Information and Computation 209 (2011) 568–579

Contents lists available at ScienceDirect

Information and Computation

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

Synchronizing automata with finitely many minimal synchronizing words

Elena V. Pribavkina [a,*,1], Emanuele Rodaro [b,2]

[a] Ural State University, 620083 Ekaterinburg, Russia
[b] Politecnico di Milano, 20133 Milano, Italy

ARTICLE INFO

Article history:
Available online 18 November 2010

Keywords:
Synchronizing automata
Minimal synchronizing words
Co-NP-hard problems

ABSTRACT

A synchronizing word for a given synchronizing DFA is called *minimal* if none of its proper factors is synchronizing. We characterize the class of synchronizing automata having only finitely many minimal synchronizing words (the class of such automata is denoted by **FG**). Using this characterization we prove that any such automaton possesses a synchronizing word of length at most $3n - 5$. We also prove that checking whether a given DFA $\mathscr{A}$ is in **FG** is co-NP-hard and provide an algorithm for this problem which is exponential in the number of states $\mathscr{A}$.

© 2010 Elsevier Inc. All rights reserved.

1. Introduction

A *synchronizing* automaton $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ is a deterministic and complete finite-state automaton (DFA) possessing a *synchronizing* word, that is a word w which takes all states of $\mathscr{A}$ to a particular one: $\delta(q, w) = \delta(q', w)$ for all $q, q' \in Q$. By $\mathrm{Syn}_{\mathscr{A}}$ we denote the language of all words synchronizing $\mathscr{A}$.

Over the past 40 years synchronizing automata and especially shortest synchronizing words have been widely studied, motivated mostly by the famous Černý conjecture [8] which states that any n-state synchronizing automaton possesses a synchronizing word of length at most $(n-1)^2$. This conjecture has been proved for a large number of classes of synchronizing automata, nevertheless in general it remains one of the most longstanding open problems in automata theory. For more details see the surveys [3,7,9].

In this paper we deal with *minimal synchronizing* words which in some sense generalize the notion of a shortest synchronizing word. Namely, a synchronizing word is called *minimal* if none of its proper factors is synchronizing. It is obvious that the language $\mathrm{Syn}_{\mathscr{A}}$ of all synchronizing words is a two-sided ideal generated by the language $\mathrm{Syn}^{min}_{\mathscr{A}}$ of all minimal synchronizing words: $\mathrm{Syn}_{\mathscr{A}} = \Sigma^* \mathrm{Syn}^{min}_{\mathscr{A}} \Sigma^*$. Thus it is rather natural to consider the class of synchronizing automata whose language of synchronizing words is a finitely generated ideal. The class of such automata is denoted by **FG**. In Section 3 we give a characterization of this class. Moreover using the characterization we prove in Section 5 that the shortest synchronizing word for such automata has length at most $3n - 5$.

Example 1. Let $\Sigma = \{a, b\}$ and consider the minimal automaton $\mathscr{A}$ recognizing the language $\mathrm{L} = \Sigma^* aba \Sigma^*$ (see Fig. 1.)

It is easy to see that $\mathscr{A}$ is synchronizing, and $\mathrm{Syn}_{\mathscr{A}} = \mathrm{L}$, thus $\mathrm{Syn}^{min}_{\mathscr{A}} = \{aba\}$, so $\mathscr{A} \in$ **FG**. Analogously for any $w \in \Sigma^*$ the minimal automaton recognizing $\Sigma^* w \Sigma^*$ is in **FG**. Moreover it is well-known that this automaton has $n = |w| + 1$ states, hence its minimal synchronizing word is of length $n - 1$. Clearly, in general, the minimal automaton recognizing the language $\Sigma^* M \Sigma^*$ for a finite language M belongs to **FG**.

* Corresponding author.
E-mail addresses: elena.pribavkina@usu.ru (E.V. Pribavkina), emanuele.rodaro@fc.up.pt (E. Rodaro).

[1] Author acknowledges support from the Federal Education Agency of Russia, Grant 2.1.1/3537, and from the Russian Foundation for Basic Research, Grants 09-01-12142 and 10-01-00793.
[2] This research was initiated with the partial support of GNSAGA during the visit of the author to the Ural State University, Russia.

0890-5401/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.ic.2010.11.020

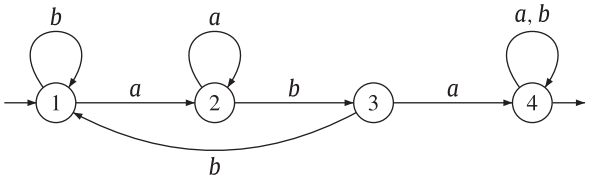

Fig. 1. Automaton $\mathscr{A}$ recognizing $L = \Sigma^* aba \Sigma^*$.

Another natural question arising in this context is whether testing $\mathscr{A} \in$ **FG** is decidable. An easy argument shows that the answer is "yes". Indeed, in general, for a given language $\mathcal{L}$, we can consider the set $\mathcal{L}^{min}$ of its minimal words. It is the set of words from $\mathcal{L}$ such that none of its proper factors belongs to $\mathcal{L}$. It is not hard to see that $\mathcal{L}^{min} = \mathcal{L} \setminus (\Sigma^* \mathcal{L} \Sigma^+ \cup \Sigma^+ \mathcal{L} \Sigma^*)$. In particular, if $\mathcal{L}$ is a two-sided ideal this expression reduces to:

$$\mathcal{L}^{min} = \mathcal{L} \setminus (\mathcal{L}\Sigma \cup \Sigma\mathcal{L}) = (\mathcal{L} \setminus \mathcal{L}\Sigma) \setminus \Sigma\mathcal{L}. \tag{1}$$

Observe that if $\mathcal{L}$ is a regular language then clearly $\mathcal{L}^{min}$ is also a regular language. If $\mathcal{L}$ is represented by an n-state DFA $\mathcal{A}$, then the language $(\mathcal{L} \setminus \mathcal{L}\Sigma)$ is recognized by a DFA with $n+1$ states obtained from $\mathcal{A}$ by adding a sink state (if there were no such states in $\mathcal{A}$) and redirecting to the sink state all the transitions outgoing from final states. The language $\Sigma\mathcal{L}$ is recognized by an $(n+1)$-state DFA obtained from $\mathcal{A}$ by adding a new initial state and transitions labeled by all the letters from this state to the old initial one. Then it is easy to see that the language $\mathcal{L}^{min}$ can be recognized by an automaton with $O(n^2)$ states, and it is easy to see that checking finiteness of $\mathcal{L}^{min}$ takes $O(n^4)$ operations. The language of synchronizing words of a given automaton $\mathscr{A}$ is well known to be regular (it is recognized by the power automaton $\mathcal{P}(\mathscr{A})$ with Q as an initial state and singletons as terminal ones). Thus the language $\mathrm{Syn}_{\mathscr{A}}$ of an n-state synchronizing automaton $\mathscr{A}$ is recognized by an automaton with at most 2^n states. Hence by (1) the language $\mathrm{Syn}^{min}_{\mathscr{A}}$ is regular and checking whether $\mathscr{A} \in$ **FG** takes $O(2^{4n})$ operations. Therefore one may ask whether checking finiteness of the language of minimal synchronizing words is indeed a hard task and if there are better algorithms than the naive one. We formally state the FINITENESS problem:

- *Input*: A synchronizing DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$.
- *Question*: Is the language $\mathrm{Syn}^{min}_{\mathscr{A}}$ finite?

Our characterization gives rise to another algorithm for solving FINITENESS which is a slight improvement of the naive one. It is discussed in Section 3. Furthermore in Section 7 we show that FINITENESS is co-NP-hard. So the problem is not likely to have a polynomial time algorithm. Gawrychowski [1] has shown that this problem is in PSPACE, but we are not aware whether it belongs to some lower complexity class.

2. Preliminaries

We fix a synchronizing DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$. The action of the transition function δ can naturally be extended to the free monoid Σ^*. This extension will still be denoted by δ. For convenience for each $v \in \Sigma^*$ and $q \in Q$ we will write $q \cdot v = \delta(q, v)$ and put $S \cdot v = \{q \cdot v \mid q \in S\}$ for any $S \subseteq Q$. A subset S of Q is called *reachable* if there is a word $v \in \Sigma^*$ with $S = Q \cdot v$. Given a subset S of Q by $\mathrm{Fix}(S)$ we denote the set of all words *fixing* S:

$$\mathrm{Fix}(S) = \{w \in \Sigma^* \mid S \cdot w = S\}.$$

By $\mathrm{Syn}(S)$ we denote the set of all words bringing S to a singleton:

$$\mathrm{Syn}(S) = \{w \in \Sigma^* \mid |S \cdot w| = 1\}.$$

In this notations we have $\mathrm{Syn}_{\mathscr{A}} = \mathrm{Syn}(Q)$. Besides $\mathrm{Syn}_{\mathscr{A}}$ is contained in $\mathrm{Syn}(S)$ for any S.

Lemma 1. *Given a word $w \in \Sigma^*$ there exists an integer $\beta \geq 0$ such that the set $m(w) = Q \cdot w^\beta$ is fixed by w. Moreover $m(w)$ is the largest subset of Q with this property.*

Proof. Consider the sets $Q \cdot w^\alpha \subseteq Q$ for any $\alpha \geq 0$. Since the number of subsets of Q is finite, there are $\beta \geq 0$ and $\gamma > 0$ such that $Q \cdot w^\beta = Q \cdot w^{\beta+\gamma}$. It is easy to see that

$$Q \cdot w^\beta \supseteq Q \cdot w^{\beta+1} \supseteq \cdots \supseteq Q \cdot w^{\beta+\gamma} = Q \cdot w^\beta.$$

Hence all inclusions are in fact equalities and in particular $Q \cdot w^{\beta+1} = Q \cdot w^{\beta}$, so the set $Q \cdot w^{\beta}$ is fixed by the word w. On the other hand take any $S \subseteq Q$ fixed by w, then applying w we obtain

$$S = S \cdot w \subseteq Q \cdot w, \ldots, S = S \cdot w^{\beta} \subseteq Q \cdot w^{\beta},$$

so $m(w)$ is the largest subset fixed by the word w. □

Given a word w, the subset $m(w)$ of Q from the previous Lemma is called *the maximal fixed set with respect to w*.
Let $k(w)$ be the least integer with the property $Q \cdot w^{k(w)} = m(w)$. Then we have the following

Lemma 2. *Given a word* $w \in \Sigma^*$

$$k(w) \leq |Q| - |m(w)|.$$

Proof. We have $|Q \cdot w^{\beta+1}| < |Q \cdot w^{\beta}|$ for any $0 \leq \beta < k(w)$. Indeed, suppose $|Q \cdot w^{\beta+1}| = |Q \cdot w^{\beta}|$ for some $0 \leq \beta < k(w)$. Since $Q \cdot w^{\beta+1} \subseteq Q \cdot w^{\beta}$ we have $Q \cdot w^{\beta+1} = Q \cdot w^{\beta}$, hence $Q \cdot w^{\beta} \subseteq m(w)$. On the other hand, $m(w) = Q \cdot w^{k(w)} \subseteq Q \cdot w^{\beta}$, thus $m(w) = Q \cdot w^{\beta}$, which is a contradiction with the choice of $k(w)$. Therefore $|Q| > |Q \cdot w| > \cdots > |Q \cdot w^{k(w)}|$, and $|Q \cdot w^{k(w)}| \leq |Q| - k(w)$, hence $k(w) \leq |Q| - |m(w)|$. □

Lemma 3. *Given a word* $w \in \Sigma^*$ *and* $\alpha \in \mathbb{N}$

$$m(w^{\alpha}) = m(w).$$

Proof. Obviously $w^{\alpha} \in \mathrm{Fix}(m(w))$ for any $\alpha \in \mathbb{N}$, hence $m(w) \subseteq m(w^{\alpha})$. Conversely, by Lemma 1 $m(w^{\alpha}) = Q \cdot w^{\alpha\beta}$ for some $\beta \geq 0$. Since $m(w^{\alpha})$ is fixed by w^{α}, we have $|m(w^{\alpha}) \cdot w^{\alpha}| = |m(w^{\alpha})|$ in particular; thus $|m(w^{\alpha}) \cdot w| = |m(w^{\alpha})|$ and hence $|Q \cdot w^{\alpha\beta+1}| = |Q \cdot w^{\alpha\beta}|$. On the other hand, $Q \cdot w^{\alpha\beta+1} \subseteq Q \cdot w^{\alpha\beta}$, thus $Q \cdot w^{\alpha\beta+1} = Q \cdot w^{\alpha\beta}$, so $m(w^{\alpha})$ is fixed by w as well, and $m(w^{\alpha}) \subseteq m(w)$. Therefore $m(w^{\alpha}) = m(w)$. □

3. Characterization of the class FG

In this section we characterize synchronizing automata having only finitely many minimal synchronizing words in terms of properties of their power automata.

Theorem 1. *Given a synchronizing DFA* $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ *the following are equivalent:*

(i) $\mathscr{A} \in \boldsymbol{FG}$
(ii) *for any reachable subset* $S \subseteq Q$ *such that* $1 < |S| < |Q|$, *for each* $w \in \mathrm{Fix}(S)$

$$\mathrm{Syn}(S) = \mathrm{Syn}(m(w)).$$

Proof. $(i) \Rightarrow (ii)$. Let S be a reachable subset of Q with $1 < |S| < |Q|$. If $\mathrm{Fix}(S) = \varnothing$ then there is nothing to prove, so we can assume that $\mathrm{Fix}(S) \neq \varnothing$ and take an arbitrary $w \in \mathrm{Fix}(S)$. Note that the inclusion $\mathrm{Syn}(m(w)) \subseteq \mathrm{Syn}(S)$ is always true. Indeed, the set S is contained in the maximal fixed set $m(w)$, and all the words synchronizing the set $m(w)$ synchronize also S. In case $\mathrm{Syn}(S) = \mathrm{Syn}_{\mathscr{A}}$ we have $\mathrm{Syn}_{\mathscr{A}} \subseteq \mathrm{Syn}(m(w)) \subseteq \mathrm{Syn}_{\mathscr{A}}$, therefore $\mathrm{Syn}(S) = \mathrm{Syn}(m(w))$. Suppose now that $\mathrm{Syn}(S) \neq \mathrm{Syn}_{\mathscr{A}}$. It means that there exists a word v which brings the set S to a singleton, but does not synchronize the whole automaton. On the other hand, since the set S is reachable there exists a word u such that $S = Q \cdot u$. Consider now the infinite sequence of words $uw^i v$ for $i \geq 0$. Note that these words are synchronizing, and since the language $\mathrm{Syn}^{min}_{\mathscr{A}}$ is finite, among them there can be only a finite number of minimal synchronizing words. Since $Q \cdot uw^i = S$ for any $i \geq 0$, all minimal synchronizing words always contain a prefix of v as a suffix. Note also that if all the minimal synchronizing words in the given sequence would start with some suffix of u, then there would be an infinite sequence of minimal synchronizing words $u''_i w^i v'_i$ where $u = u'_i u''_i$, $v = v'_i v''_i$ for $i = 0, 1, 2 \ldots$ Therefore there exists a positive integer β such that the word $w^{\beta} v$ is synchronizing (not necessarily minimal, see Fig. 2).

By Lemma 1 we can choose β such that $Q \cdot w^{\beta} = m(w)$. Then the word v brings this set to a singleton. Thus if the language $\mathrm{Syn}^{min}_{\mathscr{A}}$ is finite then $\mathrm{Syn}(S) \subseteq \mathrm{Syn}(m(w))$. Since the opposite inclusion always holds true, we have $\mathrm{Syn}(S) = \mathrm{Syn}(m(w))$.

$(ii) \Rightarrow (i)$. Arguing by contradiction suppose that the condition (ii) holds, but the language $\mathrm{Syn}^{min}_{\mathscr{A}}$ is infinite. Since this language is regular, applying the pumping lemma we have that any long enough word in $\mathrm{Syn}^{min}_{\mathscr{A}}$ can be factorized as uwv so that $w \neq 1$ (where 1 denotes the empty word) and $uw^{\alpha} v$ is in $\mathrm{Syn}^{min}_{\mathscr{A}}$ for any integer $\alpha \geq 0$. If $u = 1$ then $w^{\alpha} v \in \mathrm{Syn}^{min}_{\mathscr{A}}$ for all $\alpha \geq 0$. In particular for $\alpha = 0$ we obtain that the word v is synchronizing, but this means that the words $w^{\alpha} v$ do not belong to $\mathrm{Syn}^{min}_{\mathscr{A}}$, a contradiction. Analogously we get a contradiction in case $v = 1$. Thus we may assume that both u and v are non-empty words. Consider sets $Q \cdot uw^{\alpha}$ for $\alpha \geq 0$. Since $uw^{\alpha} v$ is minimal synchronizing, we have $1 < |Q \cdot uw^{\alpha}| < |Q|$.

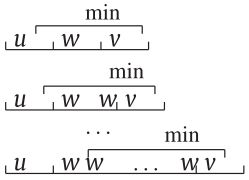

Fig. 2. Finite sequence of minimal synchronizing words.

Since the number of subsets of Q is finite there are integers $\alpha_0 \geq 0$ and $\gamma > 0$ such that $Q \cdot uw^{\alpha_0} = Q \cdot uw^{\alpha_0+\gamma}$. Put $S = Q \cdot uw^{\alpha_0}$. This set is fixed by w^γ, so by (ii) we have $\mathrm{Syn}(S) = \mathrm{Syn}(m(w^\gamma))$, with $m(w^\gamma) = Q \cdot (w^\gamma)^\beta$ for some $\beta \geq 0$. Since $v \in \mathrm{Syn}(S)$ we have $v \in \mathrm{Syn}(m(w^\gamma))$, so $|Q \cdot w^{\gamma\beta} v| = 1$. Therefore the word $w^{\gamma\beta} v$ is synchronizing and for $\alpha > \gamma\beta$ the word $uw^\alpha v$ does not belong to the language $\mathrm{Syn}^{min}_{\mathscr{A}}$, again a contradiction. Thus the language $\mathrm{Syn}^{min}_{\mathscr{A}}$ is finite. □

Corollary 1. *Let $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ be a synchronizing automaton such that there is a letter $a \in \Sigma$ with $Q \cdot a = Q$ and there are no letters $b \in \Sigma$ with $|Q \cdot b| = 1$. Then $\mathrm{Syn}^{min}_{\mathscr{A}}$ is infinite.*

Proof. Consider the shortest synchronizing word w for the automaton $\mathscr{A}$. Since no letter is synchronizing we have $w = xv$ with $x \in \Sigma$, $v \in \Sigma^+$, and $1 < |Q \cdot x| < Q$. Since a is a permutation letter it is clear that $a^\alpha \in \mathrm{Fix}(Q \cdot x)$ for some positive integer α. On the other hand $Q \cdot a^\alpha = Q$, hence $m(a^\alpha) = Q$. Note that $\mathrm{Syn}(Q \cdot x) \neq \mathrm{Syn}(Q)$ since v is in $\mathrm{Syn}(Q \cdot x)$ but not in $\mathrm{Syn}(Q) = \mathrm{Syn}_{\mathscr{A}}$ (otherwise it would be a shorter synchronizing word). Thus by Theorem 1 the language $\mathrm{Syn}^{min}_{\mathscr{A}}$ is infinite. □

4. Algorithm for the FINITENESS problem

Theorem 1 gives rise to the algorithm FinCheck slightly better than the straight-forward one presented in Section 1 for the FINITENESS problem. Actually Theorem 1 says that for a given reachable subset S of Q for all the words $w \in \mathrm{Fix}(S)$ we must check whether $\mathrm{Syn}(S) = \mathrm{Syn}(m(w))$. The problem is that the set $\mathrm{Fix}(S)$ might be infinite. On the other hand there are only finitely many subsets of Q of the form $m(w)$ for all $w \in \mathrm{Fix}(S)$. So for a given S we can check the property of Theorem 1 for all the subsets T of Q containing S with $\mathrm{Fix}(S) \cap \mathrm{Fix}(T) \neq \varnothing$. Indeed, obviously among such subsets there are those of the form $m(w)$ for all possible w. So if $\mathrm{Syn}(S) = \mathrm{Syn}(T)$ for all such T, then the condition of the theorem holds. If $\mathrm{Syn}(S) \neq \mathrm{Syn}(T)$ for some $T \supseteq S$, then it does not hold for $m(w)$ either, $w \in \mathrm{Fix}(S) \cap \mathrm{Fix}(T) \neq \varnothing$, since $T \subseteq m(w)$. Now we present the algorithm.

FinCheck($\mathscr{A}$):

```
1 From the DFA 𝒜 = ⟨Q, Σ, δ⟩ construct its power automaton 𝒫(𝒜)
  consisting only of subsets reachable from Q.
2 For each state S of 𝒫(𝒜) do:
 2.1 For each state T of 𝒫(𝒜) with S ⊆ T do:
    2.2 If Fix(T) ∩ Fix(S) ≠ ∅, then
        2.3 If Syn(T) ≠ Syn(S), then exit and return NO
3 Otherwise exit and return YES
```

We now analyze the cost of this algorithm. Let n be the number of states in an input automaton $\mathscr{A}$, then the subset construction can be performed in $O(2^n)$ operations. Step 2 consists of two nested loops and two checking procedures. It is easy to see that the language $\mathrm{Syn}(S)$ is recognized by the automaton $\mathscr{A}_{\mathrm{Syn}(S)} = \langle Q_{\mathrm{Syn}(S)}, \Sigma, \delta, S, \sigma\rangle$ with the state set $Q_{\mathrm{Syn}(S)}$ consisting of all subsets S' of Q with $|S'| \leq |S|$, with S as the initial state, and the set of final states σ consisting of singletons. The language $\mathrm{Fix}(S)$ is recognized by the automaton $\mathscr{A}_{\mathrm{Fix}(S)} = \langle Q_{\mathrm{Fix}(S)}, \Sigma, \delta, S, \{S\}\rangle$, where $Q_{\mathrm{Fix}(S)}$ consists of all $S' \subseteq Q$ with $|S'| = |S|$. Clearly $\mathscr{A}_{\mathrm{Syn}(S)}$ and $\mathscr{A}_{\mathrm{Fix}(S)}$ can be obtained from the power automaton $\mathcal{P}(\mathscr{A})$ using $O(2^n)$ operations.

Since $\mathrm{Syn}(T)$ is always contained in $\mathrm{Syn}(S)$, checking condition 2.3 is equivalent to checking emptiness of the language $\mathrm{Syn}(S) \setminus \mathrm{Syn}(T)$. Construction of an automaton recognizing this language takes

$$\sum_{j \leq |S|} \binom{n}{j} \sum_{k \leq |T|} \binom{n}{k}$$

operations in the worst case. Hence checking $\mathrm{Syn}(S) \setminus \mathrm{Syn}(T) = \varnothing$ takes

$$O\left(\sum_{j \leq |S|} \binom{n}{j} \sum_{k \leq |T|} \binom{n}{k}\right)$$

operations. Analogously, construction of an automaton recognizing the language $\mathrm{Fix}(T) \cap \mathrm{Fix}(S)$ takes in the worst case $\binom{n}{|S|}\binom{n}{|T|}$ operations, therefore checking emptiness of this language can be performed in time $O\left(\binom{n}{|S|}\binom{n}{|T|}\right)$. Thus procedures 2.2 and 2.3 take $O\left(\sum_{j\le|S|}\binom{n}{j}\sum_{k\le|T|}\binom{n}{k}\right)$ operations, and Step 2 can be performed in time

$$\sum_{S\subseteq Q}\sum_{S\subseteq T} O\left(\sum_{j\le|S|}\binom{n}{j}\sum_{k\le|T|}\binom{n}{k}\right) \le O\left(\sum_{S\subseteq Q}\sum_{S\subseteq T} 2^{2n}\right)$$
$$= O\left(2^{2n}\sum_{k=1}^{n}\binom{n}{k}2^{n-k}\right) = O(2^{2n}3^n).$$

Thus we get the following result:

Theorem 2. *Let n be the number of states of a synchronizing automaton $\mathscr{A}$. The algorithm* FinCheck$(\mathscr{A})$ *checks finiteness of the language* $\mathrm{Syn}^{min}_{\mathscr{A}}$ *in time* $O(12^n)$.

5. Upper bound on the length of shortest synchronizing words for the class FG

Using the characterization from the previous section here we prove a linear upper bound on the length of shortest synchronizing words for the class **FG**.

Recall that *the deficiency* of a word $w \in \Sigma^*$ with respect to a given automaton $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ is the difference $\mathrm{df}(w) = |Q| - |Q\cdot w|$. We make use of the following result from [5] (see also [2]):

Theorem 3. *Given a synchronizing automaton $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ and the words $u, v \in \Sigma^+$ such that $\mathrm{df}(u) = \mathrm{df}(v) = k > 1$, there exists a word τ, with $|\tau| \le k+1$, such that $\mathrm{df}(u\tau v) > k$.*

Theorem 4. *Let $\mathscr{A} = \langle Q, \Sigma, \delta\rangle \in$ **FG** with $|Q| = n$. Then $\mathscr{A}$ possesses a synchronizing word of length at most $3n-5$.*

Proof. Take any $a \in \Sigma$. By Corollary 1, if a acts as a permutation on the state set Q, then there exists a synchronizing letter b, so the statement of the theorem trivially holds. Thus we can assume that a is not a permutation letter, thus $Q\cdot a^{k(a)} = m(a) \subsetneq Q$. Suppose first that $|m(a)| = 1$. Then $1 = |m(a)| = |Q\cdot a^{k(a)}|$ and by Lemma 2, $k(a) \le n - |m(a)| = n-1 \le 3n-5$ for $n \ge 2$.

Now suppose that $|m(a)| > 1$, and consider non-singleton subsets of $m(a)$ reachable from $m(a)$:

$$\textsc{Reach}(a) = \{S \subseteq m(a) \mid S = m(a)\cdot u, u \in \Sigma^*, |S| > 1\}.$$

Obviously $m(a) \in \textsc{Reach}(a)$, so $\textsc{Reach}(a)$ is not empty. Also note that for any $S \in \textsc{Reach}(a)$ it holds

$$\mathrm{Syn}(S) = \mathrm{Syn}(m(a)). \tag{2}$$

Indeed, since $S \subseteq m(a)$ we have $a^\alpha \in \mathrm{Fix}(S)$ for some $\alpha \in \mathbb{N}$. Then since $\mathscr{A} \in$ **FG** applying Theorem 1 and Lemma 3 we get $\mathrm{Syn}(S) = \mathrm{Syn}(m(a^\alpha)) = \mathrm{Syn}(m(a))$. Now let $H = Q\cdot a^{k(a)}u$ be an element of $\textsc{Reach}(a)$ of minimal cardinality and let $k' = n - |H| = \mathrm{df}(a^{k(a)}u)$. Since $|H| > 1$ we have $k' \le n-2$. Since $\mathscr{A}$ is synchronizing, there exists a word of deficiency $n-1$, therefore by Theorem 3 there is a word τ with $|\tau| \le k'+1$ such that $\mathrm{df}(a^{k(a)}u\tau a^{k(a)}u) > k'$, i.e. $|Q\cdot a^{k(a)}u\tau a^{k(a)}u| < |H|$. Next we prove that the word $a^{k(a)}u\tau a^{k(a)}ua^{ka}$ is synchronizing. Indeed, suppose on the contrary that $|Q\cdot a^{k(a)}u\tau a^{k(a)}ua^{k(a)}| > 1$. It is easy to see that $Q\cdot a^{k(a)}u\tau a^{k(a)}ua^{k(a)} \subseteq m(a)$, hence $Q\cdot a^{k(a)}u\tau a^{k(a)}ua^{k(a)} \in \textsc{Reach}(a)$. However the inequality

$$|Q\cdot a^{k(a)}u\tau a^{k(a)}ua^{k(a)}| \le |Q\cdot a^{k(a)}u\tau a^{k(a)}u| < |H|$$

contradicts the choice of H. In fact even the word $a^{k(a)}u\tau a^{k(a)}$ is synchronizing. Indeed, consider the set $S = Q\cdot a^{k(a)}u\tau a^{k(a)}$. If $|S| > 1$ then $S \in \textsc{Reach}(a)$, hence $ua^{k(a)} \in \mathrm{Syn}(S) = \mathrm{Syn}(m(a))$, so $1 = |m(a)\cdot ua^{k(a)}| = |H\cdot a^{k(a)}|$. But by the choice of H we have $|H\cdot a^{k(a)}| = |H| > 1$, which is a contradiction.

Thus

$$\tau a^{k(a)} \in \mathrm{Syn}(Q\cdot a^{k(a)}u) = \mathrm{Syn}(H) = \mathrm{Syn}(m(a)) = \mathrm{Syn}(Q\cdot a^{k(a)}),$$

hence the word $a^{k(a)}\tau a^{k(a)}$ is synchronizing and $|a^{k(a)}\tau a^{k(a)}| \le 2k(a) + k' + 1 \le 2(n-2) + n - 1 = 3n-5$. □

Remark 1. Under conditions of the Theorem 4 if $k = \min_{a\in\Sigma} k(a)$ then there is a synchronizing word of length at most $2k+n-1$.

6. Maximal length of minimal synchronizing words

Since automata in **FG** have only finitely many minimal synchronizing words, it is rather natural to consider the length $\overline{\ell}(\mathscr{A})$ of the longest minimal synchronizing word. This quantity gives rise to the trivial upper bound $|\Sigma|^{\overline{\ell}(\mathscr{A})}$ for the number of elements in $\mathrm{Syn}^{min}_{\mathscr{A}}$. In this section we give an upper bound for $\overline{\ell}(\mathscr{A})$.

Theorem 5. *Let $\mathscr{A} \in$ **FG** and let N be the number of non-singleton states of its power automaton $\mathcal{P}(\mathscr{A})$ consisting of only reachable subsets. Then the length of any minimal synchronizing word is at most $N^2 - N + 1$.*

Proof. Let us show that any synchronizing word w for the automaton $\mathscr{A}$ of length greater than $N(N-1)+1$. For this purpose we consider the action of all non-empty prefixes $w[1\cdots i]$ of the word w on the state set Q:

$$Q_i = Q \cdot w[1\cdots i], \quad 1 \le i \le |w|.$$

If $Q_{i_0} = Q$ for some $1 \le i_0 < |w|$, then the word w is not minimal synchronizing, since its proper suffix $w[i_0+1\cdots|w|]$ is synchronizing. From the other hand $|Q_{|w|}| = |Q\cdot w| = 1$. If $|Q_{i_0}| = 1$ for some $1 \le i_0 < |w|$, then again the word w is not minimal synchronizing, since its proper prefix $w[1\cdots i_0]$ is synchronizing. Hence it remains to consider the case, in which the set Q does not appear among the subsets of the form Q_i, and all the subsets Q_i (except for $Q_{|w|}$) are not singletons. Then there can be only N different subsets of the form Q_i. From the other hand the number of such subsets equals $|w| > N(N-1)+1$. It is not hard to see that there exists a subset $S \subseteq Q$, $1 < |S| < |Q|$, appearing among the subsets Q_i at least $N+1$ times:

$$S = Q_{i_0} = Q_{i_1} = \cdots = Q_{i_m},\ m \ge N.$$

For $0 \le j \le m-1$ consider words $t_j = w[i_j+1\cdots i_{j+1}]$ and subsets $H_0 = Q$, $H_j = H_{j-1}\cdot t_{j-1}$. Note that there are more than N such subsets. Since $t_j \in \mathrm{Fix}(S)$ for all $0 \le j \le m-1$, we have $S \subseteq H_j$. Thus all the subsets of the form H_j are not singletons, therefore among them there can be at most N different subsets. So we have $H_k = H_\ell$ for some $0 \le k < \ell \le m$, and the word $u = t_k\cdots t_{\ell-1} \in \mathrm{Fix}(H_k)$. Besides, $u \in \mathrm{Fix}(S)$ as a product of words from $\mathrm{Fix}(S)$. Since $\mathscr{A} \in$ **FG**, by Theorem 1 we have $\mathrm{Syn}(S) = \mathrm{Syn}(m(u)) = \mathrm{Syn}(H_k)$. The word $v = w[i_\ell+1\cdots|w|] \in \mathrm{Syn}(S)$, hence $v \in \mathrm{Syn}(H_k)$. Thus the word $t_0t_1\cdots t_{\ell-1}v = w[i_0+1\cdots|w|]$ is a proper suffix of the word w and it synchronizes the automaton $\mathscr{A}$, so w is not minimal synchronizing. □

Obviously if n is the number of states of a finitely generated synchronizing automaton, then $N \le 2^n - n - 1$. Thus we get the following corollary:

Corollary 2. *Given a synchronizing automaton $\mathscr{A} \in$ **FG** with n states, $\overline{\ell}(\mathscr{A}) \le (2^n - n - 1)^2 - 2^n - n$.*

7. Co-NP-hardness

In this section we prove that the FINITENESS problem for a given DFA is co-NP-hard. To prove the result we introduce an auxiliary problem which we refer to as REACHABILITY. Formally:
Input: A DFA $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ and a subset $H \subseteq Q$.
Question: Is there a word $w \in \Sigma^*$ such that $Q\cdot w = H$?

The proof proceeds in two stages. First we show that any instance from a particular set I of instances of REACHABILITY can be polynomially reduced to an instance of the complement of FINITENESS. Next we complete the proof by polynomially reducing any instance of SAT to an instance of REACHABILITY belonging to I.

In our reductions we essentially make use of a particular class of automata, called *nilpotent* automata. This notion was introduced by Perles et al. in 1962 under the name of *definite table* [4]. Later such automata were studied by Rystsov in [6] in view of Černý's conjecture. In the present paper we use the definition from [6]. Namely, we say that a DFA $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ is *nilpotent* if there is a state $s \in Q$ and a positive integer n such for any word $w \in \Sigma^*$ of length at least n it holds $Q\cdot w = \{s\}$.

Obviously any nilpotent automaton is synchronizing with a *sink state*, i.e. the state fixed by all the letters of the alphabet (s in the definition). In the following lemma we state without proof some simple properties of nilpotent automata.

Lemma 4. *Let $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ be a DFA with a unique sink state.*

(1) *If $\mathscr{A}$ is nilpotent, then for any word $w \in \Sigma^+$ there exists a positive integer m such that $Q\cdot w^m = \{s\}$.*
(2) *$\mathscr{A}$ is nilpotent iff there are no cycles or loops passing through non-sink states.*

Next we introduce a particular subclass of nilpotent automata and a particular set I of instances of REACHABILITY.

Consider a nilpotent automaton $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ with a sink state $\mathfrak{s}$ and a non-empty set $\mathcal{T}_{\mathscr{A}}$ of states $\mathfrak{t}$ satisfying the following conditions:

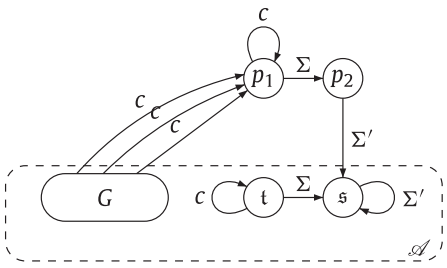

Fig. 3. Automaton $\mathscr{A}'$.

- $\mathfrak{t} \cdot \Sigma = \mathfrak{s}$;
- if there exists a word w such that $Q \cdot w = \{\mathfrak{s}, \mathfrak{t}\}$, then there exists such a word of length at least 2.

We denote this class of automata by $\mathcal{N}$. Note that the language $\mathrm{Syn}^{min}_{\mathscr{A}}$ is finite for every $\mathscr{A} \in \mathcal{N}$. It is an easy consequence of Theorem 1 and Lemma 4.

The set I is defined as follows:

$$I = \{(\mathscr{A}, H) \mid \mathscr{A} \in \mathcal{N}, H = \{\mathfrak{s}, \mathfrak{t}\}, \mathfrak{t} \in \mathcal{T}_{\mathscr{A}}\}.$$

The next proposition polynomially reduces any instance of REACHABILITY belonging to I to an instance of the complement of FINITENESS.

Proposition 1. *Let $(\mathscr{A}, H) \in I$ be an instance of* REACHABILITY, *with $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ and $H = \{\mathfrak{s}, \mathfrak{t}\}$, $\mathfrak{t} \in \mathcal{T}_{\mathscr{A}}$. Then there is a synchronizing automaton $\mathscr{A}'$ such that the language $\mathrm{Syn}^{min}_{\mathscr{A}'}$ is infinite if and only if there exists $w \in \Sigma^+$ such that $Q \cdot w = H$.*

Proof. Given an automaton $\mathscr{A} = \langle Q, \Sigma, \delta\rangle \in \mathcal{N}$ with $H = \{\mathfrak{s}, \mathfrak{t}\}$, $\mathfrak{t} \in \mathcal{T}_{\mathscr{A}}$, we modify it to obtain a new automaton $\mathscr{A}' = \langle Q', \Sigma', \delta'\rangle$ with $Q' = Q \cup \{p_1, p_2\}$, $\Sigma' = \Sigma \cup \{c\}$, where c is a new symbol, and the following transition function (Fig. 3):
$\delta'(q, x) = \delta(q, x)$ for all $x \in \Sigma$ and $q \in Q$,
let $G = Q \setminus \{\mathfrak{s}, \mathfrak{t}\}$, then $\delta'(G, c) = \{p_1\}$,
$\delta'(p_1, c) = p_1$, $\delta'(p_1, x) = p_2$ for all $x \in \Sigma$,
$\delta'(p_2, y) = \mathfrak{s}$ for all $y \in \Sigma'$, and
$\delta'(\mathfrak{t}, c) = \mathfrak{t}$, $\delta'(\mathfrak{s}, c) = \mathfrak{s}$.

Suppose there exists $w \in \Sigma^+$ such that $Q \cdot w = \{\mathfrak{s}, \mathfrak{t}\}$. By the choice of $\mathscr{A}$ there is such a word of length at least 2. Hence the image $Q' \cdot w$ is equal to $\{\mathfrak{s}, \mathfrak{t}\}$, and so this set is reachable also in $\mathscr{A}'$. Since $\{\mathfrak{s}, \mathfrak{t}\} \cdot c = \{\mathfrak{s}, \mathfrak{t}\}$, then $c \in \mathrm{Fix}(\{\mathfrak{s}, \mathfrak{t}\})$ and easily $m(c) = \{p_1, \mathfrak{s}, \mathfrak{t}\}$. On the other hand, if we take any $a \in \Sigma$, then $\{\mathfrak{s}, \mathfrak{t}\} \cdot a = \{\mathfrak{s}\}$ but $\{p_1, \mathfrak{s}, \mathfrak{t}\} \cdot a = \{p_2, \mathfrak{s}\}$. Hence $\mathrm{Syn}(\{\mathfrak{s}, \mathfrak{t}\}) \neq \mathrm{Syn}(\{p_1, \mathfrak{s}, \mathfrak{t}\})$, thus by the Theorem 1, the language $\mathrm{Syn}^{min}_{\mathscr{A}'}$ is infinite.

Conversely, if the language $\mathrm{Syn}^{min}_{\mathscr{A}'}$ is infinite then by Theorem 1 there is a reachable subset $S \subseteq Q'$ with $1 < |S| < |Q'|$ and a word $u \in \mathrm{Fix}(S)$ over $\Sigma \cup \{c\}$ such that $\mathrm{Syn}(m(u)) \subsetneq \mathrm{Syn}(S)$.

Suppose first $u \in \Sigma^+$. Since $\mathscr{A}$ is nilpotent, by Lemma 4 there is an integer n such that $Q \cdot u^n = \{\mathfrak{s}\}$. Then it is clear that $Q' \cdot u^{n+1} = \{\mathfrak{s}\}$. Thus by Lemma 1 we have $m(u) = \{\mathfrak{s}\}$, and $S \subseteq m(u) = \{\mathfrak{s}\}$, so $|S| = 1$, which is a contradiction. Therefore we may assume that u contains a factor of c^+. Since $Q' \cdot c = \{p_1, \mathfrak{s}, \mathfrak{t}\}$, then if u contains a factor of Σ^+, it is not difficult to see that $Q' \cdot u^2 = \{\mathfrak{s}\}$ which again leads to a contradiction $S \subseteq m(u) = \{\mathfrak{s}\}$. Thus we have $u = c^\beta$ for some positive integer $\beta > 0$, and so $m(u) = \{p_1, \mathfrak{s}, \mathfrak{t}\}$. Observe that, since $\mathfrak{s}$ is a sink state, then any reachable subset $S \subseteq Q'$ contains $\mathfrak{s}$. Therefore, since $|S| \geq 2$, $S \neq m(u)$ (otherwise $\mathrm{Syn}(m(u)) = \mathrm{Syn}(S)$) and $\mathfrak{s} \in S$, we have only two possibilities for S: either $S = \{\mathfrak{s}, \mathfrak{t}\}$ or $S = \{p_1, \mathfrak{s}\}$.

Case 1. Let $S = \{\mathfrak{s}, \mathfrak{t}\}$. Let $v \in (\Sigma \cup c)^+$ be a word such that $Q' \cdot v = \{\mathfrak{s}, \mathfrak{t}\}$. If c does not appear in v then $Q' \cdot v = \{\mathfrak{s}, \mathfrak{t}\}$ implies also $Q \cdot v = \{\mathfrak{s}, \mathfrak{t}\}$ and we are done. Suppose v contains c, so $v = wc^\alpha w'$ with $w \in \Sigma^*$, $\alpha > 0$, $w' \in (\Sigma \cup c)^*$. If $Q \cdot w \neq \{\mathfrak{s}, \mathfrak{t}\}$, then either $\mathfrak{t}$ is not in $Q \cdot w$ or $Q \cdot w$ contains a state different from both $\mathfrak{s}$ and $\mathfrak{t}$. In the first case by the construction of $\mathscr{A}'$ we get that $\mathfrak{t} \notin Q' \cdot v$, which is impossible. In the other case we get $Q' \cdot wc^\alpha = \{p_1, \mathfrak{s}, \mathfrak{t}\}$, however there is no $w' \in (\Sigma \cup c)^*$ such that $\{p_1, \mathfrak{s}, \mathfrak{t}\} \cdot w' = \{\mathfrak{s}, \mathfrak{t}\}$, which is again impossible. Thus $Q \cdot w = \{\mathfrak{s}, \mathfrak{t}\}$.

Case 2. Let $S = \{p_1, \mathfrak{s}\}$. We prove that in this case $\mathrm{Syn}(m(u)) = \mathrm{Syn}(S)$ which contradicts the initial assumption. Clearly it is enough to show that $\mathrm{Syn}(\{p_1, \mathfrak{s}\}) \subseteq \mathrm{Syn}(\{p_1, \mathfrak{s}, \mathfrak{t}\})$. Consider a word $z \in \mathrm{Syn}(\{p_1, \mathfrak{s}\})$. Since c fixes both $\{p_1, \mathfrak{s}\}$ and $\{p_1, \mathfrak{s}, \mathfrak{t}\}$, we can assume that z starts with a letter $a \in \Sigma$, so $z = az'$. Then $\{p_1, \mathfrak{s}\} \cdot a = \{p_2, \mathfrak{s}\} = \{p_1, \mathfrak{s}, \mathfrak{t}\} \cdot a$, thus $\{p_1, \mathfrak{s}\} \cdot z = \{p_1, \mathfrak{s}, \mathfrak{t}\} \cdot z$, i.e. $z \in \mathrm{Syn}(\{p_1, \mathfrak{s}, \mathfrak{t}\})$. □

In the sequel we polynomially reduce an instance of SAT to an instance of REACHABILITY belonging to I. To this end we present another auxiliary construction. Recall that a cartesian product of n deterministic finite automata $\mathscr{A}_i = \langle Q_i, \Sigma, \delta_i\rangle$ is

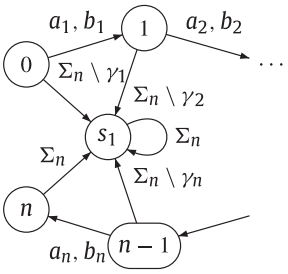

Fig. 4. Automaton A_n.

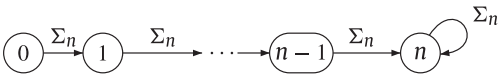

Fig. 5. Automaton B_n.

a DFA $\times_{i=1}^{n} \mathscr{A}_i = \langle Q, \Sigma, \delta \rangle$ with $Q = Q_1 \times Q_2 \times \cdots \times Q_n$, and a transition function $\delta : Q \times \Sigma \to Q$ defined componentwise

$$\delta((q_1, q_2, \ldots, q_n), a) = (\delta_1(q_1, a), \delta_2(q_2, a), \ldots, \delta_n(q_n, a)).$$

Clearly this action extends in a natural way to the free monoid Σ^*. It is not difficult to check that the following statement holds

Lemma 5. *Let $\mathscr{A}_1, \ldots, \mathscr{A}_m$ be nilpotent automata over a fixed alphabet with the sink states $q_1, \ldots, q_m$ respectively. Then their cartesian product $\times_{i=1}^{m} \mathscr{A}_i$ is also nilpotent, and $q = (q_1, \ldots, q_m)$ is its sink state.*

Next we fix a positive integer $n \geq 2$ and for each such n construct two particular nilpotent automata over the alphabet $\Sigma_n = \{a_1, b_1, \ldots, a_n, b_n\}$.

For convenience let $\gamma_i = \{a_i, b_i\}$ (in the sequel γ_i will correspond to the two possible values for the i^{th} variable). Consider an automaton $A_n = \langle Q_A, \Sigma_n, \delta_A \rangle$ with $Q_A = \{0, 1, \ldots, n, s_1\}$ and the transition function defined as follows (Fig. 4):
$i \cdot \gamma_{i+1} = i + 1$ for $0 \leq i \leq n - 1$,
$i \cdot (\Sigma_n \setminus \gamma_{i+1}) = s_1$ for $0 \leq i \leq n - 1$,
$n \cdot \Sigma_n = s_1$, $s_1 \cdot \Sigma_n = s_1$.

Next let $B_n = \langle Q_B, \Sigma_n, \delta_B \rangle$ with $Q_B = \{0, 1, \ldots, n\}$ and the following transition rules (Fig. 5): $i \cdot \Sigma_n = i + 1$ for $0 \leq i \leq n - 1$ and $n \cdot \Sigma_n = n$.

Now we construct their cartesian product $\mathcal{V}_n = A_n \times B_n$ which has $(n + 2)(n + 1)$ states and will serve as a gadget encoding the correct assignment of the values to variables. The following proposition establishes the main properties of this automaton.

Proposition 2. *The automaton $\mathcal{V}_n = A_n \times B_n = \langle Q_{A,B}, \Sigma_n, \delta_{A,B} \rangle$ is nilpotent with the sink state $s = (s_1, n)$ and the state $t = (n, n)$ satisfies $t \cdot \Sigma_n = s$. Moreover $Q_{A,B} \cdot w = \{s, t\}$ if and only if $w = x_1 \cdots x_n$ with $x_i \in \gamma_i$.*

Proof. By Lemma 5 $\mathcal{V}_n$ is nilpotent with the sink state $s = (s_1, n)$. The state $t = (n, n)$ satisfies $t \cdot \Sigma_n = s$.

Suppose that $Q_{A,B} \cdot w = \{s, t\} = \{s_1, n\} \times \{n\}$. It is obvious that for any word w it holds $Q_{A,B} \cdot w = Q_A \cdot w \times Q_B \cdot w$. Thus we have $Q_A \cdot w = \{s_1, n\}$ and $Q_B \cdot w = \{n\}$. An inspection on the automaton B_n shows that $Q_B \cdot w = \{n\}$ if and only if $|w| \geq n$. Suppose that $w = x_j w'$ for some $x_j \in \gamma_j$, then by the definition of the automaton A_n, we get $Q_A \cdot x_j = \{s_1, j\}$. Hence, not to 'kill' the state j and to lead it till the state n, we must have $w = x_j x_{j+1} \cdots x_n$, $x_i \in \gamma_i$ for all $i \geq j$. Combining this with the condition $|w| \geq n$ we get $j = 1$, i.e. $w = x_1 \cdots x_n$ with $x_i \in \gamma_i$ for all $i \geq 1$.

Conversely, let $w = x_1 \cdots x_n$ with $x_i \in \gamma_i$. An easy computation shows that $Q_A \cdot w = \{s_1, n\}$ and $Q_B \cdot w = \{n\}$. Thus $\{s, t\} = \{s_1, n\} \times \{n\} = Q_A \cdot w \times Q_B \cdot w = Q_{A,B} \cdot w = \{s, t\}$. □

Let an instance of SAT consist of m clauses $\chi = \{C_1, \ldots, C_m\}$ over n variables $X_1, \ldots, X_n$. Without loss of generality we can assume $n \geq 2$.

Next for each $x_i \in \gamma_i = \{a_i, b_i\}$, $1 \leq i \leq n$ we define $\chi(x_i) = \{C_{i_1}, \ldots, C_{i_k}\}$ to be the subset of χ consisting of clauses which contain positive literal X_i if $x_i = a_i$, and of clauses containing negative literal $\neg X_i$ if $x_i = b_i$. Without loss of generality we can assume that $\chi(a_i) \cap \chi(b_i) = \varnothing$, otherwise the common clause $C_k \in \chi(a_i) \cap \chi(b_i)$ would contain both X_i and $\neg X_i$, and so it would be trivially satisfied. Moreover we can assume that all such subsets are non-empty. Indeed if some $\chi(a_i) = \varnothing$, then all the clauses contain only negative literal $\neg X_i$, hence we can put $X_i = 0$ and reduce the problem to one with less variables.

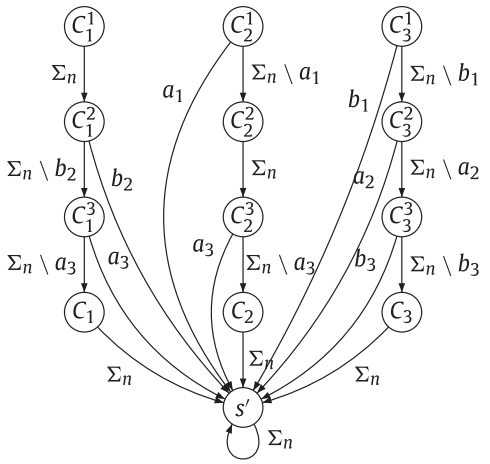

Fig. 6. Automaton $\mathscr{C}(C_1, C_2, C_3)$ for $C_1 = \neg X_2 \vee X_3$, $C_2 = X_3 \vee X_1$, $C_3 = X_2 \vee \neg X_1 \vee \neg X_3$.

We say that the set $\{x_1, \ldots, x_n\}$ with $x_i \in \gamma_i$ for $i = 1, \ldots, n$ is a *satisfiable assignment* for χ if and only if

$$\bigcup_{i=1}^{n} \chi(x_i) = \chi.$$

With these definitions, it is clear that to a satisfiable assignment $\{x_1, \ldots, x_n\}$ corresponds a satisfiable assignment for the Boolean formula $\wedge_{i=1}^{m} C_i$ given by $X_i = 1$ if $x_i = a_i$, and $X_i = 0$ if $x_i = b_i$, and vice versa.

We now define a nilpotent automaton $\mathscr{C}(C_1, \ldots, C_m) = \langle Q_C, \Sigma_n, \delta_C \rangle$ associated to the clauses $C_1, \ldots, C_m$. This automaton is a gadget which chooses satisfiable assignments from all possible ones. This automaton has $m(n+1)+1$ states with $Q_C = \chi^1 \cup \chi^2 \cup \cdots \cup \chi^n \cup \chi \cup \{s'\}$ where $\chi^i = \{C_1^i, \ldots, C_m^i\}$ for $i = 1, \ldots, n$ are copies of $\chi = \{C_1, \ldots, C_m\}$. We denote by $\chi^i(x_k)$ the subset of χ^i which is the corresponding copy of $\chi(x_k)$. The action of δ_C on Q_C is defined as follows. For each $1 \le i \le n-1$, each $1 \le j \le m$, and $C_j^i \in \chi^i$ we have

if $C_j^i \in \chi^i(a_i)$ then $C_j^i \cdot a_i = s'$, and $C_j^i \cdot (\Sigma_n \setminus \{a_i\}) = C_j^{i+1} \in \chi^{i+1}$,
if $C_j^i \in \chi^i(b_i)$ then $C_j^i \cdot b_i = s'$ and $C_j^i \cdot (\Sigma_n \setminus \{b_i\}) = C_j^{i+1} \in \chi^{i+1}$,
if $C_j^i \in \chi^i \setminus (\chi^i(a_i) \cup \chi^i(b_i))$ then $C_j^i \cdot \Sigma_n = C_j^{i+1} \in \chi^{i+1}$,
if $C_j^n \in \chi^n(a_n)$ then $C_j^n \cdot a_n = s'$ and $C_j^n \cdot (\Sigma_n \setminus \{a_n\}) = C_j \in \chi$,
if $C_j^n \in \chi^n(b_n)$ then $C_j^n \cdot b_n = s'$ and $C_j^n \cdot (\Sigma_n \setminus \{b_n\}) = C_j \in \chi$,
if $C_j^n \in \chi^n \setminus (\chi^n(a_n) \cup \chi^n(b_n))$ then $C_j^n \cdot \Sigma_n = C_j \in \chi$,
$\chi \cdot \Sigma_n = s'$ and $s' \cdot \Sigma_n = s'$.

An example of such an automaton for $\chi = \{C_1 = \neg X_2 \vee X_3, C_2 = X_3 \vee X_1, C_3 = X_2 \vee \neg X_1 \vee \neg X_3\}$ is shown in Fig. 6.

Clearly $\mathscr{C}(C_1, \ldots, C_m)$ is nilpotent with the sink state s'. The next lemma describes the main properties of the automaton $\mathscr{C}(C_1, \ldots, C_m)$.

Lemma 6. *Let $\mathscr{C}(C_1, \ldots, C_m) = \langle Q_C, \Sigma_n, \delta_C \rangle$ be constructed as above and $w = x_1 \cdots x_n \in \Sigma_n^+$ be a word with $x_i \in \gamma_i$ for $i = 1, \ldots, n$, then $Q_C \cdot w = \{s'\}$ if and only if $\{x_1, \ldots, x_n\}$ is a satisfiable assignment.*

Proof. Observe that any word of length n brings all the states (probably except for χ^1) of the automaton $\mathscr{C}(C_1, \ldots, C_m)$ to s'. Consider the image of the set χ^1 under the action of the word w:

$$\begin{aligned}
\chi^1 \cdot x_1 &= \{s'\} \cup \chi^2 \setminus \chi^2(x_1) \\
\chi^1 \cdot x_1 x_2 &= \{s'\} \cup \chi^3 \setminus (\chi^3(x_1) \cup \chi^3(x_2)) \\
&\vdots \\
\chi^1 \cdot x_1 x_2 \cdots x_{n-1} &= \{s'\} \cup \chi^n \setminus (\chi^n(x_1) \cup \chi^n(x_2) \cup \ldots \cup \chi^n(x_{n-1})) \\
\chi^1 \cdot x_1 x_2 \cdots x_n &= \{s'\} \cup \chi \setminus (\chi(x_1) \cup \chi(x_2) \cup \cdots \cup \chi(x_n)).
\end{aligned}$$

Hence

$$Q_C \cdot w = \{s'\} \cup \chi \setminus (\chi(x_1) \cup \chi(x_2) \cup \cdots \cup \chi(x_n)). \tag{3}$$

Suppose $\{x_1, \ldots, x_n\}$ is a satisfiable assignment for the clauses $C_1, \ldots, C_m$, whence $\bigcup_{i=1}^n \chi(x_i) = \chi$ and so, by equation (3), we get $Q_C \cdot w = \{s'\}$. Conversely, suppose $Q_C \cdot w = \{s'\}$, then, since $s' \notin \chi$, from equation (3) we deduce that $\chi \setminus (\chi(x_1) \cup \chi(x_2) \cup \cdots \cup \chi(x_n)) = \varnothing$, which implies that $\bigcup_{i=1}^n \chi(x_i) = \chi$. □

Proposition 3. *Let $C_1, \ldots, C_m$ be clauses over $n \geq 2$ variables. The automaton $\mathscr{A} = \mathcal{V}_n \times \mathscr{C}(C_1, \ldots, C_m) = \langle Q, \Sigma_n, \delta \rangle$ belongs to $\mathcal{N}$. Moreover, putting $\mathfrak{s} = (s, s')$, $\mathfrak{t} = (t, s')$, and $H = \{\mathfrak{s}, \mathfrak{t}\}$, we have $(\mathscr{A}, H) \in I$ and there is a word $w \in \Sigma_n^+$ such that $Q \cdot w = H$ if and only if the Boolean formula $\wedge_{i=1}^m C_i$ is satisfiable.*

Proof. By Lemma 5, the automaton $\mathscr{A} = \langle Q, \Sigma_n, \delta \rangle$ is nilpotent since both $\mathcal{V}_n$ and $\mathscr{C}(C_1, \ldots, C_m)$ are nilpotent automata. Moreover, with the notation of Proposition 2, $\mathfrak{s} = (s, s')$ is the sink state for $\mathscr{A}$ and since $t \cdot \Sigma_n = s$ and $s' \cdot \Sigma_n = s'$, we obtain that the state $\mathfrak{t} = (t, s')$ satisfies $\mathfrak{t} \cdot \Sigma_n = \mathfrak{s}$.

Let us first prove that $Q \cdot w = H = \{\mathfrak{s}, \mathfrak{t}\}$ if and only if $\wedge_{i=1}^m C_i$ is satisfiable. Since $\{\mathfrak{s}, \mathfrak{t}\} = \{s, t\} \times \{s'\}$, then $Q \cdot w = Q_{A,B} \cdot w \times Q_C \cdot w$. Thus $Q \cdot w = H$ if and only if $Q_{A,B} \cdot w = \{s, t\}$ and $Q_C \cdot w = \{s'\}$. On the other hand, by Proposition 2, $Q_{A,B} \cdot w = \{s, t\}$ if and only if $w = x_1 \cdots x_n$ with $x_i \in \gamma_i$ for $i = 1, \ldots, n$ and, by Lemma 6, $Q_C \cdot w = \{s'\}$ if and only if $\{x_1, \ldots, x_n\}$ is a satisfiable assignment.

Now we prove that $\mathcal{T}_{\mathscr{A}}$ is not empty. The word w from the previous argument satisfies $Q \cdot w = \{\mathfrak{s}, \mathfrak{t}\}$. Moreover since $n \geq 2$ we have $|w| \geq 2$. So $\mathfrak{t} \in \mathcal{T}_{\mathscr{A}}$, hence $\mathscr{A} \in \mathcal{N}$. □

Now we are ready to state the main result of this section.

Theorem 6. *The problem* FINITENESS *is co-NP-hard.*

Proof. Let $\{C_1, \ldots, C_m\}$ be an instance of SAT with $X_1, \ldots, X_n$ variables, $n \geq 2$. Combining Propositions 1 and 3 we obtain the automaton $\mathscr{A}' = \langle Q', \Sigma_n, \delta' \rangle$ over an alphabet with $2n + 1$ symbols and having $(n + 1)(n + 2)(m(n + 1) + 1) + 2$ states, such that $\mathrm{Syn}^{min}_{\mathscr{A}'}$ is infinite if and only if $\wedge_{i=1}^m C_i$ is satisfiable. □

8. Co-NP-hardness in case of a constant alphabet

Note that in our reduction the size of the alphabet is not constant and depends on the input. In this section we give a proof of our co-NP-hardness result for an alphabet Δ of a fixed size $|\Delta| > 2$.

To this purpose we encode the letters of the alphabet Σ_n in words of length n over the binary alphabet $\Sigma = \{a, b\}$ in the following way:

$$\varphi(a_i) = a^i b^{n-i}, \quad \varphi(b_i) = b^i a^{n-i}$$

and modify automata $A_n, B_n, \mathscr{C}(C_1, \ldots, C_m)$ in order to obtain automata $\tilde{A}_n$ $\tilde{B}_n$ and $\tilde{\mathscr{C}}(C_1, \ldots, C_m)$ over the binary alphabet, preserving their initial properties essential for the proof of Theorem 7. Namely, between each pair i, $i + 1$ of states in the automaton A_n $(0 \leq i < n)$ we add $2(n - 1)$ new states and transitions so that the following property holds:

$$i \cdot u = \begin{cases} i + 1 & \text{if } u = \varphi(x_{i+1}), \\ s_1, & \text{otherwise.} \end{cases}$$

The new automaton has $2n^2 - n + 2$ states. For an example of such a construction see Fig. 7 (new states are shown as small filled nodes, and transitions from new states to s_1 are not shown).

The automaton $\tilde{B}_n$ has $n^2 + 1$ states with $i \cdot \Sigma = i + 1$ for $0 \leq i < n^2$ and $n^2 \cdot \Sigma = n^2$. Then it is easy to check that the cartesian product $\tilde{\mathcal{V}}_n = \tilde{A}_n \times \tilde{B}_n$ satisfies a statement analogous to the Proposition 2.

Proposition 4. *The automaton $\tilde{\mathcal{V}}_n = \tilde{A}_n \times \tilde{B}_n = \langle \tilde{Q}_{A,B}, \Delta, \tilde{\delta}_{A,B} \rangle$ is nilpotent with the sink state $s = (s_1, n^2)$, and the state $t = (n, n^2)$ satisfies $t \cdot \Delta = s$. Moreover $\tilde{Q}_{A,B} \cdot w = \{s, t\}$ if and only if $w = \varphi(x_1 x_2 \cdots x_n)$ with $x_i \in \gamma_i$.*

To modify automaton $\mathscr{C}_n(C_1, \ldots, C_m)$, we add $2(n - 1)$ new states between each pair of states of the form C_j^i, C_j^{i+1} $(1 \leq i \leq n-1)$ and C_j^{n-1}, C_j, for $1 \leq j \leq m$, and the transitions are defined so that for each $1 \leq i \leq n-1$ the state C_j^i in the new automaton $\tilde{\mathscr{C}}(C_1, \ldots, C_m)$ goes to C_j^{i+1} with $u = \varphi(x_i)$ if and only if $C_j^i \cdot x_i = C_j^{i+1}$ in the old automaton, and to s' with all the other words u. Analogously, C_j^{n-1} in $\tilde{\mathscr{C}}(C_1, \ldots, C_m)$ goes to C_j with $u = \varphi(x_{n-1})$ if and only if $C_j^{n-1} \cdot x_{n-1} = C_j$ in

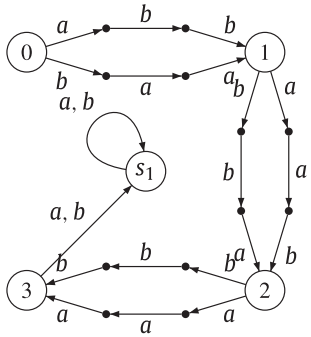

Fig. 7. $\tilde{A}_3$.

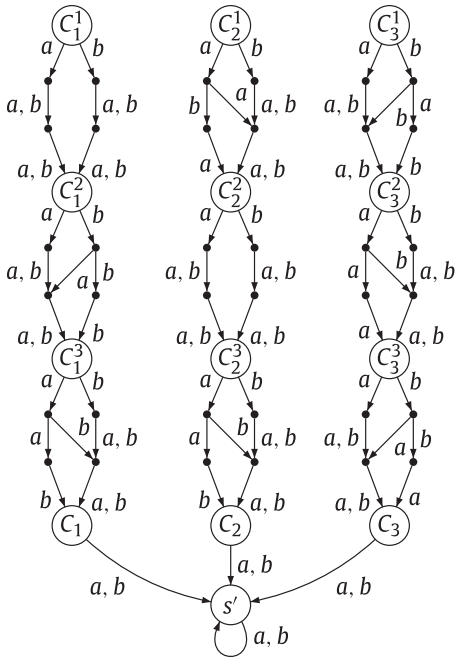

Fig. 8. The automaton $\tilde{\mathscr{C}}(C_1, C_2, C_3)$.

$\mathscr{C}(C_1 \ldots, C_m)$. The application of all the other words results in s'. The automaton $\tilde{\mathscr{C}}(C_1, C_2, C_3)$ obtained from the automaton on Fig. 6 is shown on Fig. 8 (the transitions not shown lead to s').

The constructed automaton has $(2n^2 - n + 1)m + 1$ states. A statement analogous to Lemma 6 holds:

Lemma 7. *The word $w = \varphi(x_1 \cdots x_n)$ with $x_i \in \gamma_i$ for $i = 1, \ldots, n$ synchronizes the automaton $\tilde{\mathscr{C}}(C_1, \ldots, C_m)$ if and only if $\{x_1, \ldots, x_n\}$ satisfies χ.*

Thus the automaton $\tilde{\mathscr{A}} = \tilde{\mathscr{V}}_n \times \tilde{\mathscr{C}}(C_1, \ldots, C_m)$ satisfies properties analogous to Proposition 3. Combining this with Proposition 1 we obtain the automaton $\tilde{\mathscr{A}}'$ over the 3-lettered alphabet $\Sigma' = \{a, b, c\}$ with $(2n^2 - n + 2)(n^2 + 1)((2n^2 - n + 1)m + 1) + 2 = \Theta(n^6 m)$ states such that the language $\mathrm{Syn}^{min}_{\tilde{\mathscr{A}}'}$ is infinite if and only if $\wedge_{i=1}^{m} C_i$ is satisfiable. It results in the following theorem:

Theorem 7. *The problem* FINITENESS *is co-NP-hard for all automata over a constant alphabet Δ, $|\Delta| > 2$.*

9. Open problems

Theorem 4 shows that every n-state finitely generated synchronizing automaton has a synchronizing word of length at most $3n - 5$. On the other hand, as discussed in the introduction, there are examples of automata in **FG** having shortest

synchronizing words of length $n-1$. We are interested in finding a series of finitely generated synchronizing automata whose shortest synchronizing word has length exactly $3n-5$ or proving that this length is always not greater than $n-1$.

We have proved that the FINITENESS problem is co-NP-hard in case of at least 3-letter alphabet. What about the complexity of this problem in case of a binary alphabet? Another interesting question concerns the precise complexity class of FINITENESS. In particular, is it PSPACE-complete?

Acknowledgments

The authors thank professor Mikhail V. Volkov for proposing the problem and for the precious suggestions. They deeply appreciate useful remarks by professor Juhani Karhumäki and are grateful to Pawel Gawrychowski for communicating them his PSPACE result. The authors thank anonymous referees for careful reading of the paper and a number of helpful remarks.

References

[1] P. Gawrychowski, private communication, 2008.
[2] S.W. Margolis, J.E. Pin, M.V. Volkov, Int. J. Found. Comput. Sci. 15 (2004) 259–276.
[3] A. Mateescu, A. Salomaa, EATCS Bull. 68 (1999) 134–150.
[4] M. Perles, M.O. Rabin, E. Shamir, IEEE Trans. Electr. Comp. 12 (1963) 233–243.
[5] J.E. Pin, Actes du 1er Colloque AFCET-SMF de Mathématiques Appliquées, AFCET, Tome II, 1978, pp. 85–92.
[6] I.K. Rystsov, Cybernet. Systems Anal. 30 (1994) 807–811.
[7] S. Sandberg, in: M. Broy et al. (Eds.), Model-based Testing of Reactive Systems, Lecture Notes in Computer Science, vol. 3472, Springer, Berlin, 2005, pp. 5–33.
[8] J. Černý, Mat.-Fyz. Čas. Slovensk. Akad. Vied. 14 (1964) 208–216.
[9] M.V. Volkov, in: C. Martín-Vide, F. Otto, H. Fernau (Eds.), Languages and Automata: Theory and Applications. LATA 2008, Lecture Notes in Computer Science, vol. 5196, Springer, Berlin, 2008, pp. 11–27.