International Journal of Foundations of Computer Science
Vol. 27, No. 5 (2016) 557–577

DOI: 10.1142/S0129054116500167

Subset Synchronization and Careful Synchronization of Binary Finite Automata*

Vojtěch Vorel

Department of Theoretical Computer Science and Mathematical Logic
Charles University
Malostranské nám. 25, Prague, Czech Republic
vorel@ktiml.mff.cuni.cz

Received 27 September 2014
Accepted 20 August 2015
Communicated by Arto K. Salomaa

We present a strongly exponential lower bound that applies both to the subset synchronization threshold for binary deterministic automata and to the careful synchronization threshold for binary partial automata. In the later form, the result finishes the research initiated by Martyugin (2013). Moreover, we show that both the thresholds remain strongly exponential even if restricted to strongly connected binary automata. In addition, we apply our methods to computational complexity. Existence of a subset reset word is known to be PSPACE-complete; we show that this holds even under the restriction to strongly connected binary automata. The results apply also to the corresponding thresholds in two more general settings: D1- and D3-directable nondeterministic automata and composition sequences over finite domains.

Keywords: Reset word; directing word; synchronizing word; composition sequence; Černý conjecture.

1. Introduction

Questions about synchronization of finite automata have been studied since the early times of automata theory. The basic concept is very natural: For a given machine, we want to find an input sequence that would get the machine to a unique state, no matter in which state the machine was before. Such sequence is called a *reset word*.[a] If an automaton has some reset word, we call it a *synchronizing automaton*. For deterministic automata these definitions are clear, while for more general types of automata (partial, nondeterministic, probabilistic, weighted,...) there are multiple variants, each of its own importance. Several fields of mathematics and

*Research supported by the Czech Science Foundation grant GA14-10799S and the GAUK grant No. 52215.
[a]Some authors use the terms *synchronizing word* or *directing word*.

engineering deal with such notions. Classical applications (see [28]) include model-based testing of sequential circuits, robotic manipulation, symbolic dynamics, and design of noise-resistant systems [7], but there are important connections also with information theory [27] and with formal models of biomolecular processes [4].

Two particular problems concerning synchronization has gained some publicity: the Road Coloring Problem and the Černý Conjecture. The first has been solved by Trahtman [25] in 2007 by proving that the edges of any aperiodic directed multi-graph with constant out-degree can be colored such that a synchronizing deterministic automaton arises. Motivation for the Road Coloring Problem comes from symbolic dynamics [1]. On the other hand, the Černý Conjecture remains open since 1971 [6]. It claims that any n-state synchronizing deterministic automaton has a reset word of length at most $(n-1)^2$. It is known that there is always a reset word of length at most $\frac{n^3-n}{6}$ [22].[b]

We focus on two specific synchronization problems that both generalize the problem represented by the Černý Conjecture. It may be surprising that in both the cases the corresponding bounds become exponential:

- In a *partial automaton*, each transition is either undefined or defined unambiguously. A *careful reset word* of such automaton maps all the states to one unique state (using only the defined transitions). The problem is, for given n, how long a shortest careful reset word of an n-state partial automaton may be? It is known that in general it may have strongly exponential length, i.e., $2^{\Omega(n)}$.
- Given a deterministic automaton and a subset S of its states, a *reset word of S* maps all the states of S to one unique state. The problem is, for given n, how long a shortest reset word of a subset of states in an n-state deterministic automaton may be? Again, it is known that in general it may have length $2^{\Omega(n)}$. Note that the deterministic automata under consideration do not need to be synchronizing.

Both these questions, concerning also some variants and restrictions, have been studied since 1970's but until recently none of the presented bad cases (i.e., lower bounds) have consisted of automata with two-letter or any other fixed-size alphabets. The definitions above allow the alphabet to grow with growing number of states, which offers a very strong tool for constructing series that witness strongly exponential lower bounds.

In 2013 Martyugin [19] gives a lower bound of the form $2^{\Omega\left(\frac{n}{\log n}\right)}$ that applies to both the problems restricted to automata with two-letter alphabets (i.e., *binary automata*). Moreover, for each fixed alphabet size $m \geq 2$, the author of [19] provides a specific multiplicative constant in the exponent. There is a simple construction (Lemma 6) guaranteeing that existence of a lower bound of the form $2^{\Omega(n)}$ for any particular alphabet size m implies a lower bound of the same form for any other

[b]An improved bound published by Trahtman [26] in 2011 has turned out to be proved incorrectly, see [10].

$m \geq 2$ as well. However it remained as an open question if the lower bounds can be raised to $2^{\Omega(n)}$ in such a way.

In the present paper we give the answer: We present a lower bound of the form $2^{\Omega(n)}$ that applies to both the problems restricted to binary automata. Moreover, we introduce a technique for applying the lower bounds even under the restriction to strongly connected binary automata. The main results are expressed by Theorem 8 in Sec. 5.

2. Preliminaries

2.1. *Partial finite automata*

A *partial finite automaton* (*PFA*) is a triple $A = (Q, X, \delta)$, where Q and X are finite sets and $\delta : Q \times X \to Q$ is a partial function. Elements of Q are called *states*, X is the *alphabet*. The *transition function* δ can be naturally extended to $Q \times X^{\star} \to Q$ by defining

$$\delta(s, vx) = \delta(\delta(s, v), x)$$

inductively for $x \in X, v \in X^{\star}$ if the right-hand side is defined. We extend δ also with

$$\delta(S, w) = \{\delta(s, w) \mid s \in S,\ \delta(s, w) \text{ defined}\}$$

for each $S \subseteq Q$ and $w \in X^{\star}$. A PFA is *deterministic* (*DFA*) if δ is a total function. A PFA (Q, X, δ) is said to be *strongly connected*[c] if

$$(\forall r, s \in Q)\,(\exists w \in X^{\star})\,\delta(r, w) = s.$$

A state $s \in Q$ is a *sink state* if $\delta(s, x) = s$ for each $x \in X$. Clearly, if a nontrivial PFA has a sink state, it is impossible for the PFA to be strongly connected. The class of all strongly connected PFA and the class of all PFA with k-letter alphabets are denoted by $\mathcal{SC}$ and $\mathcal{AL}_k$ respectively. Automata from $\mathcal{AL}_2$ are called *binary*.

2.2. *Careful synchronization*

For a given PFA, we call $w \in X^{\star}$ a *careful reset word* if

$$(\exists r \in Q)\,(\forall s \in Q)\,\delta(s, w) = r.$$

If such a word exists, the automaton is *carefully synchronizing*. There are also notions that describe “less careful” variants of synchronization of PFA. E.g., both the definition of *D*2*-directing* (see Sec. 6.2) and the definition of a *reset word* from [3] give conditions that may hold for words that are not careful reset words. However, all the notions are identical if we consider only DFA. In such cases we can just use terms *reset word* and *synchronizing*, without the adjective *careful*. In DFA, each

[c]Some authors use the term *transitive automaton*.

word having a reset word as a factor is also a reset word. Note that in a carefully synchronizing PFA, there is always at least one $x \in X$ such that $\delta(s, x)$ is defined on each $s \in Q$.

We use the following notation consistent with [19]. For a PFA A, let $\mathrm{car}(A)$ denote the length of a shortest careful reset word of A. If there is no such word, we put $\mathrm{car}(A) = 0$. For each $n \geq 1$, let $\mathrm{car}(n)$ denote the maximum value of $\mathrm{car}(A)$ taken over all n-state PFA A. It is easy to see that $\mathrm{car}(n) \leq 2^n - n - 1$ for each n, but this upper bound has been pushed down to $\mathrm{car}(n) = O\left(n^2 \cdot 4^{\frac{n}{3}}\right)$ by Gazdag, Iván, and Nagy-György [9] in 2009.

2.3. *Subset synchronization*

Even if a PFA is not synchronizing, there could be various subsets $S \subseteq Q$ such that

$$(\exists r \in Q)\,(\forall s \in S)\,\delta(s, w) = r$$

for some word $w \in X^\star$. We say that such S is *carefully synchronizable* in A and in the opposite case we say it is *blind* in A. The word w is called a *careful reset word of* S in A. If A is a DFA, we call w just a *reset word* of a *synchronizable* subset S in A. Such words concerning DFA are of our main interest. They lack some of the elegant properties of classical reset words of DFA (i.e., reset words of $S = Q$), particularly a word w having a factor v which is a reset word of S need not to be itself a reset word of S. In fact, if we choose a subset S and a word w, it is possible for the set $\delta(S, w)$ to be blind even if the set S is synchronizable.

For a PFA A, and $S \subseteq Q$ let $\mathrm{csub}(A)$ denote the minimum length of a careful reset word of S in A. If S is blind, we set $\mathrm{csub}(A, S) = 0$. If A is a DFA we write $\mathrm{sub}(A)$ instead of $\mathrm{csub}(A)$. For each $n \geq 1$, let $\mathrm{csub}(n)$ (and $\mathrm{sub}(n)$) denote the maximum value of $\mathrm{csub}(A)$ taken over all n-state PFA A (or n-state DFA A respectively) and all their subsets of states. It is easy to see that

$$\mathrm{sub}(n) \leq \mathrm{csub}(n) \leq 2^n - n - 1$$

for each n. Our strongly exponential lower bound applies to $\mathrm{sub}(n)$ and thus to $\mathrm{csub}(n)$ as well. The values $\mathrm{csub}(n)$ play only an auxiliary role in the present paper.

If an automaton $A = (Q, X, \delta)$ and a subset $S \subseteq Q$ are given (possibly with $S = Q$), we say that $s \in Q$ is *active after* (or *during*) *the application of* $u \in X^\star$ if $s \in \delta(S, u)$ (or $s \in \delta(S, v)$ for a prefix v of u, respectively).

3. Previously Known Lower Bounds

Let $\mathcal{M}$ be a class of automata. For each n let $\mathcal{M}_{\leq n}$ be the class of all automata lying in $\mathcal{M}$ and having at most n states. For each $n \geq 1$ we extend our notation of $\mathrm{car}^{\mathcal{M}}(n)$, $\mathrm{sub}^{\mathcal{M}}(n)$, and $\mathrm{csub}^{\mathcal{M}}(n)$, denoting the maximum values of $\mathrm{car}(A)$, $\mathrm{sub}(A)$ and $\mathrm{csub}(A)$ taken over $A \in \mathcal{M}_{\leq n}$. In the cases of $\mathrm{sub}^{\mathcal{M}}(n)$ and $\mathrm{csub}^{\mathcal{M}}(n)$, we use the notion in the obvious way even if $\mathcal{M}$ is a class of pairs automaton-subset. All such notions we informally call *synchronization thresholds.*

In 1976 Burkhard [5] showed that for any $n \geq 2$ and $k \leq n-2$ it is not hard to produce an n-state, $\binom{n-2}{k-1}$-letter DFA with a k-state subset S such that $\mathrm{sub}(A, S) \geq \binom{n-2}{k-1}$. If we set $k = \frac{n}{2}$ and use Stirling's approximation to check that

$$\binom{n}{\frac{n}{2}} \approx \sqrt{\frac{2}{\pi}} \cdot \frac{2^n}{\sqrt{n}},$$

we get

$$\mathrm{sub}(n) = \Omega\left(\frac{2^n}{\sqrt{n}}\right).$$

The threshold $\mathrm{car}(n)$ was initially studied in 1982 by Goralčík *et al.* [11], together with several related problems. The authors show that for infinitely many n there is a permutation of n states having order at least $(\sqrt[3]{n})!$ and they use it to prove that $\mathrm{car}(n) \geq (\sqrt[3]{n})!$. The construction can be easily (e.g., using our Lemma 1) modified to establish $\mathrm{sub}(n) \geq (\sqrt[3]{n})!$ as well, as it was later re-discovered in the paper [16]. Though exceeded by $\Omega\left(\frac{2^n}{\sqrt{n}}\right)$, the later lower bound of $\mathrm{sub}(n)$ remains interesting since the proof uses binary alphabets only.

In [14], Ito and Shikishima-Tsuji proved that $\mathrm{car}(n) \geq 2^{\frac{n}{2}}$ and the construction was subsequently improved by Martyugin [18] in order to reach $\mathrm{car}(n) \geq 3^{\frac{n}{3}}$. Again, the construction can be applied to subsets, so we get $\mathrm{sub}(n) \geq 3^{\frac{n}{3}}$. However, the last proofs seem to use very artificial examples of automata:

- In the series, the alphabet size grows linearly with the growing number of states - the proofs rely on the convention of measuring the *size* of an automaton only by the number of states. The results say nothing about the thresholds $\mathrm{sub}^{\mathcal{AL}_k}(n)$ or $\mathrm{car}^{\mathcal{AL}_k}(n)$ for any $k \geq 2$. In 2013, Martyugin [19] proves that

  $$\mathrm{car}^{\mathcal{AL}_2}(n) > 3^{\frac{n}{6 \cdot \log_2 n}}$$

  and

  $$\mathrm{car}^{\mathcal{AL}_k}(n) > 3^{\frac{n}{3 \cdot \log_{m-1} n}}$$

  for each $k \geq 3$, which applies in a similar form also to subset synchronization. However, it remained unclear whether $\mathrm{car}^{\mathcal{AL}_k}(n) = 2^{\Omega(n)}$ or $\mathrm{sub}^{\mathcal{AL}_k}(n) = 2^{\Omega(n)}$ for some $k \geq 2$. Here we confirm this for $k = 2$, so for any greater k the claim follows easily.
- In the case of subset synchronization, the DFA have sink states, typically two of them in each automaton. Use of sink states is a very strong tool for designing automata having given properties, but in practice such automata seem very special. They represent unstable systems balancing between different deadlocks. The very opposite are strongly connected automata. Does the threshold remain so high if we consider only strongly connected DFA? Unfortunately, we show below that it does, even if we restrict the alphabet size to a constant. We introduce *swap congruences* as an alternative to sink states.

Note that in the case of careful synchronization, any lower bound of $\mathrm{car}(n)$ applies easily to $\mathrm{car}^{\mathcal{SC}}(n)$ using a simple trick from Lemma 2. Moreover, for suitable series the alphabet size is increased only by a constant.

In short, in the present paper we prove that

$$\mathrm{sub}^{\mathcal{AL}_2 \cap \mathcal{SC}}(n) = 2^{\Omega(n)},$$
$$\mathrm{car}^{\mathcal{AL}_2 \cap \mathcal{SC}}(n) = 2^{\Omega(n)}.$$

The new bounds are tight in the sense of $\mathrm{car}(n) = 2^{\theta(n)}$ and $\mathrm{sub}(n) = 2^{\theta(n)}$.

4. Reductions Between Thresholds

This section prepares the ground for the results presented in Sec. 5 by introducing basic principles and relationships concerning the studied thresholds. The principles are not innovative, except for the method using *swap congruences* described in the paragraph 4.2, dealing with strong connectivity in subset synchronization.

As noted above, many of the lower bounds of $\mathrm{car}(n)$ and $\mathrm{sub}(n)$ found in the literature were formulated for only one of the notions but used ideas applicable to the other as well. The key method used in the present paper is of this kind again. However, we are not able to calculate any of the thresholds from the other exactly, so we at least show several related inequalities and then use some of them in Sec. 5 to prove the main results. We use the term *reduction* since we prove the inequalities by transforming an instance of a problem to an instance of another problem.

4.1. *Determinization by adding sink states*

The following inequality is not a key tool of the present paper; we prove it in order to illustrate that even careful subset synchronization is not much harder than subset synchronization itself. Recall that trivially $\mathrm{car}(n) \le \mathrm{csub}(n)$ and $\mathrm{sub}(n) \le \mathrm{csub}(n)$ for each n.

Lemma 1. *For each $n \ge 1$ it holds that*

$$\mathrm{csub}(n) \le \mathrm{sub}(n+2) - 1.$$

Proof. Take any PFA $A = (Q_A, X_A, \delta_A)$ with a carefully synchronizable subset $S_A \subseteq Q_A$ and choose a shortest careful reset word $w \in X^\star$ of S_A with $\delta_A(s, w) = r_0$ for each $s \in S_A$. We construct a DFA $B = (Q_B, X_B, \delta_B)$ and a synchronizable subset $S_B \subseteq Q_B$ such that $\mathrm{sub}(B, S_B) \ge |w| + 1$. Let us set

$$Q_B = Q_A \cup \{\mathrm{D}, \overline{\mathrm{D}}\}, \qquad \delta_B(\mathrm{D}, x) = \mathrm{D},$$
$$X_B = X_A \cup \{\omega\}, \qquad \delta_B(\overline{\mathrm{D}}, x) = \overline{\mathrm{D}}$$

for each $x \in X_B$, and

$$\delta_B(s, x) = \begin{cases} \delta_A(s, x) & \text{if defined,} \\ \overline{\mathrm{D}} & \text{otherwise,} \end{cases} \qquad \delta_B(s, \omega) = \begin{cases} \mathrm{D} & \text{if } s = r_0, \\ \overline{\mathrm{D}} & \text{otherwise} \end{cases}$$

for each $s \in Q_A, x \in X_A$. Denote $S_B = S_A \cup \{\mathrm{D}\}$. The word $w\omega$ witnesses that the subset S_B is synchronizable. On the other hand, let v be any reset word of S_B. Since D is a sink state and $\mathrm{D} \in S_B$, we have $\delta_B(v, s) = \mathrm{D}$ for each $s \in S_B$. Thus:

- The state $\overline{\mathrm{D}}$ is not active during the application of v.
- There need to be an occurrence of ω in v.

Denote $v = v_0 \omega v_1$, where $v_0 \in X_A^\star$ and $v_1 \in X_B^\star$. If $|\delta_B(S_B, v_0) \cap Q_A| = 1$, we are done since v_0 maps all the states of S_A to a unique state using only the transitions defined in A, so $|v| \geq |w| + 1$. Otherwise, there is some $s \in \delta_B(Q_B, v_0) \cap Q_A$ such that $s \neq r_0$, but then $\delta_B(\omega, s) = \overline{\mathrm{D}}$, which is a contradiction. □

4.2. *Strong connectivity*

First, we show an easy reduction concerning careful synchronization of strongly connected PFA. We use a simple trick: A letter that is defined only on a single state cannot appear in a shortest careful reset word, so one can make a PFA strongly connected by adding such letters. The number of new letters needed may be reduced by adding special states, but the simple variant described by Lemma 2 is illustrative and strong enough for our purpose.

For each $j \geq 0$ we define the class $\mathcal{C}_j$ of PFA as follows. A PFA $A = (Q, X, \delta)$ belongs to $\mathcal{C}_j$ if there are j pairs $(r_1, q_1), \ldots, (r_j, q_j) \in Q \times Q$ such that adding transitions of the form $r_i \longrightarrow q_i$ for each $i = 1, \ldots, j$ makes the automaton strongly connected. Note that $\mathcal{C}_0 = \mathcal{SC}$.

Lemma 2. *For each $n, k, j \geq 1$ it holds that*

$$\mathrm{car}^{\mathcal{AL}_k \cap \mathcal{C}_j}(n) \leq \mathrm{car}^{\mathcal{AL}_{k+j} \cap \mathcal{SC}}(n).$$

Proof. Take any PFA $A = (Q, X_A, \delta_A) \in \mathcal{AL}_k \cap \mathcal{C}_j$ together with the pairs $(r_1, q_1), \ldots, (r_j, q_j) \in Q \times Q$ from the definition of $\mathcal{C}_j$. We construct a PFA $B = (Q, X_B, \delta_B)$ where $X_B = X_A \cup \{\psi_1, \ldots, \psi_j\}$, $\delta_B(s, x) = \delta_A(s, x)$ for $x \in X_A$ and $s \in Q$, and

$$\delta_B(s, \psi_i) = \begin{cases} q_i & \text{if } s = r_i, \\ \text{undefined} & \text{otherwise} \end{cases}$$

for $i = 1 \ldots, , j'$ and $s \in Q$. Now it is easy to check that B is strongly connected and that $\mathrm{car}(B) = \mathrm{car}(A)$. □

Second, we present an original method concerning subset synchronization of strongly connected DFA. All the lower bounds applicable to $\mathrm{sub}(n)$ that we have found in the literature used two sink states (deadlocks) to force application of particular letters during a synchronization process. A common step in such proof looks like *"The letter x cannot be applied since that would make the sink state* $\overline{\mathrm{D}}$ *active, while another sink state* D *is active all the time"*. In order to prove a lower bound of $\mathrm{sub}^{\mathcal{SC}}(n)$, we have to develop an alternative mechanism. Our mechanism relies on *swap congruences*:

Recall that, given a DFA $A = (Q, X, \delta)$, an equivalence relation $\rho \subseteq Q \times Q$ is a *congruence* if

$$r \rho s \Rightarrow \delta(r, x)\, \rho\, \delta(s, x)$$

for each $x \in X$. We say that a congruence ρ is a *swap congruence* of a DFA if, for each equivalence class C of ρ and each letter $x \in X$, the restricted function $\delta : C \times \{x\} \to Q$ is injective. The key property of swap congruences is the following.

Lemma 3. *Let $A = (Q, X, \delta)$ be a DFA, let $\rho \subseteq Q \times Q$ be a swap congruence and take any $S \subseteq Q$. If there are any $r, s \in S$ with $r \neq s$ and $r \rho s$, the set S is blind.*

Proof. Because r and s lie in a common equivalence class of ρ, by the definition of a swap congruence we have $\delta(r, w) \neq \delta(s, w)$ for any $w \in X^\star$. □

Thus, the alternative mechanism relies on arguments of the form *"The letter x cannot be applied since that would make both the states r, s active, while it holds that $r \rho s$"*. It turns out that our results based on the method can be derived from more transparent but not strongly connected constructions by the following reduction principle:

Lemma 4. *For each $n \geq 1$ it holds that*

$$\mathrm{sub}(n) \leq \mathrm{sub}^{\mathcal{SC}}(2n + 2) - 1.$$

Moreover, for each $n, k \geq 1$ and $j \geq 2$ it holds that

$$\mathrm{sub}^{\mathcal{AL}_k \cap \mathcal{C}_j}(n) \leq \mathrm{sub}^{\mathcal{AL}_{k+j} \cap \mathcal{SC}}(2n + 2) - 1.$$

Proof. The first claim follows easily from the second one. So, take any DFA $A = (Q_A, X_A, \delta_A) \in \mathcal{AL}_k \cap \mathcal{C}_j$ together with the pairs $(r_1, q_1), \ldots, (r_j, q_j) \in Q_A \times Q_A$ from the definition of $\mathcal{C}_j$ and let $S \subseteq Q_A$ be synchronizable. We construct a strongly connected DFA $B = (Q_B, X_B, \delta_B)$ and a subset $S_B \subseteq Q_B$ such that $\mathrm{sub}(B, S_B) \geq \mathrm{sub}(A, S_A) + 1$. Let us set

$$Q_B = \{s, \overline{s} \mid s \in Q_A\} \cup \{\mathrm{E}, \overline{\mathrm{E}}\},$$
$$X_B = X_A \cup \{\psi_1, \ldots, \psi_j\}.$$

We want the relation

$$\rho = \langle (s, \overline{s}) \mid s \in Q_A \cup \{\mathrm{E}\} \rangle,$$

where $\langle \ldots \rangle$ denotes an equivalence closure, to be a swap congruence. Regarding this requirement, it is enough to define δ_B on $Q_A \cup \{\mathrm{E}\}$. The remaining transitions are forced by the injectivity on the equivalence classes. We set

$$\delta_B(s, x) = \delta_A(s, x), \qquad \delta_B(\mathrm{E}, x) = \mathrm{E}$$

for any $s \in Q_A, x \in X_A$, while the letters $\psi_1, \ldots, \psi_j$ act as follows:

$$\delta_B(s, \psi_I) = \begin{cases} q_I & \text{if } s = r_I, \\ \overline{q_I} & \text{otherwise,} \end{cases} \qquad \delta_B(\mathrm{E}, \psi_I) = q_I,$$
$$\delta_B(s, \psi_i) = \begin{cases} q_i & \text{if } s = r_i, \\ \overline{\mathrm{E}} & \text{otherwise,} \end{cases} \qquad \delta_B(\mathrm{E}, \psi_i) = \mathrm{E}$$

for $s \in Q_A$ and $i \neq I$, where I is chosen such that for a reset word w of S_A in A with $\delta_A(s, w) = r_0$, the state r_I is reachable from r_0. It is easy to see that such I exists for any $r_0 \in S_A$. We set $S_B = S_A \cup \{\mathrm{E}\}$.

- First, note that the set S_B is synchronizable in B by the word $wu\psi_I$ where $u \in X_A^\star$ such that $\delta_A(r_0, u) = r_I$.
- On the other hand, let v be a reset word of S_B in B. The word v necessarily contains some ψ_i for $i \in \{1, \ldots, j\}$, so we can write $v = v_0 \psi_i v_1$, where $v_0 \in X_A^\star, v_1 \in X_B^\star$. If v_0 is a reset word of S_A in A, $|v| \geq \mathrm{sub}(A, S_A) + 1$ and we are done. Otherwise there is a state $s \neq r_i$ in $\delta_B(S, v_0)$ and we see that both q_i and $\overline{q_i}$ (if $i = I$) or both E and $\overline{\mathrm{E}}$ (if $i \neq I$) lie in $\delta_B(S, v_0\psi_1)$, which is a contradiction with properties of the swap congruence ρ.

The automaton B is strongly connected since the transitions $r_i \xrightarrow{\psi_i} q_i$ and $\overline{r_i} \xrightarrow{\psi_i} \overline{q_i}$ for each $i = 1, \ldots, j$ make both the copies of A strongly connected and there are transitions $\mathrm{E} \xrightarrow{\psi_I} q_I$, $s \xrightarrow{\psi_i} \overline{\mathrm{E}}$, $\overline{\mathrm{E}} \xrightarrow{\psi_I} \overline{q_2}$, and $\overline{s} \xrightarrow{\psi_i} \mathrm{E}$ for some $i \neq I$ and $s \neq r_i$. □

4.3. *A special case of subset synchronization*

We are not aware of any general bad-case reduction from subset synchronization to careful synchronization. Here we suggest a special class (denoted by $\mathcal{M}_{\mathrm{P}}$) of pairs automaton-subset such that the instances from the class are in certain sense reducible to careful synchronization. The main construction of the present paper (i.e., the proof of Lemma 7) yields instances of subset synchronization that fit to this class. We use the following definitions:

- Given a PFA $A = (Q, X, \delta)$ and a carefully synchronizable subset $S \subseteq Q$, the *S-relevant part* of A is
$$Q_{A,S} = \bigcup_{w \in W_S} \delta(S, w)\,,$$
where W_S is the set of prefixes of careful reset words of S in A. The *S-relevant automaton* of A is $R_{A,S} = (Q_{A,S}, X, \delta_{A,S})$, where
$$\delta_{A,S}(s, x) = \begin{cases} \delta(s, x) & \text{if } \delta(s, x) \in Q_{A,S}, \\ \text{undefined} & \text{otherwise} \end{cases}$$
for each $s \in Q_{A,S}$ and $x \in X$.

- The class $\mathcal{M}_{\mathrm{P}}$ is defined as follows. For any PFA $A = (Q, X, \delta)$ and any carefully synchronizable $S \subseteq Q$, the pair $\langle A, S \rangle$ lies in $\mathcal{M}_{\mathrm{P}}$ if there are subsets $P_1, \ldots, P_{|S|} \subseteq Q$ such that:
  (a) The sets $P_1, \ldots, P_{|S|}$ are disjoint and $\bigcup_{i=1}^{|S|} P_i = Q_{A,S}$.
  (b) For each $v \in X^\star$ such that $\delta(s, u) \in Q_{A,S}$ for any prefix u of v and any $s \in S$, it holds that v is a careful reset word of S, or
  $$|\delta(S, v) \cap P_i| = 1$$
  for each $i = 1, \ldots, |S|$. In particular, the choice of empty v implies that
  $$|S \cap P_i| = 1$$
  must hold for each $i = 1, \ldots, |S|$.
- The class $\mathcal{C}_j^{\mathrm{R}}$ for $j \geq 0$ is defined as follows. For any PFA $A = (Q, X, \delta)$ and any carefully synchronizable $S \subseteq Q$, the pair $\langle A, S \rangle$ lies in $\mathcal{C}_j^{\mathrm{R}}$ if $R_{A,S} \in \mathcal{C}_j$.

Lemma 5. *For each $n \geq 1$ it holds that*

$$\mathrm{csub}^{\mathcal{M}_{\mathrm{P}}}(n) \leq \mathrm{car}(n)\,.$$

Moreover, for each $n, k, j \geq 1$ it holds that

$$\mathrm{csub}^{\mathcal{AL}_k \cap \mathcal{C}_j^{\mathrm{R}} \cap \mathcal{M}_{\mathrm{P}}}(n) \leq \mathrm{car}^{\mathcal{AL}_{k+1} \cap \mathcal{C}_j}(n)\,.$$

Proof. The first claim follows easily from the second. So, take any $\langle A, S \rangle \in \mathcal{M}_{\mathrm{P}}$ with $A = (Q, X_A, \delta_A)$ and $S \subseteq Q$, together with the sets $P_1, \ldots, P_{|S|}$ from the definition of $\mathcal{M}_{\mathrm{P}}$. By adding a letter α to the automaton $R_{A,S}$, we construct a carefully synchronizing PFA $B = (Q_{A,S}, X_B, \delta_B)$ with $\mathrm{car}(B) \geq \mathrm{csub}(A, S)$. Let $X_B = X_A \cup \{\alpha\}$. For each $s \in Q_{A,S}$ we find the i such that $s \in P_i$ and define

$$\delta_B(s, \alpha) = q_i,$$

where q_i is the only state lying in $S \cap P_i$, as guaranteed by the membership in $\mathcal{M}_{\mathrm{P}}$. The letters of X_A act in B as they do in $R_{A,S}$.

- It is easy to check that the automaton B is carefully synchronizing by αw for any $w \in X_A^\star$ that is a careful reset word of S in A.
- On the other hand, take a shortest careful reset word v of B. If α does not occur in v, then v is a careful reset word of S in A, so $|v| \geq \mathrm{csub}\,(A, S)$. Otherwise, denote $v = v_0 \alpha v_1$ where $v_0 \in X_B^\star$ and $v_1 \in X_A^\star$. By the membership in $\mathcal{M}_{\mathrm{P}}$ we have $|\delta(S, v_0) \cap P_i| = 1$ for each $i = 1, \ldots, |S|$ and thus $\delta_B(S, v_0 \alpha) = S$. It follows that v_1 is a careful reset word of S in A, so $|v| \geq \mathrm{csub}\,(A, S)$. □

4.4. *Decreasing the alphabet size*

The following method is quite simple and has been already used in the literature [2]. It modifies an automaton in order to decrease the alphabet size while preserving high synchronization thresholds.

Lemma 6. *For each $n, k \geq 1$ it holds that*

(i) $\mathrm{sub}^{\mathcal{AL}_k}(n) \leq \mathrm{sub}^{\mathcal{AL}_2}(k \cdot n)$ and $\mathrm{sub}^{\mathcal{AL}_k \cap \mathcal{SC}}(n) \leq \mathrm{sub}^{\mathcal{AL}_2 \cap \mathcal{SC}}(k \cdot n)$,
(ii) $\mathrm{car}^{\mathcal{AL}_k}(n) \leq \mathrm{car}^{\mathcal{AL}_2}(k \cdot n)$ and $\mathrm{car}^{\mathcal{AL}_k \cap \mathcal{SC}}(n) \leq \mathrm{car}^{\mathcal{AL}_2 \cap \mathcal{SC}}(k \cdot n)$.

Proof. Take a PFA $A = (Q_A, X_A, \delta_A)$ with $X_A = \{a_0, \ldots, a_m\}$. We define a PFA $B = (Q_B, X_B, \delta_B)$ as follows: $Q_B = Q_A \times X_A$, $X_B = \{\alpha, \beta\}$, and

$$\delta_B((s, a_i), \alpha) = \begin{cases} (\delta_A(s, a_i), a_0) & \text{if } \delta_A(s, a_i) \text{ is defined,} \\ \text{undefined} & \text{otherwise,} \end{cases}$$

$$\delta_B((s, a_i), \beta) = \begin{cases} (s, a_{i+1}) & \text{if } i < m, \\ (s, a_m) & \text{if } i = m \end{cases}$$

for each $i = 0, \ldots, m$. The construction of B applies to both the claims:

(i) Let A be a DFA. We choose a synchronizable $S_A \subseteq Q_A$ and denote $S_B = S_A \times \{a_0\}$. It is not hard to see that reset words of S_B in B are in a one-to-one correspondence with reset words of S_A in A. A word $a_{i_1} \ldots a_{i_d} \in X_A^\star$ corresponds to $(\beta^{i_1}\alpha) \ldots (\beta^{i_d}\alpha) \in X_B^\star$.
(ii) Let A be carefully synchronizing. We can suppose that $\delta_A(s, a_m)$ is defined on each $s \in Q_A$ since for a carefully synchronizing PFA there always exists such letter. For any careful reset word $a_{i_1} \ldots a_{i_d}$ of A, the word $\beta^m\alpha (\beta^{i_1}\alpha) \ldots (\beta^{i_d}\alpha)$ is a careful reset word of B. On the other hand, any careful reset word of B is also a careful reset word of the subset $Q_A \times \{a_0\} \subseteq Q_B$, whose careful reset words are in a one-to-one correspondence with careful reset words of A, like in the previous case.

Since $\delta_B((s, a_m), \alpha)$ is defined for each $s \in S_A$, it is not hard to check that if A is strongly connected, so is B. □

5. The New Lower Bounds

5.1. *The key construction*

Let us present the central construction of the present paper. We build a series of DFA with a constant-size alphabet and a constant structure of strongly connected components, together with subsets that require strongly exponential reset words. Moreover, the pairs automaton-subset are of the special kind represented by $\mathcal{M}_\mathrm{P}$, so a reduction to careful synchronization of PFA, as introduced in Lemma 5, is possible.

Lemma 7. *For infinitely many $m \geq 1$ it holds that*

$$\mathrm{sub}^{\mathcal{AL}_4 \cap \mathcal{C}_2 \cap \mathcal{C}_2^{\mathrm{R}} \cap \mathcal{M}_{\mathrm{P}}}(5m + \log m + 3) \geq 2^m \cdot (\log m + 1) + 1.$$

Proof. Suppose $m = 2^k$. For each $t \in 0, \ldots, m-1$ we denote by $\tau = \mathrm{bin}(t)$ the standard k-digit binary representation of t, i.e., a word from $\{\mathbf{0}, \mathbf{1}\}^k$. By a classical result proved in [8] there is a *De Bruijn sequence* $\xi = \xi_0 \ldots \xi_{m-1}$ consisting of letters $\xi_i \in \{\mathbf{0}, \mathbf{1}\}$ such that each word $\tau \in \{\mathbf{0}, \mathbf{1}\}^k$ appears exactly once as a cyclic factor of ξ (i.e., it is a factor or begins by a suffix of ξ and continues by a prefix of ξ). Let us fix such ξ. By $\pi(i)$ we denote the number t, whose binary representation $\mathrm{bin}(t)$ starts by ξ_i in ξ. Note that π is a permutation of $\{0, \ldots, m-1\}$. Set

$$\begin{aligned} Q &= \left(\{0, \ldots, m-1\} \times \left\{\mathbf{0}, \mathbf{0}^{\downarrow}, \mathbf{1}, \mathbf{1}^{\downarrow}, \mathbf{1}^{\uparrow}\right\}\right) \cup \left\{\mathrm{C}_0, \ldots, \mathrm{C}_k, \mathrm{D}, \overline{\mathrm{D}}\right\}, \\ X &= \{\mathbf{0}, \mathbf{1}, \kappa, \omega\}, \\ S &= (\{0, \ldots, m-1\} \times \{\mathbf{0}\}) \cup \{\mathrm{C}_0, \mathrm{D}\}. \end{aligned}$$

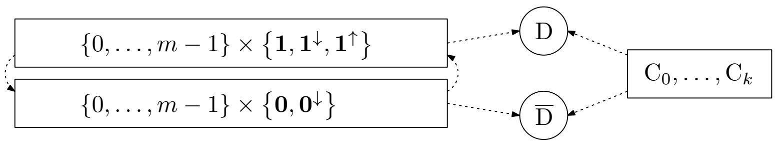

Fig. 1. A connectivity pattern of the automaton A.

Figure 1 visually distinguishes main parts of A. The states D and $\overline{\mathrm{D}}$ are sink states. Together with $\mathrm{D} \in S$ it implies that any reset word of S takes the states of S to D and that the state $\overline{\mathrm{D}}$ must not become active during its application. The states $\mathrm{C}_0, \ldots, \mathrm{C}_k$ guarantee that any reset word of S lies in

$$\left(\{\mathbf{0}, \mathbf{1}\}^k \kappa\right)^{\star} \omega X^{\star}. \tag{1}$$

Indeed, as defined by Fig. 2, no other word takes C_0 to D. Let the letter ω act as follows:

$$\begin{aligned} \{0, \ldots, m-1\} \times \{\mathbf{1}\}, \mathrm{C}_0, \mathrm{D} &\xrightarrow{\omega} \mathrm{D}, \\ \{0, \ldots, m-1\} \times \left\{\mathbf{0}, \mathbf{0}^{\downarrow}, \mathbf{1}^{\downarrow}, \mathbf{1}^{\uparrow}\right\}, \mathrm{C}_1, \ldots, \mathrm{C}_{\log m}, \overline{\mathrm{D}} &\xrightarrow{\omega} \overline{\mathrm{D}}. \end{aligned}$$

We see that ω maps each state to D or $\overline{\mathrm{D}}$. This implies that once ω occurs in a reset word of S, it must complete the synchronization. In order to map C_0 to D, the letter ω *must* occur, so any shortest reset word of S is exactly of the form

$$w = (\tau_1 \kappa) \ldots (\tau_d \kappa)\, \omega, \tag{2}$$

where $\tau_j \in \{\mathbf{0}, \mathbf{1}\}^k$ for each j.

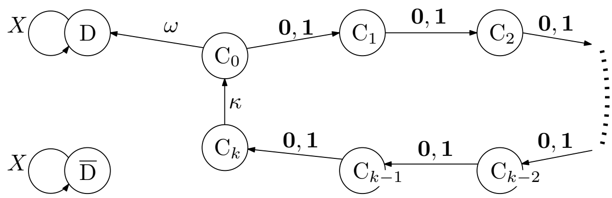

Fig. 2. A part of A. All the outgoing transitions that are not depicted lead to $\overline{\mathrm{D}}$.

The two biggest parts depicted by Fig. 1 are very similar to each other. The letters $\mathbf{0}$ and $\mathbf{1}$ act on them as follows:

$$(i,\mathbf{0}) \xrightarrow{\mathbf{0}} \begin{cases} (i+1,\mathbf{0}) & \text{if } \xi_i = \mathbf{0}, \\ \left(i+1,\mathbf{0}^{\downarrow}\right) & \text{if } \xi_i = \mathbf{1}, \end{cases} \qquad (i,\mathbf{1}) \xrightarrow{\mathbf{0}} \begin{cases} (i+1,\mathbf{1}) & \text{if } \xi_i = \mathbf{0}, \\ \left(i+1,\mathbf{1}^{\downarrow}\right) & \text{if } \xi_i = \mathbf{1}, \end{cases}$$

$$(i,\mathbf{0}) \xrightarrow{\mathbf{1}} \begin{cases} \overline{\mathrm{D}} & \text{if } \xi_i = \mathbf{0}, \\ (i+1,\mathbf{0}) & \text{if } \xi_i = \mathbf{1}, \end{cases} \qquad (i,\mathbf{1}) \xrightarrow{\mathbf{1}} \begin{cases} \left(i+1,\mathbf{1}^{\uparrow}\right) & \text{if } \xi_i = \mathbf{0}, \\ (i+1,\mathbf{1}) & \text{if } \xi_i = \mathbf{1}, \end{cases}$$

and $(i,\mathbf{b}) \xrightarrow{\mathbf{0},\mathbf{1}} (i+1,\mathbf{b})$ for each $\mathbf{b} = \mathbf{0}^{\downarrow}, \mathbf{1}^{\downarrow}, \mathbf{1}^{\uparrow}$, using the addition modulo m everywhere. For example, Fig. 3 depicts a part of A for $m = 8$ and for a particular De Bruijn sequence ξ. Figure 4 defines the action of κ on the states $\{i\} \times \left\{\mathbf{0}, \mathbf{0}^{\downarrow}, \mathbf{1}, \mathbf{1}^{\downarrow}, \mathbf{1}^{\uparrow}\right\}$ for any i, so the automaton A is completely defined.

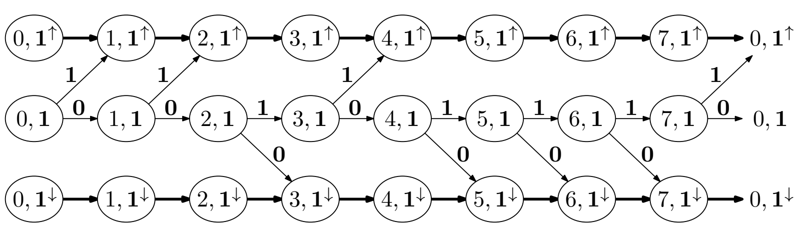

Fig. 3. A part of A assuming $m = 8$ and $\xi = \mathbf{00101110}$. Bold arrows represent both $\mathbf{0}, \mathbf{1}$.

Let w be a shortest reset word of S in A. It is necessarily of the form (2), so it makes sense to denote $v_t = \mathrm{bin}(t)\,\kappa$ and treat w as

$$w = v_{t_1} \ldots v_{t_d} \omega \in \{v_0, \ldots, v_{m-1}, \omega\}^{\star}. \tag{3}$$

The action of each v_t is depicted by Fig. 5. It is a key step of the proof to confirm that Fig. 5 is correct. Indeed:

- Starting from a state $(i, \mathbf{1})$, a word $\mathrm{bin}(t)$ takes us through a kind of decision tree to one of the states $\left(i+k, \mathbf{1}^{\downarrow}\right), (i+k, \mathbf{1}), \left(i+k, \mathbf{1}^{\uparrow}\right)$, depending on whether t is less than, equal to, or greater than $\pi(i)$, respectively. This is guaranteed by wiring the sequence ξ into the transition function, see Fig. 3. The letter κ then takes us back to $\{i\} \times \{\ldots\}$, namely to $(i, \mathbf{0})$ or $(i, \mathbf{1})$.

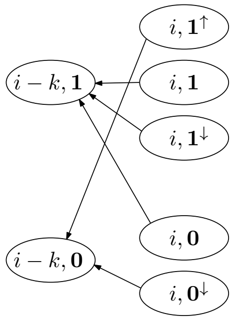

Fig. 4. The action of the letter κ, with subtraction modulo m.

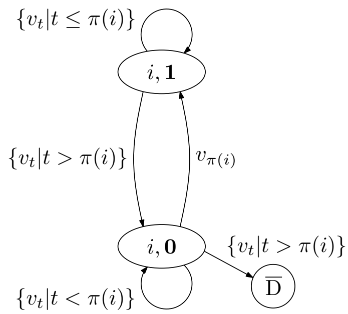

Fig. 5. The action of $v_0, \ldots, v_{m-1}$ on the i-th switch.

- Starting from a state $(i, \mathbf{0})$, we proceed similarly, but in the case of $t > \pi(i)$ we fall into $\overline{\mathrm{D}}$ during the application of $\mathrm{bin}(t)$.

It follows that after applying any prefix $v_{t_1} \ldots v_{t_j}$ of w, exactly one of the states $(i, \mathbf{0}), (i, \mathbf{1})$ is active for each i. We say that *the i-th switch is set to* $\mathbf{0}$ *or* $\mathbf{1}$ *at time j*. Note that $Q_A \backslash \{\overline{\mathrm{D}}\}$ is the S-relevant part of A and that the sets $\{i\} \times \{\mathbf{0}, \mathbf{0}^{\downarrow}, \mathbf{1}, \mathbf{1}^{\downarrow}, \mathbf{1}^{\uparrow}\}$ for $i = 0, \ldots, m-1$, together with the sets $\{\mathrm{D}\}$ and $\{\mathrm{C}_0, \ldots, \mathrm{C}_k\}$, can play the role of $P_1, \ldots, P_{m+2}$ in the definition of $\mathcal{M}_{\mathrm{P}}$.

Observe that at time d all the switches are necessarily set to $\mathbf{1}$ because otherwise the state $\overline{\mathrm{D}}$ would become active by the application of ω. On the other hand, at time 0 all the switches are set to $\mathbf{0}$. We are going to show that in fact during the synchronization of S the switches together perform a binary counting from 0 (all the switches set to $\mathbf{0}$) to $2^m - 1$ (all the switches set to $\mathbf{1}$). For each i the *significance* of the i-th switch is given by the value $\pi(i)$. So the $\pi^{-1}(m-1)$-th switch carries the most significant digit, the $\pi^{-1}(0)$-th switch carries the least significant digit and so on. The number represented in this manner by the switches at time j is denoted by $\mathfrak{b}_j \in \{0, \ldots, 2^m - 1\}$. We claim that $\mathfrak{b}_j = j$ for each j. Indeed:

- At time 0, all the switches are set to $\mathbf{0}$, we have $\mathfrak{b}_0 = 0$.
- Suppose that $\mathfrak{b}_{j'} = j'$ for each $j' \leq j-1$. We denote

$$\overline{t_j} = \min \{\pi(i) \mid i\text{-th switch is set to } \mathbf{0} \text{ at time } j-1\} \quad (4)$$

and claim that $t_j = \overline{t_j}$. Note that $\overline{t_j}$ is defined to be the least significance level at which there occurs a $\mathbf{0}$ in the binary representation of $\mathfrak{b}_{j-1}$. Suppose for a contradiction that $t_j > \overline{t_j}$. By the definition of $\overline{t_j}$ the state $\left(\pi^{-1}\left(\overline{t_j}\right), \mathbf{0}\right)$ lies in $\delta\left(S, v_{t_1} \ldots v_{t_{j-1}}\right)$. But v_{t_j} takes this state to $\overline{\mathrm{D}}$, which is a contradiction. Now suppose that $t_j < \overline{t_j}$. In such case the application of v_{t_j} does not turn any switch from $\mathbf{0}$ to $\mathbf{1}$, so $\mathfrak{b}_j \leq \mathfrak{b}_{j-1}$ and thus at time j the configuration of switches is the same at it was at time $\mathfrak{b}_j$. This contradicts the assumption that w is a shortest reset word. We have proved that $t_j = \overline{t_j}$ and it remains only to show that the application of v_{t_j} performs the addition of 1 and so makes the switches represent the value $\mathfrak{b}_{j-1}+1$.

(a) Consider an i-th switch with $\pi(i) < t_j$. By the definition of $\overline{t_j}$, it is set to **1** at time $j-1$ and the word v_{t_j} sets it to **0** at time j. This is what we need because such switches represent a continuous leading segment of **1**s in the binary representation of $\mathfrak{b}_{j-1}$.

(b) The $\pi^{-1}(t_j)$-th switch is set from **0** to **1** by the word v_{t_j}.

(c) Consider an i-th switch with $\pi(i) > t_j$. The switch represents a digit of $\mathfrak{b}_{j-1}$ which is more significant than the $\overline{t_j}$-th digit. As we expect, the word v_{t_j} leaves such switch unchanged.

Because $\mathfrak{b}_d = 2^m$, we deduce that $d = 2^m$ and thus $|w| = 2^m \cdot (k+1)+1$, assuming that a shortest reset word w exists. But in fact we have also shown that there is only one possibility for such w and that it is a true reset word for S. The unique w is of the form (3), where t_j is the position of the least significant **0** in the binary representation of $j-1$.

The automaton A lies in $\mathcal{C}_2 \cap \mathcal{C}_2^{\mathrm{R}}$ since the addition of $\mathrm{D} \longrightarrow \mathrm{C}_0$ and $\overline{\mathrm{D}} \longrightarrow (0, \mathbf{0})$ makes A strongly connected, while the addition of $\mathrm{D} \longrightarrow \mathrm{C}_0$ and $\mathrm{C}_0 \longrightarrow (0, \mathbf{0})$ makes $R_{A,S}$ strongly connected. □

5.2. *The results*

The following theorem presents the main results of the present paper:

Theorem 8. *For infinitely many $n \geq 1$ it holds that*

(i) $\mathrm{sub}^{\mathcal{AL}_2 \cap \mathcal{SC}}(n) \geq 2^{\frac{n}{61}}$,

(ii) $\mathrm{car}^{\mathcal{AL}_2 \cap \mathcal{SC}}(n) \geq 2^{\frac{n}{36}}$.

Proof. Lemma 7 says that

$$2^m \cdot (\log m + 1) + 1 \leq \mathrm{sub}^{\mathcal{AL}_4 \cap \mathcal{C}_2 \cap \mathcal{C}_2^{\mathrm{R}} \cap \mathcal{M}_{\mathrm{P}}}(5m + \log m + 3) \tag{5}$$

for infinitely many $m \geq 1$. Now we apply some of the lemmas from Sec. 4:

(i) Lemma 4 extends (5) with

$$\mathrm{sub}^{\mathcal{AL}_4 \cap \mathcal{C}_2}(5m + \log m + 3) \leq \mathrm{sub}^{\mathcal{AL}_6 \cap \mathcal{SC}}(10m + 2 \cdot \log m + 8) - 1$$

and Lemma 6 adds

$$\mathrm{sub}^{\mathcal{AL}_6 \cap \mathcal{SC}}(10m + 2 \cdot \log m + 8) - 1 \leq \mathrm{sub}^{\mathcal{AL}_2 \cap \mathcal{SC}}(60m + 12 \cdot \log m + 48) - 1.$$

We chain the three inequalities and deduce

$$\begin{aligned} \mathrm{sub}^{\mathcal{AL}_2 \cap \mathcal{SC}}(60m + 12 \cdot \log m + 48) &\geq 2^m \cdot (\log m + 1) + 2, \\ \mathrm{sub}^{\mathcal{AL}_2 \cap \mathcal{SC}}(61m) &\geq 2^m, \\ \mathrm{sub}^{\mathcal{AL}_2 \cap \mathcal{SC}}(n) &\geq 2^{\frac{n}{61}}. \end{aligned}$$

(ii) Lemma 5 extends (5) with

$$\mathrm{csub}^{\mathcal{AL}_4 \cap \mathcal{C}_2^{\mathrm{R}} \cap \mathcal{M}_{\mathrm{P}}}(5m + \log m + 3) \leq \mathrm{car}^{\mathcal{AL}_5 \cap \mathcal{C}_2}(5m + \log m + 3)\,,$$

while Lemma 2 adds

$$\mathrm{car}^{\mathcal{AL}_5 \cap \mathcal{C}_2}(5m + \log m + 3) \leq \mathrm{car}^{\mathcal{AL}_7 \cap \mathcal{SC}}(5m + \log m + 3)$$

and Lemma 6 adds

$$\mathrm{car}^{\mathcal{AL}_7 \cap \mathcal{SC}}(5m + \log m + 3) \leq \mathrm{car}^{\mathcal{AL}_2 \cap \mathcal{SC}}(35m + 7 \cdot \log m + 21)\,.$$

We chain the four inequalities and deduce:

$$\mathrm{car}^{\mathcal{AL}_2 \cap \mathcal{SC}}(35m + 7 \cdot \log m + 21) \geq 2^m \cdot (\log m + 1) + 1,$$
$$\mathrm{car}^{\mathcal{AL}_2 \cap \mathcal{SC}}(36m) \geq 2^m,$$
$$\mathrm{car}^{\mathcal{AL}_2 \cap \mathcal{SC}}(n) \geq 2^{\frac{n}{36}}.$$ □

Note that there are more subtle results for less restricted classes of automata:

Proposition 9. *It holds that* $\mathrm{sub}^{\mathcal{AL}_2}(n) \geq 2^{\frac{n}{21}}$, $\mathrm{car}^{\mathcal{AL}_2}(n) \geq 2^{\frac{n}{26}}$, $\mathrm{sub}^{\mathcal{SC}}(n) \geq 3^{\frac{n}{6}}$, *and* $\mathrm{car}^{\mathcal{SC}}(n) \geq 3^{\frac{n}{3}}$ *for infinitely many* $n \geq 1$.

Proof. The first claim follows easily from Lemmas 7 and 6, the second one requires also using Lemma 5 first. The third and the last claim follow from applying Lemmas 1 and 4 (or Lemma 2 respectively) to the construction from [18]. □

6. Consequences and Reformulations

6.1. *Computational problems*

It is well known that the decision about synchronizability of a given DFA is a polynomial time task, even if we also require an explicit reset word on the output. A lot of work has been done on such algorithms in effort to make them produce short reset words in short running time. However, it has been proven that it is both NP-hard and coNP-hard (it is actually DP-complete) to recognize the length of *shortest* reset words for a given DFA, while it is still NP-hard to recognize its upper bounds or to approximate it with a constant factor, see references in [21] and [2].

On the other hand, there has not been done much research in computational complexity of problems concerning subset synchronization and careful synchronization, although they does not seem to have less chance to emerge in practice. Namely, the first natural problems in these directions are

Subset synchronizability
Input: n-state DFA $A = (Q, X, \delta)$, $S \subseteq Q$
Output: is there some $w \in X^\star$ such that $|\delta(S, w)| = 1$?

CAREFUL SYNCHRONIZABILITY
Input: n-state PFA $A = (Q, X, \delta)$,
Output: is there some $w \in X^\star$ such that $(\exists r \in Q)\,(\forall s \in Q)\,\delta(s, w) = r$?

Both these problems, in contrast to the synchronizability of DFA, are known to be PSPACE-complete:

Theorem 10. [20, 24] SUBSET SYNCHRONIZABILITY *is a PSPACE-complete problem.*

Theorem 11. [17] CAREFUL SYNCHRONIZABILITY *is a PSPACE-complete problem.*

Note that such hardness is not a consequence of any lower bound of synchronization thresholds, because an algorithm does not need to produce an explicit reset word. The proofs of both the theorems above make use of a result of Kozen [15], which establishes that it is PSPACE-complete to decide if given finite acceptors with a common alphabet accept a common word. This problem is polynomially reduced to our problems using the idea of two sink states. Is it possible to avoid the non-connectivity here?

In the case of CAREFUL SYNCHRONIZABILITY, the simple trick from Lemma 2 easily reduces the general problem to the variant restricted to strongly connected automata, and it turns out that the method of swap congruences is general enough to perform such reduction also in the case of SUBSET SYNCHRONIZABILITY:

Theorem 12. *The following problems are PSPACE-complete:*

(i) SUBSET SYNCHRONIZABILITY restricted to binary strongly connected DFA
(ii) CAREFUL SYNCHRONIZABILITY restricted to binary strongly connected PFA.

Proof. There are polynomial reductions from the general problems SUBSET SYNCHRONIZABILITY and CAREFUL SYNCHRONIZABILITY: Perform the construction from Lemma 4 (or Lemma 2 respectively) and then the one from Lemma 6. □

6.2. *Synchronization thresholds of NFA*

In 1999, Imreh and Steinby [13] introduced three different synchronization thresholds concerning general non-deterministic finite automata (NFA). We define an NFA as a pair $A = (Q, X, \delta)$ where Q is a finite set of states, X is a finite alphabet and $\delta : Q \times X \to 2^Q$ is a total function, extended in the canonical way to $\delta : Q \times X^\star \to 2^Q$. For any $S \subseteq Q$ we denote $\delta(S, w) = \bigcup_{s \in S} \delta(s, w)$.

The key definitions are the following. For an NFA A, a word $w \in X^\star$ is:

- *D1-directing* if there is $r \in Q$ such that $\delta_A(s, w) = \{r\}$ for each $s \in Q$,
- *D2-directing* if $\delta_A(s_1, w) = \delta_A(s_2, w)$ for each $s_1, s_2 \in Q$,
- *D3-directing* if there is $r \in Q$ such that $r \in \delta_A(s, w)$ for each $s \in Q$.

By $\mathrm{d}_1(A), \mathrm{d}_2(A), \mathrm{d}_3(A)$ we denote the length of shortest D1-, D2-, and D3-directing words for A, or 0 if there is no such word. By $\mathrm{d}_1(n), \mathrm{d}_2(n), \mathrm{d}_3(n)$ we denote the maximum values of $\mathrm{d}_1(A), \mathrm{d}_2(A), \mathrm{d}_3(A)$ taken over all NFA A with at most n states. Possible restrictions are marked by superscripts as usual.

It is clear that PFA are a special kind of NFA. Any careful reset word of a PFA A is D1-, D2-, and D3-directing. On the other hand, any D1- or D3-directing word of a PFA is a careful reset word. Thus, we get

$$\mathrm{d}_2^{\mathrm{PFA}}(n) \leq \mathrm{d}_1^{\mathrm{PFA}}(n) = \mathrm{d}_3^{\mathrm{PFA}}(n) = \mathrm{car}(n)$$

and

$$\mathrm{d}_1(n) \geq \mathrm{car}(n), \qquad \mathrm{d}_3(n) \geq \mathrm{car}(n). \tag{6}$$

Note that a D2-directing word w of a PFA A is either a careful reset word of A or satisfies that $\delta(\{s\}, w) = \emptyset$ for each $s \in Q$. PFA are of a special importance for the threshold $\mathrm{d}_3(n)$ since due to a key lemma from [14], for any n-state NFA A there is a n-state PFA B such that $\mathrm{d}_3(B) \geq \mathrm{d}_3(A)$, so we have $\mathrm{d}_3(n) = \mathrm{d}_3^{\mathrm{PFA}}(n)$ for each n.

It is known that $\mathrm{d}_1(n) = \Omega(2^n)$ [14] and $\mathrm{d}_3(n) = \Omega\left(3^{\frac{n}{3}}\right)$ [18] (for upper bounds and further details see [9]). Due to the easy relationship similar to (6), our strongly exponential lower bounds apply directly to the thresholds $\mathrm{d}_1(n)$ and $\mathrm{d}_3(n)$ with the restriction to binary strongly connected NFA:

$$\mathrm{d}_1^{\mathcal{AL}_2 \cap \mathcal{SC}}(n) = 2^{\Omega(n)}, \qquad \mathrm{d}_3^{\mathcal{AL}_2 \cap \mathcal{SC}}(n) = 2^{\Omega(n)}.$$

6.3. *Compositional depths*

It has been pointed out by Arto Salomaa [23] in 2001 that very little is known about the minimum length of a composition needed to generate a function by a given set of generators. To be more precise, let us adopt and slightly extend the notation used in [23]. We denote by $\mathcal{T}_n$ the semigroup of all functions from $\{1, \ldots, n\}$ to itself. Given $\mathbf{G} \subseteq \mathcal{T}_n$, we denote by $\langle \mathbf{G} \rangle$ the subsemigroup generated by $\mathbf{G}$. Given $\mathbf{F} \subseteq \mathcal{T}_n$, we denote by $\mathrm{D}(\mathbf{G}, \mathbf{F})$ the length k of a shortest sequence $g_1, \ldots, g_k$ of functions from $\mathbf{G}$ such that $g_1 \circ \cdots \circ g_k \in \mathbf{F}$. Finally, denote

$$\mathrm{D}_n = \max_{\overline{n} \leq n} \max_{\substack{\mathbf{F}, \mathbf{G} \subseteq \mathcal{T}_{\overline{n}} \\ \mathbf{F} \cap \langle \mathbf{G} \rangle \neq \emptyset}} \mathrm{D}(\mathbf{G}, \mathbf{F}). \tag{7}$$

Note that in Group Theory, thresholds like D_n are studied in the scope of permutations, see [12].

From basic connections between automata and transformation semigroups it follows that various synchronization thresholds can be defined alternatively by putting additional restrictions to the space of considered sets $\mathbf{G}$ and $\mathbf{F}$ in the definition (7) of the threshold D_n:

(i) For the basic synchronization threshold of DFA (may be denoted by $\mathrm{car}^{\mathrm{DFA}}(n)$), we restrict $\mathbf{F}$ to be exactly the set of $\overline{n}$-ary constant functions.

Recall that a set $\mathbf{G} \subseteq \mathcal{T}_{\overline{n}}$ corresponds to a DFA $A = (\{1, \ldots \overline{n}\}, X, \delta)$: Each $g \in \mathbf{G}$ just encodes the action of certain $x \in X$. Finding a reset word of A then equals composing transitions from $\mathbf{G}$ in order to get a constant.

(ii) For the threshold $\mathrm{sub}(n)$, we restrict $\mathbf{F}$ to be some of the sets

$$\mathbf{F}_S = \{f \in \mathcal{T}_{\overline{n}} \mid (\forall r, s \in S)\, f(r) = f(s)\}$$

for $S \subseteq \{1, \ldots, \overline{n}\}$. Therefore it holds that $\mathrm{D}_n \geq \mathrm{sub}(n)$.

(iii) For $\mathrm{car}(n)$, we should consider an alternative formalism for PFA, where the "undefined" transitions lead to a special error sink state. Let the largest number stand for the error state. A careful reset word should map all the states except for the error state to one particular non-error state. So, here we restrict

$$\mathbf{F} = \{f \in \mathcal{T}_{\overline{n}} \mid (\forall r, s \in \{1, \ldots, \overline{n} - 1\})\, f(r) = f(s) \neq \overline{n}\},$$
$$\mathbf{G} \subseteq \{g \in \mathcal{T}_n \mid g(n) = n\}.$$

However, in the canonical formalism such $\mathbf{G} \subseteq \mathcal{T}_n$ corresponds to a $(n-1)$-state PFA, so we get $\mathrm{D}_n \geq \mathrm{car}(n-1)$. Allowing suitable sets $\mathbf{F}_S$ for $S \subseteq \{1, \ldots, \overline{n} - 1\}$, we get $\mathrm{D}_n \geq \mathrm{csub}(n-1)$ as well.

Arto Salomaa refers to a single nontrivial bound of D_n, namely $\mathrm{D}_n \geq (\sqrt[3]{n})!$. In fact, he omits a construction of Kozen [15, Theorem 3.2.7] from 1977, which deals with lengths of *proofs* rather than compositions but witnesses easily that $\mathrm{D}_n = 2^{\Omega\left(\frac{n}{\log n}\right)}$. However, the lower bound of $\mathrm{car}(n)$ from [14] revealed soon that $\mathrm{D}_n = 2^{\Omega(n)}$.

Like in the case of $\mathrm{car}(n)$ and $\mathrm{sub}(n)$, the notion of D_n does not concern the size of $\mathbf{G}$, thus providing a ground for artificial series of bad cases based on growing alphabets. Our results show that actually the growing size of $\mathbf{G}$ is not necessary: a strongly exponential lower bound of D_n holds even if we restrict $\mathbf{G}$ to any nontrivial fixed size.

7. Conclusions and Future Work

We have proved that both the considered thresholds (subset synchronization of DFA and careful synchronization of PFA) are strongly exponential even under two heavy restrictions (binary alphabets and strong connectivity). We have improved the lower bounds of Martyugin, 2013 [19]. However, the multiplicative constants in the exponents do not seem to be the largest possible.

For now there is no method giving upper bounds concerning the alphabet size, so it may happen that binary cases are the hardest possible. Such situation appears in the classical synchronization of DFA if the Černý Conjecture holds.

From a more general viewpoint, our results give a partial answer to the informal question: "*Which features of automata are needed for obtaining strongly exponential thresholds?*" However, for many interesting restrictions we do not even know whether the corresponding thresholds are superpolynomial. Namely, such restricted classes include monotonic and aperiodic automata, cyclic and one-cluster automata, Eulerian automata, commutative automata and others. For each such class it is also an open question whether Subset synchronizability or Careful synchronizability is solvable in polynomial time with the corresponding restriction.

As it was noted before, for the general threshold car(n) there is a gap between $\mathcal{O}\left(n^2 \cdot 4^{\frac{n}{3}}\right)$ and $\Omega\left(3^{\frac{n}{3}}\right)$, which is subject to an active research.

References

[1] R. Adler, L. Goodwyn and B. Weiss, Equivalence of topological Markov shifts, *Israel Journal of Mathematics* **27**(1) (1977) 49–63.

[2] M. V. Berlinkov, Approximating the minimum length of synchronizing words is hard, *Theory of Computing Systems* **54**(2) (2014) 211–223.

[3] M. V. Berlinkov, On two algorithmic problems about synchronizing automata, *Developments in Language Theory*, eds. A. Shur and M. Volkov, *Lecture Notes in Computer Science* **8633** (Springer International Publishing, 2014), pp. 61–67.

[4] P. Bonizzoni and N. Jonoska, Existence of constants in regular splicing languages, *Information and Computation* **242** (2015) 340–353.

[5] H. Burkhard, Zum Längenproblem homogener Experimente an determinierten und nicht-deterministischen Automaten, *Elektronische Informationsverarbeitung und Kybernetik* **12**(6) (1976) 301–306.

[6] J. Černý, A. Pirická and B. Rosenauerová, On directable automata, *Kybernetica* **7** (1971) 289–298.

[7] B. Delyon and O. Maler, On the effects of noise and speed on computations, *Theoretical Computer Science* **129**(2) (1994) 279–291.

[8] C. Flye Sainte-Marie, Solution to question nr. 48, *L'intermédiaire des Mathématicians* **1** (1894) 107–110.

[9] Z. Gazdag, S. Iván and J. Nagy-György, Improved upper bounds on synchronizing nondeterministic automata, *Inf. Process. Lett.* **109** (2009) 986–990.

[10] F. Gonze, R. M. Jungers and A. Trakhtman, A note on a recent attempt to improve the Pin-Frankl bound, *Discrete Mathematics & Theoretical Computer Science* **17**(1) (2015) 307–308.

[11] P. Goralčík, Z. Hedrlín, V. Koubek and J. Ryšlinková, A game of composing binary relations, *RAIRO - Theoretical Informatics and Applications - Informatique Theorique et Applications* **16**(4) (1982) 365–369.

[12] H. A. Helfgott and Á. Seress, On the diameter of permutation groups, *Annals of Mathematics* **179** (2014) 611–658.

[13] B. Imreh and M. Steinby, Directable nondeterministic automata, *Acta Cybern.* **14** (1999) 105–115.

[14] M. Ito and K. Shikishima-Tsuji, Some results on directable automata, *Theory is Forever*, eds. J. Karhumäki *et al.*, *Lecture Notes in Computer Science* **3113** (Springer Berlin Heidelberg, 2004), pp. 125–133.

[15] D. Kozen, Lower bounds for natural proof systems, *Proceedings of the* 18*th Annual Symposium on Foundations of Computer Science* (IEEE Computer Society, Washington, 1977), pp. 254–266.

[16] D. Lee and M. Yannakakis, Testing finite-state machines: State identification and verification, *Computers, IEEE Transactions on* **43**(3) (1994) 306–320.

[17] P. V. Martugin, Computational complexity of certain problems related to carefully synchronizing words for partial automata and directing words for nondeterministic automata, *Theory of Computing Systems* **54**(2) (2014) 293–304.

[18] P. V. Martyugin, A lower bound for the length of the shortest carefully synchronizing words, *Russian Mathematics* **54**(1) (2010) 46–54.

[19] P. V. Martyugin, Careful synchronization of partial automata with restricted alphabets, *Computer Science - Theory and Applications*, eds. A. A. Bulatov and A. M. Shur, *Lecture Notes in Computer Science* **7913** (Springer Berlin Heidelberg, 2013), pp. 76–87.

[20] B. K. Natarajan, An algorithmic approach to the automated design of parts orienters, *Proceedings of the* 27*th Annual Symposium on Foundations of Computer Science* (IEEE Computer Society, Washington, 1986), pp. 132–142.

[21] J. Olschewski and M. Ummels, The complexity of finding reset words in finite automata, *Mathematical Foundations of Computer Science* 2010, eds. P. Hliněný and A. Kučera, *Lecture Notes in Computer Science* **6281** (Springer Berlin Heidelberg, 2010), pp. 568–579.

[22] J.-E. Pin, On two combinatorial problems arising from automata theory, *Annals of Discrete Mathematics* **17** (1983) 535–548.

[23] A. Salomaa, Compositions over a finite domain: From completeness to synchronizable automata, *A Half-Century of Automata Theory* (World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2001), pp. 131–143.

[24] S. Sandberg, Homing and synchronizing sequences, *Model-Based Testing of Reactive Systems*, eds. M. Broy *et al.*, *Lecture Notes in Computer Science* **3472** (Springer Berlin Heidelberg, 2005), pp. 5–33.

[25] A. Trahtman, The road coloring problem, *Israel Journal of Mathematics* **172**(1) (2009) 51–60.

[26] A. Trahtman, Modifying the upper bound on the length of minimal synchronizing word, *Fundamentals of Computation Theory*, eds. O. Owe, M. Steffen and J. A. Telle, *Lecture Notes in Computer Science* **6914** (Springer Berlin Heidelberg, 2011), pp. 173–180.

[27] N. Travers and J. Crutchfield, Exact synchronization for finite-state sources, *Journal of Statistical Physics* **145**(5) (2011) 1181–1201.

[28] M. V. Volkov, Synchronizing automata and the Černý conjecture, *Language and Automata Theory and Applications*, eds. C. Martín-Vide *et al.*, *Lecture Notes in Computer Science* **5196** (Springer Berlin Heidelberg, 2008), pp. 11–27.