Algebraic Synchronization Criterion and Computing Reset Words

Mikhail Berlinkov[1](✉) and Marek Szykuła[2]

[1] Institute of Mathematics and Computer Science,
Ural Federal University, Yekaterinburg, Russia
berlm@mail.ru

[2] Institute of Computer Science, University of Wrocław, Wrocław, Poland

Abstract. We refine results about relations between Markov chains and synchronizing automata. We express the condition that an automaton is synchronizing in terms of linear algebra, and obtain upper bounds for the reset thresholds of automata with a short word of a small rank. The results are applied to make several improvements in the area.

We improve the best general upper bound for reset thresholds of finite prefix codes (Huffman codes): we show that an n-state synchronizing decoder has a reset word of length at most $O(n \log^3 n)$. Also, we prove the Černý conjecture for n-state automata with a letter of rank at most $\sqrt[3]{6n-6}$. In another corollary, based on the recent results of Nicaud, we show that the probability that the Černý conjecture does not hold for a random synchronizing binary automaton is exponentially small in terms of the number of states. It follows that the expected value of the reset threshold of an n-state random synchronizing binary automaton is at most $n^{7/4+o(1)}$.

Moreover, reset words of the lengths within our bounds are computable in polynomial time. We present suitable algorithms for this task for various classes of automata for which our results can be applied. These include (quasi-)one-cluster and (quasi-)Eulerian automata.

1 Introduction

We deal with *deterministic finite automata* (*DFA*) $\mathscr{A} = (Q, \Sigma, \delta)$, where Q is a non-empty *set of states*, Σ is a non-empty *alphabet*, and $\delta \colon Q \times \Sigma \mapsto Q$ is the complete *transition function*. We extend δ to $Q \times \Sigma^*$ and $2^Q \times \Sigma^*$ as usual, and for the image (resp. preimage) of a set S under a word w we write shortly $S.w$ (resp. $S.w^{-1}$). We denote $\Sigma^{\leq c} = \{w \in \Sigma^* \colon |w| \leq c\}$, the set of all words over Σ

M. Berlinkov—Supported by the Presidential Program "Leading Scientific Schools of the Russian Federation", project no. 5161.2014.1, the Russian Foundation for Basic Research, project no. 13-01-00852, the Ministry of Education and Science of the Russian Federation, project no. 1.1999.2014/K, and the Competitiveness Program of Ural Federal University.

M. Szykuła—Supported in part by Polish NCN grant DEC-2013/09/N/ST6/01194.

G.F. Italiano et al. (Eds.): MFCS 2015, Part I, LNCS 9234, pp. 103–115, 2015.
DOI: 10.1007/978-3-662-48057-1_8

of length at most c. The empty word is denoted by ε. Throughout the paper, by n we denote the cardinality $|Q|$, and by k we denote $|\Sigma|$.

A word w *compresses* a subset $S \subseteq Q$ if $|S.w| < |S|$. Then we say that S is *compressible*. The *rank* of a word w is $|Q.w|$. A *reset word* or a *synchronizing word* is a word $w \in \Sigma^*$ of rank 1, that is, w takes the automaton to a particular state no matter of the current state. An automaton is called *synchronizing* if it possesses a reset word. An example of a synchronizing automaton from the Černý series [12] is presented in Fig. 1 (left). One can verify that its shortest reset word is ba^3ba^3b. The length of the shortest reset word is called the *reset threshold* and is denoted by $\mathrm{rt}(\mathscr{A})$.

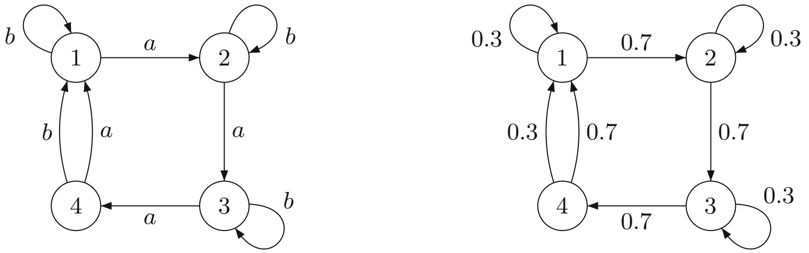

Fig. 1. The automaton $\mathscr{C}_4$ and the associated Markov chain for $P(a) = 0.7, P(b) = 0.3$

For detailed introduction to the theory of synchronizing automata we refer reader to the surveys [19,27], and for the review of relations with coding theory to [17]. For various applications, reset words allow to reestablish the control under the system modeled by an automaton. So, the reset threshold serves as a natural measure of synchronization. Thus, it is important to compute the reset threshold from both theoretical and practical points of view.

The Černý conjecture, which is arguably the most longstanding open problem in the combinatorial theory of finite automata, states that the reset threshold of a synchronizing automaton is at most $(n-1)^2$. This bound would be tight, since Černý [12] constructed for each n a synchronizing automaton $\mathscr{C}_n$ with this reset threshold. Moreover, the best upper bound known so far for the reset threshold of a synchronizing n-state automaton is equal to $\frac{n^3-n}{6} - 1$ (for $n \geq 4$) so is cubic in n (see Pin [24]). Thus it is of certain importance to prove specific upper bounds for various classes of synchronizing automata.

In this paper, we improve several results concerning reset thresholds. First, we express the condition that an automaton is synchronizing in terms of linear algebra, and derive upper bounds for automata with a word of a small rank (Sect. 2). Then, we apply the results to improve upper bounds in several cases. In Sect. 3 we show that the Černý conjecture holds for automata with a letter of rank $\sqrt[3]{6n-6}$, which improves the previous logarithmic result [22]. Also, basing on the recent results of Nicaud [20], we show that the Černý conjecture holds for a random synchronizing binary automaton with probability exponentially (in n) close to 1, and that the expected reset threshold is at most $n^{7/4+o(1)}$.

The next important application of our results is an upper bound for the length of the shortest reset words of finite prefix codes (Huffman codes), which are one of the most popular methods of data compression. One of the problems with compressed data is reliability in case of presence of errors in the compressed text. Eventually, a single error may possibly destroy the whole encoded string. One of the proposed solutions to this problem (for Huffman codes) are codes that can be synchronized by a reset word, regardless of the possible errors. The reset thresholds of binary Huffman codes was first studied by Biskup and Plandowski [8,9], who showed a general upper bound of order $O(n^2 \log n)$, where n is the number of states of the decoder (equivalently, the number of words in the code). They also proved that a word of this length can be computed in polynomial time. The bound was later improved to $O(n^2)$ for a wider class of *one-cluster* automata [2]. In Sect. 4 we prove an upper bound of order $O(n \log^3 n)$. Note that for some applications it can be also important to get bounds in terms of the maximal length of the words in the code (see e.g. [11]).

Unlike the general case, the Černý conjecture has been approved for various classes of automata such as circular [13,23], Eulerian [18] and one-cluster automata with prime length cycle [26]. Later specific quadratic upper bounds for some generalizations of these classes were obtained in [2,5]. However, no efficient algorithm for finding reset words with lengths within the specified bounds has been presented for these classes. Moreover, there is no hope to get a polynomial algorithm for finding the shortest reset words in the general case, since this problem has been shown to be $\mathrm{FP}^{\mathrm{NP}[\log]}$-hard [21]. Also, unless $\mathrm{P} = \mathrm{NP}$, there is no polynomial algorithm for computing the reset threshold for a given automaton within the approximation ratio n^{ε} for a certain $\varepsilon > 0$ even in the case of a binary alphabet [15] (cf. also [6,16]).

In Sect. 5 we present polynomial algorithms for finding reset words of length within the proven bounds. Our algorithms can be applied in particular to the classes of decoders of finite prefix codes, and also to generalized classes of quasi-Eulerian and quasi-one-cluster automata. Since from our results it is possible to derive the bounds from [2,5,10,18,25,26], our algorithms apply to these bounds as well.

The full version of this paper is available at [7].

2 Algebraic Synchronization Criterion

In this section we refine some results from [5], formulate the algebraic synchronization criterion, and derive upper bounds for reset thresholds of automata with a word of a small rank. For this purpose, we associate a natural linear structure with an automaton $\mathscr{A}$. By $\mathbb{R}^n$ we denote the real n-dimensional linear space of row vectors. Without loss of generality, we assume that $Q = \{1, 2, \ldots, n\}$ and then assign to each subset $K \subseteq Q$ its *characteristic vector* $[K] \in \mathbb{R}^n$, whose i-th entry is 1 if $i \in K$, and 0, otherwise. For $q \in Q$ we write $[q]$ instead of $[\{q\}]$ to simplify the notation. By $\langle S \rangle$ we denote the linear span of $S \subseteq \mathbb{R}^n$. The $n \times n$ identity matrix is denoted by I_n.

Each word $w \in \Sigma^*$ corresponds to a linear transformation of $\mathbb{R}^n$. By $[w]$ we denote the matrix of this transformation in the standard basis $[1], \ldots, [n]$ of $\mathbb{R}^n$. For instance, if $\mathscr{A} = \mathscr{C}_4$ from Fig. 1 (left), then

$$[a] = \begin{pmatrix} 0&1&0&0 \\ 0&0&1&0 \\ 0&0&0&1 \\ 1&0&0&0 \end{pmatrix}, \ [b] = \begin{pmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 1&0&0&0 \end{pmatrix}, \ [ba] = \begin{pmatrix} 0&1&0&0 \\ 0&0&1&0 \\ 0&0&0&1 \\ 0&1&0&0 \end{pmatrix}.$$

Clearly, the matrix $[w]$ has exactly one non-zero entry in each row. In particular, $[w]$ is *row stochastic*, that is, the sum of entries in each row is equal to 1. In virtue of row-vector notation (apart from [5]), we get that $[uv] = [u][v]$ for every two words $u, v \in \Sigma^*$. By $[w]^T$ we denote the transpose of the matrix $[w]$. One easily verifies that $[S.w^{-1}] = [S][w]^T$. Let us also notice that within this definition the (adjacency) matrix of the underlying digraph of $\mathscr{A}$ is equal to $\sum_{a \in \Sigma}[a]$.

Recall that a word w is a reset word if $q.w^{-1} = Q$, for some state $q \in Q$. Thus, in the language of linear algebra, we can rewrite this fact as $[q][w]^T = [Q]$. For two vectors $g_1, g_2 \in \mathbb{R}^n$, we denote their usual inner (scalar) product by (g_1, g_2). We say that a vector (matrix) is *positive* (*non-negative*) if it contains only positive (non-negative) entries. Let $p \in \mathbb{R}^n_+$ be a positive row stochastic vector. Then $([Q], p) = 1$, and a word w is a reset word if and only if there exists $q \in Q$ such that

$$([q.w^{-1}], p) = ([q][w]^T, p) = ([q], p[w]) = 1.$$

Now we need to recall a few properties of Markov chains. A *Markov chain* of an automaton $\mathscr{A}$ is the random walk process of an agent on the underlying digraph of $\mathscr{A}$ where each time an edge labeled by a_i is chosen according to a given probability distribution $P\colon \Sigma \mapsto R$. The matrix $S(\mathscr{A}, P) = \sum_{i=1}^{k} P(a_i)[a_i]$ is called the *transition matrix* of this Markov chain. An example of a Markov chain associated with the automaton $\mathscr{A} = \mathscr{C}_4$ is presented in Fig. 1 (right) for $P(a) = 0.7, P(b) = 0.3$ and its stationary distribution is $\alpha = (\frac{10}{37}, \frac{10}{37}, \frac{10}{37}, \frac{7}{37})$.

A non-negative square matrix M is *primitive* if for some $d > 0$, the matrix M^d is positive. It is well known that if $\mathscr{A}$ is strongly connected and synchronizing, then the matrix of the underlying digraph of $\mathscr{A}$ is primitive, and so is the matrix of a Markov chain of $\mathscr{A}$ for any positive probability distribution P (see [1,5]). The following proposition is due to the well known Perron-Frobenius theorem.

Proposition 1. *Let M be a row stochastic $n \times n$ matrix. Then there exists a* stationary distribution *$\alpha \in \mathbb{R}^n$, that is, a non-negative stochastic vector satisfying $\alpha M = \alpha$. Moreover, if M is primitive then α is unique and positive.*

Call a set of words $W \subseteq \Sigma^*$ *complete* for a subspace $V \leq \mathbb{R}^n$, with respect to a vector $g \in V$, if

$$\langle g[w] \mid w \in W \rangle = V.$$

For a subset $S \subseteq Q$ we define $V_S = \langle [p] \mid p \in S \rangle \leq \mathbb{R}^n$.

We aim to strengthen [5, Theorem 9]. Namely, we show that the condition that $\mathscr{A}$ is synchronizing is not necessary if we require completeness for the corresponding set of words, and that only completeness with respect to the stationary distribution of $\mathscr{A}$ is required. As in [5] we construct an auxiliary automaton. We fix two positive integers d_1, d_2 and two non-empty sets of words $W_1 \subseteq \Sigma^{\leq d_1}$, $W_2 \subseteq \Sigma^{\leq d_2}$. Consider the automaton

$$\mathscr{A}_c(W_1, W_2) = (R, W_2W_1, \delta_{\mathscr{A}_c}),$$

where $R = \{q.w \mid q \in Q,\ w \in W_1\}$ and $W_2W_1 = \{w_2w_1 \in \Sigma^* \mid w_2 \in W_2, w_1 \in W_1\}$. The transition function $\delta_{\mathscr{A}_c}$ is defined in compliance with the actions of words in $\mathscr{A}$, i.e. $\delta_{\mathscr{A}_c}(q, w) = \delta(q, w)$, for all $q \in R$ and $w \in W_2W_1$. Note that $\delta_{\mathscr{A}_c}$ is well defined because $q.w \in R$ for all $q \in Q$ and $w \in \Sigma_{\mathscr{A}_c}$. Without loss of generality we may assume that $R = \{1, 2, \ldots, r\}$ where $r = |R|$.

Let P_1 and P_2 be some positive probability distributions on the sets W_1 and W_2, respectively, and denote $[P_i] = \sum_{w \in W_i} P_i(w)[w]$ for $i = 1, 2$. Then the $r \times r$ submatrix formed by the first r rows and the first r columns of the matrix

$$S(\mathscr{A}_c, P_2P_1) = [P_2][P_1] = \sum_{w_1 \in W_1, w_2 \in W_2} P_1(w_1)P_2(w_2)[w_2][w_1]$$

is the transition matrix of the Markov chain on $\mathscr{A}_c$. By Proposition 1 there exists a steady state distribution $\alpha = \alpha(\mathscr{A}_c) \in V_R$, that is, a stochastic vector (with first r non-negative entries) satisfying $\alpha S(\mathscr{A}_c, P_2P_1) = \alpha$.

For a vector $g \in \mathbb{R}^n_+$, by $\mathrm{DS}(g)$ we denote the number of different positive sums of entries of g, i.e. $\mathrm{DS}(g) = |\{(g, z) \mid z \in \{0, 1\}^n\}| - 1$.

Theorem 1. *Let $\mathscr{A} = (Q, \Sigma, \delta)$ be an automaton and let*

$$\mathscr{B} = \mathscr{A}_c(W_1, W_2) = (R, W_2W_1, \delta_{\mathscr{B}}),$$

be the automaton defined as above. If W_2W_1 is complete for V_R with respect to α, and $w_0 \in \Sigma^*$ is a word with $Q.w_0 = R$, then:

1. *If $x \in V_R \setminus \langle [R] \rangle$, then there exists $w \in W_2W_1$ such that $(x, \alpha[w]) > (x, \alpha)$;*
2. *$\mathscr{B}$ is synchronizing and $\mathrm{rt}(\mathscr{B}) \leq DS(\alpha) - 1$;*
3. *$\mathscr{A}$ is synchronizing and*

$$\mathrm{rt}(\mathscr{A}) \leq \begin{cases} |w_0| + \mathrm{rt}(\mathscr{B})(d_1 + d_2) \leq |w_0| + (DS(\alpha) - 1)(d_1 + d_2) & \text{if } R \neq Q, \\ 1 + (DS(\alpha) - 2)(d_1 + d_2) & \text{if } R = Q. \end{cases}$$

Proof. Let $x \in V_R \setminus \langle [R] \rangle$. We have

$$(x, [q]) \neq (x, \alpha) \text{ for some } q \in R. \tag{1}$$

Since $[q] \in V_R$ and W_2W_1 is complete for V_R with respect to α, we can represent it as follows:

$$[q] = \sum_{w_1 \in W_1, w_2 \in W_2} \lambda_{w_1,w_2} \alpha[w_2][w_1] \text{ for some } \lambda_{w_1,w_2} \in \mathbb{R}. \tag{2}$$

Multiplying (2) by the vector $[Q]$ we obtain

$$1 = ([q],[Q]) = \sum_{w_1 \in W_1, w_2 \in W_2} \lambda_{w_1,w_2} (\alpha[w_2][w_1],[Q]) = \sum_{w_1 \in W_1, w_2 \in W_2} \lambda_{w_1,w_2}. \quad (3)$$

Multiplying (2) by the vector x we obtain

$$([q],x) = \sum_{w_1 \in W_1, w_2 \in W_2} \lambda_{w_1,w_2} (\alpha[w_2][w_1],x). \quad (4)$$

Arguing by contradiction, suppose $(x,\alpha[u_2][u_1]) = (x,\alpha)$ for every $u_1 \in W_1$, $u_2 \in W_2$. Then by (3) and (4) we get that $([q],x) = (x,\alpha)$ contradicts (1). Hence $(x,\alpha[u_2][u_1]) \neq (x,\alpha)$, for some $u_1 \in W_1$, $u_2 \in W_2$.

Since $\alpha[P_2][P_1] = \alpha$, we have either $(x,\alpha[u_2][u_1]) > (x,\alpha)$ or $(x,\alpha[v_2][v_1]) > (x,\alpha)$ for some other $v_1 \in W_1, v_2 \in W_2$. Thus Claim 1 follows.

The proof of Claims 2 and 3 follows from an application of the *greedy extension algorithm* from Sect. 5. □

Remark 1. If W_2 is complete for $\mathbb{R}^n$ with respect to some vector g, then W_2W_1 is complete for V_R with respect to g.

Criterion 1. *Let α be a stationary distribution of the Markov chain associated with a strongly connected n-state automaton $\mathscr{A}$ by a given positive probability distribution P on the alphabet Σ. Then $\mathscr{A}$ is synchronizing if and only if there exists a set of words W which is complete for $\mathbb{R}^n$ with respect to α.*

Proof. If $\mathscr{A}$ is synchronizing then for each state $q \in Q$ there is a reset word w_q such that $Q.w_q = q$. Hence, $W = \{w_q \mid q \in Q\}$ is complete for $\mathbb{R}^n$ with respect to α, because $\alpha[w_q] = [q]$.

Let us prove the opposite direction. Set

$$W_1 = \{\varepsilon\},\ W_2 = \Sigma^{\leq n-1},\ \text{and}\ [P_2] = \frac{1}{n}\sum_{i=0}^{n-1}[P]^i.$$

Then $\alpha[P_2] = \alpha$, and W_2 is complete for $\mathbb{R}^n$ with respect to α. Hence $\mathscr{A}$ is synchronizing by Theorem 1. □

Now we can provide an upper bound for the reset threshold, if we can find a short word of a small rank.

Theorem 2. *Let $\mathscr{A} = (Q,\Sigma,\delta)$ be a synchronizing automaton. Then there is a unique (strongly connected)* sink component *$\mathscr{S} = (S,\Sigma,\delta)$. Let w be a word and denote $r = |Q.w|$. Let $0 < d < n$ be the smallest positive integer such that $\Sigma^{\leq d}$ is complete for V_S with respect to any stochastic vector $g \in V_S$ and for each $q \in Q$ there is a word $u_q \in \Sigma^{\leq d}$ such that $q.u_q \in S \cap Q.w$. Then*

$$\mathrm{rt}(\mathscr{A}) \leq \begin{cases} (|w|+d)\left(\frac{r^3-r}{6}\right) - d & \text{if } r \geq 4; \\ |w| + (|w|+d)(r-1)^2 & \text{if } r \leq 3. \end{cases}$$

Proof. Let $W_1 = \{w\}$, $W_2 = \Sigma^{\leq d}$, $w_0 = w$, and let P_1, P_2 be arbitrary positive distributions on W_1 and W_2, respectively. We define $\mathscr{B} = \mathscr{A}_c(W_1, W_2)$ as in Theorem 1, and consider its sink component $\mathscr{C} = \mathscr{S}_c(W_1, W_2) = (Q_C, \Sigma, W_2W_1)$. Clearly $Q_C = Q.w \cap S$, and W_2W_1 is complete for $V_{Q_C} \leq V_S$ with respect to any stochastic vector $g \in V_{Q_C}$. By Criterion 1 we obtain that $\mathscr{C}$ is synchronizing. Since for each $q \in Q.w$ there is a word $u_q \in W_2$ and so $w_q \in W_2W_1$ (a letter of $\mathscr{B}$) which takes q to Q_C, the automaton $\mathscr{B}$ is synchronizing.

Since $\mathscr{B}$ is synchronizing, $|Q.w_0| = r$, and $|u| \leq |w| + d$ for each $u \in W_2W_1$, we have that $\mathrm{rt}(\mathscr{A}) \leq |w| + \mathrm{rt}(\mathscr{B})(|w|+d)$. By Pin's bound for the reset threshold in the general case [24], $\mathrm{rt}(\mathscr{B}) \leq \frac{r^3-r}{6} - 1$ for $r \geq 4$. □

3 The Černý Conjecture and Random Automata

Using the new bound, we can extend the class of automata for which the Černý conjecture is proven. In particular, we can improve the result from [22], where the Černý conjecture is proven for automata with a letter of rank at most $1 + \log_2 n$.

Corollary 1. *Let $\mathscr{A} = (Q, \Sigma, \delta)$ be a synchronizing automaton. If there is a letter of rank $r \leq \sqrt[3]{6n-6}$, then $\mathscr{A}$ satisfies the Černý conjecture.*

Another corollary concerns random synchronizing automata. We consider the uniform distribution P_s on all synchronizing binary automata with n states, which is formally defined by $P_s(\mathcal{A}) = P(\mathcal{A})/P_n$, where P is the uniform distribution on all n^{2n} binary automata, and P_n is the probability that a uniformly random binary automaton is synchronizing. It is known that P_n tends to 1 as n goes to infinity [4,20].

Given an arbitrary small $\varepsilon > 0$ and n large enough, Nicaud [20] proved that with probability at least $1 - O(n^{-1/8+\varepsilon})$ a uniformly random binary automaton has a reset word of length $n^{1+\varepsilon}$. He also proved that with probability at least $1 - O(\exp(n^{-\varepsilon/4}))$, some word of length $n^{3/4+3\varepsilon}(1 + o(1))$ has rank at most $n^{1/4+2\varepsilon}$. Since the probability that a uniformly random binary automaton is synchronizing tends to 1, this also holds with asymptotically at least the same probability for random synchronizing binary automata. The following statement is a straightforward consequence of this result and our Theorem 2.

Corollary 2. *For any $\varepsilon > 0$ and n large enough, with probability at least $1 - O(\exp(n^{-\varepsilon/4}))$, a random n-state synchronizing automaton with at least two letters has a reset word of length at most $n^{7/4+6\varepsilon}(1 + o(1))$, and so satisfies the Černý conjecture. Therefore, the expected value of the reset threshold is at most $n^{7/4+o(1)}$.*

4 Synchronizing Finite Prefix Codes

A *finite prefix code* (Huffman code) $\mathcal{T}$ is a set of N ($N > 0$) non-empty words $\{w_1, \ldots, w_N\}$ from Σ^*, such that no word in $\mathcal{T}$ is a prefix of another word in $\mathcal{T}$.

A finite prefix code $\mathcal{T}$ is *maximal* if adding any word $w \in \Sigma^*$ to $\mathcal{T}$ does not result in a finite prefix code. We consider only maximal prefix codes. A *reset word* for the code $\mathcal{T}$ is a word w such that for any $u \in \Sigma^*$ the word uw is a sequence of words from $\mathcal{T}$.

One can easily see that a finite prefix code corresponds naturally to a DFA called the *decoder*, whose states are proper prefixes of words from this code [9]. Formally, for a finite prefix code $\mathcal{T}$ we have the corresponding *decoder* $\mathscr{A}_{\mathcal{T}}$, which is the DFA (Q, Σ, δ) with $Q = \{q_v \mid v \text{ is a proper prefix of a word in } \mathcal{T}\}$, and δ defined as follows:

$$\delta(q_v, a) = \begin{cases} q_{va} & \text{if } va \notin \mathcal{T}; \\ q_\varepsilon & \text{otherwise.} \end{cases}$$

Clearly, a reset word w for a code is a reset word for its decoder, and $Q.w = \{q_\varepsilon\}$. A decoder naturally corresponds to a rooted k-ary tree, thus the number of states $n = (kN-1)/(k-1)$, and it does not depend on the length of the words in the code.

In [8,9] Biskup and Plandowski gave an $O(nh \log n)$ upper bound for the reset thresholds of binary decoders, where h is the maximum length of a word from the code. Since h can be linear in terms of n, this is an $O(n^2 \log n)$ general bound. Later, it was improved to $O(n^2)$ in [2]. However, in the worst case, only decoders with a reset threshold in $\Theta(n)$ are known [9], and it was conjectured that every synchronizing decoder possess a synchronizing word of length $O(n)$. Thus, there was a big gap between the upper and lower bounds for the worst case. The following lemma is a simple generalization of [9, Lemma 14] to k-ary decoders.

Lemma 1. *Let $\mathscr{A}_{\mathcal{T}} = (Q, \Sigma, \delta)$ be the n-state k-ary synchronizing decoder of a finite prefix code $\mathcal{T}$. There is a word w of rank $r \le \lceil \log_k n \rceil$ and length r.*

Since there exists a short word of small rank r, we can apply Theorem 2 to improve the general upper bounds for the reset threshold of decoders.

Corollary 3. *Let $\mathscr{A}_{\mathcal{T}} = (Q, \Sigma, \delta)$ be the n-state k-ary synchronizing decoder of a finite prefix code $\mathcal{T}$, and let $r = \lceil \log_k n \rceil$. Then*

$$\operatorname{rt}(\mathscr{A}_{\mathcal{T}}) \le \begin{cases} 2 + (r+n-1)(\frac{r^3-r}{6} - 1) & \text{if } r \ge 4; \\ 2 + (r+n-1)(r-1)^2 & \text{if } r \le 3. \end{cases}$$

If the size k of the alphabet is fixed, Corollary 3 yields $O(n \log^3 n)$ upper bound for the reset threshold, and $O(n \log^2 n)$ upper bound for the length of a word compressing a pair of states of a decoder.

Note that the word w from Lemma 1 can be easily computed in $O(n^2)$ time, since there are $O(n)$ words of length at most $\lceil \log_k n \rceil$. Then a reset word within the bound of Corollary 3 can be computed in polynomial time by the algorithm discussed in Sect. 5.

5 Finding Reset Words of the Bounded Lengths

Throughout this section suppose we are given a strongly connected automaton $\mathscr{A}$, a word w_0 such that $Q.w_0 = R$ for some $R \subseteq Q$, a non-empty polynomial set of words W_1 with a positive distribution P_1, and a set of words W_2 with a positive distribution P_2, which satisfy Theorem 1.

Consider the case when W_2 is of polynomial size. Then we can calculate the dominant eigenvector $\alpha \in \mathbb{R}^n$ of the matrix $[P_2][P_1]$. Under certain assumptions on rationality of the distributions, it can be done in polynomial time. Next, depending on whether the bound is obtained by Theorem 2 or Claim 2 of Theorem 1, we use either a greedy compressing algorithm (such as in [14]), or the following *greedy extension algorithm*, respectively.

The Greedy Extension Algorithm. We start from $x_0 = [q]$ for $q \in R$ and by Claim 1 of Theorem 1 find $u_0 \in W_2W_1$ such that $(x_0, \alpha[u_0]) > (x_0, \alpha)$. For $i = 0, 1, \ldots$ following this way until $x_i \in \langle [R] \rangle$, find for $x_{i+1} = x_i[u_i]^t$ a word $u_{i+1} \in W_2W_1$ such that $(x_{i+1}, \alpha[u_{i+1}]) > (x_{i+1}, \alpha)$. Since x_i is a 1-0 vector, we need at most $\mathrm{DS}(\alpha) - 1$ steps until $x_i = [q]([u_i u_{i-1} \ldots u_0])^t = [R]$. As the result we return the word $w_0 u_i u_{i-1} \ldots u_0$. Notice that in the case when $R = Q$ we can choose q such that for some letter $a \in \Sigma$, we have $|q.a^{-1}| > 1$ and set $u_0 = a$. □

The problem is that usually W_2 is given by $\Sigma^{\leq d}$ for some $d = \mathrm{poly}(n)$. The following reduction procedure allows to replace potentially exponential set W_2 with a polynomial set of words W, whose the longest words are not longer than those of W_2.

The Reduction Procedure. The procedure takes a number $d \geq 0$, and returns a polynomial subset $W \subseteq \Sigma^{\leq d}$ such that $\langle W \rangle = \langle \Sigma^{\leq d} \rangle$ and the maximum length of words from W is the shortest possible.

We start with $V_0 = \{I_n\}$ and $W = \{\varepsilon\}$. In each iteration $i \in \{1, 2, \ldots\}$ we first set $V_{i+1} = V_i$. Then we subsequently check each letter $a \in \Sigma$ and each word $u \in W$ of length i: If the matrix $[ua]$ does not belong to the subspace V_{i+1}, we add the word ua to W and the matrix $[ua]$ to the basis of V_{i+1}. We stop the procedure at the first iteration where nothing is added.

Since in an i-th iteration we have considered $a \in \Sigma$ and $u \in W$ of length less than i in the previous iterations, by induction we get

$$V_i = \langle I_n(W \cap \Sigma^{\leq i}) \rangle = \langle I_n \Sigma^{\leq i} \rangle.$$

It follows from the ascending chain argument (see e.g. [18,26]) that for some $j < n$ we have

$$V_j = V_{j+1} = \ldots.$$

Thus the procedure is stopped at the first such j, and $j \leq \min\{d, n-1\}$. We get that $\langle W \rangle = V_j = \langle \Sigma^d \rangle$. Since in each step we add only independent matrices as the basis of V_{i+1}, we get $|W| = \dim(V_j)$. Also the lengths of words in W are at most $j \leq \min\{d, n-1\}$. □

Using the reduction procedure for total completeness we can replace Σ^d from Theorem 2 by a polynomial W which is also complete for V_S with respect to any stochastic vector $g \in V_S$. Hence, this yields a polynomial time algorithm finding reset words of lengths within the bound of Theorem 2.

In some situations we are interested only in completeness with respect to a given vector α. Then we can find a reduced set W of potentially shorter words than that obtained by the general reduction procedure.

The Reduction Procedure for α-Completeness. The procedure takes a number $d \geq 0$ and a vector $\alpha \in \mathbb{R}^n$, and returns a polynomial subset $W \subseteq \Sigma^{\leq d}$ such that $\langle \alpha W \rangle = \langle \alpha \Sigma^{\leq d} \rangle$ and the maximum length of words from W is the shortest possible.

We just follow the general reduction procedure, where instead of matrix spaces we consider vector spaces. It is enough to replace I_0 by α, and we obtain $\langle \alpha W \rangle = V_j = \langle \alpha \Sigma^{\leq d} \rangle$. □

Remark 2. Instead of $\Sigma^{\leq d}$ the reduction procedures can also reduce any set of words $W' \subset \Sigma^*$ that is factor-closed. A set of words W' is *factor-closed* if $uvw \in W'$ implies that $uw \in W'$, for each $u, v, w \in \Sigma^*$.

5.1 Synchronizing Quasi-Eulerian Automata

Let α be the probability distribution on $\Sigma^{\leq d}$ induced by a probability distribution $P \colon \Sigma \mapsto \mathbb{R}^+$ on the alphabet, that is, $[P_2] = \frac{1}{d+1} \sum_{i=0}^{d} [P]^i$. Suppose that $d < \mathrm{poly}(n)$ is such that $\Sigma^{\leq d}$ is complete for $\mathbb{R}^n$ with respect to α. Using the reduction procedure, we can construct a set U of at most n words such that $\langle \alpha U \rangle = \langle \alpha \Sigma^{\leq d} \rangle = \mathbb{R}^n$. However, α is not necessarily the stationary distribution for some positive probability distribution on U. The following lemma solves this problem.

Lemma 2. *Let $W = \{au \mid u \in \mathrm{Suff}(U), a \in \Sigma\}$, where $\mathrm{Suff}(U)$ is the set of proper suffixes of U. Then there exists a positive probability distribution on W such that α is the corresponding stationary distribution.*

As an application we get a polynomial algorithm for finding a reset word for the class of *quasi-Eulerian* automata, a generalization of Eulerian automata. We call an automaton $\mathscr{A}$ *quasi-Eulerian* with respect to an integer $c \geq 0$ if it satisfies the following two conditions:

1. there is a subset $E_c \subseteq Q$ containing $n - c$ states such that only one of these states, say s, can have incoming edges from the set $Q \setminus E_c$;
2. there exists a positive probability distribution P on Σ such that the columns of the matrix $[P]$ that correspond to the states from $E_c \setminus \{s\}$ sum up to 1.

Within this definition, for $c = 0$ we get so-called *pseudo-Eulerian* automata, and if additionally P is uniform on Σ, then we get Eulerian automata. The upper bound $1 + (n-2)(n-1)$ on the reset thresholds of Eulerian automata was found by Kari [18], and extended to the class of pseudo-Eulerian automata by Steinberg [25]. These results were generalized in [5, Corollary 11] by showing the upper bound $2^c(n-c+1)(n-1)$ for the class of quasi-Eulerian automata with respect to a non-negative integer c. The following theorem gives a polynomial time algorithm for finding reset words satisfying these bounds.

Theorem 3. *Given a synchronizing automaton $\mathscr{A}$ which is quasi-Eulerian with respect to an integer $c \geq 0$, there is a polynomial time algorithm for finding a reset word of length at most:*

$$\begin{cases} 2^c(n-c+1)d & \text{if } c > 0; \\ 1+(n-2)d & \text{if } c = 0, \end{cases}$$

where $d \leq n-1$ is the smallest integer such that $\Sigma^{\leq d}$ is complete.

5.2 Synchronizing Quasi-One-Cluster Automata

The *underlying digraph* of a letter $a \in \Sigma$ is the digraph with edges labeled by a. Every connected component, called *cluster*, in the underlying digraph of a letter has exactly one cycle, and possible some trees rooted on this cycle. An automaton $\mathscr{A} = (Q, \Sigma, \delta)$ is called *one-cluster* if there is a letter $a \in \Sigma$ whose underlying digraph has only one cluster. An automaton $\mathscr{A}$ is *quasi-one-cluster* with respect to an integer $c \geq 0$ if it has a letter whose underlying digraph has a cluster such that there are at most c states in the cycles of all other clusters. Clearly, one-cluster automata are quasi-one-cluster with respect to $c = 0$.

The Černý conjecture was proved for one-cluster automata with prime length cycle [26]. Also, quadratic bounds for the reset thresholds in the general case of one-cluster automata were presented [2,3,10,25]. In [5] the upper bound $2^c(2n-c-2)(n-c+1)$ was proved for quasi-one-cluster with respect to c.

The following theorem gives a polynomial algorithm finding a reset word for quasi-one-cluster automata, whose length is of the mentioned bounds. It can be also easily modified to deal with the bounds from [10] for one-cluster automata.

Theorem 4. *Let $\mathscr{A}$ be a synchronizing automaton that is quasi-one-cluster with respect to a letter a and $c \geq 0$. Let C be the largest cycle of a and h be the maximal height of the trees labeled by a. Let $W_1 = \{a^{h+i} \mid i \in \{0, \ldots, |C|-1\}\}$. Then there is a polynomial algorithm for finding a reset word for $\mathscr{A}$ of length at most*

$$\begin{cases} 2^c(2n-c)(n-c+1) & \text{if } c > 0; \\ 1+(2n-r)(n-2) & \text{if } c = 0, \end{cases}$$

where r is the smallest dimension of $\langle W_1 \beta \rangle$ for $\beta \in V_C \setminus \langle [C] \rangle$. In particular, if $|C|$ is prime then $r = |C|$.

References

1. Ananichev, D., Gusev, V., Volkov, M.: Slowly synchronizing automata and digraphs. In: Hliněný, P., Kučera, A. (eds.) MFCS 2010. LNCS, vol. 6281, pp. 55–65. Springer, Heidelberg (2010)
2. Béal, M.P., Berlinkov, M.V., Perrin, D.: A quadratic upper bound on the size of a synchronizing word in one-cluster automata. Int. J. Found. Comput. Sci. **22**(2), 277–288 (2011)
3. Béal, M.-P., Perrin, D.: A quadratic upper bound on the size of a synchronizing word in one-cluster automata. In: Diekert, V., Nowotka, D. (eds.) DLT 2009. LNCS, vol. 5583, pp. 81–90. Springer, Heidelberg (2009)
4. Berlinkov, M.V.: On the probability to be synchronizable (2013). http://arxiv.org/abs/1304.5774
5. Berlinkov, M.V.: Synchronizing quasi-eulerian and quasi-one-cluster automata. Int. J. Found. Comput. Sci. **24**(6), 729–745 (2013)
6. Berlinkov, M.V.: On two algorithmic problems about synchronizing automata. In: Shur, A.M., Volkov, M.V. (eds.) DLT 2014. LNCS, vol. 8633, pp. 61–67. Springer, Heidelberg (2014)
7. Berlinkov, M.V., Szykuła, M.: Algebraic synchronization criterion and computing reset words (2014). http://arxiv.org/abs/1412.8363
8. Biskup, M.T.: Shortest synchronizing strings for huffman codes. In: Ochmański, E., Tyszkiewicz, J. (eds.) MFCS 2008. LNCS, vol. 5162, pp. 120–131. Springer, Heidelberg (2008)
9. Biskup, M.T., Plandowski, W.: Shortest synchronizing strings for huffman codes. Theoret. Comput. Sci. **410**(38–40), 3925–3941 (2009)
10. Carpi, A., D'Alessandro, F.: Independent sets of words and the synchronization problem. Adv. Appl. Math. **50**(3), 339–355 (2013)
11. Carpi, A., D'Alessandro, F.: Černý-like problems for finite sets of words. In: Proceedings of the 15th Italian Conference on Theoretical Computer Science, Perugia, Italy, September 17–19, 2014. pp. 81–92 (2014)
12. Černý, J.: Poznámka k homogénnym eksperimentom s konečnými automatami. Matematicko-fyzikálny Časopis Slovenskej Akadémie Vied **14**(3), 208–216 (1964)
13. Dubuc, L.: Sur les automates circulaires et la conjecture de Černý. Informatique Théorique et Applications **32**, 21–34 (1998)
14. Eppstein, D.: Reset sequences for monotonic automata. SIAM J. Comput. **19**, 500–510 (1990)
15. Gawrychowski, P., Straszak, D.: Strong inapproximability of the shortest reset word. In: Italiano, G.F., et al. (eds.) MFCS 2015, Part I, LNCS 9234, pp. 243–255. Springer, Heidelberg (2015)
16. Gerbush, M., Heeringa, B.: Approximating minimum reset sequences. In: Domaratzki, M., Salomaa, K. (eds.) CIAA 2010. LNCS, vol. 6482, pp. 154–162. Springer, Heidelberg (2011)
17. Jürgensen, H.: Synchronization. Inform. Comput. **206**(9–10), 1033–1044 (2008)
18. Kari, J.: Synchronizing finite automata on Eulerian digraphs. Theoret. Comput. Sci. **295**(1–3), 223–232 (2003)
19. Kari, J., Volkov, M.V.: Černý's conjecture and the road coloring problem. In: Handbook of Automata. European Science Foundation (to appear)
20. Nicaud, C.: Fast synchronization of random automata (2014). http://arxiv.org/abs/1404.6962

21. Olschewski, J., Ummels, M.: The complexity of finding reset words in finite automata. In: Hliněný, P., Kučera, A. (eds.) MFCS 2010. LNCS, vol. 6281, pp. 568–579. Springer, Heidelberg (2010)
22. Pin, J.E.: Utilisation de l'algèbre linéaire en théorie des automates. In: Act. Colloq. AFCET-SMF Math. Appl. II. pp. 85–92. AFCET (1978)
23. Pin, J.E.: Sur un cas particulier de la conjecture de Černý. Automata, Languages and Programming. LNCS, pp. 345–352. Springer, Heidelberg (1978). in French
24. Pin, J.E.: On two combinatorial problems arising from automata theory. In: Proceedings of the International Colloquium on Graph Theory and Combinatorics, vol. 75, pp. 535–548. North-Holland Mathematics Studies (1983)
25. Steinberg, B.: The averaging trick and the Černý conjecture. Int. J. Found. Comput. Sci. **22**(7), 1697–1706 (2011)
26. Steinberg, B.: The Černý conjecture for one-cluster automata with prime length cycle. Theoret. Comput. Sci. **412**(39), 5487–5491 (2011)
27. Volkov, M.V.: Synchronizing automata and the Černý conjecture. In: Martín-Vide, C., Otto, F., Fernau, H. (eds.) LATA 2008. LNCS, vol. 5196, pp. 11–27. Springer, Heidelberg (2008)