Combinatorics, Words and Symbolic Dynamics

Edited by
Valérie Berté and Michel Rigo

Contents

Contributors

Valérie Berthé *CNRS, LIAFA, Bât. Sophie Germain, Univ. Paris Diderot, Paris 7 - Case 7014, F-75205 Paris Cedex 13, France.*
`berthe@liafa.univ-paris-diderot.fr`

Michel Rigo *University of Liège, Department of Mathematics, Grande Traverse 12 (B37), B-4000 Liège, Belgium.*
`M.Rigo@ulg.ac.be`

Marie-Pierre Béal *your affiliation.*
`your mail`

Dominique Perrin *your affiliation.*
`your mail`

1

Preliminaries

Valérie Berthé and Michel Rigo

1.1 A first test

We let $\mathbb{N}$ denote the set of non-negative integers.

Definition 1.1.1 Let A be an alphabet. An *infinite word* over A is a map $w : \mathbb{N} \to A$.

See for instance (Berthé and Rigo, 2010) or (Lothaire, 1983).

1.2 Another section

Here again, we consider an example...

Remark 1.2.1 Observe the following.

Theorem 1.2.2 *Over a binary alphabet, any finite word of length at least* 4 *contains a square.*

Proof The proof is trivial. □

2

Synchronized automata

Marie-Pierre Béal and Dominique Perrin

2.1 Introduction

The notions of synchronizing word and synchronized automaton are simple to define and occur in many applications of automata. It is remarkable that they are linked with difficult combinatorial problems. Some of them (like the Černý Conjecture) are open since a long time. One (the Road Coloring Problem) was solved by Trahtman in 2009. A reason to explain the difficulty of these questions may be the fact that there is no simple description of the class of automata which are not synchronized. It is relatively simple to verify whether an automaton is synchronized since it can be checked with a polynomial algorithm. On the contrary, it was proved by Eppstein **?** that finding a synchronizing word of minimal length is NP-hard.

In this chapter, we present a survey of results concerning synchronized automata. We first define the notions of synchronizing words and synchronized automata for deterministic automata. We extend these notions to the more general class of unambiguous automata.

In a second section, we present Černý Conjecture and discuss several particular cases where a positive answer is known. This includes the important case of aperiodic automata solved by Trahtman in 2007.

In the next section, we present some results which give sufficient conditions for the existence of synchronizing words.

In the last section, we present the Road Coloring Theorem (*i.e.* the solution of Trahtman to the Road Coloring Problem). We also present a quadratic algorithm to find a synchronized coloring and a generalization of the Road Coloring Theorem to automata with a periodic graph.

2.2 Definitions

A (finite) *automaton* $\mathscr{A}$ over some (finite) alphabet A is composed of a finite set Q of states and a finite set E of edges which are triples (p,a,q) where p,q are states

and a is a symbol from A called the *label* of the edge. Note that no initial or terminal states are specified. This means that all states are both initial and terminal.

Let $\mathscr{A} = (Q,E)$ be an automaton on the alphabet A. We denote by $\varphi_{\mathscr{A}}$ the morphism from A^* into the monoid of binary relations on Q which associates to the word $w \in A^*$ the relation $\varphi_{\mathscr{A}}(w) = \{(p,q) \mid p \xrightarrow{a} q\}$. The monoid $\varphi_{\mathscr{A}}(A^*)$ is the *transition monoid* of $\mathscr{A}$.

An automaton is *irreducible* if its underlying graph is strongly connected.

An automaton is *deterministic* if, for each state p and each letter a, there is at most one edge starting in p and labeled with a. It is *complete deterministic* if, for each state p and each letter a, there is exactly one edge starting in p and labeled with a. This implies that, for each state p and each word w, there is exactly one path starting in p and labeled with w. If this unique path ends in a state q, we denote by $p \cdot w$ the state q.

A *synchronizing word* of a deterministic automaton is a word w such that there is least one path labeled by w and all paths labeled by w have the same terminal state. A synchronizing word is also called a *reset sequence* or a *magic sequence*, or also a *homing word*. An automaton which has a synchronizing word is called *synchronized* A pair of states (p,q) is *synchronizable* if there is a word w such that $p \cdot w = q \cdot w$. More generally, a set of states S is *synchronizable* if there is a word w such that $\mathrm{card}(S \cdot w) = 1$. The word w is said to *synchronize* S.

Proposition 2.2.1 *A complete deterministic automaton is synchronized if and only if any pair of states is synchronizable.*

Proof Let $\mathscr{A} = (Q,E)$ be a complete deterministic automaton. Let $S \subset Q$ be a set of the form $Q \cdot u$ of minimal cardinality for all word $u \in A^*$. If S contains two distinct elements p,q, there is a word v which synchronizes the pair (p,q). Then $\mathrm{card}(Q \cdot uv) = \mathrm{card}(S \cdot v) < \mathrm{card}\, S$, a contradiction. Thus $\mathrm{card}(S) = 1$ and u is a synchronizing word. □

2.3 Černý's Conjecture

Černý (1964) constructed synchronized n-state deterministic complete automata for which the length of a shortest synchronizing word is $(n-1)^2$. He formulated the Černý's Conjecture which asserts the existence of a synchronizing word of length at most $(n-1)^2$ for any synchronized n-state deterministic complete automaton.

The best upper bound known for the length $\ell(n)$ of a shortest synchronizing word in an n-state deterministic complete automaton obtained so far is cubic (the bound $\ell(n) = n(7n^2+6n-16)/48$ is obtained in Trahtman (2011)). A simple proof of the existence of synchronizing word of length at most cubic in a synchronized n-state deterministic complete automata is the following.

Proposition 2.3.1 *A synchronized n-state deterministic complete automaton has a synchronizing word of length at most $n(n-1)^2/2$.*

Proof Let $\mathscr{A} = (Q,E)$ be a synchronized n-state deterministic complete automaton. Since $\mathscr{A}$ is synchronized, any pair of states is synchronizable by Proposition 2.2.1. Let P be a subset of Q. If $|P| > 1$, let p,q in P and u be such that $p \cdot u = q \cdot u$. One can choose a word u satifying this equality of minimal length. Let us show that the length of u is at most $n(n-1)/2$. Indeed, if $|u| > n(n-1)/2$, let $u = u_1 \cdots u_k$, with $k > n(n-1)/2$. Then the pairs $(p \cdot u_1 \cdots u_i, q \cdot u_1 \cdots u_i)$, for $0 \le i < k$, are k pairs of distinct states. As a consequence, two of them are equal, which contradicts the fact that u is as short as possible. Hence there is a word u of length at most $n(n-1)/2$ such that $|P \cdot u| < |P|$. Starting with $P = Q$ and iterating the construction, we get a word $v = u_1 u_2 \cdots u_{n-1}$ of length at most $n(n-1)^2/2$ such that $P \cdot v = 1$.

□

Though the Černý's Conjecture is still open in general, it has been settled for important and large particular classes of automata.

2.3.1 Aperiodic automata

A deterministic automaton is *aperiodic* if there is a nonnegative integer k such that for any word w and any state q, one has

$$q \cdot w^k = q \cdot w^{k+1}.$$

Note that, since the automaton is finite, there always exists a nonnegative integer k and a positive integer p such that for any word w and any state q, one has

$$q \cdot w^k = q \cdot w^{k+p}.$$

The condition expresses the fact that one can choose $p = 1$ in the above condition. Aperiodic automata can be defined equivalently by the fact that their transition monoid does not contain any non trivial group, see Eilenberg (1976). An equivalent formulation is the following. For a subset S of the set of states, let $\mathrm{Stab}(S) = \{u \in A^* \mid S \cdot u = S\}$ and let $G(S)$ denote the set of restrictions to S of the elements of $\mathrm{Stab}(S)$. The set $G(S)$ is a permutation group on S. Indeed, the restriction to S of an element of $\mathrm{Stab}(S)$ is a permutation on S.

Then the automaton is aperiodic if and only if for each S, the permutation group $G(S)$ is reduced to the identity. This condition can be used in practice to check that an automaton is aperiodic as in the following example.

Example 2.3.1 Consider the automaton on the set of states $Q = \{1,2,3,4\}$ represented in Figure 2.1. To check that the automaton is aperiodic, we compute in Figure 2.2 the action of the alphabet on the subsets of Q of the form $Q \cdot u$ for $u \in A^*$ with more than one element. Then $ab \in \mathrm{Stab}(\{1,2,3\})$ and the restriction of ab to $\{1,2,3\}$ is the identity. It implies that the restriction of ab to $\{1,2\}$ and $\{2,3\}$ is also the identity and thus that the automaton is aperiodic.

Note that any strongly connected aperiodic automaton is synchronized. Indeed,

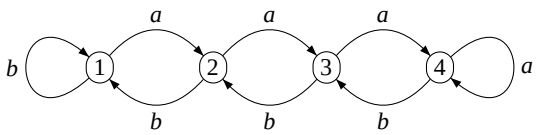

Figure 2.1 An aperiodic automaton

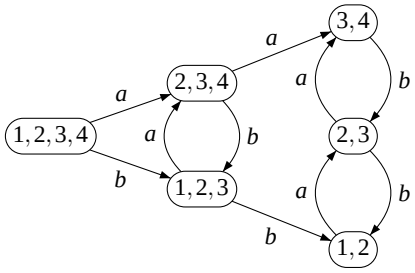

Figure 2.2 The action on subsets of Q.

assume the $\mathscr{A} = (Q,E)$ is a strongly connected aperiodic automaton. Let S be a set of the form $Q \cdot u$ for $u \in A^*$ of minimal cardinality. Since $\mathscr{A}$ is strongly connected, the group $G(S)$ is a transitive permutation group. Since $\mathscr{A}$ is aperiodic, it is reduced to the identity. This forces $\mathrm{card}(S) = 1$ and thus $\mathscr{A}$ is synchronized.

The following result was proved by Trahtman (2007). It shows that Černý Conjecture is true for aperiodic automata.

Theorem 2.3.2 *Let $\mathscr{A}$ be a synchronized n-state deterministic complete automaton. If $\mathscr{A}$ is aperiodic, then it has a synchronizing word of length at most $n(n-1)/2$.*

For an automaton $\mathscr{A} = (Q,E)$, we denote by $\mathscr{A}^2$ the automaton on the set of pairs (p,q) of *distinct* states whose edges are the $(p,q) \xrightarrow{a} (p \cdot a, q \cdot a)$ for $a \in A$ whenever $p \cdot aq \cdot a$. Let C be a set of pairs of states, which we may consider as a binary relation on Q. We denote by $<_C$ the transitive closure of the relation C, by $\leq_C$ the reflexive closure of $<_C$ and by $\equiv_C$ the equivalence closure of $<_C$.

A strongly connected component of an automaton is a strongly connected component of its associated graph. A strongly connected component is *maximal* if it is a maximal element for the order induced by the accessibility relation. Thus a strongly connected component of $\mathscr{A}^2$ is maximal if for any edge $(p,q) \xrightarrow{a} (r,s)$ of $\mathscr{A}^2$, $(p,q) \in C$ implies $(r,s) \in C$.

Lemma 2.3.3 *For any maximal strongly connected component C of $\mathscr{A}^2$, the relations $\leq_C$ and $\equiv_C$ are stable.*

Proof Let $p \leq_C q$ and let $a \in A$ be a letter. Let p_i for $0 \leq i \leq n$ be such that $p_0 = p$, $p_n = q$ and $(p_i, p_{i+1}) \in C$ for $0 \leq i \leq n-1$. For each i we have either $p_i \cdot a = p_{i+1} \cdot a$ or $(p_i \cdot a, p_{i+1} \cdot a) \in C$ since C is a maximal strongly connected component. In both

cases $p_i \cdot a \leq_C p_{i+1} \cdot a$. Thus $p \cdot a \leq_C q \cdot a$. This shows that $\leq_C$ is stable. As the equivalence closure of a stable relation, $\equiv_C$ is stable. □

Proposition 2.3.2 *Let C be a maximal strongly connected component of $\mathscr{A}^2$. If $\mathscr{A}$ is an aperiodic automaton, then $\leq_C$ is a partial order.*

Proof Let us prove by induction on $n \geq 1$ that if $(p_0, p_1, \ldots, p_{n+1})$ is such that $(p_i, p_{i+1}) \in C$ for $0 \leq i \leq n$ then $p_0 p_{n+1}$.

Since C is strongly connected and since $(p_0, p_1), (p_1, p_2)$ are in C, there is a word u such that $p_0 \cdot u = p_1$ and $p_1 \cdot u = p_2$. Since $\mathscr{A}$ is aperiodic, there is a $k \geq 1$ such that $p_0 \cdot u^k = p_0 \cdot u^{k+1}$.

For $n = 1$, $p_0 = p_2$ implies that u is in $\mathrm{Stab}(\{p_0, p_1\})$ and that $G(\{p_0, p_1\})$ contains a transposition, a contradiction. Consider now $n \geq 2$. Set $q_i = p_i \cdot u^{k-1}$ for

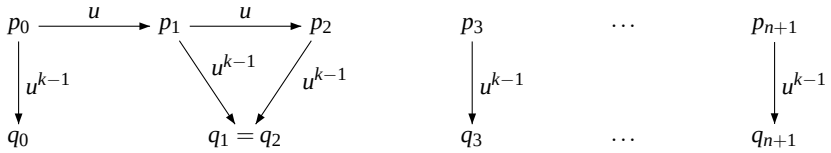

Figure 2.3

$0 \leq i \leq n+1$. We may assume that k is chosen minimal in such a way that $q_0 q_1$. We have $q_2 = p_2 \cdot u^{k-1} = p_0 \cdot u^{k+1} = p_0 \cdot u^k = p_1 \cdot u^{k-1} = q_1$. Since C is maximal, the sequence $(q_0, q_1, q_3, q_4, \ldots, q_{n+1})$ (see Figure 2.3) is such that any two consecutive elements are either equal or form a pair in C. By induction hypothesis, we cannot have $q_0 = q_{n+1}$ and thus $p_0 = p_{n+1}$ is also impossible. □

Example 2.3.4 Consider the aperiodic automaton of Example 2.3.1. The part of automaton $\mathscr{A}^2$ corresponding to the pairs (i, j) with $i < j$ is represented in Figure 2.4. The set $C = \{(1,2), (2,3), (3,4)\}$ is a maximal strongly connected component. The

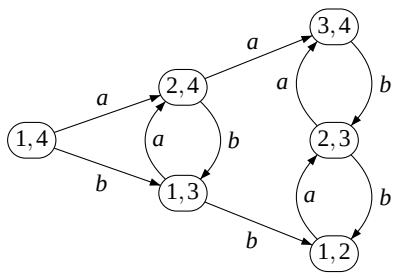

Figure 2.4 The automaton $\mathscr{A}^2$.

corresponding partial order is $1 <_C 2 <_C 3 <_C 4$.

Let $\mathscr{A}$ be an aperiodic automaton with n states. Let C be a maximal strongly connected component of $\mathscr{A}^2$. We say that a set R of states is *linked* with respect to

C if for any $r,s \in R$ there exist a sequence $(p_0, p_1, \ldots, p_m)$ of elements of R such that $p_0 = r$, $p_m = s$ and for $0 \leq i \leq m-1$, $p_i \leq_C p_{i+1}$ or $p_{i+1} \leq_C p_i$. Thus all the elements of R are in particular equivalent modulo $\equiv_C$ and thus contained in an equivalence class. Conversely, any equivalence class of $\equiv_{\mathbb{C}}$ is linked.

Proposition 2.3.3 *Let $\mathscr{A}$ be an aperiodic automaton with n states. Let C be a maximal strongly connected component of $\mathscr{A}^2$. Let R be a set of states linked with respect to C and let M be the set of minimal elements of R for the partial order $\leq_C$. Then R is synchronized by a word of length at most* $\mathrm{card}(M)(n-1)$.

Proof If $\mathrm{card}(R) = 1$, there is nothing to prove since the empty word synchronizes R. Thus we may assume that $\mathrm{card}(R) \geq 2$. We prove the statement by induction on $\mathrm{card}(M)$.

XXXXXXXXXXXXXXXXX

the integer

$$k(R) = \begin{cases} \mathrm{card}(M) & \text{if } \mathrm{card}(R) \geq 2 \\ 0 & \text{otherwise.} \end{cases}$$

If $k(R) = 0$, then $\mathrm{card}(R) = 1$ and the empty word synchronizes R. Assume that $k(R) \geq 1$. Choose some $p \in M$ and a word u such that $q = p \cdot u$ is a maximal element of Q. We may choose u of length at most $n-1$. Let $S = R \cdot u$ and let N be the set of minimal elements of S. We verify that the induction hypothesis can be applied to S. We may assume that $k(S) \geq 1$.

First the set S is linked with respect to C since $\leq_C$ is stable. Next, $k(S) \leq k(R) - 1$. Indeed, we first note that $N \subset M \cdot u$ because the relation $\leq_C$ is stable. Let $n \in N$. There is $s \in R$ such that $s \cdot u = n$. There is $p \in M$ such that $p \leq_C s$. Then $p \cdot u \leq_C s \cdot u$. We cannot have $p \cdot u <_C s \cdot u$ since $n \in N$. Thus $p \cdot u = s \cdot u$ which implies that $n \in M \cdot u$.

Next, we have $q \notin N$. Let indeed r be an element of N and let $s \in S$ be distinct of r. Since S is linked, there exist a sequence $(p_0, p_1, \ldots, p_m)$ of elements of S such that $p_0 = r$, $p_m = s$ and for $0 \leq i \leq m-1$, $p_i \leq_C p_{i+1}$ or $p_{i+1} \leq_C p_i$. All the p_i cannot be equal since rs. Let j be the least index such that $p_j p_{j+1}$. Then $r = p_j <_C p_{j+1}$ or $p_{j+1} <_C p_j = r$. The second case is not possible since r is minimal in S. Thus r is not maximal. This shows that a minimal element of S cannot be maximal in S and therefore neither maximal in Q. Thus $q \notin N$. This shows that N is strictly contained in $M \cdot u$ and thus that $k(S) < k(R)$. Thus the induction hypothesis can be applied to S. Let v be a word of length at most $\mathrm{card}(N)(n-1)$ which synchronizes S. Then uv is a word of length at most $(\mathrm{card}(N)+1)(n-1) \leq \mathrm{card}(M)(n-1)$ which synchronizes R. □

Observe that the same statement holds for the set of maximal elements of R. Indeed, we may exchange the role of minimal and maximal elements in the proof.

We can now complete the proof of Trahtman's theorem.

Proof of Theorem 2.3.2. Assume first that $\mathscr{A} = (Q, E)$ is strongly connected.

We prove the statement by induction on $\mathrm{card}(Q)$. If $\mathrm{card}(Q) = 1$, the property

holds trivially. Let C be a maximal strongly connected component of $\mathscr{A}^2$. The equivalence $\equiv_C$ is stable and is not the equality relation. Let r be the number of its classes. We may apply the induction hypothesis to the quotient $\mathscr{B} = \mathscr{A}/\equiv_C$. Let u be a word of length at most $r(r-1)/2$ which is synchronizing for $\mathscr{B}$. Let $R = Q \cdot u$. The set R is linked. We may assume that the set of minimal elements of R satisfies $\mathrm{card}(M) \leq \mathrm{card}(R)/2 \leq (n-r+1)/2$ (otherwise we use the set of maximal elements of R).

By Proposition 2.3.3, there is a word v of length at most $\mathrm{card}(M)(n-1) \leq (n-r+1)(n-1)/2$ which synchronizes R. Then uv is a synchronizing word for $\mathscr{A}$ of length at most

$$\frac{r(r-1)}{2} + \frac{(n-r+1)(n-1)}{2} \leq \frac{(n-1)(r-1)}{2} + \frac{(n-r+1)(n-1)}{2} = \frac{n(n-1)}{2}$$

since $r \leq n-1$.

In the general case, since $\mathscr{A}$ is synchronized, there is a unique maximal strongly connected component C. Set $m = \mathrm{card}(C)$. By the previous case, there is a word u of length at most $m(m-1)/2$ which synchronizes C. Let $\mathscr{B}$ be the automaton obtained from $\mathscr{A}$ by merging the states in C. Then $\mathscr{B}$ is an aperiodic synchronizing automaton with $n-m$ states. We may assume by induction that it has a synchronizing word v of length at most $(n-m)(n-m-1)/2$. Thus vu is a synchronizing word of length at most $n(n-1)/2$. ∎

The bound of Theorem 2.3.2 has been improved in several ways. Trahtman (2008b) has obtained a better bound of $(n-1)^2/2$ for all aperiodic automata. Next, Volkov (2007) has obtained a bound of $n(n+1)/6$ for strongly connected aperiodic automata. Actually no example is known of an aperiodic strongly connected automaton which does not have a synchronizing word of length at most $n-1$.

2.3.2 Independent sets

In this section, as in the previous one, $\mathscr{A} = (Q,E)$ denotes an n-state deterministic and complete automaton over an alphabet A. We fix a particular letter $a \in A$. Other letters are called b-letters.

The subgraph of the graph of $\mathscr{A}$ made of the edges labeled by a is a disjoint union of connected component called *a-clusters*. Since each state has exactly one outgoing edge in this subgraph, each a-cluster contains a unique cycle, called an a-cycle, with trees attached to the cycle at their root.

For each state p of an a-cluster, we define the *level* of p as the distance between p and the root of the tree containing p. If p belongs to the cycle, its level is null. The *level* of the automaton is the maximal level of its states.

A *one-cluster automaton* with respect to a letter a is a complete deterministic automaton which has only one a-cluster. Equivalently, an automaton is one-cluster if it satisfies the following condition: for any pair of states p,q, one has $p \cdot a^r = q \cdot a^s$ for some integers r,s.

A one-cluster automaton of level 0 is called a *circular automaton*.

The Černý conjecture was proved by Dubuc (1998) for circular automata.

Proposition 2.3.4 *A circular synchronized n-state deterministic complete automaton has a synchronizing word of size at most* $(n-1)^2$.

Let $\mathscr{A} = (Q,E)$ be a complete deterministic automaton. If P is a subset of Q and W a set of words, we denote by $P \cdot W = \{p \cdot w \mid p \in P, w \in W\}$. An *independent set* of a complete deterministic automaton $\mathscr{A} = (Q,E)$ is a set of words W of size r for which there is a set of states R of size r such that $p \cdot W = R$ for any state p of Q. The set R is called the *range* of the independent set.

Equivalently, W is an independent set with range R, if for each $p \in Q$ and $s \in R$, there is a unique $w \in W$ such that $p \cdot w = s$. Note that, if W is an independent set with range R, then, for any word u, the set uW is an independent set with the same range.

A complete deterministic automaton is *locally strongly transitive* if it has an independent set. The automaton is said to be strongly transitive if the independent set is the set of all states. Note that a one-cluster automaton is locally strongly transitive. Indeed, if r is the length of the a-cycle and ℓ the level of the automaton, the set $W = \{a^\ell, a^{\ell+1}, \dots, a^{\ell+r}\}$ is an independent set with the a-cycle as the range.

The following result is due to Carpi and D'Alessandro Carpi and D'Alessandro (2012)

Proposition 2.3.5 *Let $\mathscr{A}$ be a synchronized n-state deterministic complete automaton. If $\mathscr{A}$ is locally strongly transitive with an independent set W, then it has a synchronizing word of length at most* $(r-1)(n+\ell_{\max}+2) - r\log(r) + \ell_{\min}$, *where r is the number of words in W, and $\ell_{\min}$ (resp. $\ell_{\max}$) the minimal (resp. maximal) length of words of W.*

Before proving Proposition 2.3.5, we need some preliminaries.

Let $\mathscr{A} = (Q,E)$ be a complete deterministic automaton. Let u be a word. We denote by $\varphi(u)$ the $0-1$-matrix defined by

$$\varphi(u)_{pq} = \begin{cases} 1 & \text{if } p \cdot u = q, \\ 0 & \text{otherwise.} \end{cases}$$

Note that if u, v are two words, we have $\varphi(uv) = \varphi(u)\varphi(v)$.

If P is a set of states, we denote by $\underline{P}$ its row characteristic vector and by $\underline{P}^t$ the transpose column vector of $\underline{P}$. If p is a state, we also note $\underline{p}$ the row characteristic vector of the set $\{p\}$. If u is a word, We denote $Pu^{-1} = \{q \in Q \mid q \cdot u \in P\}$. Note that $\underline{Pu^{-1}} = \varphi(u)\underline{P}^t$.

The following lemma is well known (see (Eilenberg, 1974, p. 145) or (Sakarovitch, 2003, p. 492 corollaire 4.19)). It plays a key role in the proof of Proposition 2.3.5.

Lemma 2.3.5 *Let x be a Q-row vector with coefficients in $\mathbb{Q}$ and $\mathscr{V}$ a vector space of $\mathbb{Q}^Q$. If there is word u such that $x\varphi(u) \notin \mathscr{V}$, then there is a word u of length at most the dimension of $\mathscr{V}$ such that $x\varphi(u) \notin \mathscr{V}$.*

Proof For $i \geq 0$, let $\mathscr{V}_i$ be the subspace of $\mathbb{Q}^n$ which is spanned by the vectors $x\varphi(u)$ with $|u| \leq i$. We have $\mathscr{V}_i \subset \mathscr{V}_{i+1}$ for $i \geq 0$. Let i_0 be the minimal index i such that $\mathscr{V}_i = \mathscr{V}_{i+1}$. Then $\mathscr{V}_{i_0} = \mathscr{V}_i$ for any $i \geq i_0$. Let d be the dimension of $\mathscr{V}$. Assume that, for all words u of length at most d, one has $x\varphi(u) \in \mathscr{V}$. Then $\mathscr{V}_d \subset \mathscr{V}$. This implies $i_0 \leq d-1$ and $\mathscr{V}_{i_0} \subset \mathscr{V}$, and which is contradiction. □

For $P, R \subset Q$ and $u \in A^*$, set

$$\rho_{P,R}(u) = \mathrm{card}(Pu^{-1} \cap R).$$

The following lemma follows from the definitions.

Lemma 2.3.6 *Let P, R be subsets of Q and let u be a word. We have*

$$\rho_{P,R}(u) = \underline{R}\,\varphi(u)\,\underline{P}^t.$$

Lemma 2.3.7 *If W is an independent set of range R of a complete deterministic automaton, then, for any subset P of R,*

$$\sum_{w \in W} \rho_{P,R}(w) = r\,\mathrm{card}(P).$$

Proof From Lemma 2.3.6 and since, by independency, $\underline{p}(\sum_{i=1}^{r} \varphi(w_i) = \underline{R}$, we obtain

$$\begin{aligned}\sum_{w \in W} \rho_{P,R}(w) &= \sum_{w \in W} \underline{R}\,\varphi(w)\,\underline{P}^t = \underline{R}(\sum_{w \in W} \varphi(w))\,\underline{P}^t = \sum_{p \in R} \underline{p}(\sum_{w \in W} \varphi(w))\,\underline{P}^t \\ &= \sum_{p \in R} \underline{R}\,\underline{P}^t = \mathrm{card}(R)\,\mathrm{card}(P) = r\,\mathrm{card}(P).\end{aligned}$$

□

Let W be an independent set of $\mathscr{A}$ with a range R. Set $\rho = \rho_{P,R}$. Let P be a subset of R. We say that a word u is *P-augmenting* if

$$\rho(u) > \mathrm{card}(P),$$

or equivalently, if $\underline{R}\,\varphi(u)\,\underline{P}^t > \underline{R}\,\underline{P}^t$.

We now prove Proposition 2.3.5.

Proof of Proposition 2.3.5 Let W be an independent set of $\mathscr{A}$ with a range R of size r.

We prove that there is a sequence of synchronizable subsets $(P_i)_{0 \leq i \leq t}$ of R together with P_i-augmenting words $u_i \in A^*W$ such that $P_0 \subsetneq P_1 \cdots \subsetneq P_{t-1} \subsetneq P_t = R$, and, for $i < r$, $P_{i+1} = P_i u_i^{-1} \cap R$ and

$$|u_i| \leq n + \ell_{\max} - \frac{r - p_i}{p_i},$$

where $p_i = \mathrm{card}(P_i)$. For this, we set $P_0 = \{p\}$, where p is some state in R, and assume $P_1, \ldots, P_j$ already built. We set $P = P_j$ and assume that PR. Set also $p = \mathrm{card}(P)$.

For any word v, the set vW is an independent set of $\mathscr{A}$ of range R. By Lemma 2.3.7, we have

$$\sum_{w\in W} \rho(vw) = \mathrm{card}(R)p.$$

Then either $\rho(vw) = p$ for any $w \in W$ or there is a word $w_0 \in W$ such that $\rho(vw_0) > p$. We can moreover assume that $Pw^{-1}\emptyset$ and $Pw^{-1}Q$ for any word w in W. Indeed, if $Pw^{-1} = Q$ for some word w, we choose $u_j = w$, $P_{j+1} = R$ and get the claim. If $Pw^{-1} = \emptyset$, $Pw^{-1} \cap R = \emptyset$. Thus $Pw^{-1} \cap R\,\mathrm{card}(P)$ since $\mathrm{card}(P) > 0$. We again get the claim choosing $v = \varepsilon$.

Let $w \in W$. We have the following equivalences.

$$\begin{aligned}
&\rho(vw) - p = 0\\
&\Leftrightarrow \rho(vw) - \mathrm{card}(Qv^{-1} \cap R)\frac{p}{r} = 0\\
&\Leftrightarrow \underline{R}\,\varphi(v)\varphi(w)\underline{P}^t - (\underline{R}\,\varphi(v)\,\underline{Q}^t)\,\frac{p}{r} = 0\\
&\Leftrightarrow (\underline{R}\,\varphi(v))(\varphi(w)\underline{P}^t - \frac{p}{r}\underline{Q}^t) = 0\\
&\Leftrightarrow (\underline{R}\,\varphi(v))B = 0,
\end{aligned}$$

where $B = A - (p/r)U$, A denotes the $(Q \times W)$-matrix formed by the r columns $(\varphi(w)\underline{P}^t)$ for $w \in W$, and U is the $(Q \times W)$-matrix with all entries equal to 1.

Let I be a maximal set of linearly independent rows of A. Since W is an independent set, each row of A has exactly p nonnull entries. This implies $\mathrm{rank}(A)p \geq r$. Indeed, if $\mathrm{rank}(A)p < r$, there is an index ℓ such that each row of I has a null entry at this index. As all rows of A are linear combinations of elements of I, this implies that A has a null column, contrary to the fact that $Pw^{-1}\emptyset$ for any word w of W. Hence

$$\mathrm{rank}(B) \geq \mathrm{rank}(A) - 1 \geq \frac{r}{p} - 1 = \frac{r-p}{p}.$$

Thus the equality $\rho(vw) = p$ is satisfied for any $w \in W$ if and only the vector $\underline{R}\,\varphi(v)$ belongs to the vector space $\mathscr{V} = \{x \in \mathbb{Q}^n \mid xB = 0\}$. The vector space $\mathscr{V}$ has dimension $n - \mathrm{rank}(B)$.

Since by hypothesis PR, there is a word v and a word w in W such that $\rho(vw)p$. Indeed, let v be a synchronizing word for $\mathscr{A}$, and w in W. We have $P(vw)^{-1} = Q$ and thus $\rho(vw) = r$. Hence there is a word v such that $R\varphi(v) \notin \mathscr{V}$. By Lemma 2.3.5, we deduce that there is a word v of length at most $n - \mathrm{rank}(B)$ such that $R\varphi(v) \notin \mathscr{V}$. It follows that there is P-augmenting word vw of size at most $n - \mathrm{rank}(B) + \ell_{\max}$ and get the claim.

The range R is then synchronized by the word $z = u_{t-1}\cdots u_0$. Since $t \leq r-1$, the

length of z is at most

$$\begin{aligned}|z| &\leq (r-1)(n+\ell_{\max}) - \sum_{j=1}^{r-1} \frac{r-j}{j} \\ &= (r-1)(n+\ell_{\max}+2) - r\sum_{j=1}^{r-1}\frac{1}{j} \\ &= (r-1)(n+\ell_{\max}+2) - r\log r.\end{aligned}$$

Let now w a shortest word of W, hence of length $\ell_{\min}$. The word wz is a synchronizing word of $\mathscr{A}$ of length at most $(r-1)(n+\ell_{\max}+2) - r\log r + \ell_{\min}$. □

The following result is a consequence of Proposition 2.3.5

Proposition 2.3.6 *Let $\mathscr{A}$ be a synchronized n-state deterministic complete automaton. If $\mathscr{A}$ is one-cluster, then it has a synchronizing word of length at most $2n^2 - n\log(n)$.*

Proof Since $\mathscr{A}$ is assumed to be one-cluster, its unique a-cycle is the range of size r of the idependent set $W = \{a^\ell, a^{\ell+1}, \ldots, a^{\ell+r}\}$, where ℓ is the level of $\mathscr{A}$. We can assume that $\ell_{max} \leq (n-1), \ell_{min} \leq (n-r)$. Thus $\mathscr{A}$ has a synchronizing word of length at most

$$(r-1)(2n+1) + (n-r) - r\log r \leq 2nr - n - 1 - r\log r$$

Let us finally check that

$$2nr - n - 1 - r\log r \leq 2n(n-1) + n - 1 - n\log n$$

The above inequality is equivalent to

$$n\log n - r log r \leq 2n(n-1-r) + 2n$$

We now assume that $r \leq (n-1)$ since Proposition 2.3.4 solves the case $r = n$. Using $\log x \leq (x-1)$ (with the neperian logarithm) we get

$$\begin{aligned}n\log n - r\log r &= r\log\frac{n}{r} + \log(n)(n-r) \\ &= r\frac{n-r}{r} + (n-1)(n-r) \\ &\leq n(n-r) \leq n((n-1-r)+(n-1-r)+1).\end{aligned}$$

Hence $\mathscr{A}$ has a synchronizing word of length at most $2n(n-1) + n - 1 - n\log n = 2n^2 - n - 1 - n\log n$, which completes the proof. □

2.3.3 Unambiguous automata

An automaton is *unambiguous* if, for any pair p,q of states, and any word w, there is at most one path labeled by w going from p to q. A deterministic automaton is unambiguous. The converse is not true.

One may check that an automaton is unambiguous by computing its square. The square of $\mathscr{A}$ is the automaton on $Q \times Q$ with edges $(p,q) \xrightarrow{a} (r,s)$ if $p \xrightarrow{a} r$ and $q \xrightarrow{a} s$ are edges of $\mathscr{A}$. The automaton $\mathscr{A}$ is unambiguous if and only if there is no path in its square of the form $(p,p) \xrightarrow{u} (r,s) \xrightarrow{v} (q,q)$ with rs.

Example 2.3.8 Let $\mathscr{A}$ be the automaton represented in Figure 2.5 on the left. This

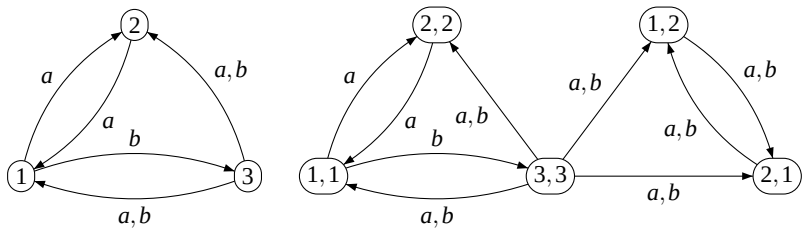

Figure 2.5 An unambiguous automaton and part of its square.

automaton is unambiguous as one may check by computing the square of the automaton $\mathscr{A}$ represented on the right in Figure 2.5 (with only the states accessible from the states (p,p) represented).

A word w is *synchronizing* for an unambiguous automaton if there is at least one path labeled by w and, for any states p,q,r,s,

$$p \xrightarrow{w} q,\ r \xrightarrow{w} s \text{ implies } p \xrightarrow{w} s \text{ and } r \xrightarrow{w} q.$$

The automaton is synchronized if there is a synchronizing word.

This is consistent with the previous definition. Indeed, if the automaton is deterministic, the condition implies that $q = s$. Moreover, if x,y are synchronizing words, there is a unique state q such that $p \xrightarrow{x} q \xrightarrow{y} r$ for some $p,r \in Q$. Indeed, if $p \xrightarrow{x} q \xrightarrow{y} r$ and $p' \xrightarrow{x} q' \xrightarrow{y} r'$, then since x,y are synchronizing, we have also $p \xrightarrow{x} q' \xrightarrow{y} r$ and thus $q = q'$ by unambiguity.

The notion of synchronizing word for an unambiguous automaton is related to the notion of rank of a binary relation on a set. The *rank* of a relation m on a set Q is the minimal cardinality of a set R such that $m = uv$ with u a $Q \times R$ relation and v an $R \times Q$ relation. A word w is synchronizing for unambiguous automaton $\mathscr{A}$ if and only if the relation $\varphi_{\mathscr{A}}(w)$ has rank one. Indeed, if w is synchronizing, then $\varphi_{\mathscr{A}}(w) = uv$ where u is the column Q-vector v is the row Q-vector defined as follows. One has $u_p = 1$ if and only if the is a path labeled w starting from p and $v_q = 1$ if and only if there is a path labeled w ending in q.

Example 2.3.9 The automaton of Figure 2.3.8 is synchronized. Indeed, the word

ab is such that

$$\varphi_{\mathscr{A}}(ab) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

and thus is synchronizing.

A unambiguous automaton is *complete* if, for any word w, there exists a pair p,q of states such that $p \xrightarrow{w} q$ is a path.

The following result is due to Carpi (1988).

Proposition 2.3.7 *If $\mathscr{A}$ is be an n-state irreducible complete unambiguous automaton. If $\mathscr{A}$ is synchronized, it has a synchronizing word of length at most $(n^2-n+2)(n-1)/2$.*

We introduce some terminology concerning monoids of relations on a set (see Berstel et al. (2007) for a more detailed presentation). For a relation m on a set Q, we denote indifferently $(p,q) \in m$ or $m_{pq} = 1$, considering the relation either as a subset of $Q \times Q$ or as a boolean $Q \times Q$ matrix. We denote by m_{p*} the row of of index p of the matrix m, which is the characteristic vector of the set $\{q \in Q \mid (p,q) \in m\}$.

A monoid of relations M on a set Q is *unambiguous* if for any $m,n \in M$ and $p,q \in Q$ there exists an most one element r of Q such that $(p,r) \in m$ and $(r,q) \in n$. The monoid is *transitive* if for any $p,q \in M$ there is an $m \in M$ such that $(p,q) \in m$.

An automaton is unambiguous if and only if its transition monoid is unambiguous. The automaton is strongly connected if and only its transition monoid is transitive.

We use the following lemmas.

A row of an element of M which is maximal among the rows of the elements of M is called a *maximal row*. The following is Exercise 9.3.5 in Berstel et al. (2007).

Lemma 2.3.10 *Let M be a transitive unambiguous monoid of relations not containing zero. The following conditions are equivalent for a row vector $\mathbf{r}$.*

(i). $\mathbf{r}$ *is a maximal row.*
(ii). $\mathbf{r}$ *is a row of $m \in M$ with a minimal number of nonzero distinct rows.*
(iii). $0 \notin \mathbf{r}M$.

Proof (i)$\Rightarrow$ (ii). Let $\mathbf{r} = m_{p*}$ be a maximal row and let $m' \in M$ have minimal number of nonzero distinct rows. Let q,s be such that $m'_{qs} = 1$ and let $n \in M$ be such that $n_{sp} = 1$. Then $(m'n)_{qp} = 1$ implies that $(m'nm)_{q*} = \mathbf{r}$ and $m'nm$ has minimal number of nonzero distinct rows.

(ii)$\Rightarrow$ (iii). Indeed let $\mathbf{r} = m_{p*}$ where m has minimal number of nonzero distinct rows. If $\mathbf{r}n = 0$, then mn has less nonzero distinct rows than m.

(iii)$\Rightarrow$ (i). Indeed, assume that $\mathbf{r}$ is not maximal and let $\mathbf{r}' > \mathbf{r}$ be a row of an element of M. Let q be such that $(\mathbf{r}' - \mathbf{r})_q = 1$. Let $m \in M$ be such that $\mathbf{s} = m_{p*}$ is a maximal row and let $n \in M$ be such that $n_{qp} = 1$. Then $\mathbf{r}'nm \geq w$ and thus $\mathbf{r}' = \mathbf{s}$. This forces $\mathbf{r}nm = 0$ and thus $0 \in \mathbf{r}M$. $\square$

Note that Lemma 2.3.10 implies in particular that, in a transitive unambiguous monoid of relations, the set of maximal rows in closed by multiplication on the right by an element of M. Indeed, this is clearly true for the set of row vectors satisfying condition (iii).

Lemma 2.3.11 *Let M be a transitive unambiguous monoid of relations not containing zero. For two elements m, m' of M, if $m \leq m'$ then $m = m'$.*

Proof Suppose that $m'_{pq} = 1$ for some $p, q \in Q$. Since M is transitive and does not contain zero, there exists a maximal row $\mathbf{r}$ such that $\mathbf{r}_p = 1$. Let us assume that $\mathbf{r} = n_{s*}$ for some $n \in M$. Then $nm \leq nm'$ and $(nm)_{s*} = n_{s*}m$ is a maximal row. Thus $(nm)_{s*} = (nm')_{s*}$. This forces $m_{pq} = 1$ since $m \leq m'$. □

Lemma 2.3.12 *For a state $p \in Q$ and a word $u \in A^*$, if $\varphi(u)_{p*}$ is not a maximal row, there is a state q and a word v of length at most $n(n-1)/2$ such that $\varphi(u)_{p*} < \varphi(vu)_{q*}$.*

Proof Let $p \in Q$ and $u \in A^*$, be such that $\varphi(u)_{p*}$ is not a maximal row. Since $\mathscr{A}$ is strongly connected, there exists a maximal row $\mathbf{r}$ such that $\mathbf{r}_p = 1$. There is at least a state p' distinct of p such that $r_{p'} = 1$ and $\varphi(u)_{p'*}0$ since otherwise $\mathbf{r}\varphi(u)$ is not maximal. Hence there is a state $q \in Q$ and a word v of length at most $n(n-1)/2$ such that $q \xrightarrow{v} p$ and $q \xrightarrow{v} p'$. Then $\varphi(u)_{p*} < \varphi(vu)_{q*}$. This proves the claim. □

Lemma 2.3.13 *Let M be a transitive unambiguous monoid of relations of minimal rank 1. Then for any maximal row $\mathbf{r}$ and any maximal column $\mathbf{c}$, one has $\mathbf{cr} \in M$ and $\mathbf{rc} = 1$.*

Proof The minimal number of nonzero distinct rows of an element of M is 1 and the same holds for columns. By a Lemma 2.3.10, there exists an $m \in M$ with all its nonzero rows equal to $\mathbf{r}$. Then $m = \mathbf{tr}$ with $\mathbf{t}$ a maximal column. Similarly, there is an $n \in M$ such that $n = \mathbf{us}$ for some maximal row $\mathbf{s}$. Since M is unambiguous, one has $\mathbf{st} \in \{0,1\}$. Since $0 \notin M$ one has $\mathbf{st} = 1$. Thus $nm = \mathbf{uv}$ and $\mathbf{uv} \in M$.

Finally, $\mathbf{vu} \in \{0,1\}$ because M is unambiguous and $\mathbf{vu}0$ since $0 \notin M$. Thus $\mathbf{vu} = 1$. □

Proof of Proposition 2.3.7. By Lemma 2.3.12 and its symmetric form, there exist pairs $(p_1,u_1), (p_2,u_2), \ldots, (p_s,u_s)$ in $Q \times A^*$ and $(v_1,q_1), (v_2,q_2), \ldots, (v_t,q_t)$ in $A^* \times Q$ such that, with $x_i = \varphi(u_i \cdots u_1)_{p_i*}$ and $y_j = \varphi(v_1 \cdots v_j)_{*q_i}$,

(i). $u_1 = v_1 = 1$ and $p_1 = q_1$.
(ii). for $2 \leq i \leq s$, the word u_i has length at most $n(n-1)/2$ and $x_i > x_{i-1}$.
(iii). for $2 \leq j \leq t$, the word v_j has length at most $n(n-1)/2$ and $y_j > y_{j-1}$.
(iv). x_s is a maximal row and y_t is a maximal column.

Let $u = u_s \cdots u_1$ and $v = v_1 \cdots v_t$. We have $|u| \leq (s-1)n(n-1)/2$ and $|v| \leq (t-1)n(n-1)/2$. Thus $|uv| \leq (s+t-2)n(n-1)/2$.

Let finally $z \in A^*$ be such that $q_t \xrightarrow{z} p_s$ with $|z| \leq n-1$. Then $w = vzu$ is such

that $y_t x_s \leq \varphi(w)$. By Lemma2.3.13, $y_t x_s \in M$. Thus by Lemma2.3.11, this implies $\varphi(w) = y_t x_s$.

By Lemma2.3.13, we have $x_s y_t = 1$. Thus $s+t \leq \sum_{q\in Q}(x_s)_q + \sum_{q\in Q}(y_t)_q \leq n+1$.

This shows that w is a synchronizing word with

$$\begin{aligned}|w| &\leq (s+t-2)n(n-1)/2+n-1 \leq \frac{1}{2}n(n-1)^2+n-1\\ &\leq \frac{1}{2}(n^2-n+2)(n-1)\end{aligned}$$

and this concludes the proof. ■

Example 2.3.14 We illustrate the proof of Proposition 2.3.7 on the automaton of Example 2.3.8. We start with $u_1 = v_1 = \varepsilon$ and $p_1 = q_1 = 1$. The row $\varphi(\varepsilon)_{1*} = \begin{bmatrix}1 & 0 & 0\end{bmatrix}$ is not maximal but $\varphi(\varepsilon)_{1*} < \varphi(a)_{3*} = \begin{bmatrix}1 & 1 & 0\end{bmatrix}$. Thus we choose $u_2 = a$ and $p_2 = 3$. Symmetrically, the column $\varphi(\varepsilon)_{*1} = \begin{bmatrix}1 & 0 & 0\end{bmatrix}^t$ is not maximal but $\varphi(\varepsilon)_{*1} < \varphi(b)_{*3} = \begin{bmatrix}1 & 1 & 0\end{bmatrix}^t$. Thus we choose $v_2 = b$ and $q_2 = 3$. Then ab is a synchronizing word as already observed (Example 2.3.9).

2.3.4 The generalized conjecture

The *rank* of a word w in a deterministic automaton $\mathscr{A} = (Q,E)$ is the size of its image $Q \cdot w$. A synchronizing word has rank 1.

It has been conjectured by Pin (1978) that if an automaton admits a word of rank at most k, then there exists such a word of length at most $(n-k)^2$. This generalizes the Černý conjecture. Pin's conjecture was proved to be false by Kari (2001). The counterexample corresponds to $n = 6$ and $k = 2$. It is represented in Figure 2.6.

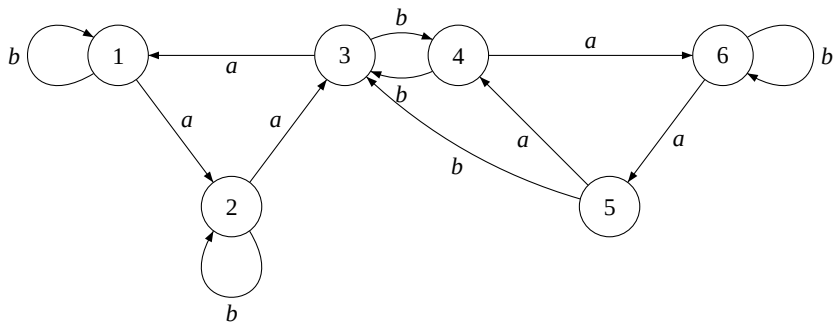

Figure 2.6 Kari's automaton.

The shortest words of rank 2 are $x = baababababbabaab$ and $y = baababaabaababaab$ of length 17. The shortest synchronizing word is $xaababaab$ of length $24 = 5^2 - 1$.

A new conjecture proposed by Volkov states that an automaton of minimal rank k admits a word of rank k of length at most $(n-k)^2$.

2.4 Divisibility conditions

In this section we present some results which give indications of the possible values of the minimal rank of an automaton. In particular, it gives sufficient conditions for an automaton to be synchronized.

2.4.1 Weights

Let $\mathscr{A} = (Q,E)$ be a strongly connected complete deterministic automaton on the alphabet A. The *adjacency matrix* of $\mathscr{A}$ is the $Q \times Q$ matrix M defined by

$$\varphi(pq) = \mathrm{card}\{a \in A \mid p \cdot a = q\}.$$

Let $k = \mathrm{card} A$. As a consequence of the Perron-Frobenius Theorem, there exists a positive left integer eigenvector v of M for the eigenvalue k. For any subset P of Q, we define the *weight* of P as $v(P) = \sum_{q \in P} v_q$.

The *nuclear equivalence* of a word w is the equivalence on Q defined by $p \equiv q$ if $p \cdot w = q \cdot w$. A *maximal class* is any class of some maximal nuclear equivalence.

The following result is due to Friedman (1990).

Proposition 2.4.1 *Any two maximal classes have the same weight.*

It follows from this result that the weight of a maximal class divides the weight V of the set of states. Thus the minimal rank divides V.

2.4.2 Minimal rank

A *transformation monoid* on a set Q is a submonoid M of the monoid of partial maps from Q into Q. It is *transitive* if for any p,q in Q, there is an m in M such that $pm = q$.

For a deterministic automaton $\mathscr{A} = (Q,E)$ we denote by $\varphi_{\mathscr{A}}$ the morphism which associates to a word w the partial map $p \mapsto p \cdot w$. Thus $\varphi_{\mathscr{A}}(A^*)$ is a transformation monoid on Q.

For a set $P \subset Q$, let

$$\mathrm{Stab}(P) = \{m \in \varphi_{\mathscr{A}}(A^*) \mid Pm \subset P\}$$

and let $\mathrm{Res}(P)$ be the restriction to P of the semigroup $\mathrm{Stab}(P)$.

The following result is due to Shin and Yoo (2010).

Theorem 2.4.1 *Let $\mathscr{A} = (Q,E)$ be a deterministic complete automaton. Let $R \subset Q$ be a set such that*

(i). the transformation monoid $\mathrm{Res}(P)$ *is transitive,*
(ii). there is a word w such that $Q \cdot w \subset P$.

Then the minimal rank of $\mathscr{A}$ divides the minimal rank of $\mathrm{Res}(P)$.

This generalizes a result of O'Brien (1981) which corresponds to the case where $\mathrm{Res}(P)$ is a circular permutation. The proof uses a generalization of Friedman's result (Proposition 2.4.1).

2.5 Road coloring

2.5.1 The Road Coloring Theorem

Imagine a map with roads which are colored in such a way that a fixed sequence of colors, called a homing sequence, leads the traveler to a fixed place whatever be the starting point. Such a coloring of the roads is called synchronized and finding a synchronized coloring is called the Road Coloring Problem. In terms of graphs, it consists in finding a synchronized labeling in a directed graph.

The Road Coloring Theorem states that every aperiodic directed graph with constant out-degree has a synchronized coloring (a graph is aperiodic if it is strongly connected and the gcd of its cycles is equal to 1). It has been conjectured under the name of the Road Coloring Problem by Adler et al. (1977), and solved for many particular types of automata (see for instance Adler et al. (1977), O'Brien (1981), Carbone (2001), Kari (2003), Friedman (1990), Perrin and Schützenberger (1992)). Trahtman (2009) settled the conjecture.

In the domain of coding, automata with outputs (*i.e.* transducers) can be used either as encoders or as decoders. When they are synchronized, the behavior of the coder (or of the decoder) is improved in the presence of noise or errors. For instance, the well known Huffman compression scheme leads to a synchronized decoder provided the lengths of the codewords of the Huffman code are mutually prime. It is also a consequence of the Road Coloring Theorem that coding schemes for constrained channels can have sliding block decoders and synchronized encoders (see Adler et al. (1983) and Lind and Marcus (1995)).

Trahtman's proof is constructive and leads to an algorithm that finds a synchronized labeling with a cubic worst-case time complexity Trahtman (2008a, 2009). The algorithm consists in a sequence of flips of edges going out of some state so that the resulting automaton is synchronized. One first searches a sequence of flips leading to an automaton which has a so-called stable pair of states (*i.e.* with good synchronizing properties). One then computes the quotient of the automaton by the congruence generated by the stable pairs. The process is then iterated on this smaller automaton. Trahtman's method for finding the sequence of flips leading to a stable pair has a worst-case quadratic time complexity, which makes his algorithm cubic.

The *period* of an automaton is the gcd of the cycles of its graph. An automaton is *aperiodic* if it is irreducible and of period 1[1].

Two automata which have isomorphic underlying graphs are called *equivalent*.

[1] Note that this notion, which is usual for graphs, is not the notion of aperiodic automata used in Section 2.3.1.

Hence two equivalent automata differ only by the labeling of their edges. In this section, we shall consider only complete deterministic automata.

Proposition 2.5.1 *A synchronized irreducible complete deterministic automaton is aperiodic.*

Proof Let d be the period of $\mathscr{A}$. We define a relation ρ on the set of states as follows. Two states p,q are related by ρ if there is a path of length multiple of d from p to q. This relation is an equivalence relation. Indeed, if $(p,q) \in \rho$, there is a path from p to q of length kd. Since the graph is strongly connected, there is a path from q to p of length ℓ. Since d divides $kd+\ell$, the length ℓ is a multiple of d.

Let us show now that any two states p,q are equivalent. Let ℓ be the length of a path from p to q. Let w be a synchronizing word in $\mathscr{A}$. Then d divides $\ell + |w| - |w| = \ell$. Just by considering one edge ($\ell = 1$), this implies $d = 1$. □

The *Road Coloring Theorem* can be stated as follows.

Theorem 2.5.1 (A. Trahtman Trahtman (2009)) *Any aperiodic complete deterministic automaton is equivalent to a synchronized one.*

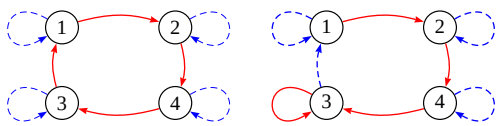

Figure 2.7 Two complete aperiodic deterministic automata over the alphabet $A = \{a,b\}$. A thick red plain edge is an edge labeled by a while a thin blue dashed edge is an edge labeled by b. The automaton on the left is not synchronized. The one on the right is synchronized. For instance, the word aaa is a synchronizing word. The two automata are equivalent since they share the same graph.

A trivial case for solving the Road Coloring Theorem is the case where the automaton has a loop edge around some state r. Indeed, since the graph of the automaton is strongly connected, there is a spanning tree rooted at r (with the edges of the tree oriented towards the root). Let us label the edges of this tree and the loop by the letter a. This coloring is synchronized by the word a^{n-1}, where n is the number of states.

2.5.2 An algorithm for finding a synchronized coloring

Trahtman's proof of Theorem 2.5.1 is constructive and gives an algorithm for finding a labeling (also called a coloring) which makes the automaton synchronized provided it is aperiodic.

In the sequel $\mathscr{A}$ denotes an n-state deterministic and complete automaton over an alphabet A. We fix a particular letter $a \in A$. Edges labeled by a are also called *red edges* or *a-edges*. The other ones are called *b-edges*. An a-path is sequence of consecutive edges labelled by a.

A pair (p,q) of states in an automaton is *stable* if, for any word u, the pair $(p\cdot u, q\cdot u)$ is synchronizable. In a synchronized automaton, any pair of states is stable. Note that if (p,q) is a stable pair, then for any word u, $(p\cdot u, q\cdot u)$ also is a stable pair, hence the terminology. Note also that, if (p,q) and (q,r) are stable pairs, then (p,r) also is a stable pair. It follows that the relation defined on the set of states by $p \equiv q$ if (p,q) is a stable pair, is an equivalence relation. It is actually a congruence (*i.e.* $p\cdot u \equiv q\cdot u$ whenever $p \equiv q$) called *the stable pair congruence*. More generally, a congruence is *stable* if any pair of states in the same class is stable. The congruence *generated* by a stable pair (p,q) is the least congruence such that p and q belong to the same class. It is a stable congruence. A set of edges going out of a state p is called a *bunch* if these edges all end in a same state q. Note that if a state q has two incoming bunches from two states p, p', then (p,p') is a stable pair.

If $\mathscr{A} = (Q,E)$ is an automaton, the *quotient* of $\mathscr{A}$ by a stable pair congruence is the automaton $\mathscr{B}$ whose states are the classes of Q under the congruence. The edges of B are the triples $(\overline{p}, c, \overline{q})$ where (p,c,q) is an edge of $\mathscr{A}$. The automaton $\mathscr{B}$ is complete deterministic when $\mathscr{A}$ is complete deterministic. The automaton $\mathscr{B}$ is irreducible (resp. aperiodic) when $\mathscr{A}$ is irreducible (resp. aperiodic).

The following Lemma was obtained by Culik et al. (2002).

Lemma 2.5.2 *If the quotient of an automaton $\mathscr{A}$ by a stable pair congruence is equivalent to a synchronized automaton, then there is a synchronized automaton equivalent to $\mathscr{A}$.*

Proof Let $\mathscr{B}$ be the quotient of $\mathscr{A}$ by a stable congruence and let $\mathscr{B}'$ be a synchronized automaton equivalent to $\mathscr{B}$. If p is a state of $\mathscr{A}$, we denote by $\overline{p}$ the class of p for the given stable congruence. We define an automaton $\mathscr{A}'$ equivalent to $\mathscr{A}$ as follows. The number of edges of $\mathscr{A}$ going out of p and ending in states belonging to a same class $\overline{q}$ is equal to the number of edges of $\mathscr{B}$ (and thus $\mathscr{B}'$) going out of $\overline{p}$ and ending in $\overline{q}$. We define $\mathscr{A}'$ by labeling these edges according to the labeling of corresponding edges in $\mathscr{B}'$. The automaton $\mathscr{B}'$ is a quotient of $\mathscr{A}'$.

Let us show that $\mathscr{A}'$ is synchronized. Let w be a synchronizing word of $\mathscr{B}'$ and r the state ending any path labeled by w in $\mathscr{B}'$. Let p,q be two states of $\mathscr{A}'$. Then $p\cdot w$ and $q\cdot w$ belong to the same congruence class r. Hence $(p\cdot w, q\cdot w)$ is a stable pair of $\mathscr{A}'$. Therefore (p,q) is a synchronizable pair of $\mathscr{A}'$. All pairs of $\mathscr{A}'$ being synchronizable, $\mathscr{A}'$ is synchronized. □

Trahtman's algorithm for finding a synchronized coloring of an aperiodic automaton $\mathscr{A}$ consists in finding an equivalent automaton $\mathscr{A}'$ of $\mathscr{A}$ which has at least one stable pair (s,t), then in recursively finding a synchronized coloring $\mathscr{B}'$ for the quotient automaton $\mathscr{B}$ by the congruence generated by (s,t), and finally in lifting up this coloring to the initial automaton as follows. If there is an edge (p,c,q) in $\mathscr{A}$ but no edge $(\overline{p}, c, \overline{q})$ in $\mathscr{B}'$, then there is an edge $(\overline{p}, d, \overline{q})$ in $\mathscr{B}'$ with cd. Then we flip the labels of the two edges labeled c and d going out of p in $\mathscr{A}$.

The algorithm is described in the following pseudocode. The procedure FIND-

STABLEPAIR, which finds an equivalent automaton which has a stable pair of states, is described in the next section. The procedure MERGE computes the quotient of an automaton by the stable congruence generated by a stable pair of states, while UPDATE updates some data needed for the computation.

```
FINDCOLORING (aperiodic automaton 𝒜)
𝒜 ← 𝒜
while (size(ℬ) > 1)
    do UPDATE(ℬ)
       ℬ, (s,t) ← FINDSTABLEPAIR(ℬ)
       lift the coloring up from ℬ to the automaton 𝒜
       ℬ ← MERGE(ℬ, (s,t))
return 𝒜
```

2.5.3 Finding a stable pair

In this section, we consider an aperiodic complete deterministic automaton $\mathscr{A}$ over the alphabet A. We present Trathaman's cubic-time algorithm for finding an equivalent automaton which has a stable pair.

In order to describe the algorithm, we give some definitions and notation.

The subgraph of the graph of $\mathscr{A}$ made of the red edges is a disjoint union of connected component called *clusters*. Since each state has exactly one outgoing edge in this subgraph, each cluster contains a unique (red) cycle with trees attached to the cycle at their root. If r is the root of such a tree, its *children* are the states p such that p is not on the a red cycle and (p,a,r) is an edge. If p,q belong to a same tree, p is an *ancestor* of q (or q is a *descendant* of p) in the tree if there is an a-path from q to p. If q belongs to some red cycle, its *predecessor* is the unique state p belonging to the same cycle such that (p,a,q) is an edge.

For each state p belonging to some cluster, we define the *level* of p as the distance between p and the root of the tree containing p. If p belongs to the cycle of the cluster, its level is thus null. The *level of an automaton* is the maximal level of its states. A *maximal state* is a state of maximal level. A *maximal tree* is a tree rooted at a state of level 0 containing at least one maximal state.

The algorithm for finding a coloring which has a stable pair relies on the following key lemma due to Trahtman Trahtman (2009). It uses the notion of minimal images in an automaton. An *image* in an automaton $\mathscr{A} = (Q,E)$ is a set of states $I = Q \cdot w$, where w is a word and $Q \cdot w = \{q \cdot w \mid q \in Q\}$. A *minimal image* in an automaton is an image which is a minimal element of the set of images for set inclusion. In an irreducible automaton two minimal images have the same cardinality which is called the *minimal rank* of $\mathscr{A}$. Also, if I is a minimal image and u is a word, then $I \cdot u$ is again a minimal image and the map $p \to p \cdot u$ is one-to-one from I onto $I \cdot u$.

Lemma 2.5.3 (Trahtman (2009)) *Let $\mathscr{A}$ be an irreducible complete deterministic automaton with a positive level. If all maximal states in $\mathscr{A}$ belong to the same tree, then $\mathscr{A}$ has a stable pair.*

Proof Since $\mathscr{A}$ is irreducible, there is a minimal image I containing a maximal state p. Let $\ell > 0$ the level of p (*i.e.* the distance between p and the root r of the unique maximal tree). Let us assume that there is a state qp in I of level ℓ. Then the cardinal of $I \cdot a^\ell$ is strictly less than the cardinal of I, which contradicts the minimality of I. Thus all states but p in I have a level strictly less than ℓ.

Let m be a common multiple of the lengths of all red cycles. Let $\mathbb{C}$ be the red cycle containing r. Let s_0 be the predecessor of r in $\mathbb{C}$ and s_1 the child of r containing p in its subtree. Since $\ell > 0$, we have $s_0 s_1$. Let $J = I \cdot a^{\ell-1}$ and $K = J \cdot a^m$. Since the level of all states of I but p is less than or equal to $\ell - 1$, the set J is equal to $\{s_1\} \cup R$, where R is a set of states belonging to the red cycles. Since for any state q in a red cycle, $q \cdot a^m = q$, we get $K = \{s_0\} \cup R$.

Let w be a word of minimal rank. For any word v, the minimal images $J \cdot vw$ and $K \cdot vw$ have the same cardinal equal to the cardinal of I. We claim that the set $(J \cup K) \cdot vw$ is a minimal image. Indeed, $J \cdot vw \subseteq (J \cup K) \cdot vw \subseteq Q \cdot vw$, hence all three are equal. But $(J \cup K) \cdot vw = R \cdot vw \cup s_0 \cdot vw \cup s_1 \cdot vw$. This forces $s_0 \cdot vw = s_1 \cdot vw$ since the cardinality of $R \cdot vw$ cannot be less than the cardinality of R. As a consequence $(s_0 \cdot v, s_1 \cdot v)$ is synchronizable and thus (s_0, s_1) is a stable pair. □

In the sequel, we call Condition C_1 the hypothesis of Lemma 2.5.3: *all maximal states belong to the same tree*.

We now describe sequences of flips such that the resulting equivalent automaton either has a stable pair, or has strictly more states of null level. We consider several cases corresponding to the geometry of the automaton.

- Case 1. We assume that the level of the automaton is $\ell = 0$. If the set of outgoing edges of each state is a bunch, then there is only one red cycle, and the automaton is not aperiodic unless the trivial case where the length of this cycle is 1. We can thus assume that there is a state p whose set of outgoing edges is not a bunch. There exists ba and qr such that (p, a, q) and (p, b, r) are edges. We flip these two edges. We obtain an automaton which satisfies Condition C_1. Let s be the state which is the predecessor of r in its red cycle. It follows from the proof of Lemma 2.5.3 that the pair (p, s) is a stable pair.
- Case 2. We assume that the level of the automaton is $\ell > 0$. Let r be the root of a maximal tree and p a maximal state of this tree. We consider a blue edge (t, b, p) ending in p. Note that, since p is a maximal state and since the automaton is irreducible, such an edge always exists. We denote $u = t \cdot a$.
  - Case 2.1. If t is not in the same cluster as r, or if t has a positive level and does not belong to the a-path from p to r, we flip the edges (t, b, p) and (t, a, u) and get an automaton which has a unique maximal tree.
  - Case 2.2. If t belongs to the a-path from p to r, we flip the edges (t, b, p) and (t, a, u) and get an automaton which has strictly more states of null level.

– Case 2.3. We assume that t belongs to the cycle containing r. Let k_1 be the length of the simple a-path from r to t and k_2 the length of the simple a-path from u to r (see Figure 2.8).

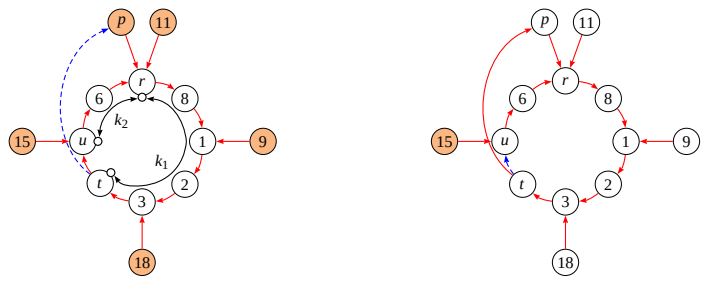

Figure 2.8 The picture on the left illustrates Case 2.3.1 ($k_2 = 2 > \ell = 1$). After flipping the edge (t,b,p) and (t,a,u), we get the automaton on the right of the figure. It has a unique maximal tree (rooted at r). Maximal states are colored and the (dashed) b-edges of the automaton are not all represented.

- Case 2.3.1. If $k_2 > \ell$, we flip the edges (t,b,p) and (t,a,u) and get an automaton which has a unique maximal tree (see Figure 2.8).
- Case 2.3.2. If $k_2 < \ell$, we flip the edges (t,b,p) and (t,a,u) and get an automaton which has strictly more states of null level since $k_1 + \ell + 1 > k_1 + k_2 + 1$ (see Figure 2.9).

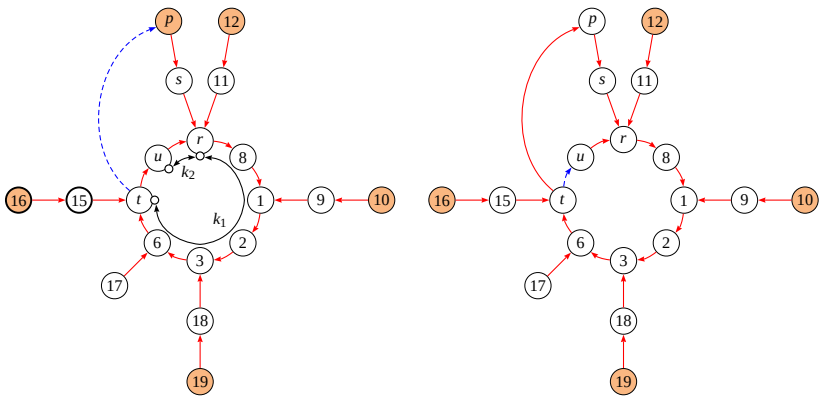

Figure 2.9 The picture on the left illustrates Case 2.3.2. We have $k_2 = 1 < \ell = 2$. After flipping the edges (t,b,p) and (t,a,u), we get the automaton which has a larger number of null level states.

- Case 2.3.3. Let us now assume that $k_2 = \ell$. Let q be the predecessor of r on the cycle and let s be the child of r ascendant of p in the maximal tree rooted at r.
  - Case 2.3.3.1. If the set of edges going out of q is not a bunch. There is a b-letter c such that $q \cdot c = vr$. Let us flip the edges (q,a,r) and (q,c,v). If r belongs to the new red cycle, then the number of null-level states has increased. In the other case, the level of r in the new automaton is at least one and thus the new automaton has a unique maximal tree (see Figure 2.10).

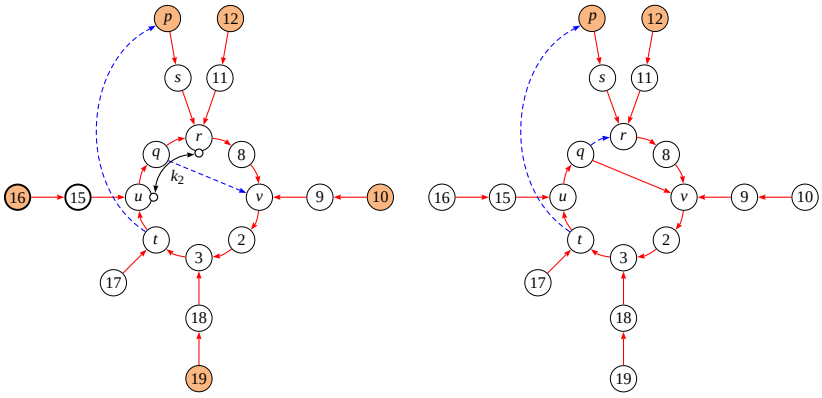

Figure 2.10 The picture on the left illustrates Case 2.3.3.1. We have $k_2 = 2$. The state q is not a bunch. After flipping the edges (q,b,v) and (q,a,r), we get an automaton which has a unique maximal tree.

  - Case 2.3.3.2. If the set of outgoing edges of q and s are bunches, then (q,s) is a stable pair.
  - Case 2.3.3.3. If the set of outgoing edges of q is a bunch and the set of outgoing edges of s is not a bunch, there is a letter ca such that $v = s \cdot c = vr$. If there is an a-path from v to s, we flip the edges (s,a,r) and (s,c,v), creating a new red cycle, which increases the number of states of level zero. If there is no a-path from v to s and the level of v is positive, we flip the edges (s,a,r) and (s,c,v) and get an automaton which has a unique maximal tree. If v has a null level and belongs to a cluster distint from the cluster of r, we flip the edges (s,a,r) and (s,c,v) and also the edges the edges (t,a,u) and (t,b,p). We again get an automaton which has a unique maximal tree.
  - Case 2.3.3.3. We assume that the set of outgoing edges of q is a bunch and the set of outgoing edges of s is not a bunch. If there is a letter ca such that $v = s \cdot c = vr$ belongs to the cycle containing r. Let us denote by k_3 the length of the simple a-path from u to v (see Figure 2.11). Since vr, we

have k_3k_2. Hence $k_3 < \ell$ or $k_3 > \ell$. We flip the edges (s,a,r) and (s,c,v) and proceed as in Case 2.3.1 or 2.3.2. respectively.

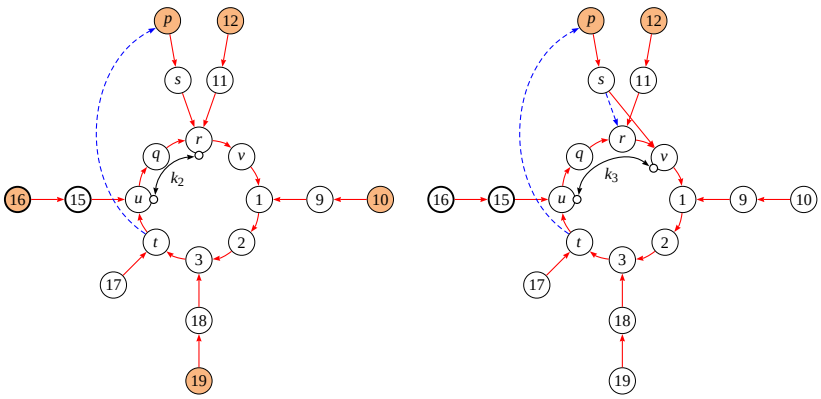

Figure 2.11 The picture on the left illustrates Case 2.3.3.3. We have $k_2 = 2$. After flipping the edges (s,b,v) and (s,a,r), we have $k_3 > k_2 = \ell$. We are back to Case 2.3.1.

The sequence of flips performed to transform the automaton into an equivalent one which has a stable pair has a quadratic time complexity. This makes Trathman's algorithm have a worst case cubic-time complexity. A quadratic-time algorithm for the Road Coloring Problem is presented in Béal and Perrin (2008). The prize to pay for decreasing the time complexity is some more complication in the choice of the flips.

2.5.4 Periodic Road Coloring

In this section, we extend the Road Coloring Theorem to periodic graphs by showing that Trahtman's algorithms provides a minimal-rank coloring. Another proof of this result using semigroup tools, obtained independently, is given in Budzban and Feinsilver (2011).

Recall that the period of an automaton is the gcd of the lengths of the cycles of its graph. If the automaton $\mathscr{A}$ is an n-state complete deterministic irreducible automaton which is not aperiodic, it is not equivalent to a synchronized automaton. Nevertheless, the previous algorithm can be modified as follows for finding an equivalent automaton with the minimal possible rank. It has a quadratic-time complexity.

```
PERIODICFINDCOLORING (automaton 𝒜)
𝒜 ← 𝒜
while (size(ℬ) > 1)
    do UPDATE(ℬ)
       ℬ, (s,t) ← FINDSTABLEPAIR(ℬ)
       lift the coloring up from ℬ to the automaton 𝒜
       if there is a stable pair (s,t)
         then ℬ ← MERGE(ℬ, (s,t))
         else return 𝒜
return 𝒜
```

It may happen that FINDSTABLEPAIR returns an automaton $\mathscr{B}$ which has no stable pair (it is made of a cycle where the set of outgoing edges of any state is a bunch). Lifting up this coloring of to the initial automaton $\mathscr{A}$ leads to a coloring of the initial automaton whose minimal rank is equal to its period.

This result can be stated as the following theorem, which extends the Road Coloring Theorem to the case of periodic graphs.

Theorem 2.5.4 *Any irreducible automaton $\mathscr{A}$ is equivalent to a an automaton whose minimal rank is the period of $\mathscr{A}$.*

Proof Let us assume that $\mathscr{A}$ is equivalent to an automaton $\mathscr{A}'$ which has a stable pair (s,t). Let $\mathscr{B}'$ be the quotient of $\mathscr{A}'$ by the congruence generated by (s,t). Let d be the period of $\mathscr{A}'$ (equal to the period of $\mathscr{A}$) and d' the period of $\mathscr{B}'$. Let us show that $d = d'$.

It is clear that d' divides d (which we denote d'/d). Let ℓ be the length of a path from s to s' in $\mathscr{A}'$, where s' is equivalent to s. Since (s,s') is stable, it is synchronizable. Thus there is a word w such that $s \cdot w = s' \cdot w$. Since the automaton $\mathscr{A}'$ is irreducible, there is a path labeled by some word u from $s \cdot w$ to s. Hence $d/(\ell + |w| + |u|)$ and $d/|(w| + |u|)$, implying d/ℓ. Let $\bar{s}$ be the class of s and z be the label of a cycle around $\bar{s}$ in $\mathscr{B}'$. Then there is a path in $\mathscr{A}'$ labeled by z from s to x, where x is equivalent to x. Thus $d/|z|$. It follows that d/d' and $d = d'$.

Suppose that $\mathscr{B}'$ has rank r. Let us show that $\mathscr{A}'$ also has rank r. Let I be a minimal image of $\mathscr{A}'$ and J be the set of classes of the states of I in $\mathscr{B}'$. Two states of I cannot belong to the same class since I would not be minimal otherwise. As a consequence I has the same cardinal as J. The set J is a minimal image of $\mathscr{B}'$. Indeed, for any word v, the set $J \cdot v$ is the set of classes of $I \cdot v$ which is a minimal image of $\mathscr{A}'$. Hence $|J \cdot v| = |J|$. As a consequence, $\mathscr{B}'$ has rank r.

Let us now assume that $\mathscr{A}$ has no equivalent automaton which has a stable pair. In this case, we know that $\mathscr{A}$ is made of one red cycle where the set of edges going out of any state is a bunch. The rank of this automaton is equal to its period which is the length of the cycle.

Hence the procedure PERIODICFINDCOLORING returns an automaton equivalent to $\mathscr{A}$ whose minimal rank is equal to its period. □

The algorithm allows to find a coloring of minimal rank for irreducible automa-

ton which has the same time and space complexity as the Road Coloring algorithm complexity for aperiodic automata.

2.5.5 Applications

In this section we show how the previous results can be applied to the automaton associated to a finite prefix code.

A *prefix code* on the alphabet A is a set X of words on A such that no element of X is a prefix of another word of X.

A prefix code is *maximal* if it is not contained in another prefix code on the same alphabet. As an equivalent definition, a prefix code X is maximal if for any word u in A^* has a prefix in X or is a prefix of a word of X.

For a deterministic automaton $\mathscr{A}$ and an initial state i, the set $X_{\mathscr{A}}$ of labels of first return paths from i to i is a prefix code. If the automaton is complete, the prefix code is maximal.

Conversely, for any finite prefix code X, there exists a deterministic automaton $\mathscr{A}$ such that $X = X_{\mathscr{A}}$. Moreover, the automaton $\mathscr{A}$ can be supposed to be irreducible. If X is a maximal prefix code, the automaton $\mathscr{A}$ is complete.

The automaton $\mathscr{A}$ can be chosen as follows. The set of states is the set Q of prefixes of the words of X. The transitions are defined for $p \in Q$ and $a \in A$ by $p \cdot a = pa$ if pa is a prefix of a word of X, and by $p \cdot a = \varepsilon$ if $pa \in X$. This automaton, denoted $\mathscr{A}_X$ is a *decoder* of X. Let indeed α be a one-to-one map from a source alphabet B onto X. Let us add an output label to each edge of $\mathscr{A}_X$ in the following way. The output label of (p,a,q) is ε if $q\varepsilon$ and is equal to $\alpha^{-1}(pa)$ if $q = \varepsilon$. With this definition, for any word $x \in X^*$, the output label of the path $i \xrightarrow{x} i$ is the word $\alpha^{-1}(x)$.

Let us show that, as a consequence of the fact that X is finite, the automaton $\mathscr{A}$ is additionally one-cluster with respect to any letter.

Indeed, let a be a letter and let C be the set of states of the form $i \cdot a^j$. For any state q, there exists a path $i \xrightarrow{u} q \xrightarrow{v} i$. We may suppose that i does not occur elsewhere on this path. Thus $uv \in X$. Since X is a finite maximal code, there is an integer j such that $ua^j \in X$. Then $q \cdot a^j = i$ belongs to C. This shows that $\mathscr{A}$ is one-cluster with respect to a.

A maximal prefix code X is *synchronized* if there is a word $x \in X^*$ such that for any word $w \in A^*$, one has $wx \in X^*$. Such a word x is called a *synchronizing word* for X.

Let X be a synchronized prefix code. Let $\mathscr{A}$ be an irreducible deterministic automaton with an initial state i such that $X_{\mathscr{A}} = X$. The automaton $\mathscr{A}$ is synchronized. Indeed, let x be a synchronizing word for X. Let q be a state of $\mathscr{A}$. Since $\mathscr{A}$ is irreducible, there is a path $i \xrightarrow{u} q$ for some $u \in A^*$. Since x is synchronizing for X, we have $ux \in X^*$, and thus $q \cdot x = i$. This shows that x is a synchronizing word for $\mathscr{A}$.

Conversely, let $\mathscr{A}$ be an irreducible complete deterministic automaton. If $\mathscr{A}$ is a synchronized automaton, the prefix code $X_{\mathscr{A}}$ is synchronized. Indeed, let x be a

synchronizing word for A. We may assume that $q \cdot x = i$ for any state q. Then x is a synchronizing word for X.

Proposition 2.5.2 *Let X be a maximal synchronized prefix code with n codewords on an alphabet of size k. The decoder of X has a synchronizing word of length at most $O((\frac{n}{k})^2)$.*

Proof The automaton $\mathscr{A}_X$ is one-cluster. The number N of its states is the number of prefixes of the words of X. Thus $N = (n-1)/(k-1)$ since a complete k-ary tree with n leaves has $(n-1)/(k-1)$ internal nodes. By Proposition 2.3.6, there is a synchronizing word of length $O(N^2)$, whence $O((\frac{n}{k})^2)$. □

Example 2.5.5 Let us consider the following Huffman code $X = (00+01+1)(0+10+11)$ corresponding to a source alphabet $B = \{a,b,c,d,e,f,g,h,i\}$ with a probability distribution $(1/16, 1/16, 1/8, 1/16, 1/16, 1/8, 1/8, 1/8, 1/4)$. The Huffman tree is pictured in the left part of Figure 2.5.5 while the decoder automaton $\mathscr{A}_X$ is given in its right part. The word 010 is a synchronizing word of $\mathscr{A}_X$.

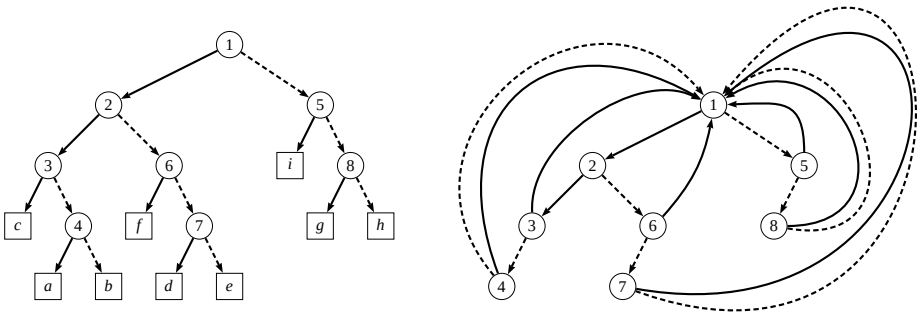

Figure 2.12 A synchronized Huffman code X on the left and its decoder $\mathscr{A}_X$ on the right.

When the lengths of the codewords in X are not relatively prime, the automaton $\mathscr{A}_X$ is never synchronized (see Example of Figure 2.5.5). When the lengths of the codewords in X are relatively prime, the code X is not necessarily synchronized. However, there is always another Huffman code X' corresponding to the same length distribution which is synchronized by a result of Schützenberger Schützenberger (1967). One can even choose X' such that the underlying graph of $\mathscr{A}_X$ and $\mathscr{A}_{X'}$ are the same. This is a particular case of the Road Coloring Theorem. The particular case corresponding to finite prefix codes was proved before in Perrin and Schützenberger (1992). Proposition 2.3.6 guarantees that the Huffman decoder has a synchronizing word of length at most quadratic in the number of nodes of the Huffman tree.

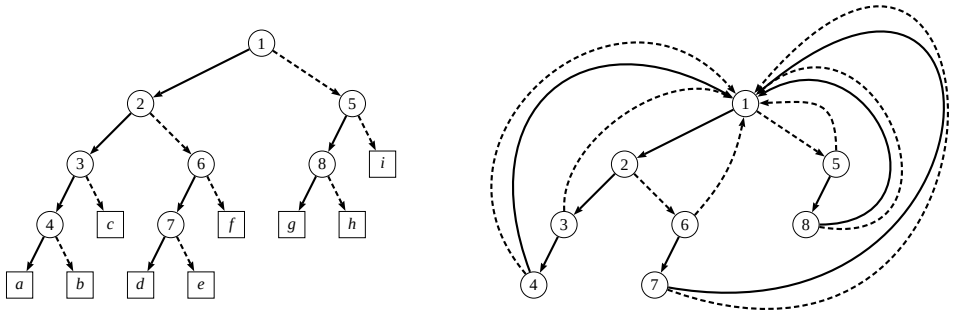

Figure 2.13 A non synchronized Huffman code X on the left and its decoder on the right. The automaton on the right is not synchronized. Indeed, for any word w, the set of states reachable by w is either $\{1,3\}$, $\{2,4\}$, $\{1,5\}$, or $\{1,6\}$.

2.5.6 The Hybrid Černý-Road Coloring Problem

The following problem, called the Hybrid Černý-Road Coloring Problem, was raised by Volkov (2008): what is the minimum length of a synchronizing word for a synchronized coloring of an aperiodic automaton? We conjecture that a synchronized coloring such that the automaton is moreover one-cluster can be obtained which guaranties a minimum length of a synchronizing word of a length at most quadratic.

Bibliography

Adler, R. L., Coppersmith, D., and Hassner, M. 1983. Algorithms for sliding block codes. *IEEE Trans. Inform. Theory*, **IT-29**, 5–22.

Adler, Roy L., Goodwyn, L. Wayne, and Weiss, Benjamin. 1977. Equivalence of topological Markov shifts. *Israel J. Math.*, **27**(1), 48–63.

Béal, Marie-Pierre, and Perrin, Dominique. 2008. A quadratic algorithm for road coloring. *CoRR*, **abs/0803.0726**.

Berstel, Jean, Perrin, Dominique, and Reutenauer, Christophe. 2007. *Codes and Automata*. Cambridge University Press.

Berthé, V., and Rigo, M. (eds). 2010. *Combinatorics, automata and number theory*. Encyclopedia Math. Appl., vol. 135. Cambridge: Cambridge University Press.

Budzban, G., and Feinsilver, Ph. 2011. The generalized road coloring problem and periodic digraphs. *Appl. Algebra Eng. Commun. Comput.*, **22**(1), 21–35.

Carbone, A. 2001. Cycles of relatively prime length and the road coloring problem. *Israel J. Math.*, **123**, 303–316.

Carpi, Arturo. 1988. On synchronizing unambiguous automata. *Theoret. Comput. Sci.*, **60**(3), 285–296.

Carpi, Arturo, and D'Alessandro, Flavio. 2012. Independent sets of words and the synchronization problem. manuscript.

Culik, II, Karel, Karhumäki, Juhani, and Kari, Jarkko. 2002. A note on synchronized automata and road coloring problem. Pages 175–185 of: *Developments in language theory (Vienna, 2001)*. Lecture Notes in Comput. Sci., vol. 2295. Berlin: Springer.

Dubuc, L. 1998. Sur les automates circulaires et la conjecture de Černý. *RAIRO Inform. Théor. Appl.*, **32**(1-3), 21–34.

Eilenberg, S. 1974. *Automata, Languages, and Machines*. Vol. A. Academic Press.

Eilenberg, Samuel. 1976. *Automata, Languages, and Machines. Vol. B*. New York: Academic Press.

Friedman, Joel. 1990. On the road coloring problem. *Proc. Amer. Math. Soc.*, **110**(4), 1133–1135.

Kari, Jarkko. 2001. A counter example to a conjecture concerning synchronizing words in finite automata. *EATCS Bulletin*, **73**, 146.

Kari, Jarkko. 2003. Synchronizing finite automata on Eulerian digraphs. *Theoret. Comput. Sci.*, **295**(1-3), 223–232.

Lind, D., and Marcus, B. 1995. *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press.

Lothaire, M. 1983. *Combinatorics on Words*. Encyclopedia of Mathematics and Its Applications, vol. 17. Addison-Wesley.

O'Brien, G. L. 1981. The road-colouring problem. *Israel J. Math.*, **39**(1-2), 145–154.

Perrin, Dominique, and Schützenberger, Marcel-Paul. 1992. Synchronizing prefix codes and automata and the road coloring problem. Pages 295–318 of: *Symbolic dynamics and its applications*. Contemp. Math., vol. 135. Amer. Math. Soc.

Pin, Jean-Eric. 1978. *Le problème de la synchronisation et la conjecture de Černý*. Thèse de 3ème cycle, Université Paris VI.

Sakarovitch, Jacques. 2003. *Éléments de théorie des automates*. Éditions Vuibert.

Schützenberger, Marcel-Paul. 1967. On synchronizing prefix codes. *Inform. and Control*, **11**, 396–401.

Shin, Sujin, and Yoo, Jisang. 2010. A note on the rank of semigroups. *Semigroup Forum*, **81**(2), 335–343.

Trahtman, A. N. 2007. Synchronization of Some DFA. Pages 234–243 of: *TAMC*.

Trahtman, A. N. 2008a. *A Subquadratic Algorithm for Road Coloring*. arXiv:0801.2838v1 [cs.DM].

Trahtman, A. N. 2008b. Synchronizing road coloring. Pages 43–53 of: *Fifth IFIP International Conference on Theoretical Computer Science—TCS 2008*. IFIP Int. Fed. Inf. Process., vol. 273. New York: Springer.

Trahtman, A. N. 2009. The road coloring problem. *Israel J. Math.*, **172**, 51–60.

Trahtman, A. N. 2011. Modifying the Upper Bound on the Length of Minimal Synchronizing Word. Pages 173–180 of: *FCT*.

Černý, J. 1964. Poznámka k homogénnym experimentom s konečnými automatmi. *Mat. fyz. čas.*, **SAV 14**, 208–215.

Volkov, Mikhail. 2008. Synchronizing strongly connected digraphs. Around theČerný Conjecture, International Workshop, University of Wroclaw.

Volkov, Mikhail V. 2007. Synchronizing Automata Preserving a Chain of Partial Orders. Pages 27–37 of: *CIAA*.

Notation index

General index