (2012), 71–81.
- [18] M. Lothaire, Combinatorics on Words, volume 17 of *Encyclopedia of Mathematics and Its Applications*, Addison Wesley, 1983.
- [19] J. Storer, Data compression: methods and theory, Computer Science Press, Rockville, MD, 1988.