ARTICLE IN PRESS
Discrete Applied Mathematics xxx (xxxx) xxx

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Generating all minimal petri net unsolvable binary words

Evgeny Erofeev [a,*,1], Kamila Barylska [b,2], Łukasz Mikulski [b,2], Marcin Piątkowski [b,2]

[a] Parallel Systems, Department of Computing Science, Carl von Ossietzky Universität, D-26111 Oldenburg, Germany
[b] Faculty of Mathematics and Computer Science, Nicolaus Copernicus University, Chopina 12/18, 87-100 Toruń, Poland

ARTICLE INFO

Article history:
Received 28 February 2017
Received in revised form 1 April 2019
Accepted 19 April 2019
Available online xxxx

Keywords:
Binary word
Labelled transition system
Petri net
Synthesis

ABSTRACT

Sets of finite words, as well as some infinite ones, can be described using finite systems, e.g. automata. On the other hand, some automata may be constructed with the use of even more compact systems, like Petri nets. We call such automata Petri net solvable. In this paper we consider the solvability of singleton languages over a binary alphabet (i.e. binary words). An unsolvable (i.e. not solvable) word w is called minimal if each proper factor of w is solvable. We present a complete language-theoretical characterisation of the set of all minimal unsolvable binary words. The characterisation utilises morphic-based transformations which expose the combinatorial structure of those words, and allows to introduce a pattern matching condition for unsolvability.

1. Introduction

To deal with infinite sets of words we need to specify them in a finite way. Finite automata which are known as a classical model for describing regular languages, are equivalent to finite labelled transition systems [8]. Some sets may be expressed with the use of even more compact system models.

In this paper we investigate the synthesis problem with a specification given in the form of labelled transition systems. The sought system model is a place/transition Petri net [11], with its reachability graph as a natural bridge between specification and implementation. Namely, we are concerned with finding a net, whose reachability graph is isomorphic to a given labelled transition system.

To address this issue one may use the general approach for net-synthesis suggested by the theory of regions [1]. For a given labelled transition system, the solution of a number of linear inequations systems provided by the theory of regions exists if and only if there exists an implementation in the form of a net. Moreover, solutions of such linear inequations systems are usually utilised during the synthesis of the resulting system (see Synet [4] and APT [12]).

Our aim is to initiate a combinatorial approach and to provide a complete characterisation of a generative nature for a special kind of labelled transition systems — non-branching and acyclic transition systems having at most two labels (i.e. binary words) [2]. More precisely, we characterise all minimal unsolvable binary words.

The paper is organised as follows. First we give some basic notions and notations concerning labelled transition systems, Petri nets and theory of regions. After that we present a necessary condition for minimal unsolvability, which

* Corresponding author.
E-mail addresses: evgeny.erofeev@informatik.uni-oldenburg.de (E. Erofeev), kamila.barylska@mat.umk.pl (K. Barylska), lukasz.mikulski@mat.umk.pl (Łu. Mikulski), marcin.piatkowski@mat.umk.pl (M. Piątkowski).

1 This work is supported by DFG (German Research Foundation) through Graduiertenkolleg GRK-1765 SCARE (System Correctness under Adverse Conditions) and grant Be 1267/16-1 ASYST (Algorithms for Synthesis and Pre-Synthesis Based on Petri Net Structure Theory).
2 This work is supported by the Polish National Science Center under the Grant No. 2013/09/D/ST6/03928.

https://doi.org/10.1016/j.dam.2019.04.023

allows to formulate possible shapes of minimal unsolvable words in the form of extended regular expressions [5]. In Section 4 we introduce the notion of (base) extendable and non-extendable binary words. In the following sections we provide the main results of this paper: a generic characterisation of all minimal unsolvable binary words and its utilisation for an efficient verifying procedure. We conclude the paper with a discussion of experimental results and a short section containing some directions for further research.

This paper is an extended and revised version of [9] presented at Prague Stringology Conference 2016.

2. Basic notions

In this section we introduce notions used throughout the paper.

Words

A *word over an alphabet* T is a finite sequence $w \in T^*$, and it is *binary* if $|T| = 2$. For a word w and a letter $t \in T$, $\#_t(w)$ denotes the number of occurrences of t in w. A word $w' \in T^*$ is called a *subword* (or a *factor*) of $w \in T^*$ if $\exists u_1, u_2 \in T^*: w = u_1 w' u_2$. In particular, w' is called a *prefix* of w if $u_1 = \varepsilon$, a *suffix* of w if $u_2 = \varepsilon$, and an *infix* of w if $u_1 \neq \varepsilon$ and $u_2 \neq \varepsilon$.

For two alphabets Σ_1 and Σ_2, a mapping $\phi : \Sigma_1^* \to \Sigma_2^*$ is called a *morphism* if we have $\phi(u \cdot v) = \phi(u) \cdot \phi(v)$ for every $u, v \in \Sigma_1^*$. A morphism ϕ is uniquely determined by its values on the alphabet. Moreover, ϕ maps the neutral element of Σ_1^* into the neutral element of Σ_2^*.

Transition systems

A *finite labelled transition system* (or simply *lts*) with initial state is a tuple $TS = (S, \to, T, s_0)$ with nodes S (a finite set of states), edge labels T (a finite set of letters), edges $\to \subseteq (S \times T \times S)$, and an initial state $s_0 \in S$. A label t is enabled at $s \in S$, denoted by $s[t\rangle$, if $\exists s' \in S: (s, t, s') \in \to$. A state s' is reachable from s through the execution of $\sigma \in T^*$, denoted by $s[\sigma\rangle s'$, if there is a directed path from s to s' whose edges are labelled consecutively by letters of σ. The set of states reachable from s is denoted by $[s\rangle$. A sequence $\sigma \in T^*$ is allowed (or firable) from a state s, denoted by $s[\sigma\rangle$, if there is a state s' such that $s[\sigma\rangle s'$.[3] Two labelled transition systems $TS_1 = (S_1, \to_1, T, s_{01})$ and $TS_2 = (S_2, \to_2, T, s_{02})$ are isomorphic if there exists a bijection $\zeta: S_1 \to S_2$ with $\zeta(s_{01}) = s_{02}$ and $(s, t, s') \in \to_1 \iff (\zeta(s), t, \zeta(s')) \in \to_2$, for all $s, s' \in S_1$.

A word $w = t_1 t_2 \ldots t_n$ of length $n \in \mathbb{N}$ uniquely corresponds to a finite transition system

$$TS(w) = (\{s_0, \ldots, s_n\}, \{(s_{i-1}, t_i, s_i) \mid 0 < i \leq n \wedge t_i \in T\}, T, s_0).$$

Petri nets

An *initially marked Petri net* is denoted as $N = (P, T, F, M_0)$ where P is a finite set of places, T is a finite set of transitions, F is the flow function $F: ((P \times T) \cup (T \times P)) \to \mathbb{N}$ specifying the arc weights, and M_0 is the initial marking (where a marking is a mapping $M: P \to \mathbb{N}$, indicating the number of tokens in each place). A *side-place* is a place p with $p^\bullet \cap {}^\bullet p \neq \emptyset$, where $p^\bullet = \{t \in T \mid F(p, t) > 0\}$ and ${}^\bullet p = \{t \in T \mid F(t, p) > 0\}$. N is *pure* or *side-place free* if it has no side-places. A transition $t \in T$ is enabled at a marking M, denoted by $M[t\rangle$, if $\forall p \in P: M(p) \geq F(p, t)$. The firing of t at marking M leads to M', denoted by $M[t\rangle M'$, if $M[t\rangle$ and $M'(p) = M(p) - F(p, t) + F(t, p)$. This can be extended, as usual, to $M[\sigma\rangle M'$ for sequences $\sigma \in T^*$, and $[M\rangle$ denotes the set of markings reachable from M. The reachability graph $RG(N)$ of a bounded (such that the number of tokens in each place does not exceed a certain finite number) Petri net N is the labelled transition system with the set of vertices $[M_0\rangle$, initial state M_0, label set T, and set of edges $\{(M, t, M') \mid M, M' \in [M_0\rangle \wedge M[t\rangle M'\}$. If a labelled transition system TS is isomorphic to the reachability graph of a Petri net N, we say that N *PN-solves* (or simply *solves*) TS, and that TS is *synthesisable* to N. We say that N solves a word w if it solves $TS(w)$. A word w is then called *solvable*, otherwise it is called *unsolvable*.

Solvability

Region theory constitutes the most common tool for proving solvability of labelled transition systems. Let $(S, \to, T, s_0)$ be an lts and $N = (P, T, F, M_0)$ be a Petri net, which we hope to synthesise. The synthesis comprises solving systems of linear inequalities in integer numbers. Those inequalities guaranty satisfiability of the following properties:

State separation property (*ssp* in short)

> For every pair $s, s' \in S$ of distinct states ($s \neq s'$) there exists a place $p \in P$ such that $M(p) \neq M'(p)$ for markings M and M' corresponding to s and s'.

Event/state separation property (*essp* in short)

> For every state–transition pair $s \in S$ and $t \in T$ with $\neg(s[t\rangle)$ there exists a place $p \in P$ such that $M(p) < F(p, t)$ for the marking M corresponding to state s.

Note that if the lts is defined by a word w then the state separation property is easy to satisfy by introducing a counter place. On the other hand, satisfiability of event/state separation property, for every state–transition pair $s \in S$ and $t \in T$

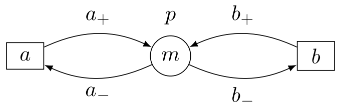


Fig. 1. A general form of a place p containing initially m tokens and preventing a transition (a or b) to satisfy essp.

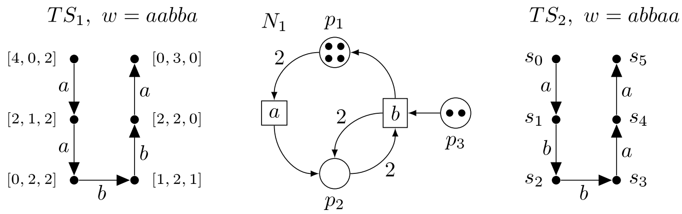


Fig. 2. N_1 solves TS_1. No solution of TS_2 exists.

with $\neg(s[t\rangle)$, requires a place preventing t at s. In the case of binary word $w \in \{a, b\}^*$ such a place $p \in P$ is of the form depicted in Fig. 1.

The labelled transition systems TS_1 and TS_2 depicted in Fig. 2 correspond to the words *aabba* and *abbaa*, respectively. The former is PN-solvable, since the reachability graph of N_1 is isomorphic to TS_1, while the latter contains an unsolvable event/state separation problem (see [2] for detailed explanation). Note that word *abbaa*, isomorphic to TS_2, is the shortest binary word (modulo swapping a/b) which is not PN-solvable. However, its reverse (*aabba*) is solvable.

Minimal unsolvable words

If w is PN-solvable, then all of its subwords w' are. To see this, let the Petri net solving w be executed up to the state before w', take this as the new initial marking, and add a pre-place with $\#_a(w')$ tokens to a and a pre-place with $\#_b(w')$ tokens to b. Thus, the unsolvability of any proper subword of w entails the unsolvability of w. For this reason, the notion of a *minimal unsolvable word* (*muw* in short) is well-defined, namely, as an unsolvable word all of whose proper subwords are solvable. A complete list of minimal unsolvable words up to length 110 can be found, amongst some other lists, in [10].

3. Structural classification of minimal unsolvable words

Throughout this section we investigate possible shapes of minimal unsolvable words in detail. In [2,3] some necessary and some sufficient properties of solvable as well as unsolvable words have already been described. In this section we shall provide known facts about minimal unsolvable words, which are true modulo swapping a and b, only in one form for the sake of succinctness.[4] From these facts we then deduce some important restrictions for the possible shapes of those words.

Proposition 1 (*[2] Sufficient Condition for Unsolvability*)**.** *If a word over $\{a, b\}$ has a subword of the form (1), then it is not PN-solvable.*

$$(\ a\ b\ \alpha\ )\ b^*\ (\ b\ a\ \alpha\ )^+\ a\ , \qquad \text{with } \alpha \in \{a, b\}^* \tag{1}$$

Remark. Let us notice that for a fixed α the language described by the expression $(ab\alpha)b^*(ba\alpha)^+a$ is regular. However in our case α is an arbitrary (but the same within the word) binary word and we consider all words of the form (1) for all possible α's. The language obtained this way is obviously not regular (nor even context-free).

In the following, $u, v \in \{a, b\}^*$. Let us consider a decomposition $w = u|_s av$. We say that b is *separable at s* if we can construct a Petri net with transitions a and b and one place p such that w can be fired completely in the net and b is not enabled at the marking corresponding to s. The lts TS in Fig. 3 corresponds to the sequence $a\ |_s\ ab$. Letter b is separable at state s since in the net N on the right of the figure, which allows the firing of aab, transition b is disabled at the corresponding marking M_s. Let us also notice that N does not solve the sequence aab, since it allows more behaviour (it allows firing of aaa, for example).

In the present paper, we rely on the main result proved in [3], which will here be used in the following form:

---

[3] For compactness, in case of long formulas we write $|_r\,\alpha\,|_s\,\beta\,|_t$ instead of $r\,[\alpha\rangle\,s\,[\beta\rangle\,t$.

[4] In the first part of this section we concentrate on words starting with a. Later we switch to words without infix aa, which are motivated by the intermediate facts.

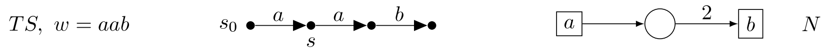


Fig. 3. b is separable at s in aab.

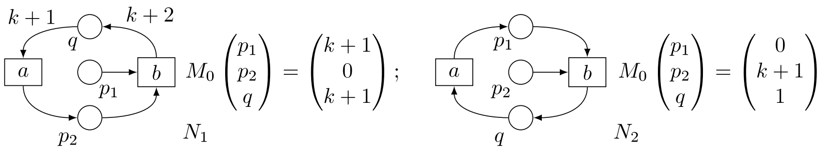


Fig. 4. N_1 solves $ab(ab)^k aa$. N_2 solves $ab(ab)^k a$.

Lemma 1 (*[3] Characterisation of Separable States*). *For a word $w \in \{a,b\}^*$ let $w = u|_s av$ be an arbitrary decomposition. Then, b is separable at s iff*

$$\forall_{\alpha,\beta,\gamma,\delta\in\{a,b\}^*}\ (w = \alpha b\beta|_s a\gamma b\delta \ \Rightarrow\ \#_b(b\beta)\cdot\#_a(a\gamma) > \#_a(b\beta)\cdot\#_b(a\gamma)).$$

In order to obtain a precise structural characterisation, we shall figure out properties of (un)solvable words. There are known conditions for (un)solvability, which allow to restrict possible shapes of minimal unsolvable words being applied step by step.

Proposition 2 (*[2] Solvability of av and vb Implies Solvability of avb*). *If both av and vb are solvable, then avb is also solvable.*

This implies that each minimal unsolvable word either starts and ends with a or starts and ends with b. Also, if a muw w starts (and ends) with a then b is always separable in w.

Lemma 2 (*Transition b is always Separable in MUW Starting with a*). *If w is a muw and starts with a, there are no violations of* essp *for b in w.*

Proof. By contraposition, assume $w = a\ldots|_s a\ldots a$, and b cannot be separated from the state s. If there is no b in w before state s, b can be separated from s with a place p having zero tokens initially, the weight of the arc from a to p is 1, and p being a side-condition for b with both arcs having weights equal to the number of occurrences of a before the first b in w. Hence, there is at least one b before s. If there is no b after state s, one can separate b from s with an input place p for b, having $\#_b(w)$ tokens initially, and the weight of the arc from p to b equal to 1. Thus, there is at least one b after s in w. As b is not separable at s, for some decomposition $w = a\alpha b\beta|_s a\gamma b\delta$, by Lemma 1, we have $\#_b(b\beta)\cdot\#_a(a\gamma) \le \#_a(b\beta)\cdot\#_b(a\gamma)$. The inequality means that the proper subword $b\beta a\gamma b$ of w is unsolvable, contradicting the minimality of w. □

From the following fact we get that minimal unsolvable word either starts with ab or with ba.

Proposition 3 (*[2] Solvable Word can be Prefixed by Starting Letter*). *If a word av is PN-solvable then aav is, too.*

Let w be a minimal unsolvable binary word starting with a. By Proposition 2 we have two possible cases: either w ends with a single a, or it has many (more than one) a's at the end. So far we know $w = abua$. Due to the following statement, we get bu either does not contain the infix aa or the infix bb.

Proposition 4 (*[2] No aa and bb Inside a Minimal Unsolvable Word*). *If a minimal non-PN-solvable word is of the form $w = a\alpha a$, then either α does not contain the factor aa or α does not contain the factor bb.*

Assume, bu has neither factors aa nor bb inside. The following two cases for a muw w are possible:

$$ab(ab)^k aa \quad \text{or} \quad ab(ab)^k a, \qquad \text{where } k \ge 0$$

Petri nets N_1 and N_2 in Fig. 4 with the corresponding initial markings solve the first and the second of these forms, respectively. From Proposition 4 and this observation we deduce that, in minimal unsolvable word $w = a\alpha a$, α has either the factor aa or the factor bb, but never both.

Thus, w has one of the following forms, where $x_i > 0$ for $1 \le i \le n$:

1. $ab^{x_1}ab^{x_2}a\ldots ab^{x_n}a$: starts and ends with a, single a at the end, no aa;
2. $aba^{x_1}ba^{x_2}b\ldots ba^{x_n}ba$: starts and ends with a, single a at the end, no bb;
3. $ab^{x_1}ab^{x_2}a\ldots ab^{x_n}aa$: starts and ends with a, many a's at the end;
4. $aba^{x_1}ba^{x_2}b\ldots ba^{x_n}a$: starts and ends with a, many a's at the end, no bb.

Please cite this article as: E. Erofeev, K. Barylska, Łu. Mikulski et al., Generating all minimal petri net unsolvable binary words, Discrete Applied Mathematics (2019), https://doi.org/10.1016/j.dam.2019.04.023.

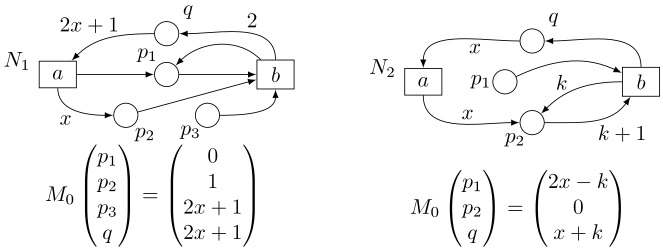


Fig. 5. N_1 solves $ab^{x+1}ab^xa$. N_2 solves $ab^{x-k}ab^xa$.

All those patterns can be comprised into the following three general forms of muw w:

$$ab^{x_1}ab^{x_2}a\ldots ab^{x_n}a \quad \text{with } x_i > 0 \text{ for } 1 \le i \le n, \quad \text{for 1;} \tag{2}$$

$$bab^{x_2}ab^{x_3}a\ldots ab^{x_n} \quad \text{with } x_i > 0 \text{ for } 2 \le i \le n, \quad \text{for swapped 2, 4;} \tag{3}$$

$$ab^{x_1}ab^{x_2}a\ldots ab^{x_n}aa \quad \text{with } x_i > 0 \text{ for } 1 \le i \le n, \quad \text{for 3.} \tag{4}$$

In the rest of this section we will try to figure out these forms more precisely.

Consider first the form (4): $w = ab^{x_1}ab^{x_2}a\ldots ab^{x_n}aa$ with $x_i > 0$ for $1 \le i \le n$. Since w necessarily has bb as a factor, $x_i \ge 2$ for some $1 \le i \le n$. If $n = 1$ then $x_1 \ge 2$. We shall prove now that if $n > 1$ then $x_1 = 2$, $x_2 = \cdots = x_n = 1$. Let $j = \max\{1 \le i \le n \mid x_i \ge 2\}$. For the subword $v = \underbrace{ab^{x_j-1}}_{\alpha}|\underbrace{ba\ldots aba}_{\beta}a$ of w, where $x_j \ge 2$ and $x_{j+1} = \cdots = x_n = 1$, we have $\#_a(\beta)\cdot\#_b(\alpha) = (n-j+1)\cdot(x_j-1) \ge 1\cdot(n-j+1) = \#_a(\alpha)\cdot\#_b(\beta)$, implying v is not solvable, due to Lemma 1. If $j > 1$, v is a proper subword of w, which contradicts minimal unsolvability of w. Hence, $x_i \le 1$ for $i > 1$. Thus, there are two possibilities for a muw w of the form (4):

$$ab^xaa, \text{ with } x > 2 \qquad \text{or} \qquad abb(ab)^kaa, \text{ with } k \ge 0 \tag{4'}$$

To understand shapes (2) and (3), the following *balancing property* will be useful.

Lemma 3 (*[3] Block Lengths Differ by at most* 1)**.** *Let $w \in a^*b^+(ab^+)^*(a|\varepsilon)$ be a word that contains both bab^xa and abb^xb with $x \ge 1$ as subwords. Then, w is not solvable.*

Let us now study pattern (2). It is easy to see that words corresponding to pattern (2) are solvable for $n = 1$. Consider the partial instance, $n = 2$, of this pattern. The words of the following two classes $ab^{x+1}ab^xa$ or $ab^{x-k}ab^xa$ with $0 \le k < x$, are solvable, and Petri nets N_1 and N_2 in Fig. 5 are possible solutions for words of the first and of the second of these forms, respectively. Thus, if $w = ab^{x_1}ab^{x_2}a$ is minimal unsolvable, then $x_1 - x_2 \ge 2$.

Lemma 4 (*[2] Side-Place-Free Solvability with Few Initial b's*)**.** *If $u = b^{x_1}ab^{x_2}a\ldots ab^{x_n}a$ is solvable and $x_1 \le \min\{x_2,\ldots,x_n\}$, then u is solvable side-place-freely.*

Lemma 5 (*[2] Solving au from u*)**.** *Suppose $u = b^{x_1}ab^{x_2}a\ldots ab^{x_n}a$ is solvable side-place-freely. Then au is solvable.*

Consider an arbitrary minimal unsolvable word $w = ab^{x_1}ab^{x_2}a\ldots ab^{x_n}a$ of the form (2) with $n \ge 3$, $x_i > 0$ for $1 \le i \le n$. Let $x = \min\{x_i \mid 2 \le i \le n\}$. Due to Lemma 3, $x_i \in \{x, x+1\}$ for $2 \le i \le n$, and then $x_1 \le x+2$. If $x_1 < x+1$, by Lemmata 4 and 5, the word w is solvable, contradicting the choice. Hence, $x+1 \le x_1 \le x+2$, and $\min\{x_i \mid 1 \le i \le n\} = x$. We now show $x_n = x$. Two cases are possible:

Case 1: $x_1 = x+2$. If $x_n = x+1$, then $x_j = x$ for some $1 < j < n$, which by Lemma 3 contradicts the minimality of w. Hence, $x_n = x$, and w follows the pattern $ab^{x+2}a(b^{x+1}a)^+b^xa$.

Case 2: $x_1 = x+1$. By contraposition, assume $x_n = x+1$. Then, $x_j = x$ for some $2 \le j \le n-1$. Let $j_1 = \max\{j \mid x_j = x\}$. Assume a cannot be separated from some state s_k in w (b is separable by Lemma 2). If $k < j_1$, then, by Lemma 1, for

$$w \;=\; \underbrace{a\,b^{x_1}\;a\ldots a\;b^{x_{k-1}}}_{\alpha}|_{s_k}\;\underbrace{\overbrace{b\;a\ldots\;b^{x_{j_1}}}^{\beta'}\;a\ldots\;b^{x_n}}_{\beta}\;a$$

we have

$$\#_a(\beta)\cdot\#_b(\alpha) \ge \#_a(\alpha)\cdot\#_b(\beta) \iff \frac{\#_b(\alpha)}{\#_a(\alpha)} \ge \frac{\#_b(\beta)}{\#_a(\beta)},$$

where $\#_a(\alpha) \neq 0$ by the form of w, and $\#_a(\beta) \neq 0$ due to $j_1 \leq n-1$. From the choice of j_1, $\#_b(\beta) \Big/ \#_a(\beta) \geq \#_b(\beta') \Big/ \#_a(\beta')$, implying

$$\frac{\#_b(\alpha)}{\#_a(\alpha)} \geq \frac{\#_b(\beta)}{\#_a(\beta)} \geq \frac{\#_b(\beta')}{\#_a(\beta')} \implies \#_a(\beta') \cdot \#_b(\alpha) \geq \#_a(\alpha) \cdot \#_b(\beta').$$

According to Lemma 1, this implies the unsolvability of the proper subword $\alpha\beta' a$ of w, which contradicts minimality of w. Assume now $k \geq j_1$. Then in

$$w \;=\; \underbrace{a\ b^{x_1}\ a\ \ldots a\ b^{x_{j_1}}\ a\ \ldots\ b^{x_k-1}}_{\alpha} |_{s_k} \underbrace{b\ a\ \ldots\ b^{x_n}}_{\beta}\ a,$$

by Lemma 1, we have

$$\#_a(\beta) \cdot \#_b(\alpha) \geq \#_a(\alpha) \cdot \#_b(\beta) \iff \frac{\#_b(\alpha)}{\#_a(\alpha)} \geq \frac{\#_b(\beta)}{\#_a(\beta)},$$

where $\#_a(\alpha) \neq 0$ by the form of w, and $\#_a(\beta) \neq 0$ because a can be separated "inside" the last group of b's with a place p having $\#_b(w) \cdot n$ tokens on it initially, the weight of the arc from p to a is $\#_b(w)$, and the weight of the arc from b to p is 1. On the other hand, thanks to the choice of x_{j_1}, we have $x+1 > \#_b(\alpha) \Big/ \#_a(\alpha)$ and $\#_b(\beta) \Big/ \#_a(\beta) > x+1$, which is a contradiction. Hence, $x_n = x$.

From the consideration above we can deduce that all minimal unsolvable words of the form (2) match one of the following three refined patterns

$$\begin{aligned}
&ab^{x+k}ab^{x}a, \text{ with } x > 0,\ k > 2 \quad \text{or}\\
&ab^{x+2}(ab^{x+1})^{*}ab^{x}a, \text{ with } x > 0 \quad \text{or}\\
&ab^{x_1}ab^{x_2}a\ldots ab^{x_n}a, \text{ with } x_1 = x+1,\ x_n = x,\ x_i \in \{x, x+1\} \text{ for } x > 0,\ n \geq 3
\end{aligned} \tag{2'}$$

The last pattern to be studied in detail is (3). Binary words of the form (3) are obviously solvable for $n = 2$. We now consider arbitrary minimal unsolvable word $w = bab^{x_2}ab^{x_3}a\ldots ab^{x_n}$ with $n \geq 3$ and $x_i > 0$ for $2 \leq i \leq n$ of the form (3). Let $x = \min\{x_i \mid 2 \leq i \leq n-1\}$. Due to Lemma 3, $x_i \in \{x, x+1\}$ for all $2 \leq i \leq n-1$, and then $x_n \leq x+2$. Assume $x_n \leq x$. Consider state s in

$$w \;=\; \underbrace{b\ a\ b^{x_2}\ a\ldots\ a\ b^{x_k}}_{\alpha} |_s \underbrace{\overbrace{a\ \ldots\ b^{x_{n-1}-1}}^{\beta'}\ b\ a\ b^{x_n-1}}_{\beta}\ b,$$

from which b cannot be separated (a can always be separated by Lemma 2). Transition b can be separated from the state right after the first b with a place p having an arc from a to p with weight equal to $\max\{x_i \mid 2 \leq i \leq n\}$, an arc from p to b with weight equal to 1, and initially 1 token on it. Hence, $k \neq 1$. Transition b can easily be separated at the very end of w by an input place p of b, having $\#_b(w)$ tokens on p initially. Hence, $k \neq n$. If $k = n-1$, we have

$$\#_a(\alpha) \cdot \#_b(\beta) = (n-2) \cdot (x_n - 1) < 1 \cdot (1 + x_2 + \cdots + x_{n-1}) = \#_a(\beta) \cdot \#_b(\alpha),$$

which, due to the minimal unsolvability of w, contradicts Lemma 1. Hence, $k < n-1$. From Lemma 1, because of minimal unsolvability of w, we have

$$\#_a(\alpha) \cdot \#_b(\beta) \geq \#_a(\beta) \cdot \#_b(\alpha) \iff \frac{\#_b(\beta)}{\#_a(\beta)} \geq \frac{\#_b(\alpha)}{\#_a(\alpha)},$$

where $\#_a(\beta) \neq 0$ because of $k < n-1$, and $\#_a(\alpha) \neq 0$ due to $k > 1$. Since we assumed $x_n \leq x$,

$$\frac{\#_b(\beta')}{\#_a(\beta')} \geq \frac{\#_b(\beta)}{\#_a(\beta)} \iff \#_a(\alpha) \cdot \#_b(\beta') \geq \#_a(\beta') \cdot \#_b(\alpha).$$

Due to Lemma 1, $\alpha\beta' b$ is not solvable. Since it is a proper subword of w, we get a contradiction to the minimality of w. Thus $x+1 \leq x_n \leq x+2$. We now demonstrate $x_2 = x$. Consider two possible cases:

Case 1: $x_n = x+2$. Take $j = \max\{i \mid x_i = x\}$. Then for the subword u

$$u \;=\; \underbrace{b\ a\ b^{x_j}\ (a\ b^{x+1})^{k}}_{\alpha} |_s \underbrace{a\ b^{x_n-1}}_{\beta}\ b.$$

of w with $k \geq 0$, the following inequality is satisfied

$$\#_b(\beta) \cdot \#_a(\alpha) = (x+1) \cdot (k+1) \geq (1 + x + (x+1) \cdot k) \cdot 1 = \#_b(\alpha) \cdot \#_a(\beta).$$

By Lemma 1, u is unsolvable. If $j > 2$, u is a proper subword of w, contradicting the minimality of w. Hence, in this case $x_2 = x$ and $x_i = x + 1$ for $2 < i < n$.

Case 2: $x_n = x + 1$. Let $j_1 = \min\{i \mid x_i = x\}$. By the definition of x, $j_1 \neq n$. By contraposition, assume $x_2 = x + 1$. Consider state s_k in w after the group of b^{x_k}, such that b cannot be separated at s_k (a is always separated by Lemma 2).

If $k > j_1$, then, by Lemma 1, for

$$w \ = \ \underbrace{b\ a\ b^{x_2}\ a\ b^{x_3-1}\ \overbrace{b\ a\ \ldots\ b^{x_{j_1}}\ a\ \ldots\ b^{x_k}}^{\alpha'}}_{\alpha}\ |_{s_k}\ \underbrace{a\ \ldots\ b^{x_n-1}}_{\beta}\ b$$

the following inequality holds

$$\#_b(\beta) \cdot \#_a(\alpha) \geq \#_a(\beta) \cdot \#_b(\alpha) \quad \iff \quad \frac{\#_b(\beta)}{\#_a(\beta)} \geq \frac{\#_b(\alpha)}{\#_a(\alpha)},$$

where $\#_a(\beta) \neq 0$ by the choice of s_k, $\#_a(\alpha) \neq 0$ due to the fact that b can be separated from the state after the first b. As $x_2 = x + 1$ and $x_{j_1} = x$, we have $\#_b(\alpha) \Big/ \#_a(\alpha) \geq \#_b(\alpha') \Big/ \#_a(\alpha')$. From

$$\frac{\#_b(\beta)}{\#_a(\beta)} \geq \frac{\#_b(\alpha')}{\#_a(\alpha')} \quad \Longrightarrow \quad \#_b(\beta) \cdot \#_a(\alpha') \geq \#_a(\beta) \cdot \#_b(\alpha'),$$

according to Lemma 1, it follows that the proper subword $\alpha'\beta b$ of w is unsolvable, contradicting minimality of w. Suppose $k \leq j_1$. Then, by Lemma 1, for

$$w \ = \ \underbrace{b\ a\ b^{x_2}\ a\ \ldots\ b^{x_k}}_{\alpha}\ |_{s_k}\ \underbrace{a\ \ldots\ b^{x_{j_1}-1}\ b\ a\ \ldots\ b^{x_n-1}}_{\beta}\ b$$

the following inequality is satisfied

$$\#_b(\beta) \cdot \#_a(\alpha) \geq \#_a(\alpha) \cdot \#_b(\beta) \quad \iff \quad \frac{\#_b(\beta)}{\#_a(\beta)} \geq \frac{\#_b(\alpha)}{\#_a(\alpha)},$$

with $\#_a(\beta) \neq 0$, by the special form of the word, and $\#_a(\alpha) \neq 0$, due to $k < n$. On the other hand, due to $x_n = x + 1$ and by the choice of j_1, we have $x + 1 > \#_b(\beta) \Big/ \#_a(\beta)$, and $\#_b(\alpha) \Big/ \#_a(\alpha) \geq x + 1$, which is a contradiction. Thus, $x_2 = x$, and we deduce the following refinement of pattern (3)

$$\begin{aligned} &bab^x(ab^{x+1})^* ab^{x+2}, \text{ with } x > 0 \quad \text{ or} \\ &bab^{x_2}ab^{x_3}a\ldots ab^{x_n}, \text{ with } x_2 = x,\ x_n = x+1,\ x_i \in \{x, x+1\} \text{ for } x > 0,\ n \geq 3 \end{aligned} \tag{3'}$$

Notice that sets of words generated by all patterns (2'), (3') and (4') are mutually disjoint. In the following section we divide them into classes of extendable and non-extendable words.

4. Generative nature of minimal unsolvable binary words

In this section we provide a complete characterisation for the class of minimal unsolvable binary words. The general idea is to split the whole set into two classes: extendable (which turn out to serve as origins for more complex minimal unsolvable words) and non-extendable (which might be also regarded as origins for more complex unsolvable, but not minimal, binary words). In the former class we distinguish the simplest extendable muw's, i.e. the words in which the factor α from (1) is of the form a^i or b^i. Such words are called base extendable. After introducing the class of base extendable words, we provide an extension operation based on simple morphisms, which are prefix codes. The code nature is used in subsequent section, where we define the converse procedure, called compression.

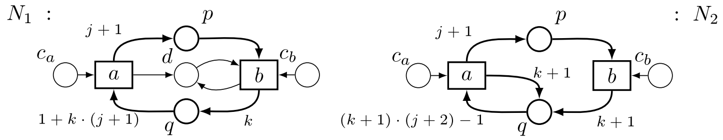
4.1. Base extendable and non-extendable words

The following definitions must be understood modulo swapping a/b (as in the second part of the previous section, we focus on words not containing the infix aa).

Definition 1 (*Base Extendable Words*)**.** A word $u \in \{a, b\}^*$ is called *base extendable* if it is of the form

$$\begin{aligned} &abw(baw)^k a \ \text{ with } \ w = b^j,\ j > 0,\ k \geq 1, \quad \text{ or} \\ &baw(abw)^k b \ \text{ with } \ w = b^j,\ j \geq 0,\ k \geq 1. \end{aligned} \tag{5}$$

The class of base extendable words is denoted by $\mathcal{BE}$. $\square$



$$M_0\begin{pmatrix}p\\q\\d\\c_a\\c_b\end{pmatrix}=\begin{pmatrix}1\\1+k\cdot(j+1)\\0\\k+1\\(j+1)(k+1)\end{pmatrix}\qquad M_0\begin{pmatrix}p\\q\\c_a\\c_b\end{pmatrix}=\begin{pmatrix}j+2\\0\\k+1\\(k+1)(j+1)\end{pmatrix}$$

Fig. 6. N_1 solves the prefix $abb^j(bab^j)^k$. N_2 solves the suffix $bb^j(bab^j)^ka$.

Definition 2 (*Non-Extendable Words*)**.** A word $u \in \{a, b\}^*$ is called *non-extendable* if it is of the form

$$abb^jb^kbab^ja \text{ with } j \geq 0,\ k \geq 1.$$

The class of all non-extendable words is denoted by $\mathcal{NE}$. □

We now establish that all words from classes $\mathcal{BE}$ and $\mathcal{NE}$ are minimal unsolvable.

Lemma 6 (*Minimal Unsolvability of base Extendable and Non-Extendable Words*)**.**
If w belongs to class $\mathcal{BE}$ or $\mathcal{NE}$, then it is unsolvable and minimal with that property.

Proof. Let us notice that a word w is a muw if and only if w is unsolvable and both every proper prefix and every proper suffix of w are solvable. Every word w from $\mathcal{BE} \cup \mathcal{NE}$ is of the form 1, hence unsolvable. We shall prove the minimality of such w by indicating Petri nets solving its maximal proper prefix and suffix.

Case 1 (base extendable words):
(a) $w = abb^j(bab^j)^ka$

Consider first an arbitrary base extendable word of the form $w = abb^j(bab^j)^ka$ with $j \geq 0$ and $k \geq 1$. This form satisfies (1) with $\alpha = b^j$, the star * being repeated zero times, and the plus $^+$ being repeated k times. Due to Proposition 1, all binary words of this form are unsolvable.

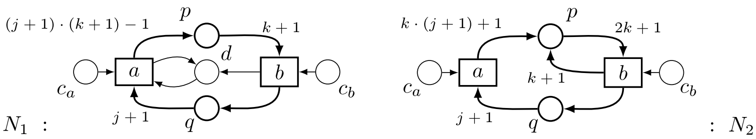
The maximal proper prefix $abb^j(bab^j)^k$ of this word can be solved by Petri net N_1 in Fig. 6. Place q in this net enables the initial a, and then disables it unless b has been fired $j+2$ times. After the execution of block bb^jb, on place q there are $k-1$ tokens more than a needs to fire. These surplus tokens allow a to be fired after each sequence b^jb, but not earlier. Place p has initially 1 token on it, which is necessary to execute block bb^jb after the first a, and this place has only $j+1$ tokens after each next a, preventing b at states where a must occur. Place d prevents premature occurrence of b at the very beginning of the prefix, and places c_a and c_b restrict the total number of firings of a and b, respectively.

For the general form of maximal proper suffix $bb^j(bab^j)^ka$ of w, one can consider Petri net N_2 on the right-hand side of Fig. 6 as a possible solution. Indeed, place q prevents premature occurrences of a in the first block bb^jb, and enables a only after this and each next block b^jb. Doing so, it collects one additional token after each b^jb, which allows this place to enable the very last a after sequence b^j. The initial marking allows to execute the sequence bb^jb at the beginning, and at most $j+1$ b's in a row after that, thanks to place p. Place c_b restricts the total number of b's allowing only block b^j at the end. Thus we deduce that any word of the form $abb^j(bab^j)^ka$ with $j > 0$ and $k \geq 1$ is a muw.

(b) $w = bab^j(abb^j)^kb$

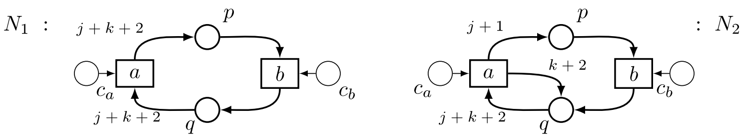
We can similarly examine arbitrary base extendable word of another form $w = bab^j(abb^j)^kb$ with $j \geq 0$ and $k \geq 1$. The word w satisfies (1) with swapped a and b, $\alpha = b^j$, the star * being repeated zero times, and the plus $^+$ being repeated k times. Due to Proposition 1, all binary words of this form are unsolvable. Petri nets N_1 and N_2 in Fig. 7 are possible solutions for maximal proper prefix and for maximal proper suffix of w, respectively.

Remark (*On Special Structure of Petri Nets which Solve Prefixes and Suffixes*)**.** Petri net N_1 in Fig. 6, which solves maximal proper prefix $abb^j(bab^j)^k$ of word $w = abb^j(bab^j)^ka$ from class $\mathcal{BE}$, has a special structure. Place d serves for preventing undesirable b in the very beginning of w, and places c_a and c_b restrict the total number of a and b, correspondingly. So, the internal structure of the word, being executed by N_1, is completely determined by two places p and q, which prevent b and a, respectively, when and only there is a necessity. In what follows, we will call the part of N_1 consisting of these two places (and transitions) a *core part*. So, Petri net N_2 in Fig. 6 has a core part made of places p and q. Similarly, such parts are formed by places p and q for both nets in Fig. 7. In future consideration we shall sometimes concentrate only on such core parts, as the other necessary places of the net may be added in an uncomplicated way and does not influence the main behaviour of the net.



$$M_0\begin{pmatrix}p\\q\\d\\c_a\\c_b\end{pmatrix}=\begin{pmatrix}k+1\\j+1\\0\\k+1\\(k+1)(j+1)\end{pmatrix}\qquad M_0\begin{pmatrix}p\\q\\c_a\\c_b\end{pmatrix}=\begin{pmatrix}0\\j+2\\k+1\\(k+1)(j+1)\end{pmatrix}$$

Fig. 7. N_1 solves the prefix $bab^j(abb^j)^k$. N_2 solves the suffix $ab^j(abb^j)^kb$.



$$M_0\begin{pmatrix}p\\q\\c_a\\c_b\end{pmatrix}=\begin{pmatrix}0\\j+k+2\\2\\2\cdot(j+1)+k\end{pmatrix}\qquad M_0\begin{pmatrix}p\\q\\c_a\\c_b\end{pmatrix}=\begin{pmatrix}j+k+2\\0\\2\\2\cdot(j+1)+k\end{pmatrix}$$

Fig. 8. N_1 solves the prefix $abb^jb^kbab^j$. N_2 solves the suffix $bb^jb^kbab^ja$.

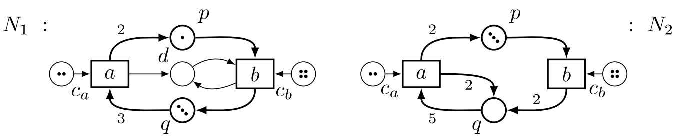


Fig. 9. N_1 solves the prefix *abbbab*. N_2 solves the suffix *bbbaba*.

Case 2 (non-extendable words): We now demonstrate that any (modulo swapping a/b) binary word of the form $w = abb^jb^kbab^ja$ with $j \geq 0$ and $k \geq 1$ from class $\mathcal{NE}$ is minimal unsolvable. w satisfies (1) with $\alpha = b^j$, the star * being repeated k times, and the plus $^+$ being repeated only once. Due to Proposition 1, w is unsolvable. To show minimality of w, we provide Petri nets N_1 and N_2 (see Fig. 8) solving its maximal proper prefix and maximal proper suffix, respectively. □

Example 1. Let us consider a word $w = abbbaba$, which is of the form (1), with $\alpha = b$, the star * being repeated zero times, and the plus $^+$ being repeated just once. By Definition 1, w is a base extendable word with $j = 1$ and $k = 1$. The word w is unsolvable (by Proposition 1) and minimal with that property. We show the minimality by introducing Petri nets solving a proper prefix *abbbab* and a proper suffix *bbbaba* of w. Those Petri nets, constructed on the basis of the proof of Lemma 6, are depicted in Fig. 9.

Notice that both Petri nets contain core parts consisting of places p and q, which are responsible for the required behaviour of the nets, as well as auxiliary places — a delay place d and counter places c_a and c_b.

4.2. Extension operation and extendable words

Let us now explain how some minimal unsolvable words can be obtained from other minimal unsolvable words. For this purpose we use the following notion of *extension operation*:

Definition 3 (*Extension Operation*)**.** For a word $v = xwx$ ($w \in \{a, b\}^*$, $x \in \{a, b\}$) an extension operation E is defined as follows:

$$E(awa) = \bigcup_{i=1}^{\infty}\Big\{abM_{a,i}(w)a^{i+1},\ aM_{b,i}(wa)\Big\},$$

$$E(bwb) = \bigcup_{i=1}^{\infty}\Big\{baM_{b,i}(w)b^{i+1},\ bM_{a,i}(wb)\Big\},$$

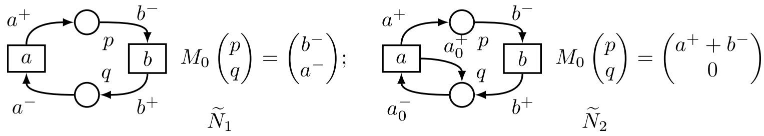


Fig. 10. Core parts of Petri nets: $\widetilde{N}_1$ for a net solving prefix, $\widetilde{N}_2$ for a net solving suffix.

where $M_{a,i}$ and $M_{b,i}$ are morphisms defined as follows

$$M_{a,i}=\begin{cases}a & \mapsto & a^{i+1}b\\ b & \mapsto & a^ib\end{cases} \quad\text{and}\quad M_{b,i}=\begin{cases}a & \mapsto & b^ia\\ b & \mapsto & b^{i+1}a\end{cases}. \quad \square$$

In what follows, for a given $w \in \{a, b\}^*$, we shall call $u \in E(w)$ an *extension* of w (when $E(w)$ is defined).
We are now ready to define the class of *extendable* minimal unsolvable words.

Definition 4 (*Extendable Words*)**.** For a word $w \in \{a, b\}^*$

1. if $w \in E(v)$ for a base extendable v, then w is extendable,
2. if $w \in E(v)$ for an extendable v, then w is extendable,
3. there are no other extendable words.

The class of all extendable words is denoted by $\mathcal{E}$. $\square$

Lemma 7 (*Unsolvability of Extendable Words*)**.** *Let $u \in \{a, b\}^*$ be of the form $abv(bav)^ka$ or $bav(abv)^kb$ ($k > 0$). Then $E(u)$ is a set of PN-unsolvable words.*

Proof. Let $u = abv(bav)^ka$ ($k > 0$). Then

$$\begin{aligned}E(u) &= \bigcup_{i\in N}\Big\{aba^ibM_{a,i}(v)\Big(a^iba^{i+1}bM_{a,i}(v)\Big)^ka^{i+1},\\ &\qquad ab^{i+1}aM_{b,i}(v)\Big(b^{i+1}ab^iaM_{b,i}(v)\Big)^kb^ia\Big\} =\\ &= \bigcup_{i\in N}\Big\{ab(a^ibM_{a,i}(v)a^i)\Big(ba(a^ibM_{a,i}(v)a^i)\Big)^ka,\\ &\qquad ab(b^iaM_{b,i}(v)b^i)\Big(ba(b^iaM_{b,i}(v)b^i)\Big)^ka\Big\} =\\ &= \bigcup_{i\in N}\Big\{abv_a\Big(bav_a\Big)^ka, abv_b\Big(bav_b\Big)^ka\Big\}.\end{aligned}$$

Therefore, by Proposition 1, $E(u)$ is a set of PN-unsolvable words. The case $u = bav(abv)^kb$ can be proved similarly. $\square$

Transformations of core part w.r.t. morphisms

As it has been demonstrated above, for every base extendable word w there are Petri nets N_1 and N_2, which solve maximal proper prefix w_1 and maximal proper suffix w_2 of w, respectively. These nets N_1 and N_2 have a special structure: so called "core" parts $\widetilde{N}_1$ and $\widetilde{N}_2$ (general patterns of $\widetilde{N}_1$ and $\widetilde{N}_2$ are depicted in Fig. 10) determine internal order of firings of a's and b's during execution of w_1 and w_2, while the remaining parts of N_1 and N_2 take responsibility of correct implementation of the beginnings and the ends of w_1 and w_2. Applying operation E to w, one can easily obtain a new minimal unsolvable word w'. Moreover, applying appropriate transformation (which is determined by the particular morphism that has been used to gain w' from w) to $\widetilde{N}_1$ or to $\widetilde{N}_2$, one derives new core part $\widetilde{N}_1'$ or $\widetilde{N}_2'$, which correctly implements the internal structure of the maximal proper prefix w_1' or the maximal proper suffix w_2' of w', respectively. In Table 1 the correspondence between morphisms from Definition 3 and such transformations of nets is provided for general forms of $\widetilde{N}_1$ and $\widetilde{N}_2$. This fact is confirmed throughout the proof of the following lemma

Lemma 8 (*Minimality of Extendable Words*)**.** *If $w \in \mathcal{E}$, then w is minimal unsolvable.*

Proof (*Sketch*[5])**.** By Lemma 7, any extendable word is unsolvable. According to Definition 4, for every $w \in \mathcal{E}$ there is a sequence $w_0, w_1, \ldots, w_r$ such that $w_0 \in \mathcal{BE}$, $w_j \in \mathcal{E}$ and $w_j \in E(w_{j-1})$ for $1 \le j \le r$, and $w_r = w$. We proceed by induction

5 For the sake of readability the complete, technical proof is given in Appendix.

Table 1
The correspondence between morphisms and transformations.

	$M_{a,i}$	$M_{b,i}$
$\widetilde{N}_1$	$a^+ \longmapsto a^+ + b^-$ $b^- \longmapsto b^- + i \cdot (a^+ + b^-)$ $b^+ \longmapsto b^+ + i \cdot (a^- + b^+)$ $a^- \longmapsto a^- + b^+$ $M(p) \longmapsto b^- + i \cdot (a^+ + b^-)$ $M(q) \longmapsto a^- + b^+$	$a^+ \longmapsto a^+ + i \cdot (a^+ + b^-)$ $b^- \longmapsto a^+ + b^-$ $b^+ \longmapsto a^- + b^+$ $a^- \longmapsto a^- + i \cdot (a^- + b^+)$ $M(p) \longmapsto a^+ + b^-$ $M(q) \longmapsto a^- + i \cdot (a^- + b^+)$
$\widetilde{N}_2$	$a^+ \longmapsto a^+ + b^-$ $b^- \longmapsto b^- + i \cdot (a^+ + b^-)$ $b^+ \longmapsto b^+ + i \cdot (a_0^- + b^+ - a_0^+)$ $a_0^- \longmapsto a_0^- + b^+$ $a_0^+ \longmapsto a_0^+$ $M(p) \longmapsto b^- + (i+1) \cdot (a^+ + b^-)$ $M(q) \longmapsto 0$	$a^+ \longmapsto a^+ + i \cdot (a^+ + b^-)$ $b^- \longmapsto a^+ + b^-$ $b^+ \longmapsto b^+ + a_0^- - a_0^+$ $a_0^- \longmapsto a_0^- + i \cdot (b^+ + a_0^- - a_0^+)$ $a_0^+ \longmapsto a_0^+$ $M(p) \longmapsto a^+ + (i+1) \cdot (a^+ + b^-)$ $M(q) \longmapsto 0$

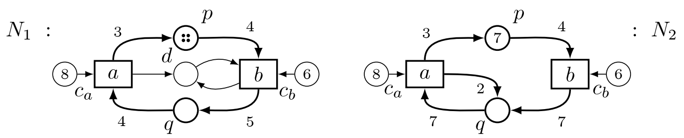


Fig. 11. N_1 solves the prefix *ababababaababa* and N_2 solves the suffix *babababaababaa* of $w_{a,1} = ababababaababaa$.

on r, and check the existence of Petri nets, solving maximal proper prefix and suffix of w. Every base extendable word w_0 is minimal unsolvable, and there are Petri nets N_1^0 and N_2^0 with core parts and additional parts, which solve the maximal proper prefix and suffix of w_0. Suppose, for $1 \leq j \leq r-1$, there are Petri nets N_1^j and N_2^j doing similar job for w_j, and which have been obtained from N_1^{j-1} and N_2^{j-1}, respectively, with the appropriate transformation of core part. The particular morphism $M_{x,i}$ with $x \in \{a, b\}$, that has been used to derive w_j from w_{j-1}, determines this transformation uniquely. Inductive step consists of proving that N_1^r and N_2^r obtained from N_1^{r-1} and N_2^{r-1}, respectively, solve maximal proper prefix and suffix of w_r. Having morphism $M_{x,i}$, the transformation and the two core parts (new and old), it can be directly checked that place p disables/enables transition b in prefix of w_{r-1} as a place of (core part of) N_1^{r-1} if and only if p does the same as a place of (core part of) N_1^r at the corresponding state in prefix of w_r. Similarly, for place q and transition a, and also for suffixes of w_{r-1} and w_r with nets N_2^{r-1} and N_2^r. Additional parts of nets N_1^r and N_2^r can be implemented with a place "from initial to non-initial" transition, having zero tokens initially and "enough many" tokens after, and a place which is a simple counter for the (total) number of firings. □

Let us note that the extension operation being applied to an extendable word, produces another extendable word which is unsolvable and minimal. On the other hand, from a non-extendable word this operation derives unsolvable but not minimal words.

Example 2. Observe again the word $w = abbbaba$. From the previous considerations (see Example 1) we know that this word is base extendable, and therefore is a muw. By the application of the extension operation, using the morphism $M_{a,1} = \begin{cases} a & \mapsto & aab \\ b & \mapsto & ab \end{cases}$, we obtain a word $w_{a,1} = ab\,ababa\,ba\,ababa\,a$, which is of the form (1) with $\alpha = ababa$, the star * being repeated zero times, and the plus $^+$ being repeated just once, hence – by Proposition 1 – unsolvable. On the basis of the Petri nets of Fig. 9, and according to Table 1 we construct Petri nets (depicted in Fig. 11) solving the maximal proper prefix *ababababaababa* and the maximal proper suffix *babababaababaa* of $w_{a,1}$. Thus, $w_{a,1}$ is a minimal unsolvable word.

Lemma 9 (*Unsolvability of Extensions of Non-Extendable Words*)**.** *If $w \in \mathcal{NE}$, then extension $u \in E(w)$ is unsolvable but not minimal unsolvable.*

Proof. Consider arbitrary $w \in \mathcal{NE}$, where $w = abb^jb^kbab^ja$ with $j \geq 0$, $k \geq 1$. Depending on a particular morphism $M_{x,i}$ with $x = a$ or $x = b$ for some $i \geq 1$, extension $u_x \in E(w)$ of $w = aw_1a$ can be

$$u_a = abM_{a,i}(w_1)a^{i+1} = ab\ a^ib\ (a^ib)^j\ (a^ib)^k\ a^ib\ a^{i+1}b\ (a^ib)^j\ a^{i+1} =$$
$$= ab\ (a^ib)^{k-1}\ a^{i-1}\ ab\ \underbrace{a^ib\ (a^ib)^j\ a^i}_{\alpha_a}|\ ba\ \underbrace{a^ib\ (a^ib)^j\ a^i}_{\alpha_a}\ a,$$

or

$$u_b = aM_{b,i}(w_1a) = a\ b^{i+1}a\ (b^{i+1}a)^j\ (b^{i+1}a)^k\ b^{i+1}a\ b^ia\ (b^{i+1}a)^j\ b^ia =$$
$$= (ab^{i+1})^k\ ab\ \underbrace{b^ia\ (b^{i+1}a)^j\ b^i}_{\alpha_b}|\ ba\ \underbrace{b^ia\ (b^{i+1}a)^j\ b^i}_{\alpha_b}\ a$$

respectively. By Proposition 1, word $ab\alpha_b ba\alpha_b a$ is unsolvable, which means unsolvability of u_b. Due to $k \geq 1$, $ab\alpha_b ba\alpha_b a$ is a proper subword of u_b. Hence, u_b is not minimal unsolvable. Analogously, unsolvability of $ab\alpha_a ba\alpha_a a$ implies non-minimal unsolvability of u_a. □

5. Generation-based classification of minimal unsolvable words

Consider minimal unsolvable words w.r.t. the classification obtained earlier. All possible patterns from (2), (3'), (4), and more precisely their refined variants from , can be distinguished into base extendable ($\mathcal{BE}$)

$ab(ba)^{k+1}a$, with $k \geq 0$, for the second pattern from (4'),
$abb^x(bab^x)^ka$, with $x > 0, k > 0$, for the second pattern from (2'),
$bab^x(abb^x)^kb$, with $x > 0, k > 0$, for the first pattern from (3'),

non-extendable ($\mathcal{NE}$)

$abb^{x-1}ba$, with $x \geq 2$ for the first pattern from (4'),
$abb^xb^{k-1}bab^xa$, with $x > 0,\ k > 2$ for the first pattern from (2'),

and the remaining ones, which we will call *compressible* ($\mathcal{C}$)

$ab^{x_1}ab^{x_2}a\dots ab^{x_n}a$, with $x_1 = x+1,\ x_n = x,\ x_i \in \{x, x+1\}, x > 0,\ n \geq 3$,
for the third pattern from (2'),
$bab^{x_2}ab^{x_3}a\dots ab^{x_n}$, with $x_2 = x,\ x_n = x+1,\ x_i \in \{x, x+1\}, x > 0,\ n \geq 3$,
for the second pattern from (3').

From this classification we derive that the class of all minimal unsolvable words $\mathcal{MUW} = \mathcal{BE} \cup \mathcal{NE} \cup \mathcal{C}$, where $\mathcal{BE}$, $\mathcal{NE}$ and $\mathcal{C}$ are mutually disjoint classes. Note, that since all words from class $\mathcal{E}$ are unsolvable and minimal with that property, and $\mathcal{E}$ is disjoint with $\mathcal{BE}$ and $\mathcal{NE}$, we have $\mathcal{E} \subseteq \mathcal{C}$.

5.1. Morphic compression and reducibility

In the previous section we showed how to construct new minimal unsolvable words on the basis of extendable words. The purpose of this section is to introduce an inverse transformation, which allows to compress longer minimal unsolvable words into shorter ones.

Definition 5 (*Compression Function*)**.** For a word $v \in \{a, b\}^*$ starting and ending with the same letter $x \in \{a, b\}$ a *compression function* C is defined as follows:

$$C(v) = C(abua^{i+1}) = aM_{a,i}^{-1}(u)a, \quad C(v) = C(baub^{i+1}) = bM_{b,i}^{-1}(u)b,$$
$$C(v) = C(auba) = aM_{b,i}^{-1}(uba), \quad C(v) = C(buab) = bM_{a,i}^{-1}(uab), \tag{6}$$

where $i \geq 1$, $u \in \{a, b\}$ and $M_{a,i}$, $M_{b,i}$ are morphisms defined as follows:

$$M_{a,i}^{-1} : \begin{cases} a^{i+1}b & \mapsto a \\ a^ib & \mapsto b \end{cases} \qquad \text{and} \qquad M_{b,i}^{-1} : \begin{cases} b^ia & \mapsto a \\ b^{i+1}a & \mapsto b. \end{cases} \quad \square$$

It is easy to see that among all possible forms from the classification of minimal unsolvable words, function C is defined exactly for patterns from class $\mathcal{C}$. Moreover, the form of the word explicitly defines the particular morphism $M_{x,i}^{-1}$ which is used when applying C to the word. Let us also notice that since $\mathcal{E} \subseteq \mathcal{C}$, all words from class $\mathcal{E}$ are compressible with function C.

From Definitions 3 and 5 it is clear that $M_{x,i}$ is reciprocal to $M_{x,i}^{-1}$ for $x \in \{a, b\}$, $i \geq 1$. The following lemma establishes that the extension operation E and the application of compression function C are complement to each other in the following sense

Lemma 10 (*Compression and Extension Functions*)**.**

1. *If $v \in \mathcal{BE} \cup \mathcal{E}$ and $u \in E(v)$, then $C(u) = v$;*
2. *If $u \in \mathcal{C}$ and $v = C(u)$, then $u \in E(v)$.*

Proof. 1. Let $v = xv_1x$, where $x \in \{a, b\}$. Hence, for distinct $x, y \in \{a, b\}$ and $i \geq 1$, we have two possible cases:

- $u = xyM_{x,i}(v_1)x^{i+1}$. By compression function definition,
$$C(u) = C(xyM_{x,i}(v_1)x^{i+1}) = xM_{x,i}^{-1}(M_{x,i}(v_1))x = xv_1x = v.$$
- $u = xM_{y,i}(v_1x)$. By compression function definition,
$$C(u) = C(xM_{y,i}(v_1x)) = C(xM_{y,i}(v_1)y^ix) = xM_{y,i}^{-1}(M_{y,i}(v_1)y^ix) = xv_1x = v.$$

2. Without the loss of generality, u starts and ends with $x \in \{a, b\}$. Due to Definition 5 of function C and class $\mathcal{C}$, u uniquely determines which compression morphism can be applied to it. Two cases are possible:

- $v = C(u) = xM_{x,i}^{-1}(u_1)x$ for $u = xyu_1x^{i+1}$, $x \neq y \in \{a, b\}$. Then,
$$E(v) = \bigcup_{j=1}^{\infty}\{xyM_{x,j}(M_{x,i}^{-1}(u_1))x^{i+1}, xM_{y,j}(M_{x,i}^{-1}(u_1)x)\}.$$
As $u = xyM_{x,j}(M_{x,i}^{-1}(u_1))x^{i+1}$ for $j = i$, hence $u \in E(v)$.
- $v = C(u) = xM_{y,i}^{-1}(u_1xy^ix)$ for $u = xu_1xy^ix$, $x \neq y \in \{a, b\}$. Then,
$$E(v) = \bigcup_{j=1}^{\infty}\{xyM_{x,j}(M_{y,i}^{-1}(u_1x))x^{i+1}, xM_{y,j}(M_{y,i}^{-1}(u_1xy^ix))\}.$$
As $u = xM_{y,j}(M_{y,i}^{-1}(u_1xy^ix))$ for $j = i$, hence $u \in E(v)$. □

5.2. Compression of a muw is an unsolvable word

Throughout this section we will demonstrate that applying the compression functions to muws (when defined) is an automorphism within the class $\mathcal{MUW}$, i.e. the results are also minimal unsolvable words. The following technical lemmata will be helpful in the further considerations.

Lemma 11. *Suppose $w = \alpha|_s b^{m-1}|_{\tilde{s}} ba\beta$, with $m \geq 1$. If a is not separable at state s, then it is not separable at state $\tilde{s}$, as well.*

Proof. By contraposition, assume there is a Petri net $N = (P, T, F, M_0)$ with a place $p \in P$ such that w can be fired completely, and $M_{\tilde{s}}(p) < F(p, a)$. Since a is enabled at the state right after $\tilde{s}$, b effectively increases the number of tokens on p. Hence, $M_s(p) \leq M_{\tilde{s}}(p) < F(p, a)$, i.e. a is separable at state s with place p, contradiction. □

Lemma 12. *If $w = ab^{x_1}ab^{x_2}a \dots ab^{x_n}a$, with $x_1 = x + 1$, $x_n = x$, $x_i \in \{x, x + 1\}$, $x > 0$, $n \geq 3$, is a minimal unsolvable word, and a separation failure occurs in group b^{x_k}, then $x_k = x + 1$.*

Proof. By Lemma 11, a is not separable at some state s in

$$w = \underbrace{\overbrace{a\ b^{x_1}\ a\ \dots\ a\ b^{x_{k-1}-1}}^{\alpha'} \mid b\ a\ b^{x_k-1}}_{\alpha} \ |_s\ \underbrace{b\ a\ \dots\ a\ b^{x_{n-1}}\ a\ b^{x_n}}_{\beta}\ a,$$

which implies, according to Lemma 1, that

$$(x_1 + \cdots + x_k - 1) \cdot (n - k) \ \geq\ (1 + x_{k+1} + \cdots + x_n) \cdot k \iff$$
$$\frac{x_1 + \cdots + x_k - 1}{k} = \frac{\#_b(\alpha)}{\#_a(\alpha)} \geq \frac{\#_b(\beta)}{\#_a(\beta)} = \frac{1 + x_{k+1} + \cdots + x_n}{n - k},$$

where $\#_a(\alpha) \neq 0$ and $\#_a(\beta) \neq 0$. Assume now, by contraposition, that $x_k = k$. Since for every $1 \leq i \leq n$ we have $x \leq x_i \leq x + 1$, then $\#_b(\alpha')\big/\#_a(\alpha') \geq \#_b(\alpha)\big/\#_a(\alpha)$, where $\#_a(\alpha') \neq 0$ because w starts with a. From $x_1 = x + 1$, it follows that $k > 1$. Due to $x_n = x = x_k$, we have $\#_b(\beta')\big/\#_a(\beta') = \#_b(\beta)\big/\#_a(\beta)$, where $\#_a(\beta') \neq 0$ since $k > 1$. Thus, $\#_b(\alpha')\big/\#_b(\alpha') \geq \#_b(\beta')\big/\#_a(\beta')$, which implies, by Lemma 1, unsolvability of $\alpha'\beta'a$, contradicting the minimality of w. □

Lemma 13. *If $w = bab^{x_2}ab^{x_3}a \dots ab^{x_n}$, with $x_2 = x$, $x_n = x + 1$, $x_i \in \{x, x + 1\}$, $x > 0$, $n \geq 3$, is a minimal unsolvable word, and separation failure occurs after group b^{x_k}, then $x_k = x$.*

Proof. For state s in w, from which b is not separable,

$$w = \underbrace{\overbrace{b\ a\ b^{x_2-1}\ b\ a\ \ldots\ a\ b^{x_{k-1}}}^{\alpha'} \mid a\ b^{x_k}}_{\alpha} \mid_s \underbrace{\overbrace{a\ b^{x_k}\ \ldots\ a\ b^{x_{n-1}-1}}^{\beta'}\ b\ a\ b^{x_n-1}}_{\beta}\ b$$

according to Lemma 1, we have

$$(k-1)\cdot(x_{k+1}+\cdots+x_n-1) \geq (1+x_2+\cdots+x_k)\cdot(n-k) \iff$$
$$\frac{x_{k+1}+\cdots+x_n-1}{n-k} = \frac{\#_b(\beta)}{\#_a(\beta)} \geq \frac{\#_b(\alpha)}{\#_a(\alpha)} = \frac{1+x_2+\cdots+x_k}{k-1},$$

where $\#_a(\beta) \neq 0$ since β starts with a, and $\#_a(\alpha) \neq 0$ because $k > 1$. By contraposition, assume $x_k = k+1$. Since for all $2 \leq i \leq n$ we have $x \leq x_i \leq x+1$, then $\#_b(\alpha')\Big/\#_a(\alpha') \leq \#_b(\alpha)\Big/\#_a(\alpha)$, where $\#_a(\alpha') \neq 0$ because of $k > 2$. From $x_n = x+1 = x_k$ it follows that $\#_b(\beta)\Big/\#_a(\beta) = \#_b(\beta')\Big/\#_a(\beta')$, where $\#_a(\beta') \neq 0$ due to β' starts with a. Hence, $\#_b(\beta')\Big/\#_a(\beta') \geq \#_b(\alpha')\Big/\#_a(\alpha')$. Due to Lemma 1, this implies unsolvability of $\alpha'\beta'b$, contradicting the minimality of w. $\square$

Consider an arbitrary minimal unsolvable word $w = aw_1 = ab^{x_1}ab^{x_2}a\ldots ab^{x_n}a$ from class $\mathcal{C}$, with $x_1 = x+1$, $x_n = x$, $x_i \in \{x, x+1\}$, $x > 0$, $n \geq 3$. According to the special form of w, compression function C can merely be applied to w in the form $C(w = aw_1) = aM_{b,x}^{-1}(w_1)$. Note that $u = C(w)$ is also unsolvable. Due to Lemma 12, for state s in

$$w = \underbrace{a\ b^{x_1}\ a\ \ldots\ a\ b^{x_k-1}}_{\alpha} \mid_s \underbrace{b\ a\ \ldots\ a\ b^{x_n}}_{\beta}\ a,$$

from which a is not separable, we have $x_k = x+1$. By Lemma 1,

$$(n-k)\cdot(x_1+x_2+\cdots+x_k-1) \geq k\cdot(x_{k+1}+\cdots+x_n+1)$$

Assume, there are l groups of b^x in α (except the part of b^{x_k}), and m groups of b^x in β. Due to the form of w, we have $0 \leq l < k-1$ and $0 < m \leq n-k$. Hence,

$$\begin{aligned}
&\#_a(\beta)\cdot\#_b(\alpha) \geq \#_a(\alpha)\cdot\#_b(\beta) \iff \\
&\iff (n-k)\cdot(k\cdot(x+1)-l-1) \geq k\cdot((n-k)\cdot(x+1)-m+1) \iff \\
&\iff k\cdot l + k\cdot m - n\cdot l - n \geq 0.
\end{aligned}$$

After applying the compression function to w, due to the definition of C and $M_{b,x}^{-1}$, for every sequence $b^x a$ and for every sequence $b^{x+1}a$ in w, we obtain a and b in u, respectively. Hence, u has $n+1$ letters at all, starts with ab and ends with a thanks to the definition of C and the shape of w, and, by Lemma 12, has b on $(k+1)$th position:

$$u = \underbrace{a\,b\ldots}_{\alpha'} \mid_{s'} \underbrace{b\ \ldots}_{\beta'}\ a,$$

where $|\alpha'| = k$, $|\beta'| = n-k$. Moreover, $\#_a(\alpha') = l+1$ and $\#_a(\beta') = m-1$. Thus, we have $\#_a(\beta')\cdot\#_b(\alpha') = (m-1)\cdot(k-l-1)$ and $\#_a(\alpha')\cdot\#_b(\beta') = (l+1)\cdot(n-k-m+1)$. Then,

$$\#_a(\beta')\cdot\#_b(\alpha') - \#_a(\alpha')\cdot\#_b(\beta') = k\cdot l + k\cdot m - n\cdot l - n \geq 0,$$

where the last inequality is because of $\#_a(\beta)\cdot\#_b(\alpha) \geq \#_a(\alpha)\cdot\#_b(\beta)$ (see above). Due to Lemma 1, this implies unsolvability of u.

Let us now consider an arbitrary minimal unsolvable word $w = bab^{x_2}ab^{x_3}a\ldots ab^{x_n}$ from class $\mathcal{C}$, with $x_2 = x$, $x_n = x+1$, $x_i \in \{x, x+1\}$, $x > 0$, $n \geq 3$, and check that $u = C(w)$ is unsolvable as well. The form of w explicitly determines that $C(w = bw_1b^{x+1}) = bM_{b,x}^{-1}(w_1)b$. By Lemma 13, for state s from which b is not separable in

$$w = \underbrace{b\ a\ b^{x_2}\ a\ \ldots\ b^{x_k}}_{\alpha} \mid_s \underbrace{a\ b^{x_{k+1}}\ a\ \ldots\ a\ b^{x_n-1}}_{\beta}\ b,$$

we have $x_k = x$. From Lemma 1,

$$(k-1)\cdot(x_{k+1}+\cdots+x_n-1) \geq (1+x_2+\cdots+x_k)\cdot(n-k).$$

Assume, there are l groups of b^{x+1} in α and m groups of b^{x+1} in β. Due to the form of w, we have $0 \leq l < k$ and $0 \leq m \leq n-k$, and

$$\begin{aligned}
&(k-1)\cdot(x\cdot(n-k)+m) \geq (1+x\cdot(k-1)+l)\cdot(n-k) \iff \\
&\iff k\cdot m - m - n + k - l\cdot n + l\cdot k \geq 0.
\end{aligned}$$

After applying the function C to w, according to the definition of $M_{b,x}^{-1}$, for every sequence $b^{x+1}a$ and every sequence b^xa in w, we obtain a and b in u, respectively. Hence, u has n letters at all, starts with ba and ends with b, by definition of the function C and the special shape of w, and, by Lemma 13, has a on kth position:

$$u = \underbrace{ba\ldots}_{\alpha'} |_{s'} \underbrace{a\ldots}_{\beta'} b,$$

where $|\alpha'| = k-1$, $|\beta'| = n-k$. Moreover, $\#_b(\alpha') = l$ and $\#_b(\beta') = m$. Thus, $\#_a(\alpha') \cdot \#_b(\beta') = (k-1-l) \cdot m$ and $\#_b(\alpha') \cdot \#_a(\beta') = l \cdot (n-k-m)$. Then,

$$\begin{aligned} \#_a(\alpha') \cdot \#_b(\beta') - \#_b(\alpha') \cdot \#_a(\beta') &= k \cdot m - m - l \cdot n + l \cdot k \geq \\ &\geq k \cdot m - m - l \cdot n + l \cdot k + k - n \geq 0. \end{aligned}$$

By Lemma 1, we deduce that u is unsolvable.

So far, we have shown that the compression-image of any word from $\mathcal{C}$ is unsolvable. Suppose that $\mathcal{C} \setminus \mathcal{E} \neq \emptyset$. Take u, one of the shortest words from $\mathcal{C} \setminus \mathcal{E}$ and let $w = C(u)$. Since w is unsolvable, two cases are possible:

Case 1: w is a minimal unsolvable word. Due to the choice of u as shortest in $\mathcal{C} \setminus \mathcal{E}$, and the fact that w is shorter than u, we have $w \notin \mathcal{C} \setminus \mathcal{E}$. Hence, w belongs to one of disjoint classes $\mathcal{BE}$, $\mathcal{NE}$, $\mathcal{E}$. If $w \in \mathcal{BE}$ or $w \in \mathcal{E}$, then, by Definition 4 and Lemma 10, $u \in E(w) \subseteq \mathcal{E}$, which contradicts the choice of $u \in \mathcal{C} \setminus \mathcal{E}$. If $w \in \mathcal{NE}$, then, by Lemma 9, $u \in E(w)$ is not a minimal unsolvable word, contradicting the minimality of u.

Case 2: w is not a minimal unsolvable word. We shall prove that u is also not a minimal unsolvable word. Assume now $w = w_1 v w_2$, where v is a minimal unsolvable word and $w_1 w_2 \neq \epsilon$, and that w has been obtained from u using the compression morphism $M_{x,i}^{-1}$, where $x \in \{a, b\}$. Since v is a proper subword of w, and w is shorter than u, then $v \notin \mathcal{C} \setminus \mathcal{E}$. From the minimal unsolvability of v we have $v \in \mathcal{BE} \cup \mathcal{E} \cup \mathcal{NE}$. Hence, any extension v' of v is unsolvable (possibly not minimal in case $v \in \mathcal{NE}$). For $x \neq y$, where $x, y \in \{a, b\}$, we have either $v = xv_1x$, or $v = yv_1y$. Consider these two possibilities.

1. $v = xv_1x$. In this case, according to Definition 3, we consider extension $v' = xyM_{x,i}(v_1)x^{i+1} \in E(v)$. Suppose both w_1 and w_2 are non-empty words. Hence, $M_{x,i}(v) = x^{i+1}yM_{x,i}(v_1)x^{i+1}y$ is a proper subword of u. As v' is a subword of $M_{x,i}(v)$, we get a contradiction to the minimal unsolvability of u. Assume, $w_1 = \epsilon$. Then, being a proper prefix of w, after extension v will be morphed to $xyM_{x,i}(v_1)x^{i+1}y$, which again has v' as a subword, implying contradiction to the minimality of u. If $w_2 = \epsilon$, extension u of w with morphism $M_{x,i}$ has a proper subword $x^{i+1}yM_{x,i}(v_1)x^{i+1}$, and hence, contains v' as well. This contradicts the minimal unsolvability of u.
2. $v = yv_1y$. Let now $v' = yM_{x,i}(v_1y) \in E(v)$. In case w_1 is non-empty word, $M_{x,i}(v) = x^iyM_{x,i}(v_1y)$ is a proper subword of u, and contains v' as a factor. This contradicts the minimality of u. If $w_1 = \epsilon$, u has v' as a proper prefix, which again contradicts the minimal unsolvability of u.

Thus, $\mathcal{C} = \mathcal{E}$, which establishes the first of main results of the paper

Theorem 1 (*Generative Nature of Minimal Unsolvable Binary Words*)**.** *Let w be a minimal Petri net unsolvable binary word. Then we have the following exclusive alternatives:*

- *w is a non-extendable word ($w \in \mathcal{NE}$),*
- *w is a base extendable word ($w \in \mathcal{BE}$),*
- *w is an extendable word ($w \in \mathcal{E}$).*

Basing on Theorem 1 and proofs of Lemmata 6 and 7 we can formulate the following

Corollary 1 (*The Necessary Condition for Unsolvability*)**.** *If a word over $\{a, b\}$ is not PN-solvable, it has a subword of the form (1).*

Generation of maximal partial solutions of minimal unsolvable words

In the last case of the alternatives from Theorem 1 (case $w \in \mathcal{E}$), applying function C to w consecutively, we can recover the (unique) sequence of minimal unsolvable words $w_0, w_1, \ldots, w_r$, such that $w_0 \in \mathcal{BE}$, $w_r = w$, $w_i \in \mathcal{E}$ and $w_{i-1} = C(w_i)$ for $1 \leq i \leq r$. Moreover, starting from a word w_0, its maximal proper prefix and maximal proper suffix, and Petri nets solving them (in special forms, that have been provided in the paper), using appropriate transformations, we can derive Petri nets solving maximal proper prefix and maximal proper suffix of w_i for all $1 \leq i \leq r$.

Example 3. Let us consider word $v = ba\,aabaaabaa\,ab\,aabaaabaa\,b$. It is unsolvable by Proposition 1, because it is of the form $ba\alpha\, a^* (ab\alpha)^+ b$ (which is exactly the form (1) — modulo swapping a/b) with $\alpha = aabaaabaa$, the star * being repeated zero times, and the plus $^+$ being repeated just once. We now aim to compress v with function C. It can be easily seen that the word could be written in the form

$v = b(aaab)(aaab)(aaab)(aab)(aaab)(aab)$, hence we need to consider the morphism $M_{a,2}^{-1} : \begin{cases} aaab & \mapsto a \\ aab & \mapsto b \end{cases}$, and by the compression we obtain a word $v_{a,2}^{-1} = baaabab$. Let us notice that $v_{a,2}^{-1}$ is dual to the word $w = abbbaba$ (see Example 1),

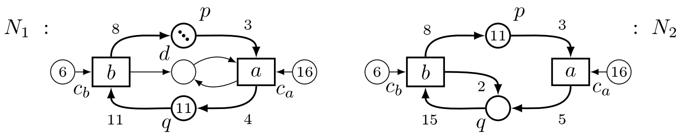


Fig. 12. N_1 solves the prefix *baaabaaabaaabaabaaabaa* and N_2 solves the suffix *aaabaaabaaabaabaaabaab* of the word $v = baaabaaabaaabaabaaabaab$.

Table 2
Comparison of the time (in nanoseconds) of different algorithms.

Word length	ABSolve	Pattern-matching algorithm	Java regular expressions
4	595.673	608.748	1 082.901
5	1 710.846	1 646.190	5 107.418
6	4 941.525	3 802.029	17 019.323
7	21 642.105	9 966.394	48 985.080
8	70 047.347	39 403.468	125 728.414
9	177 239.541	105 482.837	292 125.735
10	437 897.525	227 730.285	660 942.248
15	23 218 586.823	6 009 431.249	28 453 021.721
20	1 198 560 013.750	145 795 309.046	1 205 013 146.250

modulo swapping a/b, hence it is a minimal unsolvable word. Function C cannot be applied to $w = C(v)$, which is consistent with the fact that $w \in \mathcal{BE}$.

Moreover, starting with the word $w = abbbaba$, together with Petri nets solving its proper prefix and suffix (see Fig. 9) and applying the morphism $M_{b,2}$: $\begin{cases} a & \mapsto bba \\ b & \mapsto bbba \end{cases}$ we obtain the word $w_{b,2} = ab\,bbabbbabb\,ba\,bbabbbabb\,a$ which is dual to v modulo swapping a/b. By the previous considerations we can easily construct Petri nets solving the maximal proper prefix and the maximal proper suffix of $w_{b,2}$, hence, by swapping letters we can obtain Petri nets for a proper prefix and a proper suffix of v. Such nets are depicted in Fig. 12. Now we can state that the word v is not only unsolvable, but also minimal with that property.

5.3. Algorithm for checking unsolvability

The classification of minimal unsolvable words presented in Sections 3 and 4 leads to an efficient algorithm for verifying solvability/unsolvability of a binary word. By Definition 2 all non-extendable words are of the form (Ia) ab^xab^ya or (Ib) ba^xba^yb, where $x > y+2$, $y \geq 0$, and by Definitions 1 and 3 all extendable words (including base extendable ones) are of the form (IIa) $abw(baw)^ka$ or (IIb) $baw(abw)^kb$, where $k \geq 1$ and $w \in \{a, b\}^*$.

Recall that a word $v \in \{a, b\}^*$ containing a minimal unsolvable word as a factor is also unsolvable. Moreover, due to Theorem 1, v is unsolvable if it contains at least one of the patterns (Ia) (Ib), (IIa) or (IIb). Therefore, checking the solvability of a binary word can be reduced to a pattern-matching problem.

The algorithm described below takes a binary word v as an input; it returns true if v is solvable and false otherwise (i.e. any of the above mentioned patterns was found inside v).

As the first step we search for the patterns (Ia) and (Ib). We scan the input word from left to right comparing the sizes of the two blocks of consecutive b's between any three consecutive occurrences of a and the sizes of the two blocks of consecutive a's between any three consecutive occurrences of b. This can be done in $O(n)$ time and $O(1)$ space.

The second step is to search for the patterns (IIa) and (IIb). It utilises the Knuth–Morris–Pratt failure function called also the border table (see [6]). For any position i in v it contains the length of the longest factor u, which is at the same time a proper prefix and a proper suffix of $v[1..i]$. Such a factor is called a border of $v[1..i]$. For the relation between borders and periods of a word see for instance [7].

The search for the patterns (IIa) and (IIb) is performed as follows. For any possible pair of letters $v[i..i+1] = ab$ ($v[i..i+1] = ba$ respectively) we temporarily swap $v[i]$ with $v[i+1]$ and then build the border table for the suffix of v starting at position i. After discovering a repetition $v[i..j]$ (i.e. difference between j and the length of the border divides $j-i$) we check whether it is followed by a (b respectively) and report the occurrence of the pattern if needed.

The border table for a single suffix of the input word v can be constructed in $O(n)$ time and $O(n)$ space (see [6]). We have to process at most $O(n)$ suffixes of v, therefore the second step and the whole algorithm run in $O(n^2)$ time and $O(n)$ space, the implementation in C++ is available at [10].

6. Experimental results

In Table 2 we can see some experimental results of checking binary words for PN-solvability with different algorithms. Here we compare the algorithm ABSolve [3], which is based on other characterisation of solvability, with the algorithms

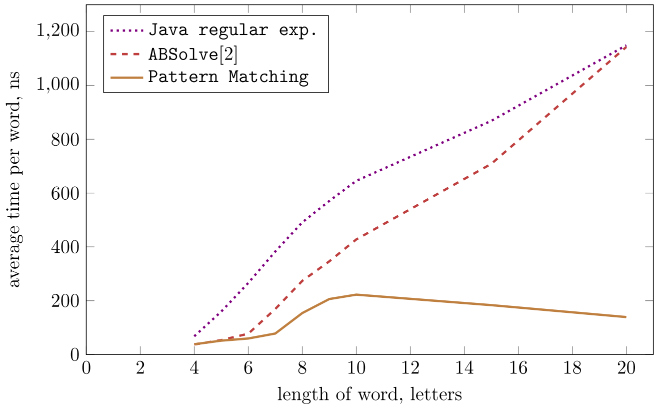


Fig. 13. Time per word of a given length to check its unsolvability.

that look for a pattern (1) as a subsequence of the word under consideration. We use the results of this search for a subsequence with the Pattern-matching algorithm described in the previous section, and with an inbuilt algorithm for regular expressions in Java. The score given in a corresponding cell of the table means time (in nanoseconds) for checking of all possible binary sequences of a fixed length for solvability. These data are normalised in Fig. 13, where one can see an average time (in nanoseconds) to check PN-solvability of a binary sequence of a fixed length. We can see that, while being pretty close in time for short sequences, the Pattern-matching algorithm essentially overtakes the ABSolve algorithm for longer sequences. Both specialised methods perform better than using inbuilt regular expressions. However, the results for longer words are almost equal for ABSolve and Java regular expressions. All the implementations are done in Java 8, and we let them run on the same machine. The data in Table 2, respectively in Fig. 13, are mean values after 10 runs of each single experiment.

7. Conclusion

In this paper we study the class of binary words which cannot be generated by any injectively-labelled Petri net, and which are minimal with that property. We examine in detail all possible shapes of such words, obtaining extended regular expressions for them. The presented classification of minimal unsolvable words results in the construction of a pattern-matching based algorithm for checking the solvability/unsolvability of binary words. Moreover, we introduce the extension and compression functions, which can be the foundation of a fixed-point procedure for the generation of the set of all minimal unsolvable binary words. The non-extendable and base extendable words are defined by simple parametrised formulas (see Definitions 1 and 2). Choosing all possible values of the parameters j and k we can generate all non-extendable and base extendable words of a given length. Then by using recursive calls of extension operation and compression function we can generate all extendable words of a given length.

Acknowledgement

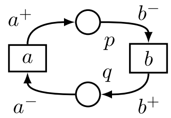
We would like to thank Jonas Prellberg for the benchmarks of the Pattern matching algorithm described in the paper.

Appendix

Lemma 8 (*Minimality of Extendable Words*)**.** *If $w \in \mathcal{E}$, then w is minimal unsolvable.*

Proof (*Complete*)**.** Let $w \in \mathcal{E}$ be an arbitrary extendable word. By Lemma 7, w is unsolvable. Let us now check its minimality. According to Definition 4, there is a sequence $w_0, w_1, \ldots, w_r$ such that $w_0 \in \mathcal{BE}$, $w_j \in \mathcal{E}$ and $w_j \in E(w_{j-1})$ for $1 \leq j \leq r$, and $w_r = w$. We will argue by induction on the length r of this sequence. From the previous consideration we know that the base extendable word w_0 is minimal unsolvable, and there are Petri nets N_1^0 and N_2^0 with core part and additional part, which are solutions for the maximal proper prefix and the maximal proper suffix of w_0. Assume now, that for every $1 \leq j \leq r-1$, there are Petri nets N_1^j and N_2^j which are solutions for the maximal proper prefix and the maximal proper suffix of w_j, and which have been obtained from N_1^{j-1} and N_2^{j-1}, respectively, with the appropriate transformation from Table 1 (this transformation is uniquely defined by the particular morphism $M_{x,i}$ with $x \in \{a, b\}$, that has been used to derive w_j from w_{j-1}). We now prove, that knowing morphism $M_{x,i}$ with $x \in \{a, b\}$, which is used for producing w_r from w_{r-1}, and using the corresponding transformation, Petri nets N_1^r and N_2^r, which are derivatives of N_1^{r-1} and N_2^{r-1}, are indeed solutions for the maximal proper prefix and the maximal proper suffix of w_r.

$\widetilde{N}_1^{r-1}$:



$M_0 \begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} b^- \\ a^- \end{pmatrix}$;

Fig. 14. Core part of Petri net N_1^{r-1} solving maximal proper prefix of w_{r-1}.

Let us consider the case of producing N_1^r from N_1^{r-1}, when $w_{r-1} = aw'a$ and $w_r = aM_{b,i}(w'a)$, for some $i \geq 1$. Having the core part $\widetilde{N}_1^{r-1}$ (see Fig. 14) of the solution N_1^{r-1} for aw', with the transformations of the arc weights and the new initial marking

$$\begin{aligned} &a^+ \longmapsto \quad a^+ + i\cdot(a^+ + b^-) \quad b^- \longmapsto \quad a^+ + b^- \\ &a^- \longmapsto \quad a^- + i\cdot(a^- + b^+) \quad b^+ \longmapsto \quad a^- + b^+ \\ &M_0\begin{pmatrix} p \\ q \end{pmatrix} = \quad \begin{pmatrix} a^+ + b^- \\ a^- + i\cdot(a^- + b^+) \end{pmatrix} \end{aligned} \tag{7}$$

for morphism $M_{b,i}$ we can construct the new core part $\widetilde{N}_1^r$ for aw'', where $aw''a = w_r$. Let us now check that the constructed core part implements the internal part of aw''. We shall show that place p prevents all undesirable b's inside aw'' and enables all b's that are to occur, and the similar for place q and transition a. Since we have used morphism $M_{b,i}$ for the extension operation, we have a special form of extension $w_r = ab^{x_1}ab^{x_2}a\ldots ab^{x_n}a \in E(w_{r-1})$, with $x_j \in \{i, i+1\}$. By contraposition, assume p disables a transition b that must occur at state s in $aw'' = ab^{x_1}a\ldots ab^{x_k-m}\,|_s\, b^m a\ldots$, where s is the leftmost state in aw'' with this property, and $k \geq 1$. By (7), each firing of a brings $a^+ + i\cdot(a^+ + b^-)$ tokens on place p, and b consumes $a^+ + b^-$ tokens on its every occurrence. Hence p can only disable the last but one b in a group b^{i+1}, i.e. $x_k = i+1$ and $m = 1$. Assume, there are l groups of b^{i+1} in $ab^{x_1}a\ldots ab^{x_k-1}\,|_s$. By the initial assumption, marking of p at state s is less than the weight of the arc from p to b, i.e.

$$\begin{aligned} M_s(p) = (a^+ + b^-) + k\cdot(a^+ +\ i\cdot(a^+ + b^-)) - \\ - k\cdot i\cdot(a^+ + b^-) - l\cdot(a^+ + b^-) < a^+ + b^- \quad \Longleftrightarrow \\ \Longleftrightarrow \quad (k-l)\cdot a^+ + (1-l)\cdot b^- < b^- \end{aligned}$$

On the other hand, every sequence $b^{i+1}a$ in w_r corresponds to b in w_{r-1}, and every sequence b^ia corresponds to a in w_{r-1}. Hence, the marking of the place p in the net $\widetilde{N}_1^{r-1}$, before applying transformation (7), at the state s_1 of $w_{r-1} = a\ldots|_{s_1} b\ldots a$, where the b right after s_1 corresponds to the block $b^{x_k}a$ in w_r, is $M_{s_1}(p) = b^- + (k-l)\cdot a^+ - l\cdot b^-$. Therefore, $M_{s_1}(p) < b^-$, which contradicts the assumption that the net $\widetilde{N}_1^{r-1}$ solves the word aw'. Thus, place p after transformation (7), allows all necessary occurrences of b. Notice here that place p allows b to fire initially also.

We now have to show that p disables b at all states where a has to occur, except the initial one. Suppose a contrary, i.e. there is a state s in $aw'' = ab^{x_1}a\ldots ab^{x_k}\ |_s\ a\ldots ab^{x_n}$, with $k \geq 1$, such that $M_s(p) \geq a^+ + b^-$. Without the loss of generality, let s be the leftmost (except the initial) state with that property. Assume $x_k = i+1$. Consider state s' in $ab^{x_1}a\ldots\ |_{s'}\ ab^{x_k}\ |_s\ a\ldots ab^{x_n}a$. Then

$$\begin{aligned} M_s(p) = M_{s'}(p) + a^+ + i\cdot(a^+ + b^-) - (i+1)\cdot(a^+ + b^-) \geq a^+ + b^- \quad \Longleftrightarrow \\ \Longleftrightarrow \quad M_{s'}(p) \geq b^- + (a^+ + b^-). \end{aligned}$$

The last inequality means that b is not separated at state s'. If $k = 1$, then, by (7), $M_{s'}(p) = M(p) = a^+ + b^-$, which contradicts the last inequality. If $k > 1$, then we get a contradiction to the choice of s. Hence, $x_k = i$. Let l be the number of blocks b^{i+1} in $ab^{x_1}a\ldots ab^{x_k}|_s$. Then

$$\begin{aligned} M_s(p) = (a^+ + b^-) + k\cdot(a^+ + i\cdot(a^+ + b^-)) - \\ -\ k\cdot i\cdot(a^+ + b^-) - l\cdot(a^+ + b^-) \geq a^+ + b^- \quad \Longleftrightarrow \\ \Longleftrightarrow \quad (k-l)\cdot a^+ + (1-l)\cdot b^- \geq b^-. \end{aligned}$$

Since w_r has been obtained using morphism $M_{b,i}$, sequence $b^{x_k}a$ corresponds to letter a in w_{r-1}. Therefore, in $w_{r-1} = a\ldots|_{s_1}a\ldots$, where s_1 fits the state right before $b^{x_k}a$ in w_r, we have b is not separated at state s_1, which contradicts the assumption that $\widetilde{N}_1^{r-1}$ solves aw'. Thus, in the net $\widetilde{N}_1^r$ that was derived from $\widetilde{N}_1^{r-1}$ by (7), p disables b whenever and only if it is necessary inside aw''.

For the separation of b at the initial marking, one can construct additional place p_1, having 0 tokens on it initially, and being a pure input place for transition b and pure output place for transition a with unit arc weights. For restricting the total number of occurrences of b, it is enough to construct place p_2 with $\#_b(w^r)$ tokens on it initially, which is a pure input place for b with the arc weight equal to 1.

Let us now consider place q and transition a. First we will show that q allows a to fire at each state where this is necessary. It is clear that initially q enables a. By contraposition, assume there is a state s in $aw'' = ab^{x_1}ab^{x_2}a\dots ab^{x_k}\mid_s ab^{x_{k+1}}a\dots$, with $k \geq 1$, such that q disables a at s. Due to (7), each firing of b brings $a^- + b^+$ tokens on q. Hence $x_k = i$. Suppose there are l blocks b^{i+1} in $ab^{x_1}a\dots ab^{x_k}\mid_s$. Then, we have

$$\begin{aligned}M_s(q) \;=\; & a^- + i\cdot(a^- + b^+) - k\cdot(a^- + i\cdot(a^- + b^+)) + \\ & + k\cdot i\cdot(a^- + b^+) + l\cdot(a^- + b^+) < a^- + i\cdot(a^- + b^+) \iff \\ & \iff a^- + l\cdot b^+ - (k-l)\cdot a^- < a^-\end{aligned}$$

Due to the fact that $aw''a$ has been obtained from $aw'a$ using morphism $M_{b,i}$, block $b^{x_k}a$ corresponds to a right after the state s' in $w_{r-1} = a\dots\mid_{s_1} a\dots$. The last inequality means $M_{s_1}(q) < a^-$ which contradicts the assumption that $\widetilde{N}_1^{r-1}$ solves the word aw'. Thus, place q after the transformation (7) allows each mandatory firing of a.

We now demonstrate that q disables a at every place, where b has to occur. By contraposition, suppose there is a state s in $aw'' = ab^{x_1}a\dots ab^{x_k-m}\mid_s b^m a\dots$, with $k, m > 0$, at which a is enabled by place q. Without the loss of generality, let s be the leftmost state in aw'' with that property. Due to the initial marking of q provided in (7), $k > 1$.

Hence, for state s and place q we have

$$\begin{aligned}M_s(q) \;=\; & a^- + i\cdot(a^- + b^+) - k\cdot(a^- + i\cdot(a^- + b^+)) \\ & + (x_1 + \dots + x_k - m)\cdot(a^- + b^+) \geq a^- + i\cdot(a^- + b^+)\end{aligned}$$

If $x_{k-1} = i$, then

$$\begin{aligned}M_{s_1}(q) \;=\; & a^- + i\cdot(a^- + b^+) - (k-1)\cdot(a^- + i\cdot(a^- + b^+)) \\ & + (x_1 + \dots + x_{k-1} - m)\cdot(a^- + b^+) \geq a^- + i\cdot(a^- + b^+) + a^-\end{aligned}$$

implying that a is enabled by q at state s_1 in $aw'' = ab^{x_1}a\dots ab^{x_{k-1}-m}\mid_{s_1} b^m a\dots$, which contradicts the choice of s. Then, $x_{k-1} = i+1$. This means, the block $b^{x_k}a$ corresponds to letter b in $aw'a$, and state s in aw'' corresponds to the state s_0 in $aw' = a\dots\mid_{s_0} b\dots$. On the other hand,

$$\begin{aligned}M_s(q) \;&= a^- + i\cdot(a^- + b^+) - k\cdot(a^- + i\cdot(a^- + b^+)) \\ & + (x_1 + \dots + x_k - m)\cdot(a^- + b^+) \geq a^- + i\cdot(a^- + b^+) \iff \\ & \iff a^- - (k-l)\cdot a^- + l\cdot b^+ \geq a^- + (m-1)\cdot(a^- + b^+)\end{aligned}$$

Since $m \geq 1$, we have $M_{s_0}(q) \geq a^-$ in the net $\widetilde{N}_1^{r-1}$, implying that a is enabled at state s_0. This contradicts the fact that $\widetilde{N}_1^{r-1}$ solves aw'. Thus, q disables a at every state in aw'' where b has to occur.

Redundant occurrence of b at the very beginning of aw'', that is not handled by p, can be easily restricted by place p_1, having zero tokens initially, the arc weight from a to p_1 is $i+1$ and the arc weight from p_1 to b is 1. The length of execution performed by $\widetilde{N}_1^r$ can be simply restricted with letter-counting place, having no inputs and a single output for each transition, and the initial number of tokens equal to the length of aw''. As a result, we have Petri net N_1^r, solving exactly aw'', with a core and additional part.

The other cases from Table 1 can be checked analogously. □

References

[1] E. Badouel, L. Bernardinello, Ph. Darondeau, Petri Net Synthesis, Springer, 2009.
[2] K. Barylska, E. Best, E. Erofeev, Ł. Mikulski, M. Piątkowski, Conditions for Petri net solvable binary words, in: T Petri Nets and Other Models of Concurrency, Vol. 11, 2016, pp. 137–159.
[3] E. Best, E. Erofeev, U. Schlachter, H. Wimmel, Characterising Petri net solvable binary words, in: Proc. of ATPN, 2016.
[4] B. Caillaud, 2002, http://www.irisa.fr/s4/tools/synet.
[5] C. Câmpeanu, K. Salomaa, S. Yu, A formal study of practical regular expressions, Internat. J. Found Comput. Sci. 6 (13) (2003).
[6] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, third ed., The MIT Press, 2009.
[7] M. Crochemore, L. Ilie, W. Rytter, Repetitions in strings: Algorithms and combinatorics, Theoret. Comput. Sci. 410 (50) (2009).
[8] M. Droste, R.M. Shortt, From Petri nets to automata with concurrency, Appl. Categ. Structures 10 (2) (2002).
[9] E. Erofeev, K. Barylska, Ł. Mikulski, M. Piątkowski, Generating all minimal Petri net unsolvable binary words, in: Proceeding of 20th Prague Stringology Conference, 2016.
[10] M. Piątkowski, et al., 2015, http://folco.mat.umk.pl/unsolvable-words.
[11] W. Reisig, UnderstandIng Petri Nets — Modeling Techniques, Analysis Methods, Case Studies, Springer, 2013.
[12] U. Schlachter, et al., 2013, http://github.com/CvO-Theory/apt.

Please cite this article as: E. Erofeev, K. Barylska, Łu. Mikulski et al., Generating all minimal petri net unsolvable binary words, Discrete Applied Mathematics (2019), https://doi.org/10.1016/j.dam.2019.04.023.