Synchronizing Automata and the Černý Conjecture

Mikhail V. Volkov

Department of Mathematics and Mechanics,
Ural State University, 620083 Ekaterinburg, Russia
Mikhail.Volkov@usu.ru

Abstract. We survey several results and open problems related to synchronizing automata. In particular, we discuss some recent advances towards a solution of the Černý conjecture.

1 History and Motivations

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a deterministic finite automaton (DFA), where Q denotes the state set, Σ stands for the input alphabet, and $\delta : Q \times \Sigma \to Q$ is the transition function defining an action of the letters in Σ on Q. The action extends in a unique way to an action $Q \times \Sigma^* \to Q$ of the free monoid Σ^* over Σ; the latter action is still denoted by δ. The automaton $\mathscr{A}$ is called *synchronizing* if there exists a word $w \in \Sigma^*$ whose action resets $\mathscr{A}$, that is to leave the automaton in one particular state no matter which state in Q it started at: $\delta(q, w) = \delta(q', w)$ for all $q, q' \in Q$. Any word w with this property is said to be a *reset* word for the automaton.

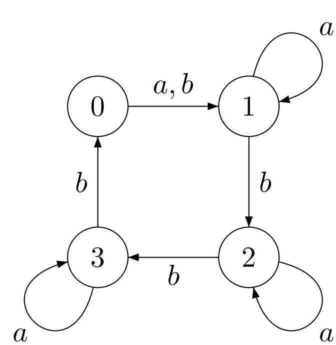

Fig. 1. A synchronizing automaton

Fig. 1 shows an example of a synchronizing automaton with 4 states. The reader can easily verify that the word ab^3ab^3a resets the automaton leaving it in the state 1. With somewhat more effort one can also check that ab^3ab^3a is the shortest reset word for this automaton. The example in Fig. 1 is due to Černý, a Slovak computer scientist, in whose pioneering paper (1964) the notion of a synchronizing automaton explicitly appeared for the first time. (Černý called such automata *directable*. The word *synchronising* in this context was probably introduced by Hennie (1964).) Implicitly, however, this concept has been around since the earliest days of automata theory. The very first synchronizing automaton that we were able to trace back in the literature appeared in Ashby's classic book (1956, pp. 60–61). There Ashby presents a puzzle dealing with taming two ghostly noises, Singing and Laughter, in a haunted mansion. Each of the noises can be either on or off, and their behaviour depends on combinations of two possible actions, playing the organ or burning incense. Under

C. Martín-Vide, F. Otto, and H. Fernau (Eds.): LATA 2008, LNCS 5196, pp. 11–27, 2008.

a suitable encoding, this leads to the following automaton with 4 states and 4 input letters:

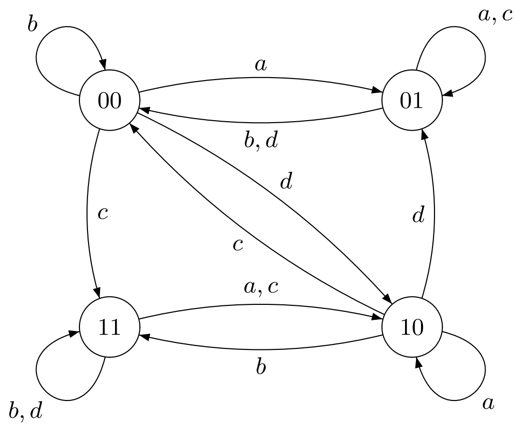

Fig. 2. Ashby's "ghost taming" automaton

Here 00 encodes the state when both Singing and Laughter are silent, 01 stands for the state when Singing is of but Laughter is on, etc. Similarly, a stands for the transition that happens when neither the organ is played nor incense is burned, b encodes the transition caused by organ-playing in the absence of incense-burning, etc. The problem is to ensure silence, in other words, to bring the automaton in Fig. 2 to the state 00. Ashby only solves the problem under the assumption that the automaton is in the state 11 and his suggested solution is encoded by the word acb. However, it is easy to check that acb is in fact a reset word for the automaton so applying the corresponding sequence of actions will get the house quiet from any initial configuration. It is not clear whether or not Ashby realized this nice feature of his automaton, and moreover, the fact that Ashby's automaton is synchronizing seems to be overlooked for many years.

Let us return to the genesis of the concept of synchronizing automata. In (Černý, 1964) this notion arose within the classic framework of Moore's "Gedanken-experiments" (1956). For Moore and his followers finite automata served as a mathematical model of devices working in discrete mode, such as computers or relay control systems. This leads to the following natural problem: how can we restore control over such a device if we do not know its current state but can observe outputs produced by the device under various actions? Moore (1956) has shown that under certain conditions one can uniquely determine the state at which the automaton arrives after a suitable sequence of actions (called an *experiment*). Moore's experiments were adaptive, that is, each next action was selected on the basis of the outputs caused by the previous actions. Ginsburg (1958) considered more restricted experiments that he called *uniform*. A uniform experiment[1] is just a fixed sequence of actions, that is, a word over

[1] After (Gill, 1961), the name *homing sequence* has become standard for the notion.

the input alphabet; thus, in Ginsburg's experiments outputs were only used for calculating the resulting state at the end of an experiment. From this, just one further step was needed to come to the setting in which outputs were not used at all. It should be noted that this setting is by no means artificial—there exist many practical situations when it is technically impossible to observe output signals. (Think of a satellite which loops around the Moon and cannot be controlled from the Earth while "behind" the Moon.)

It is not surprising that synchronizing automata were re-invented a number of times. First of all, the notion was very natural by itself and fitted fairly well in what was considered as the mainstream of automata theory in the 1960s. Second, Černý's paper (1964) published in Slovak language remained unknown in the English-speaking world for some time. As examples, we mention here the report (Laemmel & Rudner, 1969) and the paper (Fischler & Tannenbaum, 1970) both rediscovering results from (Černý, 1964). The books (Booth, 1967; Hennie, 1968; Kohavi, 1970) also present some information about synchronizing automata but do not refer to (Černý, 1964). It seems that the situation begun to change only in 1972 when the English translation of the book (Starke, 1969) appeared.

The original "Gedanken-experiments" motivation for studying synchronizing automata is still of importance, and reset words are frequently applied in model-based testing of reactive systems[2]. Rather unexpectedly, an additional source of problems related to synchronizing automata has come from *robotics* or, more precisely, from part handling problems in industrial automation such as part feeding, fixturing, loading, assembly and packing. Within this framework, the concept of a synchronizing automaton was again rediscovered in the mid-1980s by Natarajan (1986, 1989). We explain how abstract automata arise in part handling problems by means of a simple illustrative example from (Ananichev & Volkov, 2004).

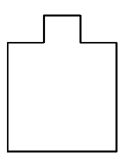

Fig. 3. A part

Suppose that a part of a certain device has the shape shown in Fig. 3. Such parts arrive at manufacturing sites in boxes and they need to be sorted and oriented before assembly. For simplicity, assume that only four initial orientations of the part shown in Fig. 3 are possible, namely, the four shown in in Fig. 4.

Fig. 4. Four possible orientations

Further, suppose that prior the assembly the part should take the "bump-left" orientation (the second one in Fig 4). Thus, one has to construct an orienter

[2] See (Cho et al, 1993; Boppana et al, 1999) as typical samples of technical contributions to the area and (Sandberg, 2005) for a recent survey.

which action will put the part in the prescribed position independently of its initial orientation.

Of course, there are many ways to design such an orienter but practical considerations favor methods which require little or no sensing, employ simple devices, and are as robust as possible. For our particular case, these goals can be achieved as follows. We put parts to be oriented on a conveyer belt which takes them to the assembly point and let the stream of the parts encounter a series of passive obstacles placed along the belt. We need two type of obstacles: high and low. A high obstacle should be high enough in order that any part on the belt encounters this obstacle by its rightmost low angle (we assume that the belt is moving from left to right). Being curried by the belt, the part then is forced to turn 90° clockwise. A low obstacle has the same effect whenever the part is in the "bump-dow" orientation (the first one in Fig. 4); otherwise it does not touch the part which therefore passes by without changing the orientation.

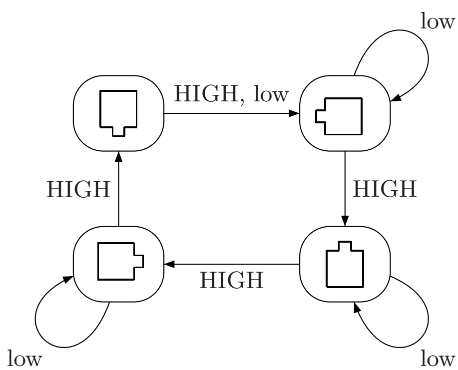

Fig. 5. The action of the obstacles

The scheme in Fig. 5 summarizes how the aforementioned obstacles effect the orientation of the part. The reader immediately recognizes the synchronizing automaton from Fig. 1. Remembering that its shortest reset word is the word ab^3ab^3a, we conclude that the series of obstacles

low–HIGH–HIGH–HIGH–low–HIGH–HIGH–HIGH–low

yields the desired sensor-free orienter.

Since the 1990s synchronizing automata usage in the area of robotic manipulation has grown into a prolific research direction but it is fair to say that publications in this direction deal mostly with implementation technicalities. However, amongst them there are papers of theoretical importance such as (Eppstein, 1990; Goldberg, 1993; Chen & Ierardi, 1994).

Speculating about further possible applications of synchronizing automata, one can think of *biocomputing*. Here we refer to recent experiments (Benenson et al, 2001, 2003) in which DNA molecules have been used as both hardware and software for finite automata of nanoscaling size. For instance, Benenson et al (2003) produced a “soup of automata”, that is, a solution containing 3×10^{12} identical automata per μl. All these molecular automata can work in parallel on different inputs, thus ending up in different and unpredictable states. In contrast to an electronic computer, one cannot reset such a system by just pressing a button; instead, in order to synchronously bring each automaton to its “ready-to-restart” state, one should spice the soup with (sufficiently many copies of) a DNA molecule whose nucleotide sequence encodes a reset word.

Clearly, from the viewpoint of applications, real or yet imaginary, algorithmic and complexity issues are of crucial importance. We discuss them in Section 2.

Putting applications aside, mathematicians since the 1960s have intensively studied synchronizing automata *per se*, as an interesting combinatorial object. These studies are mainly motivated by the *Černý conjecture*. Černý (1964) constructed for each $n > 1$ a synchronizing automaton $\mathscr{C}_n$ with n states which shortest reset word has length $(n-1)^2$ (the automaton in Fig. 1 is $\mathscr{C}_4$). Soon after that he conjectured that those automata represent the worst possible case, that is, every synchronizing automaton with n states can be reset by a word of length $(n-1)^2$. By now this simply looking conjecture is arguably the most longstanding open problem in the combinatorial theory of finite automata. We will discuss the Černý conjecture and some related partial results in Section 3.

Other mathematical motivations for studying synchronizing automata come from semigroup theory (see Ananichev & Volkov, 2004), multiple-valued logic and symbolic dynamics (see Mateescu & Salomaa, 1999). The latter connection is especially interesting in view of a recent breakthrough in the area—a (positive) solution to the Road Coloring Problem found by Trahtman (2008), but it clearly deserves a separate survey.

2 Algorithms and Complexity

It should be clear that not every DFA is synchronizing. Therefore, the very first question that we should address is the following one: *given an automaton $\mathscr{A}$, how to determine whether or not $\mathscr{A}$ is synchronizing?*

This question is in fact quite easy, and the most straightforward solution to it can be achieved via the classic power automaton construction. Recall that the *power automaton* $\mathcal{P}(\mathscr{A})$ of a given DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ has the collection $\mathcal{P}'(Q)$ of the non-empty subsets of Q as the state set and the natural extension of δ to the set $\mathcal{P}'(Q) \times \Sigma$ as the transition function (still denoted by δ). In other words, for P being a non-empty subset of Q and $a \in \Sigma$, one sets $\delta(P, a) = \{\delta(p, a) \mid p \in P\}$. Fig. 6 presents the power automaton for the DFA $\mathscr{C}_4$ shown in Fig. 1.

Now it is obvious that a word $w \in \Sigma^*$ is a reset word for the DFA $\mathscr{A}$ if and only if w labels a path in $\mathcal{P}(\mathscr{A})$ starting at Q and ending at a singleton. (For instance, the bold path in Fig. 6 represents the shortest reset word ab^3ab^3a of the automaton $\mathscr{C}_4$.) Thus, the question of whether or not a given DFA $\mathscr{A}$ is synchronizing reduces to the following reachability question in the underlying digraph of the power automaton $\mathcal{P}(\mathscr{A})$: is there a path from Q to a singleton? The latter question can be easily answered by breadth-first search (see, e.g., Corman et al, 2001, Section 22.2).

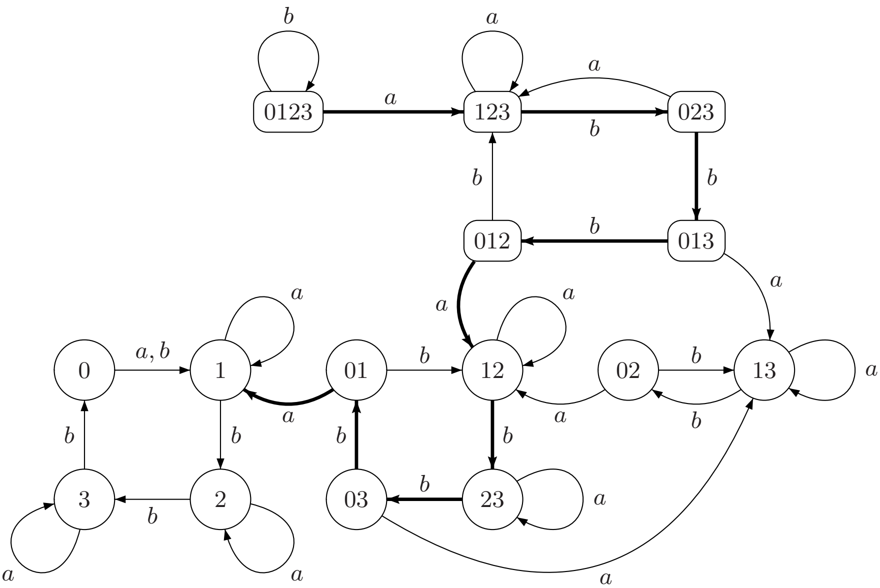

Fig. 6. The power automaton $\mathcal{P}(\mathscr{C}_4)$

The described procedure is conceptually very simple but rather inefficient because the power automaton $\mathcal{P}(\mathscr{A})$ is exponentially larger than $\mathscr{A}$. However, the following criterion of synchronizability (Černý, 1964, Theorem 2) gives rise to a polynomial algorithm.

Proposition 1. *A DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ is synchronizing if and only if for every $q, q' \in Q$ there exists a word $w \in \Sigma^*$ such that $\delta(q, w) = \delta(q', w)$.*

One can treat Proposition 1 as a reduction of the synchronizability problem to a reachability problem in the subautomaton $\mathcal{P}^{[2]}(\mathscr{A})$ of $\mathcal{P}(\mathscr{A})$ whose states are 2-element and 1-element subsets of Q. Since the subautomaton has $\dfrac{|Q|(|Q|+1)}{2}$ states, breadth-first search solves this problem in $O(|Q|^2 \cdot |\Sigma|)$ time. This complexity bound assumes that no reset word is explicitly calculated. If one requires that, whenever $\mathscr{A}$ turns out to be synchronizing, a reset word is produced, then the best of the known algorithms (which is due to (Eppstein, 1990, Theorem 6), see also (Sandberg, 2005, Theorem 1.15)) has an implementation that consumes $O(|Q|^3 + |Q|^2 \cdot |\Sigma|)$ time and $O(|Q|^2 + |Q| \cdot |\Sigma|)$ working space, not counting the space for the output which is $O(|Q|^3)$.

For a synchronizing automaton, the power automaton can be used to construct shortest reset words which correspond to shortest paths from the whole state set to a singleton. Of course, this requires exponential (of $|Q|$) time in the worst case. Nevertheless, there were attempts to implement this approach (see, e.g., Rho et al, 1993; Trahtman, 2006a). One may hope that, as above, a suitable calculation in the “polynomial” subautomaton $\mathcal{P}^{[2]}(\mathscr{A})$ may yield a polynomial algorithm. However, it is not the case, and moreover, as we will see, it is very unlikely that any reasonable algorithm may exist for finding shortest reset words in general synchronizing automata. In the following discussion we assume the reader's acquaintance with some basics of computational complexity (such as the definitions of the complexity classes NP, coNP and PSPACE) that can be found, e.g., in (Garey & Johnson, 1979; Papadimitriou, 1994).

Consider the following decision problems:

SHORT-RESET-WORD: *Given a synchronizing automaton $\mathscr{A}$ and a positive integer ℓ, is it true that $\mathscr{A}$ has a reset word of length ℓ?*

SHORTEST-RESET-WORD: *Given a synchronizing automaton $\mathscr{A}$ and a positive integer ℓ, is it true that the minimum length of a reset word for $\mathscr{A}$ is equal to ℓ?*

Clearly, SHORT-RESET-WORD belongs to the complexity class NP: one can non-deterministically guess a word $w \in \Sigma^*$ of length ℓ and then check if w is a reset word for $\mathscr{A}$ in time $\ell|Q|$. Eppstein (1990) has proved that SHORT-RESET-WORD is NP-hard by a polynomial reduction from 3-SAT. Thus, SHORT-RESET-WORD is NP-complete. Other proofs for the same result (all via reductions from 3-SAT) have been suggested in (Goralčik & Koubek, 1995; Salomaa, 2003; Samotij, 2007). From the proofs it follows easily that SHORTEST-RESET-WORD is NP-hard; recently Samotij (2007) has shown that the negation of 3-SAT can be polynomially reduced to SHORTEST-RESET-WORD whence the latter problem is also coNP-hard. As a corollary, SHORTEST-RESET-WORD cannot belong to NP unless NP = coNP which is commonly considered to be very unlikely. In other words, even non-deterministic algorithms cannot find the minimum length of a reset word for a given synchronizing automaton in polynomial time.

On the other hand, the exhaustive search for reset words through all words over Σ of length $\leq \ell$ can be performed in polynomial (in fact, linear) space since one can reuse space. Thus, the problem SHORTEST-RESET-WORD belongs to the complexity class PSPACE; the question of the precise location of this problem with respect to the polynomial hierarchy still remains open. An upper bound has been recently found by Martjugin (unpublished) who has shown that the problem lies in the complexity class $\Sigma^2 \cap \Pi^2$.

By a standard argument, the hardness of the decision problem SHORT-RESET-WORD implies that its optimization version, in which one seeks a reset word of minimum length for a given synchronizing automaton, is hard as well. This did not exclude however that the optimization problem might admit a polynomial-time approximation algorithm, and moreover, all existing proofs for NP-hardness of SHORT-RESET-WORD were consistent with the conjecture that such an algorithm exists. However, recently Berlinkov (unpublished) has shown (assuming P $\neq$ NP) that, for any given positive integer k, no polynomial algorithm can find for each synchronizing automaton $\mathscr{A}$ a reset word whose length would be bounded by k times the minimum length of reset words for $\mathscr{A}$. Thus, approximating the minimum length of reset words is hard.

We mention that Pixley et al (1992) suggested an heuristic algorithm for finding short reset words in synchronizing automata that was reported to perform rather satisfactory on a number of benchmarks from (Yang, 1991); further algorithms yielding short (though not necessarily shortest) reset words were implemented by Trahtman (2006a).

3 The Černý Conjecture

In this section we discuss results and open problems related to the following natural question: *given a positive integer n, how long can be reset words for synchronizing automata with n states?*

First of all, we recall Černý's construction (1964). For $n > 1$, let $\mathscr{C}_n$ stand for the DFA whose states are the residues modulo n and whose input letters a and b act as follows:

$$\delta(0,a) = 1,\ \delta(m,a) = m \text{ for } 0 < m < n,\ \delta(m,b) = m+1 \pmod{n}.$$

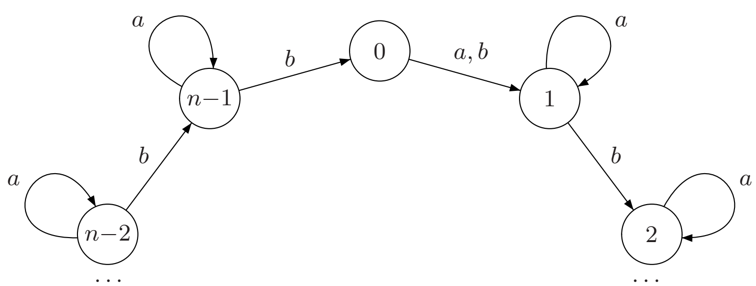

Fig. 7. The automaton $\mathscr{C}_n$

Černý (1964) has proved that $\mathscr{C}_n$ is a synchronizing automaton and that its shortest reset word is $(ab^{n-1})^{n-2}a$ of length $(n-1)^2$. (This series of automata was rediscovered many times, see, e.g., (Laemmel & Rudner, 1969; Fischler & Tannenbaum, 1970; Eppstein, 1990).) Thus, if we define the *Černý function* $C(n)$ as the maximum length of shortest reset words for synchronizing automata with n states, the above property of the series $\{\mathscr{C}_n\}$, $n = 2, 3, \ldots$, yields the inequality $C(n) \geq (n-1)^2$. The *Černý conjecture* is the claim that the equality $C(n) = (n-1)^2$ holds true.

In the literature, one often refers to (Černý, 1964) as the source of the Černý conjecture. In fact, the conjecture was not yet formulated in that paper. There Černý only observed that

$$(n-1)^2 \leq C(n) \leq 2^n - n - 1 \tag{1}$$

and concluded the paper with the following remark:

> "The difference between the bounds increases rapidly and it is necessary to sharpen them. One can expect an improvement mainly for the upper bound."

The conjecture in its present-day form was formulated a bit later, after the expectation in the above quotation was confirmed by Starke (1966). (Namely, Starke improved the upper bound in (1) to $1 + \frac{n(n-1)(n-2)}{2}$, which was the first polynomial upper bound for $C(n)$.) Černý explicitly stated the conjecture $C(n) = (n-1)^2$ in his talk at the Bratislava Cybernetics Conference held in 1969; in print the conjecture first appeared in (Černý et al, 1971).

The best upper bound for the Černý function $C(n)$ achieved so far guarantees that for every synchronizing automaton with n states there exists a reset word of length $\frac{n^3-n}{6}$. Such a reset word arises as the output of the following greedy algorithm.

Algorithm 1.

---

input $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ (a DFA)

initialization $w \leftarrow 1$ (the empty word)
$P \leftarrow Q$

while $|P| > 1$ find a word $v \in \Sigma^*$ of minimum length with $|\delta(P, v)| < |P|$; if none exists, **return** Failure
$w \leftarrow wv$
$P \leftarrow \delta(P, v)$

return w

If $|Q| = n$, then clearly the main loop of Algorithm 1 is executed at most $n - 1$ times. In order to evaluate the length of the output word w, we estimate the length of each word v produced by the main loop.

Consider a generic step at which $|P| = k > 1$ and let $v = a_1 \cdots a_\ell$ with $a_i \in \Sigma$, $i = 1, \ldots, \ell$. Then it is easy to see that the sets

$$P_1 = P,\ P_2 = \delta(P_1, a_1),\ \ldots,\ P_\ell = \delta(P_{\ell-1}, a_{\ell-1})$$

are k-element subsets of Q. Furthermore, since $|\delta(P_\ell, a_\ell)| < |P_\ell|$, there exist two distinct states $q_\ell, q'_\ell \in P_\ell$ such that $\delta(q_\ell, a_\ell) = \delta(q'_\ell, a_\ell)$. Now define 2-element subsets $R_i = \{q_i, q'_i\} \subseteq P_i$, $i = 1, \ldots, \ell$, such that $\delta(q_i, a_i) = q_{i+1}$, $\delta(q'_i, a_i) = q'_{i+1}$ for $i = 1, \ldots, \ell - 1$. Then the condition that v is a word of minimum length with $|\delta(P, v)| < |P|$ implies that $R_i \nsubseteq P_j$ for $1 \leq j < i \leq \ell$. Altogether, we arrive at the following purely combinatorial problem:

Question 1. Let Q be an n-element set, $P_1, \ldots, P_\ell$ a sequence of its k-element subsets $(k > 1)$ and $R_1, \ldots, R_\ell$ a sequence of its 2-element subsets. Suppose that $R_i \subseteq P_i$ for each $i = 1, \ldots, \ell$ but $R_i \nsubseteq P_j$ for $1 \leq j < i \leq \ell$. How big the number ℓ can be?

Question 1 was solved by Frankl (1982) who found the tight bound $\ell \leq \binom{n-k+2}{2}$. Summing up these inequalities from $k = n$ to $k = 2$, one arrives at the aforementioned bound

$$C(n) \leq \frac{n^3 - n}{6}. \tag{2}$$

In the literature the bound (2) is usually attributed to Pin who explained the above connection between Algorithm 3.1 and Question 1 and conjectured the estimation $\ell \leq \binom{n-k+2}{2}$ in his talk at the Colloquium on Graph Theory and Combinatorics held in Marseille in 1981; Frankl learned Question 1 from that talk. Accordingly, the usual reference for (2) is the paper (Pin, 1983) based on the talk. The full story is however more complicated. Actually, the bound (2) first appeared in (Fischler & Tannenbaum, 1970) where it was deduced from a combinatorial conjecture equivalent to Pin's one. Fischler & Tannenbaum presented their paper at the 11th FOCS conference but that time there was no Frankl in the audience so that the conjecture remained unproved and the paper eventually got lost in limbo. The bound (2) then reoccurred in Kohavi & Winograd (1971, 1973) but the argument justifying it in these papers was insufficient. Later both (2) and Frankl's solution to Question 1 were independently rediscovered in (Klyachko et al, 1987).

If one executes Algorithm 1 on the Černý automaton $\mathscr{C}_4$ (Fig. 6 is quite helpful here), one sees that the algorithm returns the word ab^2abab^3a of length 10 which is not the shortest reset word for $\mathscr{C}_4$. This reveals one of the main intrinsic difficulties of the synchronization problem: the standard optimality principle does not apply here since it is not true that the optimal solution behaves optimally also in all intermediate steps. In our example, the optimal solution is the word ab^3ab^3a but it cannot be found by the greedy algorithm because the algorithm chooses $v = b^2a$ rather than $v = b^3a$ on the second execution of the main loop.

Another difficulty behind the scene is that there are only very few examples of *extreme* synchronizing automata, that is n-state synchronizing automata whose shortest reset words have lengths $(n-1)^2$. In fact, the Černý series $\mathscr{C}_n$, $n = 2, 3, \dots$, is the only known infinite series of extreme synchronizing automata. Besides that, we know only a few isolated examples of such automata: up to isomorphism and adding/removing non-essential letters, there are three extreme automata with 3 states, three extreme automata with 4 states (see Fig. 8), one extreme automaton with 5 states recently found by Roman, see Fig. 9, and one extreme automaton with 6 states found by Kari (2001), see Fig. 10.

Moreover, even synchronizing automata whose shortest reset words have lengths close to the Černý bound are very rare. For instance, we are not aware of any 5-state synchronizing automaton whose shortest reset word has length 24, nor of any 6-state synchronizing automaton whose shortest reset word has length 33 or 34 or 35, etc. As for regular constructions of "almost-extreme" automata, we know just one series of n-state synchronizing automata with odd $n \geq 5$ such that the minimum length of reset words for the n^{th} automaton in the series is equal to $(n-1)(n-2)$, see (Ananichev et al, 2007).

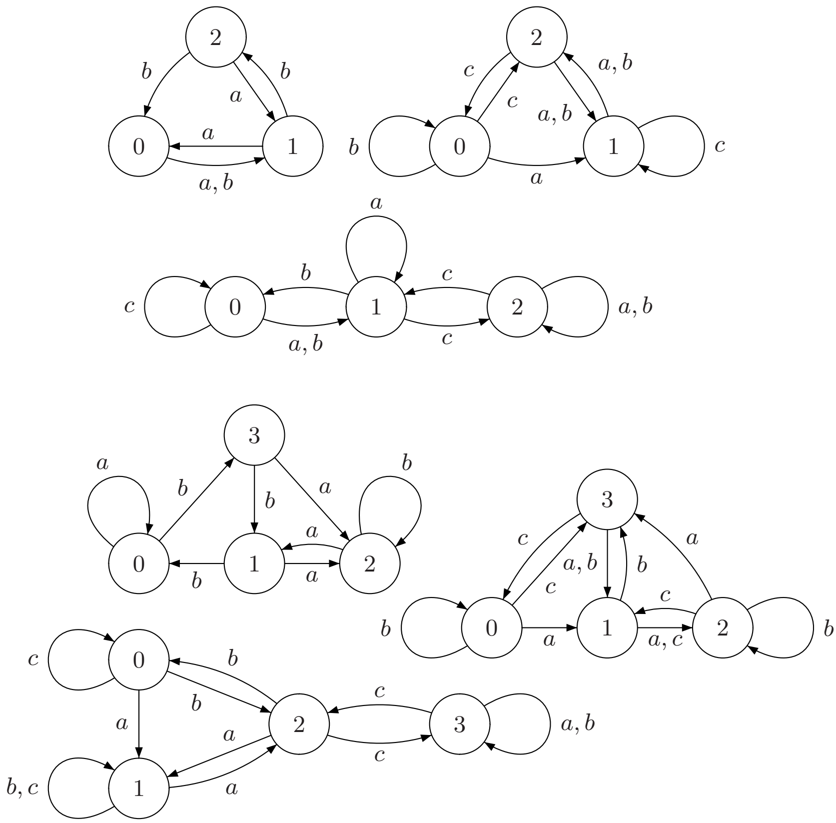

Fig. 8. Extreme synchronizing automata with 3 and 4 states

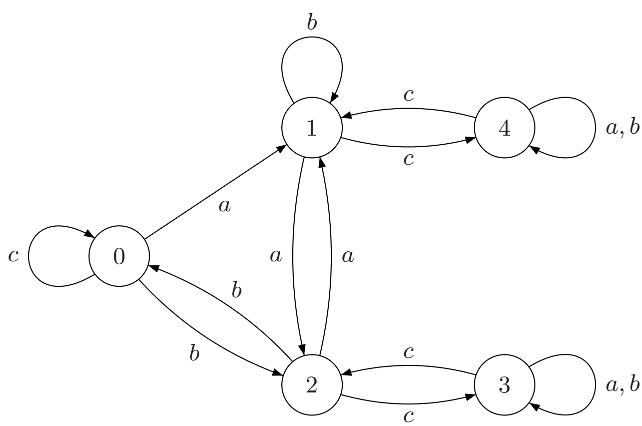

Fig. 9. Roman's automaton

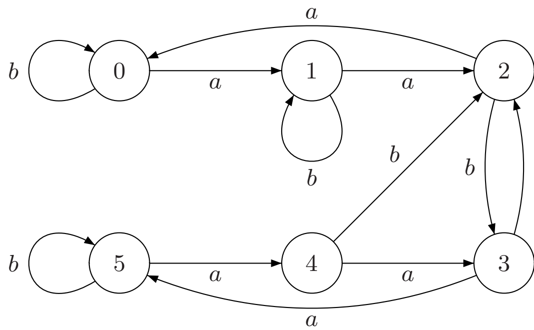

Fig. 10. Kari's automaton

In general, "slowly" synchronizing automata turn out to be rather exceptional, and this observation is supported also by probabilistic arguments. Indeed, if Q is an n-element set (with n large enough), then, on average, any product of $2n$ randomly chosen transformations of Q is known to be a constant map, see (Higgins, 1988). Being retold in automata-theoretic terms, this fact implies that a randomly chosen DFA with n states and a sufficiently large input alphabet tends to be synchronizing, and moreover, the length of its shortest reset word does not exceed $2n$. This means, in particular, that there is no hope to find new examples of "slowly" synchronizing automata, to say nothing of a counterexample to the Černý conjecture, via a random sampling experiment.

The Černý conjecture has been confirmed for various classes of synchronizing automata satisfying some additional restrictions. We conclude with a (non-complete) list of the most important results of this sort.

We begin with Eppstein's result (1990) in which restrictions are imposed on the action of the letters on the state set. Consider the set $\{0, 1, \ldots, n-1\}$ equipped with the natural cyclic order $0 \prec 1 \prec 2 \prec \cdots \prec n-1 \prec 1$ (here $k \prec \ell$ means that ℓ immediately follows k). If $i_1, i_2, \ldots, i_m$ are numbers in $\{0, 1, 2, \ldots, n-1\}$, we call the sequence $i_1, i_2, \ldots, i_m$ *cyclic* if, after removal of possible adjacent duplicate numbers, it is a subsequence of a cyclic shift of the sequence $0, 1, 2, \ldots, n-1$. In a slightly more formal language, we may say that $i_1, i_2, \ldots, i_m$ is a cyclic sequence if there exists no more than one index $t \in \{1, \ldots, m\}$ such that $i_{t+1} < i_t$ where i_{m+1} is understood as i_1 and $<$ stands for the usual strict linear order on $\{0, 1, 2, \ldots, n-1\}$. A transformation α of the set $\{0, 1, 2, \ldots, n-1\}$ is said to be *orientation preserving* if the numbers $0\alpha, 1\alpha, \ldots, (n-1)\alpha$ form a cyclic sequence. Now let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a DFA with n states. We say that $\mathscr{A}$ is *orientable* if its states can be indexed by $0, 1, 2, \ldots, n-1$ so that all the transformations $\delta(\sqcup, a) : Q \to Q$ induced by the letters $a \in \Sigma$ are orientation preserving. For instance, Černý's automata $\mathscr{C}_n$, $n = 2, 3, \ldots$, are orientable.

Eppstein's interest in orientable automata (which he called *monotonic*) was motivated by the robotics applications of synchronizing automata. Indeed, in the problem of sensor-free orientation of polygonal parts one deals with solid bodies whence only those transformations of polygons are physically meaningful that preserve relative location of the faces of these polygons. It was observed already by Natarajan (1986) that in the "flat" case (when the polygonal parts do not leave a certain plane, say, the surface of a conveyer belt) this physical requirement leads precisely to orientation preserving transformations. In (Eppstein, 1990, Theorem 2) it is proved that, in accordance with the Černý conjecture, every orientable synchronizing automaton with n states has a reset word of length $(n-1)^2$. An extension of this result to a larger class of synchronizing automata whose letter actions mimic certain "spatial" transformations of solid polygons was obtained by Ananichev & Volkov (2004).

Dubuc (1998) has proved the Černý conjecture for yet another natural class of automata also containing the Černý series: automata in which a letter acts on the state set Q as a cyclic permutation of order $|Q|$.

In Kari's elegant paper (2003) the restriction has been imposed on the underlying digraphs of automata in question, namely, the Černý's conjecture has been verified for synchronizing automata with Eulerian digraphs. Moreover, it has been proved that if the underlying digraph of an n-state synchronizing automaton is Eulerian then there exists a reset word of length $(n-2)(n-1)+1$ (Kari, 2003, Theorem 2). It is unknown whether or not this bound is tight.

Recall that the *transition monoid* of a DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ is the monoid consisting of all transformations $\delta(\sqcup, w) : Q \to Q$ induced by the words $w \in \Sigma^*$. Several authors have studied synchronization issues for automata whose transition monoids satisfy certain abstract properties. An important example of a property of automata expressed in this language is aperiodicity. A monoid is said to be *aperiodic* if all its subgroups are singletons; a DFA is called *aperiodic* (or *counter-free*) if its transition monoid is aperiodic. Aperiodic automata play a distinguished role in many aspects of formal language theory and its connections to logic, see the classic monograph by McNaughton & Papert (1971). Thus, studying synchronization within this important subclass of automata appears to be well justified, especially if one takes into account that the problem of finding short reset words is known to remain difficult when restricted to aperiodic automata. Indeed, inspecting the reductions from 3-SAT used in (Eppstein, 1990) or (Goralčik & Koubek, 1995) or (Salomaa, 2003), one can observe that in each case the construction results in an aperiodic automaton, and therefore, the question of whether or not a given aperiodic automaton admits a reset word whose length does not exceed a given positive integer is NP-complete.

Recently Trahtman (2007) has proved that every synchronizing aperiodic automaton with n states admits a reset word of length at most $\frac{n(n-1)}{2}$. Thus, the Černý conjecture holds true for synchronizing aperiodic automata. However, the problem of establishing a precise bound for the minimum length of reset words for synchronizing aperiodic automata with n states still remains open, and moreover, we do not even have a reasonably justified conjecture for this case. Denote by $C_A(n)$ the minimum length of reset words for synchronizing aperiodic automata with n states, that is, the restriction of the Černý function to the class of aperiodic automata. Then Trahtman's result can be expressed by the inequality $C_A(n) \leq \dfrac{n(n-1)}{2}$. However, the only non-trivial lower bound for

$C_A(n)$, which has been established so far, is linear, namely, $C_A(n) \geq n + \left\lfloor \frac{n}{2} \right\rfloor - 2$ for $n \geq 7$. (This bound comes from Ananichev's paper (2005).) One sees that the gap between the two bounds is fairly large. We believe that the actual value of $C_A(n)$ is closer to the lower bound.

In (Volkov, 2007) the results from (Trahtman, 2007) have been extended to a larger class of automata and improved. In particular, it has been proved that if the underlying digraph of an n-state aperiodic automaton is strongly connected, then the automaton has a reset word of length $\lfloor \frac{n(n+1)}{6} \rfloor$ (Volkov, 2007, Corollary 1).

Another large class of finite monoids which is of importance for formal language theory is known under the name **DS** and can be described as follows: a finite monoid M belongs to **DS** if and only if for all $x, y, z, t \in N$ the following condition holds:

$$MxM = MyM = MzM = MtM = MxyM \text{ implies } MxyM = MztM.$$

(For the reader acquainted with some basics of semigroup theory, we recall an equivalent but more standard description of **DS**: a finite monoid M belongs to **DS** if and only if each regular $\mathcal{D}$-class of M is a subsemigroup in M.) Recently Almeida et al (2008) have proved the Černý conjecture for synchronizing automata with transition monoids in **DS**. Again, the problem of establishing a precise bound for the minimum length of reset words for synchronizing automata in this class still remains open.

References

Almeida, J., Margolis, S., Steinberg, B., Volkov, M.V.: Representation theory of finite semigroups, semigroup radicals and formal language theory. Trans. Amer. Math. Soc. (to appear, 2008), `http://arxiv.org/abs/math/0702400v1`

Ananichev, D.S.: The mortality threshold for partially monotonic automata. In: De Felice, C., Restivo, A. (eds.) DLT 2005. LNCS, vol. 3572, pp. 112–121. Springer, Heidelberg (2005)

Ananichev, D.S., Volkov, M.V.: Some results on Černý type problems for transformation semigroups. In: Araujo, I., Branco, M., Fernandes, V.H., Gomes, G.M.S. (eds.) Semigroups and Languages, pp. 23–42. World Scientific, Singapore (2004)

Ananichev, D.S., Volkov, M.V., Zaks, Y.I.: Synchronizing automata with a letter of deficiency 2. Theoret. Comput. Sci. 376(1-2), 30–41 (2007)

Ashby, W.R.: An Introduction to Cybernetics. Chapman & Hall, London (1956), `http://pcp.vub.ac.be/books/IntroCyb.pdf`

Benenson, Y., Paz-Elizur, T., Adar, R., Keinan, E., Livneh, Z., Shapiro, E.: Programmable and autonomous computing machine made of biomolecules. Nature 414(1), 430–434 (2001)

Benenson, Y., Adar, R., Paz-Elizur, T., Livneh, Z., Shapiro, E.: DNA molecule provides a computing machine with both data and fuel. Proc. National Acad. Sci. USA 100, 2191–2196 (2003)

Booth, T.L.: Sequential Machines and Automata Theory. J. Wiley & Sons, New York (1967)

Boppana, V., Rajan, S.P., Takayama, K., Fujita, M.: Model checking based on sequential ATPG. In: Halbwachs, N., Peled, D.A. (eds.) CAV 1999. LNCS, vol. 1633, pp. 418–430. Springer, Heidelberg (1999)

Černý, J.: Poznámka k homogénnym eksperimentom s konečnými automatami. Matematicko-fyzikalny Časopis Slovensk. Akad. Vied 14(3), 208–216 (1964) (in Slovak)

Černý, J., Pirická, A., Rosenauerová, B.: On directable automata. Kybernetika 7(4), 289–298 (1971)

Chen, Y.-B., Ierardi, D.J.: The complexity of oblivious plans for orienting and distinguishing polygonal parts. Algorithmica 14, 367–397 (1995)

Cho, H., Jeong, S.-W., Somenzi, F., Pixley, C.: Synchronizing sequences and symbolic traversal techniques in test generation. J. Electronic Testing 4, 19–31 (1993)

Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms. MIT Press, Cambridge (2001)

Dubuc, L.: Sur le automates circulaires et la conjecture de Černý. RAIRO Inform. Theor. Appl. 32, 21–34 (1998) (in French)

Eppstein, D.: Reset sequences for monotonic automata. SIAM J. Comput. 19, 500–510 (1990)

Fischler, M.A., Tannenbaum, M.: Synchronizing and representation problems for sequential machines with masked outputs. In: Proc. 11th Annual Symp. Foundations Comput. Sci., pp. 97–103. IEEE, Los Alamitos (1970)

Frankl, P.: An extremal problem for two families of sets. Eur. J. Comb. 3, 125–127 (1982)

Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-completeness. Freeman, San Francisco (1979)

Gill, A.: State-identification experiments in finite automata. Information and Control 4(2-3), 132–154 (1961)

Ginsburg, S.: On the length of the smallest uniform experiment which distinguishes the terminal states of a machine. J. Assoc. Comput. Mach. 5, 266–280 (1958)

Goldberg, K.: Orienting polygonal parts without sensors. Algorithmica 10, 201–225 (1993)

Goralčik, P., Koubek, V.: Rank problems for composite transformations. Int. J. Algebra and Computation 5, 309–316 (1995)

Hennie, F.C.: Fault detecting experiments for sequential circuits. In: Proc. 5th Annual Symp. Switching Circuit Theory and Logical Design, pp. 95–110. IEEE, Los Alamitos (1964)

Hennie, F.C.: Finite-state Models for Logical Machines. J. Wiley & Sons, New York (1968)

Higgins, P.M.: The range order of a product of i transformations from a finite full transformation semigroup. Semigroup Forum 37, 31–36 (1988)

Kari, J.: A counter example to a conjecture concerning synchronizing words in finite automata. EATCS Bull. 73, 146 (2001)

Kari, J.: Synchronizing finite automata on Eulerian digraphs. Theoret. Comput. Sci. 295, 223–232 (2003)

Klyachko, A.A., Rystsov, I.K., Spivak, M.A.: An extremal combinatorial problem associated with the bound of the length of a synchronizing word in an automaton. Kibernetika 25(2), 16–20 (1987) (in Russian); English translation: Cybernetics and Systems Analysis 23(2), 165–171

Kohavi, Z.: Switching and Finite Automata Theory. McGraw-Hill, New York (1970)
Kohavi, Z., Winograd, J.: Bounds on the length of synchronizing sequences and the order of information losslessness. In: Kohavi, Z., Paz, A. (eds.) Theory of Machines and Computations, pp. 197–206. Academic Press, New York (1971)
Kohavi, Z., Winograd, J.: Establishing certain bounds concerning finite automata. J. Comput. Syst. Sci. 7(3), 288–299 (1973)
Laemmel, A.E., Rudner, B.: Study of the application of coding theory, Report PIBEP-69-034, Polytechnic Inst. Brooklyn, Dept. Electrophysics, Farmingdale (1969)
Mateescu, A., Salomaa, A.: Many-valued truth functions, Černý's conjecture and road coloring. EATCS Bull. 68, 134–150 (1999)
McNaughton, R., Papert, S.A.: Counter-free Automata. MIT Press, Cambridge (1971)
Moore, E.F.: Gedanken-experiments on sequential machines. In: Shannon, C.E., McCarthy, J. (eds.) Automata Studies, Annals of Mathematics Studies, vol. 34, pp. 129–153. Princeton University Press, Princeton (1956)
Natarajan, B.K.: An algorithmic approach to the automated design of parts orienters. In: Proc. 27th Annual Symp. Foundations Comput. Sci., pp. 132–142. IEEE, Los Alamitos (1986)
Natarajan, B.K.: Some paradigms for the automated design of parts feeders. Internat. J. Robotics Research 8(6), 89–109 (1989)
Papadimitriou, C.H.: Computational Complexity. Addison-Wesley, Reading (1994)
Pin, J.-E.: On two combinatorial problems arising from automata theory. Ann. Discrete Math. 17, 535–548 (1983)
Pixley, C., Jeong, S.-W., Hachtel, G.D.: Exact calculation of synchronization sequences based on binary decision diagrams. In: Proc. 29th Design Automation Conf., pp. 620–623. IEEE, Los Alamitos (1992)
Rho, J.-K., Somenzi, F., Pixley, C.: Minimum length synchronizing sequences of finite state machine. In: Proc. 30th Design Automation Conf., pp. 463–468. ACM, New York (1993)
Salomaa, A.: Composition sequences for functions over a finite domain. Theoret. Comput. Sci. 292, 263–281 (2003)
Samotij, W.: A note on the complexity of the problem of finding shortest synchronizing words. In: Proc. AutoMathA 2007, Automata: from Mathematics to Applications, Univ. Palermo (CD) (2007)
Sandberg, S.: Homing and synchronizing sequences. In: Broy, M., Jonsson, B., Katoen, J.-P., Leucker, M., Pretschner, A. (eds.) Model-Based Testing of Reactive Systems. LNCS, vol. 3472, pp. 5–33. Springer, Heidelberg (2005)
Starke, P.H.: Eine Bemerkung über homogene Experimente. Elektronische Informationverarbeitung und Kybernetik 2, 257–259 (1966) (in German)
Starke, P. H.: Abstrakte Automaten, Deutscher Verlag der Wissenschaften, Berlin (1969) (in German); English translation: Abstract Automata, North-Holland, Amsterdam, American. Elsevier, New York (1972)
Trahtman, A.: An efficient algorithm finds noticeable trends and examples concerning the Černý conjecture. In: Královič, R., Urzyczyn, P. (eds.) MFCS 2006. LNCS, vol. 4162, pp. 789–800. Springer, Heidelberg (2006)
Trahtman, A.: The Černý conjecture for aperiodic automata. Discrete Math. Theoret. Comp. Sci. 9(2), 3–10 (2007)
Trahtman, A.: The Road Coloring Problem. Israel J. Math. (to appear, 2008), `http://arxiv.org/abs/0709.0099`

Volkov, M.V.: Synchronizing automata preserving a chain of partial orders. In: Holub, J., Žďárek, J. (eds.) CIAA 2007. LNCS, vol. 4783, pp. 27–37. Springer, Heidelberg (2007)

Yang, S.: Logic Synthesis and Optimization Benchmarks User Guide Version 3.0, Microelectronics Center of North Carolina, Research Triangle Park, NC (1991)