Unambiguous Finite Automata

Home assignment to hand in before or on February 15, 2018.

February

			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28				

Electronic versions (PDF only) can be sent by email to ⟨sylvain.schmitz@lsv.fr⟩, paper versions should be handed in on the 15th or put in my mailbox at LSV, ENS Cachan. **No delays**. The numbers in the margins next to exercises are indications of time and difficulty, not necessarily of the points you might earn answering them.

In a nondeterministic finite automaton, a word w might have several different accepting runs. The number of these accepting runs is called the *ambiguity degree* of w. It turns out that automata with restricted ambiguity degrees have good algorithmic properties. It is also a domain of formal language theory where a bit of linear algebra comes in handy.

1 Ambiguity in Automata

Formally, let $\mathcal{A} = \langle Q, \Sigma, \delta, I, F\rangle$ be a nondeterministic finite automaton. To simplify matters, we only work in this assignment with 'real-time' automata where $\delta \subseteq Q \times \Sigma \times Q$, i.e. there are no ε-transitions.

Recall that a *run* on a word $w = a_1 \cdots a_n \in \Sigma^*$ in $\mathcal{A}$ is a finite sequence of steps $q_0 \xrightarrow{a_1} q_1 \cdots q_{n-1} \xrightarrow{a_\ell} q_\ell$ where $(q_i, a_{i+1}, q_{i+1}) \in \delta$ for all $0 \leq i < \ell$. Such a run is *accepting* if $q_0 \in I$ and $q_\ell \in F$. For a word w, let $\deg_{\mathcal{A}}(w)$ denote the number of accepting runs on w in $\mathcal{A}$; we call this the *ambiguity degree* of w.

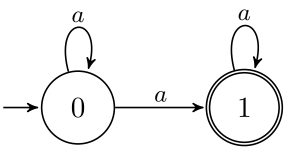
Exercise 1 (Ambiguity Degree)**.** The *ambiguity degree* $\deg_{\mathcal{A}}$ of an automaton $\mathcal{A}$ is the function from $\mathbb{N}$ to $\mathbb{N}$ mapping ℓ to $\max_{w \in \Sigma^{\leq \ell}} \deg(w)$ where $\Sigma^{\leq \ell}$ denotes the set of words of length at most ℓ over Σ. An automaton $\mathcal{A}$ is

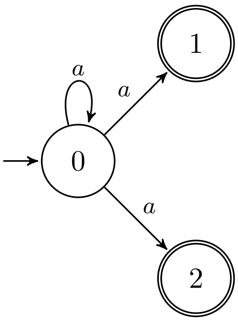
unambiguous if $\deg_{\mathcal{A}}(\ell) \leq 1$ for all ℓ,

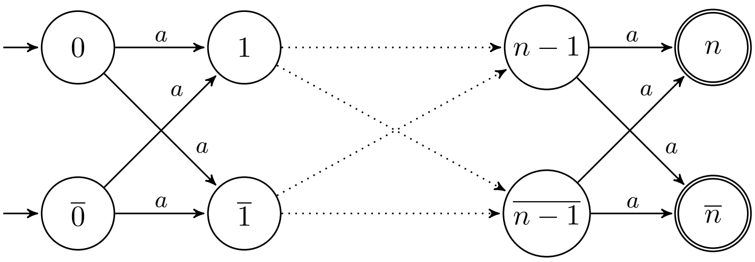
finitely ambiguous if there exists $k \in \mathbb{N}$ such that $\deg_{\mathcal{A}}(\ell) \leq k$ for all ℓ, and

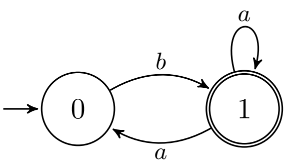
polynomially ambiguous if there exists p a polynomial function such that $\deg_{\mathcal{A}}(\ell) \leq p(\ell)$ for all ℓ.

[1] 1. What is the ambiguity degree of the following automata? Is the automaton unambiguous, finitely ambiguous, polynomially ambiguous?

(a) 

(b) 

(c) 

(d) 

[1] 2. What is the maximal ambiguity degree one can obtain with an n-states automaton? Provide a family of automata that reaches this bound.

Exercise 2 (Weighted Automata)**.** Weighted automata can be defined in general over any semiring, but for the purposes of this homework assignment, we will only consider the case of the semiring $(\mathbb{N}, +, \cdot, 0, 1)$. A *weighted finite automaton* is then a tuple $\mathcal{W} = \langle Q, \Sigma, \mathbb{N}, \delta, I, F\rangle$ where Q is a finite set of states, Σ a finite alphabet, $\delta \subseteq Q \times \Sigma \times \mathbb{N} \times Q$ is a finite transition relation, and $I, F \subseteq Q$ are sets of initial and final states.

A *run* over a word $w = a_1 \cdots a_\ell \in \Sigma^*$ in $\mathcal{W}$ is a finite sequence of steps $q_0 \xrightarrow{a_1,n_1} q_1 \xrightarrow{a_2,n_2} q_2 \cdots q_{\ell-1} \xrightarrow{a_\ell,n_\ell} q_n$ where $(q_i, a_{i+1}, n_{i+1}, q_{i+1}) \in \delta$ for all $0 \leq i < \ell$, and associates the *weight* $n_1 \cdot n_2 \cdots n_\ell$ to w. Such a run is *accepting* if $q_0 \in I$ and $q_\ell \in F$. The *weight* $\text{weight}_{\mathcal{W}}(w)$ of a word w in $\mathcal{W}$ is defined as the sum of the weights of the accepting runs over w in $\mathcal{W}$.

[1] 1. Show that, given a nondeterministic finite automaton $\mathcal{A}$, one can construct a weighted finite automaton of the same size such that $\deg_{\mathcal{A}}(w) = \mathrm{weight}_{\mathcal{W}}(w)$ for all $w \in \Sigma^*$.

[1] 2. Show that, given a weighted finite automaton $\mathcal{W}$, one can construct an equivalent normalised weighted finite automaton $\mathcal{W}'$ of the same size. Being *equivalent* means here that $\mathrm{weight}_{\mathcal{W}}(w) = \mathrm{weight}_{\mathcal{W}'}(w)$ for all $w \in \Sigma^*$; being *normalised* means here that for all $q, q' \in Q$ and $a \in \Sigma$, δ' contains exactly one transition (q, a, n, q') for some $n \in \mathbb{N}$.

3. Normalised weighted automata can also be understood in terms of matrices. We view I and F as characteristic vectors in $\mathbb{N}^Q$ with $I(q) = 1$ if $q \in I$ and $I(q) = 0$ otherwise, and similarly for F. We also define a homomorphism η from Σ^* to $\mathbb{N}^{Q \times Q}$ by letting $\eta(a)$ be the matrix with $\eta(a)(q, q') \stackrel{\text{def}}{=} n$ whenever $(q, a, n, q') \in \delta$ (since $\mathcal{W}$ is normalised there is a unique such n); then $\eta(\varepsilon) \stackrel{\text{def}}{=} \mathrm{Id}_Q$ the identity matrix and $\eta(u \cdot v) \stackrel{\text{def}}{=} \eta(u) \cdot \eta(v)$ as usual for a homomorphism.

[1] Show that $\mathrm{weight}_{\mathcal{W}}(w) = {}^t I \cdot \eta(w) \cdot F$ for all $w \in \Sigma^*$.

2 Unambiguous Automata

Unambiguous finite automata form an intermediate class between deterministic and nondeterministic ones. They strike an interesting balance between succinctness, which we examine in Exercise 3, and tractable algorithms, which we will investigate in Section 3.

Exercise 3 (Succinctness)**.** We compare in this exercise the succinctness of nondeterministic, unambiguous, and deterministic finite automata: how many states are needed in order to define a given regular language? The techniques employed in this exercise are rooted in *communication complexity*, which is a framework for proving lower bounds.

[1] 1. Provide a family of unambiguous finite automata $\mathcal{A}_n$ with $n + 1$ states such that no deterministic finite automaton for $L(\mathcal{A}_n)$ has fewer than 2^n states. *Hint: Use a family described in class; in that case it is not required to prove the lower bound on the number of states of the corresponding deterministic automata.*

2. Let $L \subseteq \Sigma^*$ be a language over Σ and $\mathfrak{C}(L) \stackrel{\text{def}}{=} \{(u, v) \in \Sigma^* \times \Sigma^* \mid u \cdot v \in L\}$ the set of pairs that build up words of L.

A *rectangular decomposition* of L is a finite collection $D = (P_k, S_k)_{1 \le k \le C}$ of pairs of languages $P_k, S_k \subseteq \Sigma^*$ such that $\mathfrak{C}(L) = \bigcup_{1 \le k \le C} P_k \times S_k$. The number of pairs $C \in \mathbb{N}$ is called the *complexity* of the decomposition D. A rectangular decomposition is *disjoint* if for all $k \neq k'$, $(P_k \times S_k) \cap (P_{k'} \times S_{k'}) = \emptyset$.

[2] Show that, if $\mathcal{A}$ is an unambiguous finite automaton with n states, then there exists a disjoint rectangular decomposition of $L(\mathcal{A})$ with complexity at most n.

3. Let $L \subseteq \Sigma^*$ be a language over Σ^* with a disjoint rectangular decomposition $D = (P_k, S_k)_{1 \le k \le C}$ of complexity C.

   Let $\vec{u} = (u_i)_{1 \le i \le m}$ and $\vec{v} = (v_j)_{1 \le j \le m}$ be two collections of m finite words in Σ^*. Consider the matrix $M^L_{\vec{u},\vec{v}}$ in $\mathbb{Z}^{m \times m}$ where entry (i, j) holds 1 if $u_i v_j \in L$ and 0 otherwise.

[2] Show that this matrix has rank $\mathrm{rk}(M^L_{\vec{u},\vec{v}}) \le C$. *Hint: Use the fact that matrix rank is subadditive.*

4. Consider the family of matrices M_n in $\mathbb{Z}^{2^n \times 2^n}$ defined inductively for all $n \in \mathbb{N}$ by

$$M_0 \stackrel{\text{def}}{=} \begin{bmatrix} 0 \end{bmatrix} \qquad\qquad M_{n+1} \stackrel{\text{def}}{=} \begin{bmatrix} M_n & M_n \\ 1 & M_n \end{bmatrix}$$

   where '1' denotes the matrix with all entries equal to 1.

[1] Show that $\mathrm{rk}(M_n) = 2^n - 1$. *Hint: You might want to consider the matrices N_n defined by*

$$N_0 \stackrel{\text{def}}{=} \begin{bmatrix} 1 \end{bmatrix} \qquad\qquad N_{n+1} \stackrel{\text{def}}{=} \begin{bmatrix} N_n & N_n \\ 0 & N_n \end{bmatrix}$$

   where '0' denotes the matrix with all entries equal to 0.

5. Recall the family of nondeterministic finite automata $\mathcal{A}_n \stackrel{\text{def}}{=} \langle Q_n, \{a, b\}, \delta_n, Q_n, Q_n \rangle$ seen in class, where

$$Q_n \stackrel{\text{def}}{=} \{0, \ldots, n-1\} \text{ and } \delta_n \stackrel{\text{def}}{=} \{(i, a, i + 1 \bmod n) \mid i < n\} \cup \{(i, b, i) \mid 0 < i < n\}$$

   and all the states are both initial and final. This is depicted in Figure 1 on the next page.

   For a subset $K \subseteq Q_n$, let $w_K \stackrel{\text{def}}{=} w_{K,0} \cdots w_{K,n-1}$ be the word defined by $w_{K,i} \stackrel{\text{def}}{=} a^{n-i} b a^i$ if $i \in K$ and $w_{K,i} \stackrel{\text{def}}{=} a^n$ otherwise.

[2] (a) Show that, for K, K' two subsets of Q_n, $w_K w_{K'} \in L(\mathcal{A}_n)$ if and only if $K \cup K' \subsetneq Q_n$.

[2] (b) Provide for each n two families $\vec{u} = (u_m)_{0 \le m < 2^n}$ and $\vec{v} = (v_m)_{0 \le m < 2^n}$ of 2^n words such that $\mathrm{rk}(M^{L(\mathcal{A}_n)}_{\vec{u},\vec{v}}) = 2^n - 1$.

[0] (c) Conclude that any unambiguous finite automaton for $L(\mathcal{A}_n)$ has at least $2^n - 1$ states.

3 Decision Problems

Decision problems on unambiguous finite automata are typically simpler than on nondeterministic ones: universality, inclusion, and equivalence are in P instead of PSPACE in the general case. We are going to show this in the case of universality in Exercise 5, but first we show that we can efficiently decide whether a finite automaton is unambiguous.

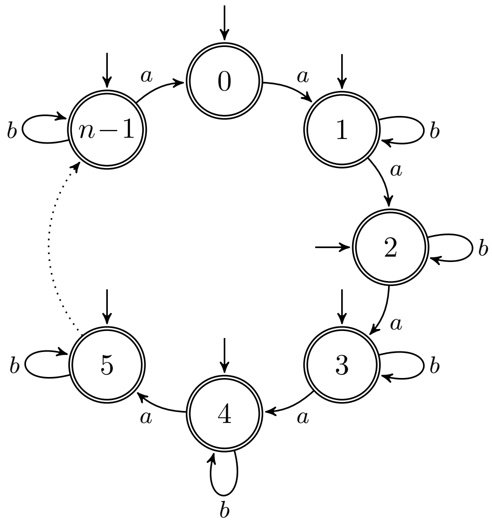


Figure 1: The automaton $\mathcal{A}_n$.

Exercise 4 (Ambiguity)**.** We consider the following decision problem:

instance a nondeterministic finite automaton $\mathcal{A}$

question is $\mathcal{A}$ unambiguous?

Since ambiguity is a semantic property of the runs of $\mathcal{A}$, the fact that this is decidable at all is not immediate.

[3] Show that this can be solved in deterministic time $O(|\mathcal{A}|^2)$.

Exercise 5 (Universality)**.** We consider the following decision problem:

instance an unambiguous finite automaton $\mathcal{A}$ over an alphabet Σ

question is $\mathcal{A}$ universal, i.e. does $L(\mathcal{A}) = \Sigma^*$?

Recall from your complexity course that the same problem on nondeterministic finite automata is PSPACE-complete. But restricting our attention to unambiguous automata allows to derive an algorithm in P, as we are going to see now. Interestingly, we are going to leverage the lower bounds of Exercise 3 to obtain good algorithmic upper bounds.

[2] 1. Show that, if $\mathcal{A}$ has n states and is not universal, then there is a word $w \notin L(\mathcal{A})$ of length at most n. *Hint: Apply Exercise 3 Question 3.*

2. We consider the weighted automaton $\mathcal{W}$ associated with $\mathcal{A}$ and its matrix representation as in Exercise 2. Let

$$E(n) \stackrel{\text{def}}{=} \sum_{w \in \Sigma^n} \eta(w) \qquad\qquad E'(n) \stackrel{\text{def}}{=} \sum_{w \in \Sigma^{\leq n}} \eta(w)$$

[2] (a) Show that $E(n)$ and $E'(n)$ can be computed by induction on n in deterministic time polynomial in n and $|\mathcal{A}|$. Deduce that $\deg_{\mathcal{A}}(n)$ can be computed in deterministic time polynomial in n and $|\mathcal{A}|$.

[1] (b) Deduce that the universality problem for unambiguous finite automata can be solved in P.