Theoretical Computer Science 615 (2016) 12–22

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Synchronizing delay for binary uniform morphisms

Karel Klouda [a,*], Kateřina Medková [b]

[a] Faculty of Information Technology, Czech Technical University in Prague, Thákurova 9, 160 00, Prague 6, Czech Republic
[b] Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague, Břehová 7, 115 19, Prague 1, Czech Republic

ARTICLE INFO

Article history:
Received 22 July 2015
Received in revised form 16 November 2015
Accepted 19 November 2015
Available online 8 December 2015
Communicated by D. Perrin

Keywords:
D0L-system
Circularity
Synchronizing delay

ABSTRACT

Circular D0L-systems are those with finite synchronizing delay. We introduce a tool called graph of overhangs which can be used to find the minimal value of synchronizing delay of a given D0L-system. By studying the graphs of overhangs, a general upper bound on the minimal value of a synchronizing delay of a circular D0L-system with a binary uniform morphism is given.

1. Introduction

Circular codes are a classical notion studied in theory of codes [1]. A set X of finite words is a code if each word in X^+ has a unique decomposition into words from X. If we slightly modify the requirement of uniqueness, we get the definition of a circular code: X is a circular code if each word in X^+ written in a circle has a unique decomposition into words from X.

An analogue to codes in the family of D0L-systems are D0L-systems that are injective on the set of all factors of their languages. Circularity is defined as slightly relaxed injectivity: a D0L-system is circular if long enough factors of its language have a unique preimage (under the respective morphism) in the language except for some prefix and suffix bounded in length by some constant. This constant is called a synchronizing delay and it is studied in this paper.

In the case of D0L-systems circularity is connected with repetitiveness. As stated in [2], a non-circular D0L-system is repetitive, i.e., for each $k \in \mathbb{N}$ there exists a word v such that v^k is a factor of the language. In fact, if a D0L-system is not pushy (which is always true if the morphism is uniform and the language is infinite), then circularity is equivalent to non-repetitiveness [3].

As explained by Cassaigne in [4], knowledge of the value of the synchronizing delay can be very helpful when analyzing the structure of bispecial factors in languages of D0L-systems. This idea was further developed by one of the authors in [5], where an algorithm for generating all bispecial factors is given. This algorithm works for circular and non-pushy D0L-systems and its computational complexity depends on the value of the synchronizing delay. This fact and the absence of any known bound on the value of synchronizing delay is the main motivation of the present work.

Unfortunately, it does not seem easy to find such a bound. Therefore we focus on the simplest case: a circular D0L-system with binary k-uniform morphism with $k \geq 2$. Using the notion of the graph of overhangs introduced in Subsection 2.2, we prove the following result. The details of the proof are given in Section 3.

* Corresponding author.
E-mail address: karel.klouda@fit.cvut.cz (K. Klouda).

http://dx.doi.org/10.1016/j.tcs.2015.11.043
0304-3975/

Theorem 1. *If the morphism φ of a circular D0L-system $(\{a, b\}, \varphi, a)$ is k-uniform, then the minimum value of its synchronizing delay, denoted by Z_{min}, is bounded as follows:*

(i) $Z_{min} \le 8$ *if* $k = 2$,
(ii) $Z_{min} \le k^2 + 3k - 4$ *if k is an odd prime number,*
(iii) $Z_{min} \le k^2 \left(\frac{k}{d} - 1\right) + 5k - 4$ *otherwise,*

where number d is the least divisor of k greater than 1.

2. Preliminaries

A finite set of symbols is an *alphabet*, denoted by $\mathcal{A}$. The set of all finite words over $\mathcal{A}$ is denoted by $\mathcal{A}^*$, the *empty word* is ε and $\mathcal{A}^+ = \mathcal{A}^* \setminus \{\varepsilon\}$. If a word $u \in \mathcal{A}^*$ is a concatenation of three words x, y and z from $\mathcal{A}^*$, i.e., $u = xyz$, the word x is a *prefix* of u, y its *factor* and z a *suffix*. We put $x^{-1}u = yz$ and $uz^{-1} = xy$. The *length* of the word u equals the number of letters in u and is denoted by $|u|$; $|u|_a$ denotes the number of occurrences of a letter a in u.

A mapping $\varphi : \mathcal{A}^* \to \mathcal{A}^*$ is a *morphism* if for every $v, u \in \mathcal{A}^*$ we have $\varphi(vu) = \varphi(v)\varphi(u)$. A triplet $G = (\mathcal{A}, \varphi, w)$ is a *D0L-system*, if φ is a morphism on $\mathcal{A}$ and $w \in \mathcal{A}^+$. The word w is called an *axiom*. The *language of* G is the set $L(G) = \{\varphi^n(w) : n \in \mathbb{N}\}$. The set of all factors of elements of $L(G)$ is denoted by $S(L(G))$. The alphabet is always considered to be the minimal alphabet necessary, i.e., $\mathcal{A} \cap S(L(G)) = \mathcal{A}$.

A D0L-system $G = (\mathcal{A}, \varphi, w)$ is *injective on* $S(L(G))$ if for every $u, v \in S(L(G))$, $\varphi(u) = \varphi(v)$ implies that $u = v$. It is clear that if φ is injective, then G is injective. If φ is *non-erasing*, i.e., $\varphi(a) \ne \varepsilon$ for all $a \in \mathcal{A}$, then G is a *propagating D0L-system*, shortly *PD0L-system*.

Given a D0L-system $G = (\mathcal{A}, \varphi, w)$, we say that a letter a is *bounded* if the set $\{\varphi^n(a) : n \in \mathbb{N}\}$ is finite. If a letter is not bounded, it is *unbounded*. The system G is *pushy* if $S(L(G))$ contains infinitely many factors containing bounded letters only.

A D0L-system G is *repetitive* if for any $k \in \mathbb{N}$ there is a non-empty word v such that v^k is a factor from $S(L(G))$. By [6], any repetitive D0L-system is *strongly repetitive*, i.e., there is a non-empty word v such that v^k is a factor for all $k \in \mathbb{N}$. We say that G is *unboundedly repetitive* if there is v containing at least one unbounded letter such that v^k is a factor for all $k \in \mathbb{N}$.

2.1. Circular D0L-systems

In [4], a circular D0L-system is defined using the notion of synchronizing point (see Section 3.2 in [4] for details). We give here an equivalent definition employing the notion of interpretation.

Definition 2. Let $G = (\mathcal{A}, \varphi, w)$ be a PD0L-system and $u \in S(L(G))$. A triplet (p, v, s), where $p, s \in \mathcal{A}^*$ and $v \in S(L(G))$, is an *interpretation of the word u* if $\varphi(v) = pus$.

Definition 3. Let $G = (\mathcal{A}, \varphi, w)$ be a PD0L-system. We say that two interpretations (p, v, s) and (p', v', s') of a word $u \in S(L(G))$ are *synchronized at position k* if there exist indices i and j such that

$$\varphi(v_1 \cdots v_i) = pu_1 \cdots u_k \quad \text{and} \quad \varphi(v'_1 \cdots v'_j) = p'u_1 \cdots u_k$$

with $v = v_1 \cdots v_n \in \mathcal{A}^n$, $v' = v'_1 \cdots v'_m \in \mathcal{A}^m$ and $u = u_1 \cdots u_\ell \in \mathcal{A}^\ell$ (if $k = 0$, we put $u_1 \cdots u_k = \varepsilon$). Two interpretations that are not synchronized at any position are called *non-synchronized*.

We say that a word $u \in S(L(G))$ has a *synchronizing point* at position k with $0 \le k \le |u|$ if all its interpretations are pairwise synchronized at position k.

Definition 4. Let $G = (\mathcal{A}, \varphi, w)$ be a PD0L-system injective on $S(L(G))$. We say that G is circular if there is a positive integer Z, called a *synchronizing delay*, such that any u from $S(L(G))$ longer than Z has a synchronizing point. The minimal constant Z with this property is denoted by $Z_{\min}$.

By the results from [2,3], non-circular systems are repetitive (and by [6] also strongly repetitive). In fact, a D0L-system injective on $S(L(G))$ is not circular if and only if it is *unboundedly repetitive* [3], i.e., there exists v containing an unbounded letter such that $v^k \in S(L(G))$ for all $k \in \mathbb{N}$. Since this property can be checked by a simple algorithm [7], we can easily verify whether a given D0L-system injective on $S(L(G))$ is circular or not.

The notion of circularity is inspired by the notion of circular code:

Definition 5. A subset X of $\mathcal{A}^*$ is called a *code* over alphabet $\mathcal{A}$ if for any word $v \in X^+$ there are uniquely given a number n and words $x_1, x_2, \ldots, x_n$ from X so that $v = x_1x_2 \ldots x_n$.

The set X is a *circular code* over $\mathcal{A}$ if for all $n, m \ge 1$, $x_1, \ldots, x_n, y_1, \ldots, y_m \in X$, $p \in \mathcal{A}^*$ and $s \in \mathcal{A}^+$ it holds that:

$$(sx_2x_3 \cdots x_np = y_1y_2 \cdots y_m \text{ and } x_1 = ps) \Longrightarrow (n = m\,,\ p = \epsilon \text{ and } x_i = y_i\ \forall\, i = 1, \ldots, n)\,.$$