

# Loughborough University Institutional Repository

---

## *Periodicity forcing words*

This item was submitted to Loughborough University's Institutional Repository by the/an author.

**Citation:** DAY, J.D., REIDENBACH, D. and SCHNEIDER, J.C., 2013. Periodicity forcing words. IN: Karhumäki, J., Lepistö, A.; Zamboni, L. (eds.). *Combinatorics on Words: 9th International Conference, WORDS 2013, Turku, Finland, September 16-20, 2013, Proceedings. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*: 8079, pp.107-118.

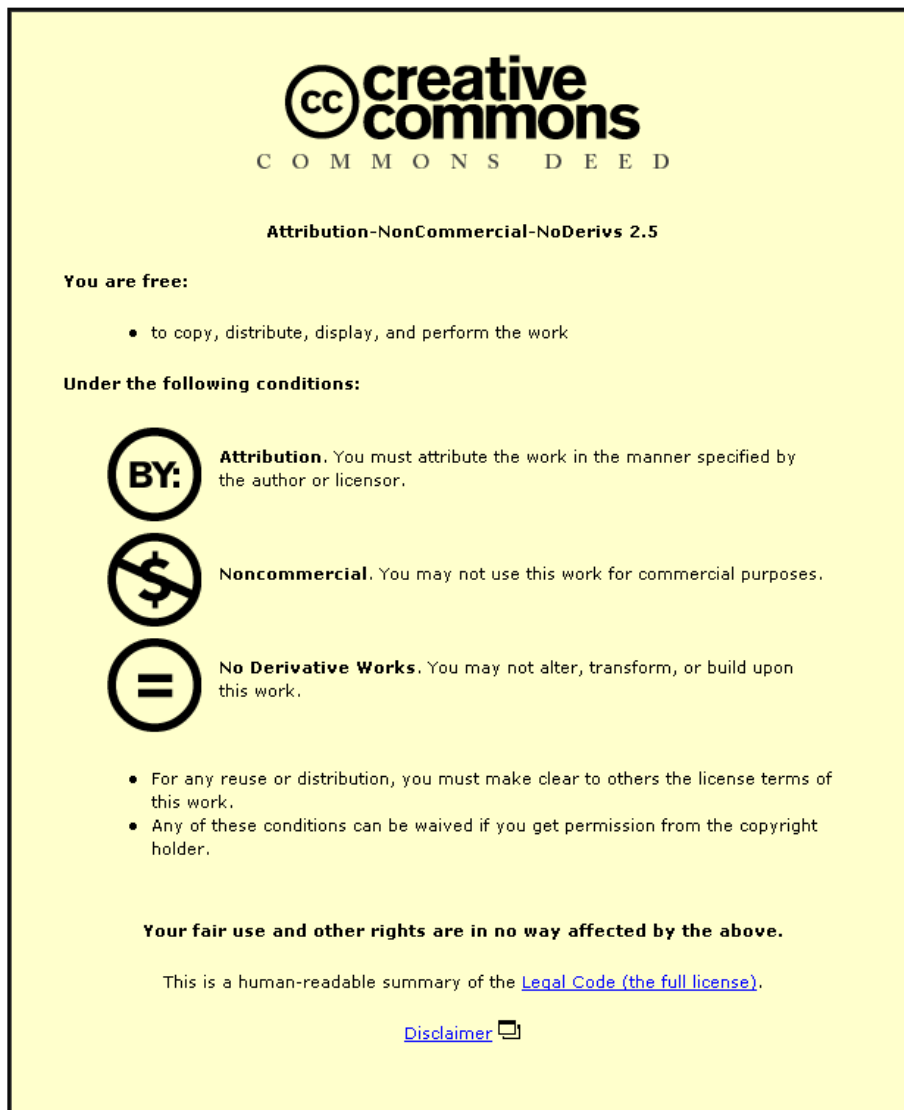
**Metadata Record:** <https://dspace.lboro.ac.uk/2134/14011>

**Version:** Accepted for publication

**Publisher:** © Springer-Verlag Berlin Heidelberg

Please cite the published version.

This item was submitted to Loughborough's Institutional Repository (<https://dspace.lboro.ac.uk/>) by the author and is made available under the following Creative Commons Licence conditions.




creative commons  
COMMONS DEED


**Attribution-NonCommercial-NoDerivs 2.5**


**You are free:**

- to copy, distribute, display, and perform the work

**Under the following conditions:**

 **Attribution.** You must attribute the work in the manner specified by the author or licensor.


 **Noncommercial.** You may not use this work for commercial purposes.

 **No Derivative Works.** You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

**Your fair use and other rights are in no way affected by the above.**

This is a human-readable summary of the [Legal Code \(the full license\)](#).

[Disclaimer](#) 

For the full text of this licence, please go to:  
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

# Periodicity Forcing Words<sup>\*</sup>

Joel D. Day<sup>\*\*1</sup>, Daniel Reidenbach<sup>1</sup>, and Johannes C. Schneider<sup>2</sup>

<sup>1</sup> Department of Computer Science, Loughborough University,  
Loughborough, Leicestershire, LE11 3TU, UK

J.Day-10@student.lboro.ac.uk

D.Reidenbach@lboro.ac.uk

<sup>2</sup> DIaLOGIKa GmbH, Pascalschacht 1,  
66125 Saarbrücken, Germany

johannes.schneider@dialogika.de

**Abstract.** The Dual Post Correspondence Problem asks, for a given word  $\alpha$ , if there exists a non-periodic morphism  $g$  and an arbitrary morphism  $h$  such that  $g(\alpha) = h(\alpha)$ . Thus  $\alpha$  satisfies the Dual PCP if and only if it belongs to a non-trivial equality set. Words which do not satisfy the Dual PCP are called *periodicity forcing*, and are important to the study of word equations, equality sets and ambiguity of morphisms. In this paper, a ‘prime’ subset of periodicity forcing words is presented. It is shown that when combined with a particular type of morphism it generates exactly the full set of periodicity forcing words. Furthermore, it is shown that there exist examples of periodicity forcing words which contain any given factor/prefix/suffix. Finally, an alternative class of mechanisms for generating periodicity forcing words is developed, resulting in a class of examples which contrast those known already.

**Keywords:** Equality sets, Morphisms, Dual Post Correspondence Problem, Periodicity forcing sets, Periodicity forcing words, Ambiguity of morphisms

## 1 Introduction

The Dual Post Correspondence Problem (Dual PCP) is a decidable variation of the famous Post Correspondence Problem (see Post [10]). It was introduced by Culik II and Karhumäki in [1], where the authors make progress towards a characterisation of binary equality sets. A word is said to satisfy the Dual PCP if it belongs to an equality set  $E(g, h)$  for two morphisms  $g, h$  where at least one morphism is non-periodic. For example, the word **abba** belongs to  $E(g, h)$  where  $g(\mathbf{a}) := \mathbf{aba}$ ,  $g(\mathbf{b}) := \mathbf{b}$ ,  $h(\mathbf{a}) := \mathbf{a}$ , and  $h(\mathbf{b}) := \mathbf{bab}$ . Thus **abba** satisfies the Dual PCP; in other words, it is a non-trivial *equality word*. In contrast, the word **abaab** does not satisfy the Dual PCP, but this claim is much harder to verify.

---

\* This work was supported by the London Mathematical Society, grant SC7-1112-02.

\*\* Corresponding author.

In this paper, words which do not satisfy the Dual PCP (often referred to as *periodicity forcing words*) are examined. Periodicity forcing words are of immediate importance to the study of equality sets, since they are words which do not belong to any non-trivial equality set. As a result, they can be viewed as being opposite to equality words. Furthermore, they are strongly related to the studies of word equations and of the ambiguity of morphisms.

Due to both the original research by Culik II and Karhumäki [1], and from more recent research into equality sets (e. g., Holub [5], Hadravova, Holub [4]) and word equations (e. g., Czeizler et al. [2], Karhumäki, Petre [7]), quite a lot is known about the binary case. Much less, however, is known about the general case. One reason for this is that although the Dual PCP is known to be decidable (due to Makanin’s algorithm [9], as shown by Culik II and Karhumäki [1]) deciding on whether a word is periodicity forcing can be a particularly intricate task, and becomes even more so as the alphabet size increases. In [3], we overcome this problem by employing the use of morphisms to generate periodicity forcing words over arbitrary alphabets.

In Sect. 3 of the present paper, we explore the structure of the set of periodicity forcing words ( $\text{DPCP}^\neg$ ) in relation to morphisms. Specifically, a ‘prime’ subset of  $\text{DPCP}^\neg$  is considered from which all periodicity forcing words may be generated using a specific type of morphism, characterised in [3]. In Sect. 4, it is shown that there exist periodicity forcing words with arbitrary factors, providing a level of generality not yet achieved. Finally, in light of the results on ‘prime’ periodicity forcing words, some alternative approaches to generating periodicity forcing words (specifically over large alphabets) are investigated.

## 2 Notation and preliminary results

Let  $\mathbb{N} = \{1, 2, \dots\}$  be the set of natural numbers, and let  $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$ . The set  $\mathbb{N}$  is used as an infinite alphabet of symbols, and words over  $\mathbb{N}$  are referred to as *patterns*. The symbols occurring in a pattern are called *variables*, and the set of variables occurring in a pattern  $\alpha$  is denoted by  $\text{var}(\alpha)$ . Symbols from words which are not patterns (referred to as letters) are indicated using typewriter font (e. g.,  $\Sigma := \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \dots\}$ ).

For an alphabet  $\Sigma := \{a_1, a_2, \dots, a_n\}$  and a word  $u \in \Sigma^*$ , the *Parikh vector*, written  $\text{P}(u)$ , is the vector  $(|u|_{a_1}, |u|_{a_2}, \dots, |u|_{a_n})$ . The result of dividing the Parikh vector by the greatest common divisor of its components is called the *basic Parikh vector*. A word  $u$  is *primitive* if it is not a repetition of a shorter word (i. e.,  $u = v^n$  implies  $n = 1$ ). Otherwise it is *imprimitive*. A word  $u \in \Sigma^*$  is *ratio-imprimitive* if there exist words  $v, w \in \Sigma^+$  such that  $u = vw$  and  $u, v$  share the same basic Parikh vector. Otherwise, it is *ratio-primitive*.

A *morphism*  $h : A^* \rightarrow B^*$  is a mapping which is compatible with concatenation (meaning  $h(uv) = h(u) \cdot h(v)$  for any words  $u, v \in A^*$ ). Thus, although a morphism maps words in  $A^*$  to words in  $B^*$ , it is fully defined once it is specified for each individual symbol in  $A$ . The composition of two morphisms  $g : A^* \rightarrow B^*$

and  $h : B^* \rightarrow C^*$  is the morphism  $g \circ h : A^* \rightarrow C^*$ , given by  $g \circ h(x) = g(h(x))$  for every  $x \in A$ .

A morphism  $g : A^* \rightarrow B^*$  is *periodic* if there exists a word  $w \in B^*$  such that for every  $x \in A$ ,  $g(x) \in \{w\}^*$ . Given another morphism  $h : A^* \rightarrow B^*$ ,  $g$  and  $h$  are said to be *distinct* if there exists an  $x \in A$  such that  $g(x) \neq h(x)$ . If, for some word  $\alpha \in A^+$ ,  $g(\alpha) = h(\alpha)$ , then  $g$  and  $h$  are said to *agree* on  $\alpha$ . A *renaming* of a word  $u \in \{a_1, a_2, \dots, a_n\}^+$  is the word  $\sigma(u)$  where  $\sigma : \{a_1, a_2, \dots, a_n\}^* \rightarrow \{b_1, b_2, \dots, b_n\}^*$  is a morphism given by  $\sigma(a_i) = b_i$ , and where  $b_1, b_2, \dots, b_n$  are distinct letters. If  $\{a_1, a_2, \dots, a_n\} \cap \{b_1, b_2, \dots, b_n\} = \emptyset$ , then the renaming is said to be *strict*. On the other hand, if  $a_i \in \{b_1, b_2, \dots, b_n\}$  for  $1 \leq i \leq n$ , then  $\sigma(u)$  is a *permutation* of  $u$ . For sets  $\Delta$  and  $V \subset \Delta$ , the morphism  $\pi_V : \Delta^* \rightarrow V^*$ , given by  $\pi_V(x) = x$  if  $x \in V$  and  $\pi_V(x) = \varepsilon$  otherwise, is called a *projection*.

A set of patterns is *periodicity forcing* if, whenever two morphisms agree on every pattern in the set, they are periodic. A set of patterns  $T$  is said to be a *test set* of another set of patterns  $S$  if any two morphisms which agree on every pattern in  $T$  also agree on every pattern in  $S$ . Note that this means any test set of a periodicity forcing set must also be periodicity forcing.

A morphism  $\sigma$  is said to be *ambiguous* with respect to a pattern  $\alpha$  if there exists another morphism  $\tau$  such that  $\sigma(\alpha) = \tau(\alpha)$  and  $\sigma, \tau$  are distinct. It is convenient to refer to the following set:  $\text{DPCP} := \{\alpha \in \mathbb{N}^+ \mid \text{there exists a non-periodic morphism } \sigma \text{ and an arbitrary morphism } \tau \text{ such that } \sigma(\alpha) = \tau(\alpha)\}$ . Note that this implies the complement  $\text{DPCP}^c$  is exactly the set of periodicity forcing words.

For a set of unknowns  $\Delta$ , a word equation is an equation  $\alpha = \beta$  for some words  $\alpha, \beta \in \Delta^+$ . It is non-trivial if  $\alpha \neq \beta$ . For a given alphabet  $\Sigma$ , solutions to the word equation are morphisms  $\sigma : \Delta^* \rightarrow \Sigma^*$  such that  $\sigma(\alpha)$  and  $\sigma(\beta)$  are equal. Unless otherwise specified,  $\Delta$  is usually a set of variables, while  $\Sigma$  is a set of letters. As a result, word equations equate patterns, and their solutions map to terminal words (words which are not patterns). The following is a well known and important result on word equations.

**Lemma 1 (Lothaire [8]).** *Non-trivial word equations in two unknowns have only periodic solutions.*

One consequence of Lemma 1 which provides a particularly useful tool is that if two words  $u$  and  $v$  commute (i.e.,  $uv = vu$ ), then  $u$  and  $v$  (and therefore also  $uv$ ) share a primitive root. Similarly, an arbitrarily large set of words  $\{u_1, u_2, \dots, u_n\}$  is said to commute if  $u_1, u_2, \dots, u_n$  all share the same primitive root.

In our investigation into the use of morphisms to generate periodicity forcing words in [3], we provide the following criterion.

**Lemma 2 ([3]).** *Let  $\Delta_1, \Delta_2$  be sets of variables. Let  $\varphi : \Delta_1^* \rightarrow \Delta_2^*$  be a morphism such that for every  $x \in \Delta_2$ , there exists a  $y \in \Delta_1$  such that  $x \in \text{var}(\varphi(y))$ , and*

- (i) *for every non-periodic morphism  $\sigma : \Delta_2^* \rightarrow \{\mathbf{a}, \mathbf{b}\}^*$ ,  $\sigma \circ \varphi$  is non-periodic, and*

(ii) for all distinct morphisms  $\sigma, \tau : \Delta_2^* \rightarrow \{\mathbf{a}, \mathbf{b}\}^*$ , where at least one is non-periodic,  $\sigma \circ \varphi$  and  $\tau \circ \varphi$  are distinct.

Then for any  $\alpha \notin \text{DPCP}$  with  $\text{var}(\alpha) = \Delta_1$ ,  $\varphi(\alpha) \notin \text{DPCP}$ .

Characterisations of morphisms which satisfy conditions (i) and (ii) of Lemma 2 are given in the following propositions respectively.

**Proposition 3 ([3]).** *Let  $\Delta_1$  and  $\Delta_2$  be sets of variables, let  $\varphi : \Delta_1^* \rightarrow \Delta_2^*$  be a morphism, and let  $\beta_i := \varphi(i)$  for every  $i \in \Delta_1$ . The morphism  $\varphi$  satisfies Condition (i) of Lemma 2 if and only if, for every non-periodic morphism  $\sigma : \Delta_2^* \rightarrow \{\mathbf{a}, \mathbf{b}\}^*$ ,*

- (i) *there are at least two patterns  $\beta_i$  such that  $\sigma(\beta_i) \neq \varepsilon$ , and*
- (ii) *there do not exist  $k_1, k_2, \dots, k_n \in \mathbb{N}$  such that*

$$\sigma(\gamma_1)^{k_1} = \sigma(\gamma_2)^{k_2} = \dots = \sigma(\gamma_n)^{k_n} \quad (1)$$

where  $\{\gamma_1, \gamma_2, \dots, \gamma_n\}$  is the set of all patterns  $\beta_i$  such that  $\sigma(\beta_i) \neq \varepsilon$ .

**Proposition 4 ([3]).** *Let  $\Delta_1, \Delta_2$  be sets of variables, and let  $\varphi : \Delta_1^* \rightarrow \Delta_2^*$  be a morphism. For every  $i \in \Delta_1$ , let  $\beta_i := \varphi(i)$ . The morphism  $\varphi$  satisfies Condition (ii) of Lemma 2 if and only if  $\{\beta_1, \beta_2, \dots, \beta_n\}$  is a periodicity forcing set.*

### 3 A ‘Prime’ Generating Subset of $\text{DPCP}^\neg$ .

In this section, the structure of the set  $\text{DPCP}^\neg$ , with respect to morphisms, is investigated. Specifically,  $\text{DPCP}^\neg$  is partitioned according to whether, for a given pattern  $\alpha$ , there exists a morphism  $\varphi$ , and a second pattern  $\beta \notin \text{DPCP}$ , such that  $\alpha = \varphi(\beta)$ .<sup>1</sup> This condition is clearly trivial if  $\beta$  is permitted to be a renaming of  $\alpha$ , so only morphisms which alter the structure of  $\beta$  are considered. Furthermore, the Dual PCP is trivial for unary alphabets, so only patterns  $\alpha$  and  $\beta$  over non-unary alphabets are considered. This partition allows  $\text{DPCP}^\neg$  to be represented as chains of patterns. It can be inferred directly from the constructions given in [3] that every periodicity forcing word is a pre-image of another, meaning these chains are infinite in one direction. In Proposition 9 below, it is shown that there exist patterns for which there does not exist a non-trivial pre-image in  $\text{DPCP}^\neg$  and therefore that some chains terminate. More generally, it can be shown that  $\text{DPCP}^\neg$  is spanned by one-sided infinite chains of this type, and thus that there exists a (strict) subset of  $\text{DPCP}^\neg$  from which all periodicity forcing words can be generated using the morphisms characterised in [3].

The proofs rely on a lower bound on the size of periodicity forcing words (relative to the number of variables), achieved by developing a strong sufficient

<sup>1</sup> It is worth noting that a characterisation of such morphisms  $\varphi$  is given in [3] (Theorem 14).

condition for a pattern to be contained in DPCP. To do this, morphisms of the following form are considered.

$$\sigma(x) := \begin{cases} \mathbf{a}^{p_y} \mathbf{b} \mathbf{a}^{q_y} & \text{if } x = y, \text{ and} \\ \mathbf{a}^{r_x} & \text{otherwise,} \end{cases}$$

for some fixed variable  $y$ , where  $p_x, q_y, r_y$  are numbers depending on the variables  $x$  and  $y$  respectively. Clearly two morphisms  $\sigma_1$  and  $\sigma_2$  of this type agree on a pattern  $\alpha$  if and only if the number of occurrences of  $\mathbf{a}$  coincide between each occurrence of  $\mathbf{b}$ . Thus the agreement of the two morphisms can be determined by solving a system of linear Diophantine equations. In the case that  $n < |\text{var}(\alpha)|$ , it is possible to show that such a system always permits a non-trivial solution – meaning the two morphisms are distinct. Furthermore, it is clear that they are non-periodic, so it is possible to conclude the following.

**Proposition 5.** *Let  $\alpha$  be a pattern, and let  $n := |\text{var}(\alpha)|$ . Suppose that  $|\alpha|_x < n$  for some  $x \in \text{var}(\alpha)$ . Then  $\alpha \in \text{DPCP}$ .*

It follows that, for a periodicity forcing word with  $n$  letters, each letter must occur at least  $n$  times, implying the next corollary which provides a lower bound on the length of the shortest periodicity forcing word with  $n$  letters.

**Corollary 6.** *Let  $\alpha \notin \text{DPCP}$ , and let  $n := |\text{var}(\alpha)|$ . Then  $|\alpha| \geq n^2$ .*

Since periodicity forcing words can be obtained as concatenations of words in a particular type of periodicity forcing set (see Sect. 5), it is possible to infer a corresponding upper bound from results in [6]. The authors provide a concise test set (containing at most  $5n$  words, each of length  $n$ ) for the set  $S_n$  consisting of all permutations of the word  $x_1 \cdot x_2 \cdots x_n$ . Although it is stated in [6] that  $S_n$  itself is not periodicity forcing, it can be verified using results from [6] and [1] that the augmented set  $S_n' := S_n \cup \{x_1 \cdot x_1 \cdot x_2 \cdot x_2 \cdots x_n \cdot x_n\}$  is. Given a test set  $T_n$  for  $S_n$ , a test set for  $S_n'$  is clearly  $T_n \cup \{x_1 \cdot x_1 \cdot x_2 \cdot x_2 \cdots x_n \cdot x_n\}$ . Thus there exists a test set for  $S_n'$  containing at most  $5n$  words of length  $n$  and one word of length  $2n$ . The periodicity forcing word resulting from concatenating these words is at most  $5n^2 + 2n$  letters long.

**Proposition 7.** *Let  $\alpha_n$  be a shortest pattern not in DPCP such that  $|\text{var}(\alpha)| = n$ . Then  $n^2 \leq |\alpha| \leq 5n^2 + 2n$ .*

The above bounds not only demonstrate the growth of periodicity forcing words with respect to alphabet size, but also provide an indication of how restrictive the set  $\text{DPCP}^\neg$  is. Furthermore, the lower bound is particularly useful when considering the following.

**Definition 8.** *Let  $\alpha \notin \text{DPCP}$  be a pattern with  $|\text{var}(\alpha)| \geq 2$ . Then  $\alpha$  is said to be a prime element of  $\text{DPCP}^\neg$  (or simply prime) if for every pattern  $\beta \notin \text{DPCP}$  with  $|\text{var}(\beta)| > 1$ , and every morphism  $\varphi : \text{var}(\beta)^* \rightarrow \text{var}(\alpha)^*$ ,  $\varphi(\beta) = \alpha$  implies  $\varphi$  is a renaming morphism.*

Showing that a pattern satisfies Definition 8 is a highly non-trivial task, since all morphisms must be accounted for with respect to every pattern  $\beta \notin \text{DPCP}$ . However, due to Proposition 5, it is possible to provide an example. Specifically, it is possible to conclude that  $1 \cdot 2 \cdot 1 \cdot 1 \cdot 2$  is a prime element of  $\text{DPCP}^\neg$ , since any pre-image  $\beta$  must contain a variable  $x$  such that  $|\beta|_x \leq 2$ . By Proposition 5, this excludes the possibility that  $|\text{var}(\beta)| \geq 3$ , and reduces the candidates for  $\beta$  to a finite number of patterns which may be checked individually with little effort. This demonstrates that it is possible to produce chains of periodicity forcing words which terminate in exactly one direction (i.e., they are not bi-infinite).

**Proposition 9.** *Prime elements of  $\text{DPCP}^\neg$  exist.*

It is possible to generalise the reasoning behind Proposition 9, and show that each periodicity forcing word is either prime, or may be obtained from a prime periodicity forcing word using morphisms. This results in a structure comprised of one-sided infinite chains which spans exactly the set  $\text{DPCP}^\neg$ .

**Theorem 10.** *Let  $S$  be the set of all prime elements of  $\text{DPCP}^\neg$ . Let  $\alpha \notin \text{DPCP}$  with  $|\text{var}(\alpha)| \geq 2$ . Then either  $\alpha \in S$ , or there exists  $\beta \in S$  and a non-trivial morphism  $\varphi$  such that  $\varphi(\beta) = \alpha$ .*

Thus, there exists a non-trivial subset of  $\text{DPCP}^\neg$  whose elements, when combined with the morphisms characterised in [3], generate the set  $\text{DPCP}^\neg$ . Moreover, it is not difficult to see that the conditions for satisfying Definition 8 are very restrictive, and therefore one can expect such a subset to be much smaller than the original set.

## 4 Patterns in $\text{DPCP}^\neg$ with Arbitrary Factors

One particular consequence of the research on periodicity forcing words in [3] is that there exist periodicity forcing sets which include any given pattern  $\alpha$  – it is sufficient to simply include a pattern  $\beta \notin \text{DPCP}$  where  $\text{var}(\beta) = \text{var}(\alpha)$ . By constructing these sets such that they satisfy the conditions for Proposition 3, it is possible to provide a morphism  $\varphi$  which satisfies Lemma 2 such that  $\varphi(\alpha)$  contains an arbitrary given factor  $\beta$  for some  $\alpha \notin \text{DPCP}$ . Thus a level of generality previously not achieved is reached: that there exist periodicity forcing words with arbitrary factors. It is worth noting that due to the properties of morphisms, the construction may be altered with little effort to guarantee that  $\beta$  appears as a prefix or suffix.

Proposition 3 is addressed in the following proposition, which demonstrates that the conditions may always be satisfied. The task is somewhat simplified by using patterns with the same Parikh vector, since any morphism  $\sigma$  either maps all, or none of them to the empty word. The result is also relevant to Theorem 19, confirming that such a construction always exists.



**Proposition 11.** *Let  $\alpha_0$  be a pattern, and let  $n := \lceil \log_2(|\text{var}(\alpha_0)|) \rceil$ . There exist patterns  $\alpha_1, \alpha_2, \dots, \alpha_n$  with  $P(\alpha_0) = P(\alpha_1) = \dots = P(\alpha_n)$  such that for any  $k_0, k_1, \dots, k_n \in \mathbb{N}$ , the system of word equations*

$$\alpha_0^{k_0} = \alpha_1^{k_1} = \dots = \alpha_n^{k_n}$$

*has only periodic solutions.*

It is now possible to show that there exists a pattern not in DPCP which has an arbitrary pattern  $\beta$  as a factor. This is achieved as follows. Let  $\beta_1$  be a pattern not in DPCP such that  $\text{var}(\beta_1) = \text{var}(\beta)$ . In accordance with Proposition 11, construct the patterns  $\beta_2, \dots, \beta_n$ , and consider the morphism  $\varphi : \{1, 2, \dots, n+1\}^* \rightarrow \text{var}(\beta)^*$  given by  $\varphi(i) := \beta_i$  for  $1 \leq i \leq n$ , and  $\varphi(n+1) = \beta$ . Since  $\beta_1 \notin \text{DPCP}$ , and  $\text{var}(\beta_1) = \text{var}(\beta_i) = \text{var}(\beta)$  for  $1 \leq i \leq n$ , the set  $\{\varphi(x) \mid 1 \leq x \leq n+1\}$  is periodicity forcing, so by Proposition 4,  $\varphi$  satisfies Condition (ii) of Lemma 2. By construction,  $\varphi$  also satisfies Condition (i). Let  $\alpha \notin \text{DPCP}$  be a pattern such that  $\text{var}(\alpha) = \{1, 2, \dots, n+1\}$ . By Lemma 2,  $\varphi(\alpha) \notin \text{DPCP}$ . It is clear that  $\beta$  appears as a factor of  $\varphi(\alpha)$ . It is therefore possible to formulate the following theorem.

**Theorem 12.** *For any pattern  $\beta \in \mathbb{N}^+$ , there exists a pattern  $\alpha \notin \text{DPCP}$  such that  $\beta$  is a factor of  $\alpha$ .*

Example 13 demonstrates how such a morphism may be constructed. Note that the patterns  $\beta_2, \dots, \beta_n$  are constructed around  $\beta$ , rather than  $\beta_1$ . This is simply to keep example more compact, and it is not difficult to see why the correctness is unaffected. In a similar way, the patterns can be ‘swapped’ around to guarantee that  $\beta$  appears as a prefix or suffix of  $\varphi(\alpha)$ .

*Example 13.* Let  $\beta := 1 \cdot 1 \cdot 2 \cdot 3$ , let  $\beta_2 := 2 \cdot 3 \cdot 1 \cdot 1$ , and let  $\beta_3 := 3 \cdot 1 \cdot 1 \cdot 2$ . Let  $\beta_1 := 1 \cdot 2 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 3 \cdot 1 \cdot 1 \cdot 3 \cdot 2 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 3 \cdot 1 \cdot 1 \cdot 3 \cdot 2 \cdot 1 \cdot 1 \cdot 2 \cdot 1$ . By [3] (Proposition 32),  $\beta_1 \notin \text{DPCP}$ . Thus, by Proposition 4, the morphism  $\varphi : \{1, 2, 3, 4\}^* \rightarrow \{1, 2, 3\}^*$  given by  $\varphi(i) := \beta_i$  for  $1 \leq i \leq 3$  and  $\varphi(4) := \beta$  satisfies Condition (ii) of Lemma 2.

Condition (i) is now considered. Let  $\sigma : \{1, 2, 3\}^* \rightarrow \{\mathbf{a}, \mathbf{b}\}^*$  be a non-periodic morphism. Note that, since  $\text{var}(\beta) = \text{var}(\beta_1) = \text{var}(\beta_2) = \text{var}(\beta_3)$ ,  $\sigma(\gamma) \neq \varepsilon$  for every  $\gamma \in \{\beta, \beta_1, \beta_2, \beta_3\}$ . Let  $k_1, k_2, k_3, k_4 \in \mathbb{N}$  and consider the equation

$$\sigma(\beta_1)^{k_1} = \sigma(\beta_2)^{k_2} = \sigma(\beta_3)^{k_3} = \sigma(\beta)^{k_4}. \quad (2)$$

Clearly, this is only satisfied if

$$\sigma(2 \cdot 3 \cdot 1 \cdot 1)^{k_2} = \sigma(3 \cdot 1 \cdot 1 \cdot 2)^{k_3} = \sigma(1 \cdot 1 \cdot 2 \cdot 3)^{k_4},$$

and therefore

$$\sigma(1 \cdot 1 \cdot 2 \cdot 3) = \sigma(2 \cdot 3 \cdot 1 \cdot 1) = \sigma(3 \cdot 1 \cdot 1 \cdot 2).$$

Assume that (2), and therefore the subsequent systems of equations, are satisfied. This implies that

$$\begin{aligned}\sigma(311)\sigma(2) &= \sigma(2)\sigma(311) \\ &= \sigma(11)\sigma(23) = \sigma(23)\sigma(11) \\ &= \sigma(112)\sigma(3) = \sigma(3)\sigma(112)\end{aligned}$$

and therefore by Lemma 1  $\sigma(1)$ ,  $\sigma(2)$ ,  $\sigma(3)$  share a primitive root. Thus  $\sigma$  is periodic. This is a contradiction; there does not exist a non-periodic morphism  $\sigma$  such that (2) is satisfied. By Proposition 3,  $\varphi$  therefore satisfies Condition (i) of Lemma 2. Thus, for any pattern  $\alpha \notin \text{DPCP}$  with  $\text{var}(\alpha) = \{1, 2, 3, 4\}$ ,  $\varphi(\alpha) \notin \text{DPCP}$ , and  $\beta$  is a factor of  $\varphi(\alpha)$ .

## 5 An Alternative Means of Finding Patterns not in DPCP

While Theorem 10 provides motivation for the further study of morphisms in the context of  $\text{DPCP}^\neg$ , it also demonstrates the need to identify periodicity forcing words by other means. In [1], Culik II and Karhumäki show that this may be done using periodicity forcing sets. Indeed, patterns not in DPCP are essentially periodicity forcing sets with a cardinality of 1. However, it is generally easier to construct periodicity forcing sets with higher cardinalities, as more patterns results in a more restricted class of (pairs of) morphisms which agree on every pattern. This is precisely the reason why the morphisms approach is useful (see Proposition 4).

It follows from the properties of morphisms that the agreement of two morphisms on a ratio-imprimitive pattern can be reduced to the agreement of those morphisms on a set of two (or more) smaller patterns. The following lemma establishes this relationship formally, providing a characterisation of when a ratio-imprimitive pattern is in DPCP.

**Lemma 14.** *Let  $\alpha = \beta_1 \cdot \beta_2 \cdot \dots \cdot \beta_n$  be a pattern such that  $\beta_1, \beta_2, \dots, \beta_n$  share a basic Parikh vector. Then  $\alpha \notin \text{DPCP}$  if and only if  $\{\beta_1, \beta_2, \dots, \beta_n\}$  is a periodicity forcing set.*

For patterns with a higher element of ratio-imprimitivity (i.e., those which have many different prefixes with the same basic Parikh vector as the whole pattern), larger values of  $n$  can be taken. This results in a larger potential simplification gained by applying Lemma 14. While this does restrict the range of patterns to which this approach may be applied, it is worth noting that any concatenation of all the patterns  $\beta_1, \beta_2, \dots, \beta_n$  is also not in DPCP. This means that relatively rich classes of patterns can be established with any single set of factors. Expressing the same result using morphisms demonstrates more clearly this trade-off. The following proposition gives a criterion for a morphism  $\varphi : \Delta_1^* \rightarrow \Delta_2^*$  which maps any pattern  $\alpha$  with  $\text{var}(\alpha) = \Delta_1$  to a pattern not in DPCP.

**Proposition 15.** *Let  $\Delta_1, \Delta_2$  be sets of variables, and let  $\varphi : \Delta_1^* \rightarrow \Delta_2^*$  be a morphism. For every  $i \in \Delta_1$ , let  $\beta_i := \varphi(i)$ . If  $\{\beta_i \mid i \in \Delta_1\}$  is a periodicity forcing set, and  $\beta_1, \beta_2, \dots, \beta_n$  share the same basic Parikh vector, then  $\varphi(\alpha) \notin \text{DPCP}$  for any pattern  $\alpha$  satisfying  $\text{var}(\alpha) = \Delta_1$ .*

While the set of patterns to which morphisms satisfying Proposition 15 can be applied is much larger than for morphisms satisfying Lemma 2, the images are more restricted. The result is a contrasting class of examples of patterns not in DPCP. The characterisation given in Lemma 14 shows that periodicity forcing sets of patterns with the same basic Parikh vectors are very closely related to sets of ratio-imprimitive patterns not in DPCP. Indeed, every ratio-imprimitive pattern not in DPCP can be decomposed into a unique periodicity forcing set of ratio-primitive patterns with the same basic Parikh vectors, and for every such set, there exists a unique corresponding set of ratio-imprimitive patterns not in DPCP, obtained by concatenating every pattern in the set at least once.

It is therefore appropriate to simply investigate periodicity forcing sets of ratio-primitive patterns with equal basic Parikh vectors, since such sets automatically yield sets of patterns not in DPCP. While it is not difficult to construct periodicity forcing sets for any set of variables, generating sets of patterns with equal basic Parikh vectors present more of a challenge. Similarly to the morphisms approach studied in [3] and Sect. 3, the following techniques produce periodicity forcing words by building on the existing knowledge in the two variable case. Strong sufficient conditions are known for a set of patterns over two variables to be periodicity forcing (see Holub [5]), so they are generally not difficult to produce. Lemma 16 provides a conveniently concise example to use as a starting point.

**Lemma 16 (Culik II, Karhumäki [1]).** *The set  $\{1 \cdot 2, 1 \cdot 1 \cdot 2 \cdot 2\}$  is periodicity forcing.*

The advantage of starting with a smaller periodicity forcing set is that strict conditions can already be imposed on factors of the larger patterns. It is not difficult to see that for any periodicity forcing set  $\Pi := \{\beta_1, \beta_2, \dots, \beta_n\}$ , and any morphism  $\varphi : (\text{var}(\beta_1) \cup \dots \cup \text{var}(\beta_n))^* \rightarrow \mathbb{N}^*$ , the set  $\varphi(\Pi) := \{\varphi(\beta_1), \varphi(\beta_2), \dots, \varphi(\beta_n)\}$  is periodicity forcing with respect to each factor  $\varphi(x)$ , where  $x \in \text{var}(\beta_1) \cup \text{var}(\beta_2) \cup \dots \cup \text{var}(\beta_n)$ . Specifically, for each pair of morphisms  $\sigma, \tau$  which agree on  $\varphi(\Pi)$ , at least one of the following cases needs to be satisfied:

- (i) There exists a primitive word  $w$  such that, for every  $x \in \text{var}(\beta_1) \cup \text{var}(\beta_2) \cup \dots \cup \text{var}(\beta_n)$ ,  $\sigma(\varphi(x)) \in \{w\}^*$  and  $\tau(\varphi(x)) \in \{w\}^*$ .
- (ii) For every  $x \in \text{var}(\beta_1) \cup \text{var}(\beta_2) \cup \dots \cup \text{var}(\beta_n)$ ,  $\sigma(\varphi(x)) = \tau(\varphi(x))$ .

This can be verified by contradiction: assuming that neither condition holds, the morphisms  $\sigma \circ \varphi$  and  $\tau \circ \varphi$  are evidence that  $\{\beta_1, \beta_2, \dots, \beta_n\}$  is not a periodicity forcing set. It can therefore be more efficient to generate new periodicity forcing sets from existing ones, by substituting individual variables for patterns as this considerably restricts the morphisms  $\sigma, \tau$  which need to be accounted for.

The first case is, generally speaking, the more difficult – and is addressed in the following two lemmas, which provide a tool for exploiting the ‘partial’ periodicity of two morphisms, and extending it to guarantee their total periodicity. This is achieved by introducing patterns which are formed by ‘splitting’ a pattern which has a periodicity constraint on it.

**Lemma 17.** *Let  $w, u, v$  be words, and let  $k_1, k_2, k_3, k_4 \in \mathbb{N}_0$  with  $k_2 \geq 1$ . If*

$$w^{k_1} \cdot u \cdot w^{k_2} \cdot v \cdot w^{k_3} = w^{k_4}$$

*then  $u, v$ , and  $w$  commute.*

**Lemma 18.** *Let  $w$  be a primitive word, and let  $u, u', v, v'$  be non-empty words such that  $u \cdot v = u' \cdot v' = w$ . Then for any  $k_1, k_2, k_3, k_4 \in \mathbb{N}_0$  and any  $q_1, q_2 \in \mathbb{N}$ , the equation*

$$w^{k_1} \cdot u \cdot w^{q_1} \cdot v \cdot w^{k_2} = w^{k_3} \cdot u' \cdot w^{q_2} \cdot v' \cdot w^{k_4} \quad (3)$$

*only has solutions if  $k_1 = k_3, k_2 = k_4, q_1 = q_2, u = u'$  and  $v = v'$ .*

It is now easier to formulate methods for generating larger periodicity sets from smaller ones, allowing for the preservation of the property of having patterns with the same basic Parikh vector. The following method relies on ‘splitting’ one variable  $y$  into two (so each occurrence of  $y$  becomes, e.g.,  $y_1 y_2$ ) in each pattern. New patterns are then introduced to ‘force’ the periodicity of  $y_1$  and  $y_2$ . Although the theorem appears very technical, it is relatively simple to apply, as example 20 shows.

**Theorem 19.** *Let  $\Delta := \{x_1, x_2, \dots, x_n\}$  be a set of variables, and let  $y \notin \Delta$  be a variable. Let  $\Pi := \{\alpha_1, \alpha_2, \dots, \alpha_m\}$  be a periodicity forcing set such that  $\bigcup_{i=1}^m \text{var}(\alpha_m) = \Delta$ . Let  $\varphi : \Delta^* \rightarrow (\Delta \cup \{y\})^*$  be the morphism given by  $\varphi(x_n) := x_n \cdot y$  and  $\varphi(x_i) := x_i$  for  $1 \leq i < n$ . Let  $t \in \mathbb{N}$ , and for  $1 \leq i \leq t$ , let  $\beta_i := x_n \cdot \gamma_i \cdot y$  for some pattern  $\gamma_i$ . Let  $\beta_{t+1} = x_1 \cdot x_1 \cdot x_2 \cdot x_2 \cdots x_n \cdot x_n \cdot y \cdot y$ . If*

- (i)  $\gamma_1, \gamma_2, \dots, \gamma_t$  are patterns such that  $\text{var}(\gamma_1) = \text{var}(\gamma_2) = \dots = \text{var}(\gamma_t) = \Delta \setminus \{x_n\}$ ,
- (ii) for any  $k_1, k_2, \dots, k_q \in \mathbb{N}$ , the series of word equations  $\gamma_1^{k_1} = \gamma_2^{k_2} = \dots = \gamma_t^{k_t}$  has only periodic solutions,

*then the set  $\{\varphi(\alpha_1), \varphi(\alpha_2), \dots, \varphi(\alpha_m), \beta_1, \beta_2, \dots, \beta_{t+1}\}$  is periodicity forcing.*

*Example 20.* Let  $\Pi := \{1 \cdot 2, 1 \cdot 1 \cdot 2 \cdot 2\}$ . It is established in Lemma 16 that  $\Pi$  is a periodicity forcing set. Let  $\varphi : \{1, 2\}^* \rightarrow \{1, 2, 3\}^*$  be the morphism given by  $\varphi(1) := 1$  and  $\varphi(2) := 2 \cdot 3$ . Consider the set  $\Pi' := \{\varphi(1 \cdot 2), \varphi(1 \cdot 1 \cdot 2 \cdot 2), \beta_1, \beta_2\}$  where  $\beta_1 := 2 \cdot 1 \cdot 3$  and  $\beta_2 := 1 \cdot 1 \cdot 2 \cdot 2 \cdot 3 \cdot 3$ . It follows from the fact that  $\Pi$  is a periodicity forcing set that, for any two morphisms  $\sigma, \tau : \{1, 2, 3\}^* \rightarrow \{\mathbf{a}, \mathbf{b}\}^*$ , if  $\sigma$  and  $\tau$  agree on every pattern in  $\Pi'$ , then

- (1)  $\sigma(1), \tau(1), \sigma(2 \cdot 3), \tau(2 \cdot 3)$  commute, or
- (2)  $\sigma(1) = \tau(1)$  and  $\sigma(2 \cdot 3) = \tau(2 \cdot 3)$ .

Assume the first case is true. It follows from Lemmas 17 and 18 that if  $\sigma$  and  $\tau$  agree on  $2 \cdot 1 \cdot 3$ , they must be periodic. Assume that the second case is true. Then if  $\sigma$  and  $\tau$  agree on  $1 \cdot 1 \cdot 2 \cdot 2 \cdot 3 \cdot 3$ , they must agree on  $2 \cdot 2 \cdot 3 \cdot 3$ . They also agree on  $2 \cdot 3$ , and  $\{2 \cdot 3, 2 \cdot 2 \cdot 3 \cdot 3\}$  is a periodicity forcing set. Thus if  $\sigma$  and  $\tau$  are distinct, they must be periodic over  $\{2, 3\}$ . Furthermore, if they agree on  $2 \cdot 1 \cdot 3$ , then since  $\sigma(1) = \tau(1) = u$  for some word  $u \in \{\mathbf{a}, \mathbf{b}\}^*$ ,

$$\sigma(2) \cdot u \cdot \sigma(3) = \tau(2) \cdot u \cdot \tau(3).$$

If  $\sigma$  and  $\tau$  are periodic over  $\{2, 3\}$ , there exist  $k_1, k_2, k_3, k_4 \in \mathbb{N}_0$  and a primitive word  $w \in \{\mathbf{a}, \mathbf{b}\}^*$  such that  $\sigma(2) = w^{k_1}$ ,  $\sigma(3) = w^{k_2}$ ,  $\tau(2) = w^{k_3}$  and  $\tau(3) = w^{k_4}$ . Thus

$$w^{k_1} \cdot u \cdot w^{k_2} = w^{k_3} \cdot u \cdot w^{k_4}$$

which is a non-trivial equation in two unknowns unless  $k_1 = k_3$  and  $k_2 = k_4$ , in which case  $\sigma$  and  $\tau$  are not distinct. Therefore by Lemma 1,  $u \in \{w\}^*$ . Consequently, if two distinct morphisms agree on every pattern in  $\Pi'$ , they are periodic, so  $\Pi'$  is a periodicity forcing set.

An alternative to splitting variables in the patterns of a periodicity forcing set is to generate a set of patterns obtained by inserting a new variable repeatedly into occurrences of a single pattern not in DPCP. It is relatively simple to establish a set of patterns with the same basic Parikh vectors in this way. The next results demonstrate how it can be shown that such a set is also periodicity forcing. The following definition is given to provide a notation for inserting a new variable  $x$  at a specified place in a pattern  $\alpha$ .

**Definition 21.** *Let  $\alpha$  be a pattern and let  $x \in \text{var}(\alpha)$  be a variable. Let  $\text{pre}_x(\alpha)$  be the prefix of  $\alpha$  up to, and including the first occurrence of  $x$ . Let  $\text{suf}_x(\alpha)$  be the suffix of  $\alpha$  starting after (not including) the first occurrence of  $x$ .*

Note that  $\text{pre}_x(\alpha) \cdot \text{suf}_x(\alpha) = \alpha$ , so the pattern  $\text{pre}_x(\alpha) \cdot y \cdot \text{suf}_x(\alpha)$  is the pattern obtained by inserting the variable  $y$  into the pattern  $\alpha$  directly after the first occurrence of  $x$ . Again, knowledge of existing periodicity forcing sets is used to impose the required conditions. For clarity, a specific example (from Lemma 16) is used; however, any periodicity forcing set of patterns with two variables would be suitable.

**Theorem 22.** *Let  $\alpha \notin \text{DPCP}$  and let  $x \notin \text{var}(\alpha)$ . Then the set  $\Pi := \{x \cdot \alpha, x \cdot x \cdot \alpha \cdot \alpha\} \cup \{\text{pre}_y(\alpha) \cdot x \cdot \text{suf}_y(\alpha) \mid y \in \text{var}(\alpha)\}$  is periodicity forcing.*

It is clear that the patterns generated in the style of Theorem 22 have the pattern  $\alpha^k$  as a sub-pattern, where  $k := |\text{var}(\alpha)| + 3$ . Thus there exists a non-trivial morphism  $\varphi$  and a pattern  $\beta \notin \text{DPCP}$  such that  $\varphi(\beta) = \alpha^k$ .

**Proposition 23.** *Let  $\alpha = \beta^k$  for some pattern  $\beta$  and number  $k \geq |\text{var}(\alpha)| + 3$ . Then  $\alpha$  is not a prime element of  $\text{DPCP}^\square$ .*

This is an interesting result since the properties associated with the Dual PCP are, due to the nature of morphisms, generally consistent for powers of the same word. It can also be interpreted that, as a result, the majority of periodicity forcing words are not prime.

## 6 Conclusion

Section 3 introduces a prime subset of  $\text{DPCP}^\neg$ , allowing the set to be described as chains of morphic images. It is shown that this subset is non-empty, and thus that  $\text{DPCP}^\neg$  can be exactly generated by the set of prime periodicity forcing words. In Section 4, a construction is given for periodicity forcing words containing any given factor/prefix/suffix. This not only produces a rich class of new examples, but demonstrates a previously unknown level of generality within the seemingly very restrictive set. Motivated by the study of the prime periodicity forcing words introduced earlier, Section 5 examines alternative methods for generating periodicity forcing words. The results give examples of periodicity forcing words which contrast those known so far, and provide further insights into the prime words considered earlier in the paper. As a by-product of results from this paper and existing literature, tight bounds on the length of the shortest periodicity forcing word over a given alphabet can be given.

**Acknowledgements.** The authors wish to thank the anonymous referees for their helpful remarks and suggestions which have provided a useful additional reference and a construction which has produced a stronger form of Proposition 7.

## References

1. K. Culik II and J. Karhumäki. On the equality sets for homomorphisms on free monoids with two generators. *Theoretical Informatics and Applications (RAIRO)*, 14:349–369, 1980.
2. E. Czeizler, S. Holub, J. Karhumäki, and M. Laine. Intricacies of simple word equations: An example. *International Journal of Foundations of Computer Science*, 18:1167–1175, 2007.
3. J.D. Day, D. Reidenbach, and J.C. Schneider. On the Dual Post Correspondence problem. In *Proc. 17th International Conference on Developments in Language Theory, DLT 2013*. To appear.
4. J. Hadravova and S. Holub. Large simple binary equality words. *International Journal of Foundations of Computer Science*, 23:1385–1403, 2012.
5. S. Holub. Binary equality sets are generated by two words. *Journal of Algebra*, 259:1 – 42, 2003.
6. S. Holub and J. Kortelainen. Linear size test sets for certain commutative languages. *Theoretical Informatics and Applications (RAIRO)*, 35:453–475, 2001.
7. J. Karhumäki and E. Petre. On some special equations on words. In *Technical Report 584, Turku Centre for Computer Science, TUCS*, 2003.
8. M. Lothaire. *Combinatorics on Words*. Addison-Wesley, Reading, MA, 1983.
9. G.S. Makanin. The problem of solvability of equations in a free semi-group. *Soviet Mathematics Doklady*, 18:330–334, 1977.
10. E.L. Post. A variant of a recursively unsolvable problem. *Bulletin of the American Mathematical Society*, 52:264–268, 1946.