doi: 10.1515/umcsmath-2015-0013

ANNALES
UNIVERSITATIS MARIAE CURIE-SKŁODOWSKA
LUBLIN – POLONIA

VOL. LXIX, NO. 1, 2015 SECTIO A 73–82

ADAM GĄGOL

Pattern avoidance in partial words over a ternary alphabet

Abstract. Blanched-Sadri and Woodhouse in 2013 have proven the conjecture of Cassaigne, stating that any pattern with m distinct variables and of length at least 2^m is avoidable over a ternary alphabet and if the length is at least $3 \cdot 2^{m-1}$ it is avoidable over a binary alphabet. They conjectured that similar theorems are true for partial words – sequences, in which some characters are left "blank". Using method of entropy compression, we obtain the partial words version of the theorem for ternary words.

1. Introduction. Let $\Sigma = \{a, b, c, \dots\}$ and $\Delta = \{A, B, C, \dots\}$ be finite alphabets. We refer to elements of Σ as *letters* and to elements of Δ as *variables*. A *word* w over some alphabet is a sequence of letters from this alphabet, an *infinite word* is an infinite sequence of letters. A factor of w is a subsequence of w consisting of consecutive letters. A *prefix* of w is a factor containing the first letter of w and a *suffix* is a factor containing its last letter. A *pattern* p is a word over Δ and a *doubled pattern* is a pattern in which every variable occurs at least twice. A word w over Σ is an *instance of* p if there exists a morphism $h : \Delta^+ \to \Sigma^+$ such that $h(p) = w$. A word w is said to *avoid* p if no factor of w is an instance of p. For example, *aabaac* contains an instance of ABA and *abaca* avoids AA.

A *partial word* over alphabet Σ is a sequence of characters from extended alphabet $\Sigma_\diamond = \Sigma \cup \{\diamond\}$, an occurrence of $\diamond$ is called a *hole*. For a partial word

2010 *Mathematics Subject Classification.* 68R15.

Key words and phrases. Formal languages, combinatorics on words, pattern avoidance, partial words, entropy compression, probabilistic method.

w we denote the set of positions of holes as holes(w). A partial word w is an instance of p if there exists a substitution of single letters from Σ to holes(w) such that the resulting word is an instance of p. For example, $w = a \diamond ab$ contains an instance of AAA but it avoids $ABBA$ and holes(w) = $\{2\}$.

The *avoidability index* $\lambda(p)$ of pattern p is the size of the smallest alphabet Σ such that there exists an infinite word over Σ that avoids p. The *partial avoidability index* $\lambda^*(p)$ of pattern p is the size of the smallest alphabet Σ such that there exists an infinite partial word W over $\Sigma_\diamond$ avoiding p and with $|\text{holes}(W)| = \infty$.

Blanchet-Sadri and Woodhouse [1] and independently Ochem and Pinlou [8] proved the following conjecture of Cassaigne [2]:

Theorem 1.1 ([2]). *Let p be a pattern with exactly k distinct variables.*

(1) *If p has length at least 2^k then $\lambda(p) \leq 3$.*
(2) *If p has length at least $3 \cdot 2^{k-1}$ then $\lambda(p) = 2$.*

It was known previously that above bounds are the best possible [6]. Blanchet-Sadri and Woodhouse conjectured that for partial avoidability the first statement remains true for doubled patterns with at least 4 variables and the second remains true without changes. Proof of the first statement for partial words is the main result of this paper, i.e. we will prove that if p is a doubled pattern with $k \geq 4$ variables and length at least 2^k, then $\lambda(p) \leq 3$.

2. Tools and notation. In this section we introduce a few classical combinatorial concepts and results which will be used in the proof.

2.1. Analytic combinatorics. First we need several concepts of analytic combinatorics. We send readers not familiar with this topic to an excellent book of Flajolet and Sedgewick [3]. We say that a number sequence $(a_i)_{i\in\mathbb{N}}$ is of exponential order K^n, which we abbreviate as $a_n \bowtie K^n$ iff:

$$\limsup_{n\to\infty} \sqrt[n]{|a_n|} = K.$$

We will also use one of the basic ordinary generating functions operator, namely SEQ. The operator corresponds to the class of objects $1 + E + 1 + E_2 + \dots$ and represents sequences, i.e. the slots are not being permuted and there is exactly one empty sequence. We have

$$\begin{aligned}\text{SEQ}(f(z)) &= 1 + \Sigma_{n\geq 1} Z(E_n)(f(z), f(z^2), \dots, f(z^n)) \\ &= 1 + \Sigma_{n\geq 1} f(z)^n = \frac{1}{1 - f(z)}.\end{aligned}$$

Analytic combinatorics will be used in the proof as a tool for bounding asymptotic growth of coefficients of the generating function $f(z)$ defined by an equation of type $f(z) = z \cdot \phi(f(z))$. The following theorem will allow us to do that:

Theorem 2.1 (Flajolet, Sedgewick [3], Proposition IV.5). *Let ϕ be a function analytic at 0, having non-negative Taylor coefficients such that $\phi(0) \neq 0$. Let $R \leq +\infty$ be the radius of convergence of the series representing ϕ at 0. Under the condition,*

$$\lim_{x \to R_-} \frac{x\phi'(x)}{\phi(x)} > 1, \tag{1}$$

there is a unique solution $\tau \in (0, R)$ of the characteristic equation:

$$\frac{\tau\phi'(\tau)}{\phi(\tau)} = 1. \tag{2}$$

Then, the formal solution $y(z)$ of the equation $y(z) = z \cdot \phi(y(z))$ is analytic at 0 and its coefficients satisfies exponential growth formula:

$$[z^n]f(z) \bowtie \left(\frac{1}{\rho}\right)^n$$

where $p = \frac{\tau}{\phi(\tau)} = \frac{1}{\phi'(\tau)}$.

Based on the above theorem we introduce the general method for bounding the exponential order of combinatorial sequences proposed by Zydroń [9]. Let $f(z) = \Sigma_{i=0}^{\infty} f_i z^i$ be a generating function satisfying an equation $f(z) = z \cdot \phi(f(z))$ where $\phi(y)$ satisfies the following conditions:

- I $\phi(0) \neq 0$,
- II $\phi(y)$ is analytic in 0,
- III $\forall_{n \geq 0}\ [y^n]\phi(y) \geq 0$,
- IV $\lim_{y \to R^-} \phi(y) = +\infty$, where R is the finite radius of convergence of power series expansion of $\phi(y)$ at 0.

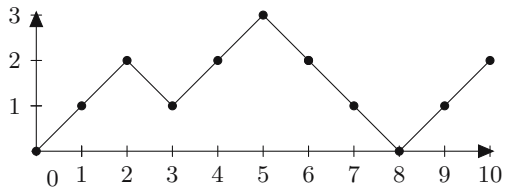
Define function $z(f) = \frac{f}{\phi(f)}$ – an inversion of f (calculated from the equation defining f). Note that the condition IV implies that:

$$0 > \lim_{f \to R^-} z'(f) = \lim_{f \to R^-} \left(\frac{1}{\phi(f)} - \frac{f \cdot \phi(f)'}{\phi(f)^2}\right)$$
$$\Downarrow$$
$$\lim_{f \to R^-} \left(\frac{f \cdot \phi(f)'}{\phi(f)}\right) > 1.$$

and hence that the condition (1) of Theorem 2.1 is satisfied. It means that $\phi(y)$ satisfies all conditions of Theorem 2.1 so there is precisely one solution of the equation (2) and hence also the equation $z'(\tau) = 0$. Note that $z(0) = \lim_{f \to R^-} z(f) = 0$ and $z(f)$ is non-negative in the interval $(0, R)$. Based on the above fact we deduce that $z(f)$'s only maximum in $(0, R)$ is the point τ. Moreover, from thesis of Theorem 2.1 we get that $z(\tau)$ is a radius of convergence of $f(z)$. Based on the above observations we are ready to propose a general method for bounding exponential order of the coefficients of f:

Step 1. Express generating function f as solution of the equation $f(z) = z \cdot \phi(f(z))$ where ϕ satisfies conditions I–IV.
Step 2. Basing on the above equation, calculate function $z(f)$ – an inversion of $f(z)$.
Step 3. Find a point f_0 inside the interval $(0, R)$ where R is a radius of convergence of $\phi(f)$ such that $z(f_0) > \frac{1}{K}$ for some K.
Step 4. Basing on Theorem 2.1, deduce that radius of convergence of the function $f(z)$ is greater than $\frac{1}{K}$ and hence that its coefficients are of exponential order at most K^n.
Note that K does not need to be the maximal value of $z(f)$, which allows us to use numerical computations in the proof.

2.2. Dyck paths. A *Generalized Dyck path* of type (n, m) is a path on the square lattice with steps $(1, 1)$ and $(1, -1)$ from $(0, 0)$ to (n, m) that never falls below the x-axis. We denote the number of all generalized Dyck paths of type (m, n) as $D_{m,n}$. Dyck paths are a standard example of structures counted by Catalan numbers - $D_{2n,0}$ is equal to C_n.



A sample generalized Dyck path of type $(10, 2)$

3. Proof. The proof follows the general framework of Moser–Tardosz algorithmisation of Lovasz local lemma [7] adjusted for application to sequences by Grytczuk, Kozik and Micek [4]. We will refer to this method as *entropy compression*. In the proof we assume that it is not possible to construct an infinite word over alphabet $\{a, b, c\}$ avoiding pattern p and therefore there is such n that every word of length n contains an instance of p. Hence a naive algorithm that given an infinite sequence of letters $S = \{a, b, c\}^*$ tries to construct a word W of length n avoiding pattern p never stops. We use this fact to get the desired contradiction by compressing initial segments of the sequence S to a better extent than it is actually possible. There are 3^M possible prefixes of sequence S of length M and we want to show that it is possible to reconstruct any such prefix from a structure created by the algorithm and that there is strictly less than 3^M such structures possible to obtain after M steps of the algorithm.

Theorem 3.1. *If p is a doubled pattern of length 2^k with $k \geq 4$ variables, then $\lambda(p)^* \leq 3$.*

Proof. Let us fix a pattern p with at least 4 variables and arbitrarily large N. We will prove by contradiction that it is possible to construct a word $W = w_1 \dots w_N$ over alphabet $\Sigma = \{a, b, c\}$ with holes$(W) = \{i : 100 \mid i\}$ avoiding p. We consider Algorithm 1 running on a random source S that tries to assign letters to all positions of W (even the ones with positions divisible by 100 for easier analysis) and retract all instances of p treating positions in holes(W) as proper holes when it comes to pattern identification and retracting letters assigned to them normally.

Algorithm 1: Avoiding pattern P

```
1   input: S : ℕ → Σ = {a, b, c}
2   i ← 1,
3   j ← 1
4   while Symbols are not assigned to entire W do
5   |   w_j ← S(i)
6   |   i++
7   |   j++
8   |   if there is an occurrence R of pattern p ending in w_j, then
9   |   |   let W_R be the positions of R
10  |   |   for k ∈ W_R do
11  |   |   |   erase the value of w_k
12  |   |   j ← index of the first point in W_R
13  |   return W
```

Note that by our assumption that appropriate assignment of letters does not exist, the algorithm never stops. Let us fix some input sequence S and run the algorithm for M steps (i.e. M iterations of the main loop). With every such run we associate some structure describing the behavior of the algorithm. Clearly such a structure depends only on M initial values of S. More importantly sequences S and S' which differ on at least one of M initial positions would produce different structures. The structure we use for description of a run of the algorithm is a tuple (P, L, R, H, F) where:

(1) $P = (p_1, \dots, p_M)$ is a sequence of numbers such that p_i is the number of places with assigned symbols after i-th step (i.e. the number of indexes i for which w_i is defined),

(2) $L = (L_1, \dots, L_s)$ is a sequence of sets of numbers such that $L_i = \{l_{i,1}, \dots, l_{i,k-1}\}$ where $l_{i,j}$ is a number of letters assigned to j-th variable in the i-th retracted occurrence of p during the runtime of the algorithm,

(3) $R = (r_1, \dots, r_r)$ is a sequence of letters such that after retraction of pattern p, letters assigned to variables $A, B, C, \dots$ are added as suffixes of S,

(4) $H = \{h_1, \ldots h_v\}$ is a sequence of letters assigned to holes in retracted instances of p. After every retraction we add to H as many letters as many holes were retracted. It is somewhat redundant with R but it does not make an asymptotic difference,
(5) $F = (f_1, \ldots, f_n)$ is a sequence of symbols left in the word W after M steps of the algorithm.

Now we need to prove that this encoding of a prefix of S is loseless and that it is an actual compression for M large enough.

Loselessness. We prove that it is possible to reconstruct the first M elements of the input sequence S from a tuple (P, L, R, H, F) constructed in M steps of the algorithm. Given (P, L, R, H, F) we are going to decode $S(M)$ and (P', L', R', H', F') - tuple constructed by the algorithm running for $M-1$ steps on the same input sequence S. Then by simple iteration we can extract all values $S(i)$ for $i \in \{1, \ldots, M\}$. We consider two cases:
Case 1. If $p_M = p_{M-1} + 1$ then no pattern instance was retracted during the last step of the algorithm. Then:

- $S(M)$ is simply the last element of F,
- P' is one element shorter,
- $L' = L$,
- $R' = R$,
- $H' = H$,
- F' is one element shorter.

Case 2. If $p_M = p_{M-1} - r + 1$ where $k > 0$, then in the last step there was a retraction of r elements that formed an instance of the pattern p. Then from the last element of L we can reconstruct the structure of this instance, i.e. numbers of letters assigned to each of the variables. From the last element of P we know in which place of the word W an instance occurred and hence the number and placement of holes present in the instance. From the last elements of R we are able to reconstruct the exact letters forming an instance (number of letters we need to subtract from R is equal to the sum of lengths of subwords substituted to variables, which we already know from L). Note that these letters were not necessarily the letters assigned to the places with holes so finally from H we reconstruct the letters assigned to all holes in the instance (number of holes is already known from P). Knowing precisely the structure of the retracted fragment, we can find the last element of S and the quintuple (P', L', R', H', F'):

- $S(M)$ is the last element of the reconstructed retracted fragment,
- P' is one element shorter,
- L' is one element (i.e. set) shorter,
- R' is shorter by all elements used to reconstruct the retracted fragment,

- H' is shorter by as many elements as many holes were in the reconstructed fragment,
- F' is equal to F with the reconstructed fragment added as a suffix without the last element added at the end.

Compression. We are concerned with the asymptotic number of descriptions when M tends to infinity. We will bound P, L, R together and then separately H and F.

Bounding P, L, R. We use analytic combinatorics to find an exponential order of sequence $(T_i)_{i\in\mathbb{N}}$ of possible tuples (P, L, R) occurring after i steps of the algorithm. Before we can use the previously presented method, we need to apply two transformations on P. First we transform P's into generalized Dyck paths by adding downsteps for every retraction – if in P number n follows number k and $n < k$ we add between them all natural numbers between n and k. For example sequence $(0, 1, 2, 3, 4, 1, 2, 0)$ would be transformed into $(0, 1, 2, 3, 4, 3, 2, 1, 2, 1, 0)$. Such modified P is a sequence in which two consecutive numbers differ by exactly 1, which clearly corresponds to a generalized Dyck path. Note that this operation makes P at most two times longer.

Second transformation we apply only to the paths ending on level other than first. Every such a path we artificially prolong by adding sequence of upward steps until it reaches level N and then sequence of downward steps until it reaches level 1. We add upward steps because we want to keep the condition that all the paths have descendances at least as long as many variables are in the pattern p.

Note that if M (numbers of steps of Algorithm 1, which is now close to half of the length of paths) will be big enough in comparison to N, then such operation will not change the exponential order of the number of our paths. We construct the desired generating function step by step.

Let $P(z)$ be the generating function encoding all Dyck paths with falls of lengths being lengths of possible retractions in the algorithm, $PL(z)$ be the generating function of such Dyck paths encoded together with possible L's and finally $t(z)$ be the desired generating function for P, L and R. We will use Flajolet's symbolic operators notation [3] for operations on combinatorial classes. We use slightly modified last passages decomposition for Dyck paths. Let $P_{0,n}(z)$ be a generating function of possible paths starting at level 0 and ending at level n. Recording the times at which each level $0, \ldots, n$ is last traversed gives us $P_{0,n}(z) = P_{0,1}(z)^{n-1}$ so summing up for all possible last descendances, we get $P(z) = z(1+\mathrm{SEQ}(P(z)))$. Since together with last descendance we want to record L we need to divide it into k parts corresponding to variables in such a way that the part corresponding to the i-th variable occurring u_i times in p is of length divisible by u_i.

We get

$$PL(z) = z\left(1 + \left(\prod_{i=1}^{k}\left(\mathrm{SEQ}\big(\mathrm{SEQ}_{u_i}(PL(z))\big)\right)\right)\right)$$

Encoding it together with R, we get construction for t and transform it into generating function equation:

$$t(z) = z\left(1 + \left(\prod_{i=1}^{k}\left(\mathrm{SEQ}\big(\mathrm{SEQ}_{u_i}(3 \cdot t(z))\big)\right)\right)\right)$$

$$\Downarrow$$

$$t(z) = z\left(1 + \left(\prod_{i=1}^{k}\left(\frac{3t(z)^{u_i}}{1-3t(z)^{u_i}}\right)\right)\right).$$

Function $\phi(z) = 1+\left(\prod_{i=1}^{k}\left(\frac{3z^{u_i}}{1-3z^{u_i}}\right)\right)$ satisfies conditions $I-VI$ necessary to use our method. Since $t(z)$ is the formal solution of the equation $t(z) = z \cdot \phi(t(z))$ and we are interested in bounding exponential order of its coefficients from above we need to investigate maximum of its inversion – $\frac{x}{\phi(x)}$. To do that we need to find for which u's the function achieves the smallest values. For this purpose we consider the function

$$\varphi(u_1, \ldots, u_k) = \frac{t}{1 + \prod_{i=1}^{k}\left(\frac{3t^{u_i}}{1-3t^{u_i}}\right)}$$

for $t \in (0, 0.6), u_1, \ldots, u_k \geq 2$ – we can use this restriction on t since we don't need to find the real maximum of $\frac{x}{\phi(x)}$ and restriction on u's comes from the fact that the pattern is doubled. Since $\varphi(u_1, \ldots, u_k)$ is convex for variables $u_1, \ldots, u_k$ and maximal value in convex set $\{(u_1, \ldots, u_k); 2 \leq u_i, \Sigma_{i=1}^{k} u_k = 2^k\}$ is one of sets extremal points we get:

$$\frac{t}{1 + \prod_{i=1}^{k}\left(\frac{3t^{u_i}}{1-3t^{u_i}}\right)} \geq \frac{t}{1 + \prod_{i=1}^{k-1}\frac{3t^2}{(1-3t^2)} \cdot \frac{3t^{2^k-2k+2}}{1-3t^{2^k-2k+2}}}$$

$$\Downarrow$$

$$\frac{t}{1 + \prod_{i=1}^{k}\frac{3t^{u_i}}{1-3t^{u_i}}} \geq \frac{t}{1 + \prod_{i=1}^{3}\frac{3t^2}{1-3t^2} \cdot \frac{3t^{10}}{1-3t^{10}}}.$$

Using Maple software, we check that the right side of the last inequality achieves 0.471 for $t = 0.487$ so exponential order of T_n is at most $1/0.487 = 2.0533$ and hence there is at most 2.0533^M possible tuples P, L, R.

Bounding H. Since a pattern p has length at least 2^k for $k \geq 4$ and its every retracted instance has at least one letter substituted to every variable then every retraction is of length at least 16. Also, since distance between two holes is 100, every retraction adds at most $\lceil \frac{|R|}{100} \rceil$ letters to H. Moreover,

the sum of retraction lengths is at most M (we couldn't retract more than we wrote) so there is at most $3^{\frac{M}{16}} < 1.08^M$ possible H's.

Bounding F. Finite sequence is of length at most $|W| - 1$ so there are less then $4^{|W|}$ possibilities of such sequence, because there can be assigned symbols $0, 1, 2$ or no symbol assigned at every place. Symbols $\diamond$ are assigned at prespecified positions so there is no need to encode them.

Bounding P, L, R, H, F. Summing all bounds together, we get that for M big enough there is at most $(2.0533 \cdot 1.08)^M \cdot 4^{|W|} < 2.2176^M \cdot 4^{|W|} < 3^M$ tuples and they fully describe 3^M possible prefixes of S, which gives us the desired contradiction. □

4. Concluding remarks. Overall, entropy compression is a useful method for proving results for partial words. It is mainly helpful to obtain bounds for bigger numbers of variables, since it needs to encode Dyck paths and takes advantage of the fact that short retractions never occur. Straightforward application of developed methods for the second part of the conjecture – about binary words, provides the desired result for doubled patterns with at least 3 variables. To prove it for shorter patterns it may be necessary to use some other, possibly deterministic methods.

References

[1] Blanchet-Sadri, F., Woodhouse, B., *Strict Bounds for Pattern Avoidance*, Theoret. Comput. Sci. **506** (2013), 17–28.

[2] Cassaigne, J., *Motifs évitables et régularités dans les mots*, PhD Thesis, Université Paris VI, July 1994.

[3] Flajolet, P., Sedgewick, R., *Analytic Combinatorics*. Cambridge University Press, 2009, ISBN 978-0-521-89806-5, electronic version.

[4] Grytczuk, J., Kozik, J., Micek, P., *A new approach to nonrepetitive sequences*, Random Structures Algorithms **42** (2013), 214–225.

[5] Krieger, D., Ochem, P., Rampersad, N., Shallit, J., *Avoiding Approximate Squares*, Lecture Notes in Computer Science, Vol. 4588, 2007, 278–289.

[6] Lothaire, M., *Algebraic Combinatorics on Words*, Cambridge University Press, Cambridge, 2002.

[7] Moser, R. A., Tardos, G., *A constructive proof of the general lovasz local lemma*, J. ACM **57** (2) (2010), Art. 11, 15 pp.

[8] Ochem, P., Pinlou, A., *Application of entropy compression in pattern avoidance*, Electron. J. Combin. **21**, P2.7 (2014).

[9] Zydroń, A., *Unikalność bezjednostkowych wzorców o dużej liczbie zmiennych*, MsC Thesis, Jagiellonian University, 2013.

Adam Gągol
Institute of Mathematics
Maria Curie-Skłodowska University
pl. M. Curie-Skłodowskiej 1
20-031 Lublin
Poland
e-mail: adam.gagol@gmail.com

Received June 9, 2014