Generalized One-Unambiguity

Pascal Caron[1], Yo-Sub Han[2,⋆], and Ludovic Mignot[1]

[1] LITIS, Université de Rouen, 76801 Saint-Étienne du Rouvray Cedex, France
{pascal.caron,ludovic.mignot}@univ-rouen.fr
[2] Dept. of Computer Science, Yonsei University, Seoul 120-749, Republic of Korea
emmous@cs.yonsei.ac.kr

Abstract. Brüggemann-Klein and Wood have introduced a new family of regular languages, the *one-unambiguous regular languages*, a very important notion in XML DTDs. A regular language L is one-unambiguous if and only if there exists a regular expression E over the operators of sum, catenation and Kleene star such that $L(E) = L$ and the position automaton of E is deterministic. It implies that for a one-unambiguous expression, there exists an equivalent linear-size deterministic recognizer. In this paper, we extend the notion of one-unambiguity to weak one-unambiguity over regular expressions using the complement operator $\neg$. We show that a DFA with at most $(n+2)$ states can be computed from a weakly one-unambiguous expression and that it is decidable whether or not a given DFA recognizes a weakly one-unambiguous language.

1 Introduction

Regular expressions are basic tools for numerous domains such as pattern-matching or electronic document specification. A regular expression is a compact representation for a set of words. A recurrent question is the membership problem which is to decide whether or not a word belongs to the language denoted by an expression. This problem can be solved by computing a recognizer called automaton from a regular expression. One of the best-known automata construction is the position construction [6,9]. If a regular expression E has n occurrences of symbols, then the corresponding position automaton, which is not necessarily deterministic, has exactly $(n+1)$ states. There always exists a deterministic recognizer equivalent to the position automaton but its size can be exponentially larger.

Brüggemann-Klein and Wood [3] introduced a subfamily of regular languages called *one-unambiguous regular languages*: A regular language is one-unambiguous if and only if there exists an equivalent regular expression the position automaton of which is deterministic. XML DTDs are specified by extended context-free grammars in which the right-hand side of the productions (content models) are one-unambiguous [2]. These content models allow efficient compiling and testing. It turned out that one-unambiguity is very important in XML DTDs. Therefore it

⋆ Han was supported by the Basic Science Research Program through NRF funded by MEST (2010-0009168).

G. Mauri and A. Leporati (Eds.): DLT 2011, LNCS 6795, pp. 129–140, 2011.

is important to investigate the properties of one-unambiguous regular languages. One-unambiguous regular languages are strictly included into regular languages. We want to expand the size of the family of expressions for which there exists a method to compute a deterministic linear-size recognizer. In this purpose, we study generalized expressions since they often enable us to write more compact regular expressions. For instance, Gelade and Neven [4,5] have shown that the size of the smallest regular expression without intersection operators for the intersection of two languages denoted by simple regular expressions can be exponential.

In Section 2, we define some basic notions. We demonstrate in Section 3 that one-unambiguous regular languages are closed neither under intersection nor under complement. Note that Brüggemann-Klein and Wood [3] have shown that one-unambiguous regular languages are closed neither under union, catenation nor under Kleene star. A new family of regular languages closed under complement and containing the one-unambiguous regular languages is then introduced in Section 4. We define weakly one-unambiguous regular expressions and show that a linear-size recognizer can be computed from such an expression. We then structurally characterize the minimal DFA of weakly one-unambiguous regular languages.

2 Preliminaries

A *deterministic finite automaton* (DFA) $A = (\Sigma, Q, i, F, \delta)$ is a 5-tuple defined by Σ a finite set of symbols called the *alphabet*, Q a finite set of *states*, $i \in Q$ the *initial state*, $F \subset Q$ the set of *final states* and $\delta : Q \times \Sigma \to Q$ the *transition function*. The function δ is equivalent to the set defined by: $(q, a, q') \in \delta$ if and only if $q' = \delta(q, a)$. The function δ is extended to $2^Q \times \Sigma^* \to 2^Q$ as follows: $\forall P \subset Q, \forall a \in \Sigma, \forall w \in \Sigma^*$, $\delta(P, \varepsilon) = P$, $\delta(P, a) = \bigcup_{p \in P} \delta(p, a)$ and $\delta(P, a \cdot w) = \delta(\delta(P, a), w)$. The automaton A *recognizes* the language $L(A) = \{w \in \Sigma^* \mid \delta(i, w) \in F\}$. The *set of recognizable languages* defined by $\{L \mid \exists A \text{ a DFA}, L(A) = L\}$ is written $\mathrm{Rec}(\Sigma^*)$. A DFA is *complete* if $\forall (q, a) \in Q \times \Sigma$, $\mathrm{Card}(\delta(q, a)) = 1$. For every DFA A, there exists an equivalent complete DFA A' such that $L(A') = L(A)$ [13]. The *left language* (resp. *right language*) of a state q of A is the set of words $L_q^{\leftarrow}(A) = \{w \in \Sigma^* \mid \delta(i, w) = q\}$ (resp. $L_q^{\rightarrow}(A) = \{w \in \Sigma^* \mid \delta(q, w) \in F\}$). A state q in Q is *accessible* (resp. *coaccessible*) if and only if $L_q^{\leftarrow}(A) \neq \emptyset$ (resp. $L_q^{\rightarrow}(A) \neq \emptyset$). We say that q is a *sink* state if $L_q^{\rightarrow}(A) = \emptyset$. The automaton A is *trim* if all the states of A are accessible and coaccessible. Two states q and q' are *equivalent* with respect to the Myhill-Nerode congruence [11,12] if and only if $L_q^{\rightarrow}(A) = L_{q'}^{\rightarrow}(A)$. We assume that A has a single sink state since all sink states are equivalent. A DFA A is *minimal* if there exists no DFA recognizing $L(A)$ with less states than A. Notice that for a given language L, all minimal DFAs recognizing $L(A)$ are isomorphic. The minimal DFA of L is computable from any trim DFA recognizing L by merging equivalent states (For computation of the minimal DFA, see [1,7,10]). For a state q in a DFA A, we denote by $[q]$ the equivalent class of q which is a state of the minimal DFA of A. A *regular expression* E over an alphabet Σ is inductively defined by

$E = a$, $E = \emptyset$, $E = \varepsilon$, $E = (F + G)$, $E = (F \cdot G)$, $E = (F^*)$, $E = \neg(F)$ with F and G two regular expressions and a a symbol of Σ. The *alphabetical width* $|E|$ of E is the number of occurrences of symbols of E. Let L_1 and L_2 be two languages. We define $L_1 \cdot L_2 = \{w = w_1 \cdot w_2 \mid w_1 \in L_1 \wedge w_2 \in L_2\}$, $(L_1)^* = \{w = w_1 \cdots w_k \mid k \in \mathbb{N} \wedge \forall j \in \{1, \ldots, k\}, w_j \in L_1\}$ and $\neg L_1 = \{w \in \Sigma^* \mid w \notin L_1\}$. The *language denoted* by a regular expression E over an alphabet Σ is inductively computed by $L(a) = \{a\}$, $L(\emptyset) = \emptyset$, $L(\varepsilon) = \{\varepsilon\}$, $L(F + G) = L(F) \cup L(G)$, $L(F \cdot G) = L(F) \cdot L(G)$, $L(F^*) = (L(F))^*$, and $L(\neg F) = \neg(L(F))$, with F and G two regular expressions and a a symbol in Σ. The regular expression E is said to be a *simple expression* if it only contains sum, catenation and Kleene star operators. As a consequence, the *set of regular languages*, $\mathrm{Rat}(\Sigma^*) = \{L \mid$ there exists a simple expression E such that $L(E) = L\}$, is the smallest family containing $\emptyset$ and $\{a\}$ for all symbol a in Σ and which is closed under catenation, Kleene star and sum. Kleene's Theorem [8] asserts that $\mathrm{Rat}(\Sigma^*) = \mathrm{Rec}(\Sigma^*)$. Moreover, given a complete DFA A, a DFA ${}_cA$ such that $L({}_cA) = \neg L(A)$ can be computed by switching non-final the final states and *vice versa*. The automaton ${}_cA$ is *the complement of* A. Therefore, the set of regular languages is closed under complement.

A subset O of states of a DFA A is an *orbit* if and only if, for every pair of states (q, q') in O^2, there exists a string w in Σ^+ such that $q' = \delta(q, w)$ and if for every state q in $Q \setminus O$, either there exists no word w in Σ^* such that $\delta(q, w) \in O$, or there exists no word w in Σ^* such that $\delta(O, w) \cap \{q\} \neq \emptyset$. Notice that a strongly connected component is not necessarily an orbit, since a singleton without a loop is a strongly connected component but not an orbit. The *gates* of an orbit O are the states belonging to the set gates(O) defined by: $\{o \in O \mid \exists a \in \Sigma, q \in Q \setminus (O \cup \mathrm{sink}(A)), \delta(o, a) = q\} \cup (O \cap F)$. Let j be a state of O. The *automaton of the orbit* O *for* j is the automaton $A_{O,j} = (\Sigma, O, j, \mathrm{gates}(O), \delta_O)$ where $\delta_O = \delta \cap (O \times \Sigma \times O)$. The language recognized by the automaton $A_{O,j}$ is called *the orbit language of* O *for* j. The orbit O is *transverse* if and only if the two following conditions are checked:

(1) $\forall q, q' \in \mathrm{gates}(O)$, $\forall a \in \Sigma$: $\delta(q, a) \notin O \setminus \mathrm{sink}(A) \Rightarrow \delta(q, a) = \delta(q', a)$.
(2) $\forall q, q' \in \mathrm{gates}(O)$, q and q' are both either final or non-final.

A symbol a in Σ is A-*consistent* if and only if there exists a state $\mathrm{f}(a)$ such that every final state of A has a transition to $\mathrm{f}(a)$ labelled by a. A set $\Sigma' \subset \Sigma$ is A-*consistent* if and only if every symbol a of Σ' is A-consistent. If Σ' is an A-consistent set of symbols, the Σ'-*cut of* A, denoted by $A_{\Sigma'}$, is obtained by deleting every transition labelled by a symbol a in Σ' and starting from a final state. The *position automaton* or *Glushkov automaton* of a simple regular expression E is an $(n+1)$-state automaton that recognizes $L(E)$ (see [6,9] for construction rules). A simple expression E is *one-unambiguous* if and only if its Glushkov automaton is deterministic [3]. A regular language is *one-unambiguous* if and only if there exists a one-unambiguous regular expression denoting it. For details on one-unambiguous regular languages and properties, refer to Brüggemann-Klein and Wood [3].

3 One-Unambiguous Languages are Not Closed under Boolean Operators

Regular languages are closed for basic operators such as catenation, union, Kleene star, intersection or complement. On the other hand, for one-unambiguous regular languages, Brüggemann-Klein and Wood [3] noticed that they are closed neither under union, catenation nor under Kleene star. We show that one-unambiguous regular languages are closed neither under intersection (Example 1) nor under complement (Example 2).

Example 1. Let $E_1 = (b(c + \varepsilon))^*$ and $E_2 = b(c(b + \varepsilon))^* + c(b + \varepsilon)(c(b + \varepsilon))^*$ be two one-unambiguous regular expressions. Their minimal DFAs A_1 and A_2 are given in Figure 1. The minimal DFA A_3 of the language $L(A_1) \cap L(A_2)$ is given Figure 2. We can see that $L(A_3)$ is not one-unambiguous, since there does not exist nonempty A-consistent subset of Σ (see Theorem **F** [3]).

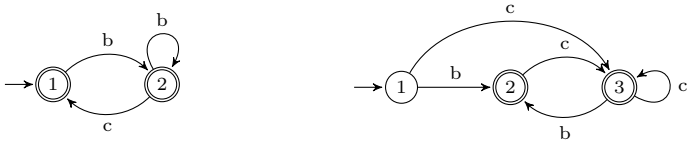

Fig. 1. The Automata A_1 and A_2

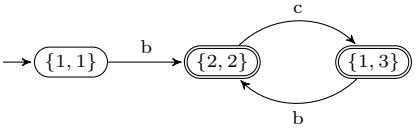

Fig. 2. The Automaton A_3

Example 2. Let A_1 and A_2 be two minimal DFAs given in Figure 3. The automaton A_2 recognizes the language $\neg(L(A_1))$. The language $L(A_1)$ is one-unambiguous, while $L(A_2)$ is not, since there does not exist nonempty A-consistent subset of Σ (see Theorem **F** [3]).

Fig. 3. The Automata A_1 and A_2

From the results (union, catenation and Kleene star) of Brüggemann-Klein and Wood [3] and Examples 1 and 2, we establish the following statement:

Proposition 1. *One-unambiguous regular languages are closed neither under union, catenation, Kleene star, intersection nor under complement.*

We have seen that one-unambiguous languages are not closed under complement. As a consequence, the characterization theorem for one-unambiguous languages has to be rewritten in order to deal with the complement operator. It leads to a new notion, the *weak one-unambiguity*.

4 The Weak One-Unambiguity

We exhibit a new family of languages closed under complement and containing the one-unambiguous languages family. The Kleene-like theorem of Brüggemann-Klein and Wood (Theorem D [3]) is transformed in order to deal with complement closure. This new family of regular languages is called the *weakly one-unambiguous languages*. We first define two particular subsets of symbols in Σ. Let L be a language over an alphabet Σ. The sets $\mathrm{First}(L)$ and $\mathrm{FollowLast}(L)$ are defined as follows: $\mathrm{First}(L) = \{a \in \Sigma \mid \exists w \in \Sigma^*, aw \in L\}$, $\mathrm{FollowLast}(L) = \{a \in \Sigma \mid \exists w, w' \in \Sigma^*, w \neq \varepsilon, w \in L \wedge waw' \in L\}$.

Definition 1. *The family of* weakly one-unambiguous expressions *over an alphabet Σ is the family $\mathcal{E}$ inductively defined as follows:*
(1) *$\emptyset$, ε, and a are in $\mathcal{E}$, for each symbol a in Σ,*
(2) *if $E_1, E_2 \in \mathcal{E}$ and $\mathrm{First}(L(E_1)) \cap \mathrm{First}(L(E_2)) = \emptyset$, then $E_1 + E_2 \in \mathcal{E}$,*
(3) *if $E_1, E_2 \in \mathcal{E}$, $\mathrm{FollowLast}(L(E_1)) \cap \mathrm{First}(L(E_2)) = \emptyset$ and $(\varepsilon \notin L(E_1) \vee \mathrm{First}(L(E_1)) \cap \mathrm{First}(L(E_2)) = \emptyset)$, then $E_1 \cdot E_2 \in \mathcal{E}$,*
(4) *if $E_1 \in \mathcal{E}$ and $\mathrm{FollowLast}(L(E_1)) \cap \mathrm{First}(L(E_1)) = \emptyset$, then $E_1^* \in \mathcal{E}$,*
(5) *if $E_1 \in \mathcal{E}$, then $\neg(E_1) \in \mathcal{E}$.*

Definition 2. *A language L is* weakly one-unambiguous *if and only if there exists a weakly one-unambiguous expression denoting L.*

The two following definitions are used to characterize the minimal DFA of a weakly one-unambiguous language.

Definition 3. *Let A be a complete DFA and ${}_cA$ be its complement. The automaton A* satisfies the transverse property *if and only if for all orbit O in A, one of the two following propositions is satisfied:*
(1) *O is transverse in A and $\forall i \in O$, $A_{O,i}$ satisfies the consistence property;*
(2) *O is transverse in ${}_cA$ and $\forall i \in O$, $({}_cA)_{O,i}$ satisfies the consistence property.*

Definition 4. *Let A be a complete DFA and ${}_cA$ be its complement. The automaton A* satisfies the consistence property *if and only if for all orbit O in A, one of the two following propositions is satisfied:*
(1) *there exists a nonempty subset Σ' of Σ such that Σ' is A-consistent and $A_{\Sigma'}$ satisfies the transverse property;*
(2) *there exists a nonempty subset Σ' of Σ such that Σ' is ${}_cA$-consistent and $({}_cA)_{\Sigma'}$ satisfies the transverse property.*

Even if the two previous definitions seem to be dependent, each one checks the other on a smaller automaton (an orbit automaton or a Σ'-cut). Since there are no more orbits at the end, this dependency stops eventually. Note that both properties are satisfied by an acyclic automaton. The transverse property characterizes the minimal DFA of weakly one-unambiguous languages.

Theorem 1. *A regular language is weakly one-unambiguous if and only if its complete minimal DFA satisfies the transverse property. Furthermore, if E is an n symbol weakly one-unambiguous expression, the minimal DFA of $L(E)$ has at most $n + 2$ states.*

In order to prove this theorem, we first show that minimization preserves the transverse property (Section 4.1), and then we demonstrate the necessity and the sufficiency of the transverse property to characterize the weak one-unambiguity (Section 4.2 and Section 4.3).

4.1 Minimization Preserves the Transverse Property

Lemma 1. *Let A be a DFA and A' be its minimal DFA. For all orbit O' in A', there exists an orbit O in A such that $\forall p \in O$, $[p] \in O' \wedge (p \in \text{gates}(O) \Leftrightarrow [p] \in \text{gates}(O'))$. The orbit O is said to be a* lift *of O'.*

Proof. Let $A = (\Sigma, Q, i, F, \delta)$ and $A' = (\Sigma, Q', i', F', \delta')$. Let O' be an orbit in A' and $[p]$ be a state in O'. There exists p in A such that $p \in [p]$. **(1)** For all $[q]$ in O', there exist two words w_1 and w_2 such that $\delta'([p], w_1) = [q]$ and $\delta'([q], w_2) = [p]$. As a consequence, there exists a state $q \in [q]$ such that $\delta(p, w_1) = q$ and $\delta(q, w_2) = p' \in [p]$. This does not imply that p and q are in a same orbit O. However, by repeating these words and since A is a DFA, an orbit O is accessible from p such that for all p' in O, $[p'] \in O'$. **(2a)** Let $[p]$ be in gates(O') and p be a state in O such that $p \in [p]$. If $[p]$ is final, so does p. If there exists a symbol a in Σ such that $\delta'([p], a) \notin O' \cup \text{sink}(A')$, it holds $\delta(p, a) \notin O \cup \text{sink}(A)$ (otherwise contradiction with $p \in [p]$). As a consequence, p is in gates(O). **(2b)** Let O be one of the last accessible orbist in A satisfying **(1)** (considering the partial order of accessible orbits). Let p be a gate of O. Suppose that $[p]$ is not a gate. Consequently, p has to be a non-final state. As a consequence, there exists a transition from p going out of O (but not in the sink state) by a letter a such that $\delta'([p], a) \in O' \cup \text{sink}(A')$. This implies either $\delta(p, a) = \text{sink}(A)$ or there is another orbit O_2 accessible from $\delta(p, a)$ satisfying **(1)**. Contradiction in both cases. ∎

Lemma 2. *Let A' be a minimal DFA and O' be a transverse orbit in A'. For every j' in O', $A'_{O',j'}$ is minimal.*

Proof. Let j' be a state in O'. We consider the automata $A' = (\Sigma, Q', i', F', \delta')$ and $A'_{O',j'} = (\Sigma, O', j', \text{gates}(O'), \delta'')$. Let p'_1 and p'_2 be two equivalent states in $(A'_{O',j'})$. For all w in Σ^*, $\delta''(p'_1, w)$ and $\delta''(p'_2, w)$ are equivalent. Since O' is transverse, for every word w such that $\delta'(p'_1, w) \notin O'$, $\delta'(p'_2, w) = \delta'(p'_1, w)$. This equivalence is preserved in A'. Therefore if $p'_1 \neq p'_2$, contradiction with the minimality of A'. ∎

Lemma 3. *Let A be a complete DFA and A' be its complete minimal DFA. Let O' be an orbit in A' and O be a lift of O'. For all j in O, if there exists an $A_{O,j}$-consistent symbol a in Σ, then a is $A'_{O',[j]}$-consistent and $(A'_{O',[j]})_{\{a\}}$ is the minimal automaton of $L((A_{O,j})_{\{a\}})$.*

Proof. Let j be a state in O. Let $A'_{O',[j]} = (\Sigma, Q', [j], F', \delta')$ and $(A'_{O',[j]})_{\{a\}} = (\Sigma, Q', [j], F', \delta'')$. **(1)** If there is a symbol a in Σ such that a is not $A'_{O',[j]}$-consistent, there exist two gates $[q'_1]$ and $[q'_2]$ in O' such that $\delta'([q'_1], a) \in O'$ and $\delta'([q'_1], a) \neq \delta'([q'_2], a)$. As a consequence, there exist two gates q_1 and q_2 of O such that $q_1 \in [q'_1]$, $q_2 \in [q'_2]$ and $\delta(q_1, a) \in O$ and $\delta(q_1, a) \neq \delta(q_2, a)$. Finally, a is not $A_{O,j}$-consistent. **(2)** Let us prove that $(A'_{O',[j]})_{\{a\}}$ is minimal and that $L((A_{O,j})_{\{a\}}) = L((A'_{O',[j]})_{\{a\}})$. If $A_{O,j}$ is a-consistent, **(I)** implies that $A'_{O',[j]}$ is a-consistent. **(a)** Let p'_1 and p'_2 be two nonequivalent states in $A'_{O',[j]}$. By definition, there exists a symbol $b \neq a$ in Σ such that $\delta'(p'_1, b) \neq \delta'(p'_2, b)$, and by construction of $(A'_{O',[j]})_{\{a\}}$, $\delta''(p'_1, b) \neq \delta''(p'_2, b)$. By Lemma 2, $A'_{O',[j]}$ is minimal, and so is $(A'_{O',[j]})_{\{a\}}$. **(b)** Let $(A_{O,j})_{\{a\}} = (\Sigma, Q, j, F, \delta)$. The word w is in $L((A_{O,j})_{\{a\}}) \Leftrightarrow \delta(j, w) \in F \Leftrightarrow \delta''([j], w) \in F'' \Leftrightarrow$ The word w is in $L((A'_{O',[j]})_{\{a\}})$. ∎

Proposition 2. *Let A be a complete DFA and A' be its complete minimal DFA. If A satisfies the transverse property, so does A'.*

Proof. Assume that A' does not satisfy the transverse property. Then there exists an orbit O' in A' such that one of the four following cases occurs: **(I)** Suppose that O' is not transverse in A'. Let p' and q' be two gates in O'. According to Lemma 1, there exists an orbit O in A containing two nonequivalent states p and q such that p is merged into p' and q is merged into q'; moreover, since p' and q' are two gates of O', so are p and q in O. **(a)** If p' and q' do not have the same finality, neither do p nor q. **(b)** Let a be a symbol in Σ such that $\delta'(p', a) \notin (O' \cup \mathrm{sink}(A'))$ and $\delta'(q', a) \neq \delta'(p', a)$. As a consequence, $\delta(q, a) \neq \delta(p, a)$ and $\delta(p, a) \notin (O \cup \mathrm{sink}(A))$. In both cases, O is not transverse in A. **(II)** If O' is an orbit which is not transverse in ${}_cA'$, the same argument as **(I)** leads to the existence of an orbit O which is not transverse in ${}_cA$. **(III)** Suppose that O' is transverse in A' and there exists $[k]$ in O' such that $A'_{O',[k]}$ does not satisfy the consistence property. Let O be a lift of O'. Let j be a state in O which is in $[k]$. Then $[k] = [j]$. **(a)** Suppose that there exists no nonempty subset Σ' of Σ which is $A'_{O',[j]}$-consistent. As a consequence, every symbol a in Σ is not $A'_{O',[j]}$-consistent. If a in Σ is not $A'_{O',[j]}$-consistent, according to Lemma 3, a is not $A_{O,j}$-consistent. **(b)** Suppose that a is a $A_{O,j}$-consistent symbol. According to Lemma 3, the automaton $(A'_{O',[j]})_{\{a\}}$ is the minimal automaton of $L((A_{O,j})_{\{a\}})$. By recurrence on the number of transitions of the automata, according to **(I)** and **(II)**, if $(A'_{O',[j]})_{\{a\}}$ does not satisfy the transverse property, neither does $(A_{O,j})_{\{a\}}$. **(IV)** If O' is an orbit which is transverse in ${}_cA'$, we let O be a lift of O' and j be a state in O such that $A'_{O',[j]}$ does not satisfy the consistence

property. A similar argument as **(III)** leads to the fact that $A_{O,j}$ does not satisfy the consistence property.

Finally, if A' does not satisfy the transverse property, neither does A. ∎

4.2 From a Weakly One-Unambiguous Expression to a Linear-Size DFA Satisfying the Transverse Property

We show how to inductively compute a minimal DFA from a weakly one-unambiguous expression E. In a complete minimal DFA, we distinguish, if they exist, two particular states: the sink state and the *c-sink*, the only state q such that $L_q^{\rightarrow} = \Sigma^*$. Notice that the c-sink of A is the sink state of ${}_cA$. For a set Q of states in A, we denote by Q^+ the set $\{q \in Q \mid \exists w \in \Sigma^+, \delta(i,w) = q \wedge q \notin \mathrm{sink}(A) \cup \mathrm{sink}({}_cA)\}$. We show that Q^+ has at most $|E|$ elements. As a consequence, since $Q = Q^+ \cup \{i\} \cup \mathrm{sink}({}_cA)$, Q has at most $|E| + 2$ elements.

Lemma 4. *Let $A_1 = (\Sigma, Q_1, i_1, F_1, \delta_1)$ and $A_2 = (\Sigma, Q_2, i_2, F_2, \delta_2)$ be two minimal DFAs satisfying the transverse property such that*

$$\mathrm{First}(L(A_1)) \cap \mathrm{First}(L(A_2)) = \emptyset.$$

Let $A = (\Sigma, Q, i, F, \delta)$ be the minimal DFA of $L(A_1) \cup L(A_2)$. The two following propositions are satisfied:

(1) *the automaton A satisfies the transverse property,*

(2) $\mathrm{Card}(Q^+) \leq \mathrm{Card}(Q_1^+) + \mathrm{Card}(Q_2^+)$.

Proof. Let us consider the automaton $A' = (\Sigma, Q', i, F', \delta')$ defined by $Q' = Q_1 \cup Q_2 \cup \{i\}$, $F' = F_1 \cup F_2 \cup \{i\}$ if $\varepsilon \in L(A_1) \cup L(A_2)$, $F' = F_1 \cup F_2$ otherwise, and $\delta' = \delta_1 \cup \delta_2 \cup \{(i,a,p) \mid (i_1,a,p) \in \delta_1 \vee (i_2,a,p) \in \delta_2\}$. As $\mathrm{First}(L(A_1)) \cap \mathrm{First}(L(A_2)) = \emptyset$, A' is deterministic. By construction, since an orbit in A' is an orbit in A_1 or in A_2, A' satisfies the transverse property; moreover, $w \in L(A') \Leftrightarrow \delta'(i,w) \in F' \Leftrightarrow \delta_1(i_1,w) \in F_1 \vee \delta_2(i_2,w) \in F_2 \Leftrightarrow w \in L(A_1) \vee w \in L(A_2)$. Consequently, A' recognizes $L(A_1) \cup L(A_2)$. According to Proposition 2, the complete minimal DFA of $L(E)$ satisfies the transverse property. By construction, $\mathrm{Card}(Q'^+) = \mathrm{Card}(Q_1^+) + \mathrm{Card}(Q_2^+)$; finally, $\mathrm{Card}(Q^+) \leq \mathrm{Card}(Q_1^+) + \mathrm{Card}(Q_2^+)$. ∎

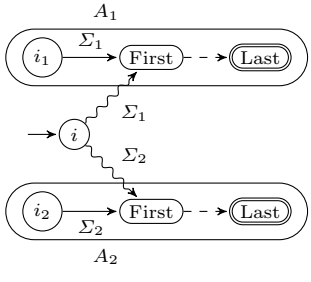

Fig. 4. The Automaton of $L(A_1) \cup L(A_2)$

Lemma 5. *Let $A_1 = (\Sigma, Q_1, i_1, F_1, \delta_1)$ and $A_2 = (\Sigma, Q_2, i_2, F_2, \delta_2)$ be two minimal DFAs satisfying the transverse property such that*

$$(\varepsilon \notin L(A_1) \vee \mathrm{First}(L(A_1)) \cap \mathrm{First}(L(A_2)) = \emptyset)$$
$$\textit{and } \mathrm{FollowLast}(L(A_1)) \cap \mathrm{First}(L(A_2)) = \emptyset.$$

Let $A = (\Sigma, Q, i, F, \delta)$ be the minimal DFA of $L(A_1) \cdot L(A_2)$. The following propositions are satisfied:

(1) *the automaton A satisfies the transverse property,*
(2) $\mathrm{Card}(Q^+) \leq \mathrm{Card}(Q_1^+) + \mathrm{Card}(Q_2^+)$.

Proof. Let us consider the automaton $A' = (\Sigma, Q', i, F', \delta')$ defined by $Q' = Q_1 \cup Q_2 \cup \{i\}$, $F' = F_1 \cup F_2 \cup \{i\}$ if $\varepsilon \in L(A_1) \wedge \varepsilon \in L(A_2)$, $F' = F_1 \cup F_2$ if $\varepsilon \notin L(A_1) \wedge \varepsilon \in L(A_2)$, $F' = F_2$ otherwise, and $\delta' = \delta_1 \cup \delta_2 \cup \{(i, a, p) \mid (i_1, a, p) \in \delta_1 \vee (i_1 \in F_1 \wedge (i_2, a, p) \in \delta_2\} \cup \{(p, a, p') \mid p \in F_1 \wedge (i_2, a, p') \in \delta_2\}$. Since $(\varepsilon \notin L(A_1) \vee \mathrm{First}(L(A_1)) \cap \mathrm{First}(L(A_2)) = \emptyset)$ and $\mathrm{FollowLast}(L(A_1)) \cap \mathrm{First}(L(A_2)) = \emptyset$, A' is deterministic. By construction, an orbit O in A' is an orbit in A_1 or in A_2; if O is in A_2, the transverse property is preserved; if O is in A_1, as the only transitions added are going out of O, the orbit languages are preserved; moreover, the new transitions preserve transversality: if a transition is added, its origin q is a final state, which is a gate in O; if O is transverse in A_1, all the other gates are final and transitions are also added.
If q is a non-final gate in ${}_cA_1$, all these gates are non final in ${}_cA_1$ and final in A_1, consequently transitions out of them are also added. If q is a final state in A_1 which is not a gate in ${}_cA_1$, for all symbol a in Σ, either $\delta'(q, a) = \mathrm{sink}({}_cA_1)$ or $\delta'(q, a) \in O$. As a consequence, every symbol a in Σ is in $\mathrm{FollowLast}(L(E_1))$. Either $L(A_2) = \emptyset$ or $L(A_2) = \{\varepsilon\}$ and there is no transition added, or contradiction with $\mathrm{FollowLast}(L(A_1)) \cap \mathrm{First}(L(A_2)) = \emptyset$.

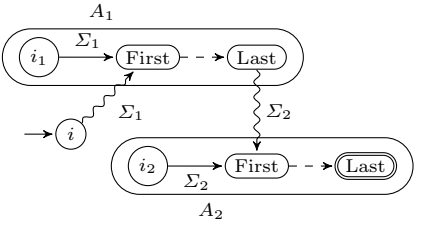

Fig. 5. The Automaton of $L(A_1) \cdot L(A_2)$ in the case where $\varepsilon \notin L(A_1)$ and $\varepsilon \notin L(A_2)$

As a consequence, A' satisfies the transverse property. Moreover, $w \in L(A) \Leftrightarrow \delta'(i, w) \in F' \Leftrightarrow (w = w'aw'' \wedge \delta_2(\delta'(\delta_1(i_1, w'), a), w'') \in F_2) \vee (\delta'(i_2, w) \in F_2 \wedge i_1 \in F_1) \vee (\delta_1(i_1, w) \in F_1 \wedge \varepsilon \in L(A_2)) \Leftrightarrow w \in L(A_1) \cdot L(A_2)$. Consequently, A' recognizes $L(A_1) \cdot L(A_2)$. According to Proposition 2, the complete minimal DFA of $L(E)$ satisfies the transverse property. By construction, $\mathrm{Card}(Q'^+) = \mathrm{Card}(Q_1^+) + \mathrm{Card}(Q_2^+)$; finally, $\mathrm{Card}(Q^+) \leq \mathrm{Card}(Q_1^+) + \mathrm{Card}(Q_2^+)$. ∎

Lemma 6. *Let $A_1 = (\Sigma, Q_1, i_1, F_1, \delta_1)$ be a minimal DFA satisfying the transverse property such that*

$$\mathrm{FollowLast}(L(A_1)) \cap \mathrm{First}(L(A_1)) = \emptyset.$$

Let $A = (\Sigma, Q, i, F, \delta)$ be the minimal DFA of $L(A_1)^*$. The two following propositions are satisfied:

(1) *the automaton A satisfies the transverse property,*
(2) $\mathrm{Card}(Q^+) \leq \mathrm{Card}(Q_1^+)$.

Proof. Let us consider the automaton $A' = (\Sigma, Q', i, F', \delta')$ defined by $Q' = Q_1 \cup \{i\}$, $F' = F_1 \cup \{i\}$, and $\delta' = \delta_1 \cup \{(i, a, p) \mid (i_1, a, p) \in \delta_1\} \cup \{(p, a, p') \mid$

$p \in F_1 \wedge (i_1, a, p') \in \delta_1\}$. Since $\text{FollowLast}(L(A_1)) \cap \text{First}(L(A_1)) = \emptyset$, A' is deterministic. By construction, there is only one orbit which is transverse. Furthermore, the set $\text{First}(L(A_1))$ is A'-consistent, and the $\text{First}(L(A_1))$-cut of A' is A_1. As a consequence, A' satisfies the transverse property. Moreover, $w \in L(A) \Leftrightarrow \delta'(i, w) \in F' \Leftrightarrow w = w_1 w_2 \cdot \cdots \cdot w_k \wedge \delta'(i, w_1) \in F' \wedge \delta'(i, w_2) \in F' \ldots \delta'(i, w_k) \in F' \Leftrightarrow w_1 \in L(A_1) \wedge w_2 \in L(A_1) \ldots w_k \in L(A_1) \Leftrightarrow w \in L(A_1)^*$. Consequently, A' recognizes $L(A_1)^*$.

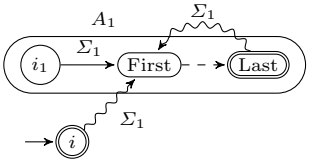

Fig. 6. The Automaton of $L(A_1)^*$

According to Proposition 2, the complete minimal DFA of $L(A')$ satisfies the transverse property. By construction, $\text{Card}(Q'^+) = \text{Card}(Q_1^+)$; finally, $\text{Card}(Q^+) \leq \text{Card}(Q_1^+)$. ∎

Lemma 7. *Let $A_1 = (\Sigma, Q_1, i_1, F_1, \delta_1)$ be a minimal DFA satisfying the transverse property. Let $A = (\Sigma, Q, i, F, \delta)$ be the minimal DFA of $\neg L(A_1)$. The two following propositions are satisfied:*

(1) *the automaton A satisfies the transverse property,*

(2) $\text{Card}(Q^+) = \text{Card}(Q_1^+)$.

Proof. By construction, ${}_cA_1$ is the complete minimal DFA of $\neg(L(A_1))$, which satisfies the transverse property. By construction, $\text{Card}(Q'^+) = \text{Card}(Q_1^+)$; finally, $\text{Card}(Q^+) = \text{Card}(Q_1^+)$. ∎

Proposition 3. *Let E be a weakly one-unambiguous expression and A' be the minimal DFA of $L(E)$. Then A satisfies the transverse property and A has at most $|E| + 2$ states.*

Proof. According to Lemma 4, Lemma 5, Lemma 6 and Lemma 7.

4.3 From a Minimal Automaton Satisfying the Transverse Property to a Weakly One-Unambiguous Expression

We show that the language denoted by a DFA satisfying the transverse property is weakly one-unambiguous. Let $A = (\Sigma, Q, i, F, \delta)$ be a DFA and q be a state in Q. The *q-starting automaton* A^q of A is the accessible part of the automaton $(\Sigma, Q, q, F, \delta)$. Note that A^q is a subautomaton of A and $L(A^q) = L_q^{\rightarrow}(A)$.

Proposition 4. *Let A be a DFA satisfying the transverse property. Then $L(A)$ is weakly one-unambiguous.*

Proof. We show how to inductively compute $L_q^A = L(A^q)$ from $A = (\Sigma, Q, i, F, \delta)$ and we show that L_q^A is weakly one-unambiguous.

(I) Suppose that Q is not an orbit.

(a) If i is in an orbit O which is transverse in A, let Σ' be the subset $\{a \in \Sigma \mid \forall q \in \text{gates}(O), \delta(q,a) \notin O\}$. Let us consider the language L' defined by $L' = (L_{O,i}) \cdot (\bigcup_{a\in\Sigma'}(\{a\} \cdot L^A_{\mathrm{f}(a)}))$ if $\text{gates}(O) \cap F = \emptyset$ or $L' = (L_{O,i}) \cdot (\{\varepsilon\} \cup \bigcup_{a\in\Sigma'}(\{a\} \cdot L^A_{\mathrm{f}(a)}))$ otherwise, where, $L_{O,i}$ is the orbit language of O beginning in the state i (see **(II)**). A word w is in L^A_i if and only if it can be split into $w = w_1 w_2$ where w_1 is a path from i to a gate g of O and w_2 a path from g to a final state of A which is a nonempty path in A^g if $w_2 \neq \varepsilon$. As a consequence, $L^A_i = L'$. By recurrence on the number of transitions and according to **(II)**, $L_{O,i}$ and $L'' = \bigcup_{a\in\Sigma'}(\{a\} \cdot L^A_{\mathrm{f}(a)})$ are weakly one-unambiguous. If $\varepsilon \in L_{O,i} \wedge \text{First}(L_{O,i}) \cap \text{First}(L'') \neq \emptyset$, then there exist two transitions labelled by the same symbol going out of i, which contradicts the determinism. If $\text{FollowLast}(L_{O,i}) \cap \text{First}(L'') \neq \emptyset$, then there exist two transitions labelled by the same symbol going out of a gate of O, one going in O, and the other in $\mathrm{f}(a)$, which contradicts the determinism. As a consequence, L' is weakly one-unambiguous.
(b) If i is in an orbit O which is transverse in ${}_cA$, then $L^A_i = \neg(L^{{}_cA}_i)$ which is weakly one-unambiguous according to **(a)**.
(c) If i is not in an orbit, $L^A_i = \bigcup_{a\in\Sigma}(\{a\} \cdot L^A_{\delta(i,a)})$ if $i \notin F$, $L^A_i = \{\varepsilon\} \cup \bigcup_{a\in\Sigma}(\{a\} \cdot L^A_{\delta(i,a)})$. By recurrence on the number of transitions, $L^A_i = \bigcup_{a\in\Sigma}(\{a\} \cdot L^A_{\delta(i,a)})$ is weakly one-unambiguous.
(II) Suppose that Q is an orbit. Let Σ' be a subset of Σ.
(a) If Σ' is A-consistent, let us consider the language $L' = L^{A_{\Sigma'}}_i \cdot (\bigcup_{a\in\Sigma'}(\{a\} \cdot L^{A_{\Sigma'}}_{\mathrm{f}(a)}))^*$. Since $A_{\Sigma'}$ satisfies the transverse property, according to **(I)**, $L^{A_{\Sigma'}}_i$ is weakly one-unambiguous. By recurrence on the number of transitions, $L'' = \bigcup_{a\in\Sigma'}(\{a\} \cdot L^{A_{\Sigma'}}_{\mathrm{f}(a)})$ is weakly one-unambiguous. Suppose that $\text{FollowLast}(L'') \cap \text{First}(L'') \neq \emptyset$; then there exist two transitions labelled by the same symbol going out of a gate of the orbit, one leading to $\mathrm{f}(a)$ and the other to another state of O. Contradiction with determinism. The same contradiction is implied if $\varepsilon \in L^{A_{\Sigma'}}_i \wedge \text{First}(L^{A_{\Sigma'}}_i) \cap \Sigma' \neq \emptyset$ or $\text{FollowLast}(L^{A_{\Sigma'}}_i) \cap \text{First}(L'') \neq \emptyset$. Finally, since a word w in $L(A)$ can be split up into $w_1 \cdot w_2$ such that w_1 is the label of a path from i to a final state and w_2 the label of a path from a final state to another final state, w is in L'.
(b) If Σ' is ${}_cA$-consistent, then $L^A_i = \neg(L^{{}_cA}_i)$, which is weakly one-unambiguous according to **(a)**. ∎

Example 3. Consider the automaton A in Figure 7. It is composed by the four orbits $O_1 = \{2,3\}$, $O_2 = \{4,5\}$ and the singleton $\{1\}$ and $\{6\}$. The singletons trivially satisfy the transverse property. The orbit O_1 is transverse in A and O_2

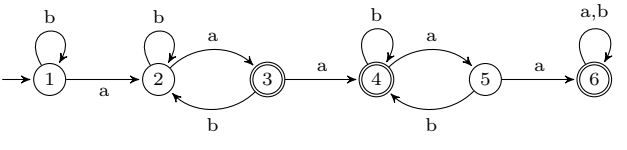

Fig. 7. The automaton A

is transverse in the complement of A. As a consequence, it can be denoted by the weakly one-unambiguous expression $E = b^*ab^*a(bb^*a)^*(\varepsilon + a\neg(b^*a(bb^*a)^*)))$.

5 Conclusion

Content models of XML DTD are one-unambiguous [3]. Usually, the size of a regular expression can be reduced using the complement operator. Thus the closure property of one-unambiguous regular languages is important in XML applications. We have demonstrated that one-unambiguous regular languages are closed neither under intersection nor under complement. We have also considered one-unambiguous regular languages with complement closure property (weak one-unambiguity). We have investigated the closure properties of weakly one-unambiguous regular languages and the state complexity of the languages on some operations. Finally, we have shown that as far as weakly one-unambiguous regular expressions are concerned, a linear-size deterministic recognizer can be computed in order to decide whether or not a word belongs to the language denoted by a non-simple regular expression.

References

1. Blum, N.: An $O(n \log n)$ implementation of the standard method for minimizing n-state finite automata. Inform. Process. Lett. 57(2), 65–69 (1996)
2. Bray, T., Paoli, J., Sperberg-Mc Queen, C.M., Maler, E., Yergeau, F.: Extensible Markup Language (XML) 1.0, 4th edn. (2006), `http://www.w3.org/TR/2006/REC-xml-20060816`
3. Brüggemann-Klein, A., Wood, D.: One-unambiguous regular languages. Inform. Comput. 140, 229–253 (1998)
4. Gelade, W.: Succinctness of regular expressions with interleaving, intersection and counting. Theor. Comput. Sci. 411(31-33), 2987–2998 (2010)
5. Gelade, W., Neven, F.: Succinctness of the complement and intersection of regular expressions. In: Albers, S., Weil, P. (eds.) STACS. Dagstuhl Seminar Proceedings, vol. 08001, pp. 325–336 (2008)
6. Glushkov, V.M.: The abstract theory of automata. Russian Mathematical Surveys 16, 1–53 (1961)
7. Hopcroft, J.E.: An n log n algorithm for minimizing the states in a finite automaton. In: Kohavi, Z. (ed.) The Theory of Machines and Computations, pp. 189–196. Academic Press, New York (1971)
8. Kleene, S.: Representation of events in nerve nets and finite automata. In: Automata Studies, Ann. Math. Studies, vol. 34, pp. 3–41. Princeton U. Press (1956)
9. McNaughton, R.F., Yamada, H.: Regular expressions and state graphs for automata. IEEE Transactions on Electronic Computers 9, 39–57 (1960)
10. Moore, E.F.: Gedanken experiments on sequential machines. In: Automata Studies, pp. 129–153. Princeton Univ. Press, Princeton (1956)
11. Myhill, J.: Finite automata and the representation of events. WADD, TR-57-624, 112–137 (1957)
12. Nerode, A.: Linear automata transformation. In: Proceedings of AMS, vol. 9, pp. 541–544 (1958)
13. Rabin, M.O., Scott, D.: Finite automata and their decision problems. IBM J. Res. 3(2), 115–125 (1959)