On Complementing Unambiguous Automata and Graphs With Many Cliques and Cocliques

Emil Indzhev, Stefan Kiefer[1]

University of Oxford, UK

Abstract

We show that for any unambiguous finite automaton with n states there exists an unambiguous finite automaton with $\sqrt{n+1} \cdot 2^{n/2}$ states that recognizes the complement language. This builds and improves upon a similar result by Jirásek et al. [Int. J. Found. Comput. Sci. 29 (5) (2018)]. Our improvement is based on a reduction to and an analysis of a problem from extremal graph theory: we show that for any graph with n vertices, the product of the number of its cliques with the number of its cocliques (independent sets) is bounded by $(n+1)2^n$.

1. Introduction

Given two finite automata $\mathcal{A}_1, \mathcal{A}_2$, recognizing languages $L_1, L_2 \subseteq \Sigma^*$, respectively, the *state complexity* of union (or intersection, or complement, etc.) is how many states may be needed for an automaton that recognizes $L_1 \cup L_2$ (or $L_1 \cap L_2$, or $\Sigma^* \setminus L_1$, etc.). The state complexity depends on the type of automaton considered, such as nondeterministic finite automata (NFAs), deterministic finite automata (DFAs), or unambiguous finite automata (UFAs). UFAs are NFAs that have, for each word $w \in \Sigma^*$, either one or zero accepting runs.

The state complexity has been well studied for various types of automata and language operations, see, e.g., [1] and the references therein for some known results. For example, it was shown in [2] that complementing an NFA with n states may require $\Theta(2^n)$ states. However, the state complexity for UFAs is not yet fully understood. It was shown only in 2018 by Raskin [3] that the state complexity for UFAs and complement is not polynomial:

Proposition 1 ([3]). *For any $n \in \mathbb{N}$ there exists a unary (i.e., $|\Sigma| = 1$) UFA $\mathcal{A}_n$ with n states such that any NFA that recognizes $\Sigma^* \setminus L(\mathcal{A}_n)$ has at least $n^{(\log\log\log n)^{\Theta(1)}}$ states.*

This super-polynomial blowup (even for unary alphabet and even if the output automaton is allowed to be ambiguous) refuted a conjecture that it may

[1] Stefan Kiefer is supported by a Royal Society University Research Fellowship.

be possible to complement UFAs with a polynomial blowup [4]. A non-trivial upper bound (for general alphabets and outputting a UFA) was shown by Jirásek et al. [1]:

Proposition 2 ([1]). *Let $\mathcal{A}$ be a UFA with $n \geq 7$ states that recognizes a language $L \subseteq \Sigma^*$. Then there exists a UFA with at most $n \cdot 2^{0.786n}$ states that recognizes the language $\Sigma^* \setminus L$.*

In this note we build and improve on this result. We show:

Theorem 3. *Let $\mathcal{A}$ be a UFA with $n \geq 0$ states that recognizes a language $L \subseteq \Sigma^*$. Then there exists a UFA with at most $\sqrt{n+1} \cdot 2^{n/2}$ states that recognizes the language $\Sigma^* \setminus L$.*

Theorem 3 is based on a tight analysis of the complementation construction due to Jirásek et al. [1]. Their construction performs two UFA complementation procedures and picks the smaller UFA among the two. The first procedure is the standard subset construction for determinizing NFAs, followed by swapping accepting and non-accepting states. The second procedure is a symmetrical "backward" variant. It was shown in [1] that the smaller of the two resulting UFAs has at most $n \cdot 2^{0.786n}$ states. We improve this upper bound to $\sqrt{n+1} \cdot 2^{n/2}$ states. We also show that this bound is tight up to a constant factor 2; i.e., for all n there is a UFA with n states where the complementation construction from [1] produces a UFA with at least $\frac{1}{2}\sqrt{n+1} \cdot 2^{n/2}$ states.

Our improvement is based on two technical contributions which may be of independent interest:

1. We reduce the analysis of the complementation construction described above to a problem from extremal graph theory: loosely speaking, we show in Section 3 that applying the construction to a UFA with n states can yield a large UFA if and only if there exists a graph with n vertices that has both many cliques and many cocliques (independent sets).
2. We solve this graph problem: we show in Section 4 that any graph with n vertices has at most $\sqrt{n+1} \cdot 2^{n/2}$ cliques or at most $\sqrt{n+1} \cdot 2^{n/2}$ cocliques. This bound is tight up to a factor of 2.

These results enable us to prove Theorem 3 in Section 5. There we also show that our analysis of the complementation construction from [1] is almost tight.

2. Preliminaries

Finite Automata

An *NFA* is a quintuple $\mathcal{A} = (Q, \Sigma, \delta, I, F)$, where Q is the finite set of states, Σ is the finite alphabet, $\delta \subseteq Q \times \Sigma \times Q$ is the transition relation, $I \subseteq Q$ is the set of initial states, and $F \subseteq Q$ is the set of accepting states. We write $q \xrightarrow{a} r$ to denote that $(q, a, r) \in \delta$. A finite sequence $q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} q_n$ is called a *run* and can also be summarized as $q_0 \xrightarrow{a_1 \cdots a_n} q_n$. The NFA $\mathcal{A}$ *recognizes* the language $L(\mathcal{A}) := \{w \in \Sigma^* \mid \exists\, q_0 \in I \,.\, \exists\, f \in F \,.\, q_0 \xrightarrow{w} f\}$. The NFA $\mathcal{A}$ is a *DFA* if $|I| = 1$ and for every $q \in Q$ and $a \in \Sigma$ there is exactly one q' with $q \xrightarrow{a} q'$. The NFA $\mathcal{A}$ is a *UFA* if for every word $w = a_1 \cdots a_n \in \Sigma^*$ there is at most one *accepting* run for w, i.e., a run $q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} q_n$ with $q_0 \in I$ and $q_n \in F$. Clearly, any DFA is a UFA.

Let $\mathcal{A} = (Q, \Sigma, \delta, I, F)$ be an NFA. For $S \subseteq Q$ and $w \in \Sigma^*$ we write

$$\begin{aligned} \delta(S, w) &:= \{r \in Q \mid \exists\, q \in S \,.\, q \xrightarrow{w} r\} \qquad \text{and} \\ \delta^{-1}(w, S) &:= \{r \in Q \mid \exists\, q \in S \,.\, r \xrightarrow{w} q\}\,. \end{aligned}$$

The *forward determinization* of $\mathcal{A}$ is the DFA obtained by the standard subset construction, i.e., the DFA $\mathcal{A}_f := (Q_f, \Sigma, \delta_f, \{I\}, F_f)$ with $Q_f := \{\delta(I, w) \mid w \in \Sigma^*\}$ and $\delta_f := \{(S, a, \delta(S, a)) \mid S \in Q_f,\ a \in \Sigma\}$ and $F_f := \{S \in Q_f \mid S \cap F \neq \emptyset\}$. Analogously, the *backward determinization* of $\mathcal{A}$ is the NFA $\mathcal{A}_b := (Q_b, \Sigma, \delta_b, I_b, \{F\})$ where $Q_f := \{\delta^{-1}(w, F) \mid w \in \Sigma^*\}$ and $\delta_b := \{(\delta^{-1}(a, S), a, S) \mid S \in Q_b,\ a \in \Sigma\}$ and $I_b := \{S \in Q_b \mid S \cap I \neq \emptyset\}$. Note that $L(\mathcal{A}) = L(\mathcal{A}_f) = L(\mathcal{A}_b)$. The NFA $\mathcal{A}_b$ is "backward-deterministic", i.e., it has a single accepting state and for every $S \in Q_b$ and $a \in \Sigma$ there is exactly one S' with $S' \xrightarrow{a} S$. It follows that $\mathcal{A}_b$ is a UFA. UFAs recognizing $\Sigma^* \setminus L(\mathcal{A})$ can be obtained from $\mathcal{A}_f$ (resp., $\mathcal{A}_b$) by swapping accepting and non-accepting (resp., initial and non-initial) states. It follows:

Lemma 4. *Let $\mathcal{A}$ be an NFA that recognizes a language $L \subseteq \Sigma^*$. Suppose that its forward determinization has k states and its backward determinization has ℓ states. Then there is a UFA with at most* $\min\{k, \ell\}$ *states that recognizes the language $\Sigma^* \setminus L$.*

Like Proposition 2 from [1], Theorem 3 is based on Lemma 4.

Graphs

An (undirected, simple, finite) *graph* is a pair $G = (V, E)$, where V is the finite set of vertices and $E \subseteq \{\{v, v'\} \subseteq V \mid v \neq v'\}$ is the set of edges. A *clique* in G is a set $X \subseteq V$ of vertices such that whenever $v, v' \in X$ and $v \neq v'$ then $\{v, v'\} \in E$. A *coclique* (or independent set) in G is a set $Y \subseteq V$ of vertices such that whenever $v, v' \in Y$ and $v \neq v'$ then $\{v, v'\} \notin E$. Note that any subset of a clique, including the empty set, is also a clique, and similarly for cocliques.

3. Reduction to a Graph Problem

The proof of our main result, Theorem 3, rests on two key auxiliary results. In this section we prove the first key lemma, Lemma 5. Loosely speaking, it says that a UFA with a large forward determinization and a large backward determinization defines a graph with many cliques and cocliques:

Lemma 5. *Let $\mathcal{A} = (Q, \Sigma, \delta, I, F)$ be a UFA. Suppose that its forward determinization has k states, and its backward determinization has ℓ states. Then there exists a graph $G = (Q, E)$ that has at least k cliques and at least ℓ cocliques.*

Proof. Define

$$E \;:=\; \{\{q,q'\} \subseteq Q \mid q \neq q',\ \exists\, q_0, q_0' \in I \,.\, \exists\, w \in \Sigma^* \,.\, q_0 \xrightarrow{w} q \wedge q_0' \xrightarrow{w} q'\}\,.$$

It follows from the definition of the forward determinization that every state in the forward determinization of $\mathcal{A}$ is a clique in G. So G has at least k cliques.

Let $Y \subseteq Q$ be a state in the backward determinization of $\mathcal{A}$. It suffices to show that Y is a coclique in G. If $Y = \emptyset$, then Y is a coclique. Let $q, q' \in Y$. It suffices to show that $\{q,q'\} \notin E$. If there is no word $w_1 \in \Sigma^*$ with $q, q' \in \delta(I, w_1)$, then $\{q,q'\} \notin E$. So we can assume that there are states $q_0, q_0' \in I$ and a word $w_1 \in \Sigma^*$ such that $q_0 \xrightarrow{w_1} q$ and $q_0' \xrightarrow{w_1} q'$. Since $Y \supseteq \{q,q'\}$ is a state in the backward determinization, there are a word $w_2 \in \Sigma^*$ and states $f, f' \in F$ such that $q \xrightarrow{w_2} f$ and $q' \xrightarrow{w_2} f'$ in $\mathcal{A}$. Thus, $\mathcal{A}$ accepts the word $w_1 w_2$ via runs

$$q_0 \xrightarrow{w_1} q \xrightarrow{w_2} f \qquad \text{and} \qquad q_0' \xrightarrow{w_1} q' \xrightarrow{w_2} f'\,.$$

Since $\mathcal{A}$ is unambiguous, these runs are equal. Thus, $q = q'$. We conclude that $\{q,q'\} \notin E$. □

Lemma 5 has the following converse:

Lemma 6. *Let (V,E) be a graph with k cliques and ℓ cocliques. Then there is a UFA with $|V|$ states whose forward determinization has at least k states and whose backward determinization has at least ℓ states.*

Proof. Let (V,E) be a graph, and let $\mathcal{X}, \mathcal{Y} \subseteq 2^V$ be the sets of its cliques and cocliques, respectively. If $V = \emptyset$, then $\mathcal{X} = \mathcal{Y} = \{\emptyset\}$, and $\emptyset$ is a state (the only one) of the forward and the backward determinization of a UFA with no states. So we can assume that there is $v_0 \in V$. Define the NFA $\mathcal{A} := (V, \Sigma, \delta, \{v_0\}, \{v_0\})$ where $\Sigma := \Sigma_1 \cup \Sigma_2$ and $\Sigma_1 := \{1\} \times \mathcal{X}$ and $\Sigma_2 := \{2\} \times \mathcal{Y}$ and $\delta := \delta_1 \cup \delta_2$ and $\delta_1 := \{(v_0, (1,X), v) \mid v \in X \in \mathcal{X}\}$ and $\delta_2 := \{(v, (2,Y), v_0) \mid v \in Y \in \mathcal{Y}\}$. For each $X \in \mathcal{X}$ we have $\delta(\{v_0\}, (1,X)) = X$, and for each $Y \in \mathcal{Y}$ we have $\delta^{-1}((2,Y), \{v_0\}) = Y$. Hence, every clique is a state of the forward determinization of $\mathcal{A}$, and every coclique is a state of the backward determinization of $\mathcal{A}$.

It remains to show that $\mathcal{A}$ is a UFA. Let $w_1, w_2 \in \Sigma^*$ and $v, v' \in V$ with $v_0 \xrightarrow{w_1} v \xrightarrow{w_2} v_0$ and $v_0 \xrightarrow{w_1} v' \xrightarrow{w_2} v_0$. It suffices to show that $v = v'$. If w_1 or w_2 is the empty word or w_1 ends with a letter of the form $(2,Y)$ or w_2 begins with a letter of the form $(1,X)$, then $v = v_0 = v'$. So we can assume that w_1 ends with, say, $(1,X)$, and w_2 begins with, say, $(2,Y)$. All transitions labelled with $(1,X)$ have v_0 as source, and all transitions labelled with $(2,Y)$ have v_0 as target, so we have

$$\begin{aligned} v, v' \;&\in\; \delta(\{v_0\}, w_1) \;\subseteq\; \delta(\{v_0\}, (1,X)) \;\subseteq\; X \qquad\qquad \text{and} \\ v, v' \;&\in\; \delta^{-1}(w_2, \{v_0\}) \;\subseteq\; \delta^{-1}((2,Y), \{v_0\}) \;\subseteq\; Y\,. \end{aligned}$$

Since X is a clique, we have $v = v'$ or $\{v,v'\} \in E$. Since Y is a coclique, we have $v = v'$ or $\{v,v'\} \notin E$. We conclude that $v = v'$. □

4. Graphs With Many Cliques and Many Cocliques

This section is about a problem from extremal graph theory. Theorem 7 serves as the second of our two key auxiliary results, but may be of independent interest.

Theorem 7. *Any graph with n vertices has at most $\sqrt{n+1} \cdot 2^{n/2}$ cliques or at most $\sqrt{n+1} \cdot 2^{n/2}$ cocliques. Moreover, for any $n \geq 0$ there is a graph with n vertices that has at least $\frac{1}{2}\sqrt{n+1} \cdot 2^{n/2}$ cliques and at least $\frac{1}{2}\sqrt{n+1} \cdot 2^{n/2}$ cocliques.*

Our proof strategy is to consider pairs of a clique and a coclique. The following lemma counts the possibilities of partitioning the vertices of a subgraph into a clique and a coclique.

Lemma 8. *Let (V, E) be a graph. For any $S \subseteq V$, let*

$$P_S \;:=\; \{X \subseteq S \mid X \text{ is a clique and } S \setminus X \text{ is a coclique}\}\,.$$

Then $|P_S| \leq |S| + 1$.

Proof. We proceed by induction on $|S|$. For $S = \emptyset$ we have $P_\emptyset = \{\emptyset\}$. Suppose that $|P_S| \leq |S| + 1$ holds for some $S \subset V$, and let $v \in V \setminus S$. For the inductive step, it suffices to show that $|P'| \leq |P_S| + 1$ where $P' := P_{S \cup \{v\}}$. Every $X' \in P'$ can be expressed as X or as $X \cup \{v\}$ for some $X \in P_S$. If there is an $X \subseteq S$ with $X \in P'$ and $X \cup \{v\} \in P'$, then X must be the set of neighbours of v in S; i.e., there is at most one $X \in P_S$ with both $X \in P'$ and $X \cup \{v\} \in P'$. It follows that $|P'| \leq |P_S| + 1$. $\square$

The following lemma is similar, but considers pairs of a clique and a coclique that may overlap.

Lemma 9. *Let (V, E) be a graph. For any $S \subseteq V$, let*

$$R_S \;:=\; \{(X, Y) \in 2^S \times 2^S \mid X \cup Y = S \text{ and } X \text{ is a clique and } Y \text{ is a coclique}\}\,.$$

Then $|R_S| \leq 2|S| + 1$.

Proof. Let $S \subseteq V$. Every $(X, Y) \in R_S$ with $X \cap Y = \emptyset$ can be uniquely written as $(X, S \setminus X)$ with $X \in P_S$, where P_S is defined as in Lemma 8. By Lemma 8, there are at most $|S| + 1$ pairs $(X, Y) \in R_S$ with $X \cap Y = \emptyset$.

Let $v \in S$. Suppose there is $(X_v, Y_v) \in R_S$ with $v \in X_v \cap Y_v$. It follows from the definition of R_S that $X_v \cap Y_v = \{v\}$ and that $X_v \setminus \{v\}$ is the set of neighbours of v in S and that $Y_v \setminus \{v\}$ is the set of non-neighbours of v in S. So (X_v, Y_v) is the only $(X, Y) \in R_S$ with $v \in X \cap Y$.

We conclude that there are at most $|S|$ pairs $(X, Y) \in R_S$ with $X \cap Y \neq \emptyset$. Hence, $|R_S| \leq (|S| + 1) + |S| = 2|S| + 1$. $\square$

Lemma 9 implies a bound on the product of the number of cliques with the number of cocliques:

Lemma 10. *Let (V, E) be a graph with $|V| = n$. Then*

$$|\{X \subseteq V \mid X \text{ is a clique}\}| \cdot |\{Y \subseteq V \mid Y \text{ is a coclique}\}| \;\leq\; (n+1)2^n\,.$$

Proof. The product on the left-hand side is equal to $|R|$, where

$$R \;:=\; \{(X, Y) \in 2^V \times 2^V \mid X \text{ is a clique and } Y \text{ is a coclique}\}\,.$$

We have $R = \bigcup_{S \subseteq V} R_S$, so by Lemma 9 we have

$$\begin{aligned}
|R| \;&\leq\; \sum_{i=0}^{n} \binom{n}{i}(2i+1) \;=\; \sum_{i=0}^{n} \left(\binom{n}{i} i + \binom{n}{i}(n-i) + \binom{n}{i}\right) \\
&=\; \sum_{i=0}^{n} \left(\binom{n}{i} n + \binom{n}{i}\right) \;=\; 2^n(n+1)\,.
\end{aligned}$$

□

The bound in Lemma 10 is tight, as a graph with n vertices and $\binom{n}{2}$ edges has 2^n cliques and $n+1$ cocliques.

Now we can prove Theorem 7.

Theorem 7. *Any graph with n vertices has at most $\sqrt{n+1} \cdot 2^{n/2}$ cliques or at most $\sqrt{n+1} \cdot 2^{n/2}$ cocliques. Moreover, for any $n \geq 0$ there is a graph with n vertices that has at least $\frac{1}{2}\sqrt{n+1} \cdot 2^{n/2}$ cliques and at least $\frac{1}{2}\sqrt{n+1} \cdot 2^{n/2}$ cocliques.*

Proof. Since $\min\{k, \ell\} \leq \sqrt{k \cdot \ell}$ holds for all k, ℓ, the upper bound follows from Lemma 10. For the lower bound, let $n \geq 0$. Define a graph $G = (V_1 \cup V_2, E)$ with $|V_1| = k$ and $|V_2| = n - k$ for some $k \in \{0, 1, \ldots, n\}$ and $E = \{\{v, v'\} \subseteq V_1 \mid v \neq v'\}$. Graph G has at least 2^k cliques and at least (in fact, exactly) $(k+1)2^{n-k}$ cocliques. Choose k as an integer nearest to $\frac{n}{2} + \frac{1}{2}\log_2 \frac{n+1}{2}$. Then we have

$$\begin{aligned}
2^k \;&\geq\; 2^{\frac{n}{2} - \frac{1}{2} + \frac{1}{2}\log_2 \frac{n+1}{2}} \;=\; \frac{1}{2}\sqrt{n+1} \cdot 2^{\frac{n}{2}} \qquad \text{and} \\
(k+1)2^{n-k} \;&\geq\; \left(\frac{n}{2} + \frac{1}{2} + \frac{1}{2}\log_2 \frac{n+1}{2}\right) 2^{\frac{n}{2} - \frac{1}{2} - \frac{1}{2}\log_2 \frac{n+1}{2}} \\
&\geq\; \frac{n+1}{2} \cdot 2^{\frac{n}{2} - \frac{1}{2} - \frac{1}{2}\log_2 \frac{n+1}{2}} \\
&=\; \frac{1}{2}\sqrt{n+1} \cdot 2^{\frac{n}{2}}\,.
\end{aligned}$$

□

5. Proof of the Main Result and Conclusions

In the previous two sections we proved our key auxiliary results, Lemma 5 and Theorem 7, respectively. We use them to show the main theorem:

Theorem 3. *Let $\mathcal{A}$ be a UFA with $n \geq 0$ states that recognizes a language $L \subseteq \Sigma^*$. Then there exists a UFA with at most $\sqrt{n+1} \cdot 2^{n/2}$ states that recognizes the language $\Sigma^* \setminus L$.*

Proof. Suppose that the forward determinization of $\mathcal{A}$ has k states, and the backward determinization of $\mathcal{A}$ has ℓ states. By Lemma 5, there is a graph with n vertices and at least k cliques and at least ℓ cocliques. By Theorem 7, it follows that $k \leq \sqrt{n+1} \cdot 2^{n/2}$ or $\ell \leq \sqrt{n+1} \cdot 2^{n/2}$. Hence, by Lemma 4, there is a UFA with at most $\sqrt{n+1} \cdot 2^{n/2}$ states that recognizes the language $\Sigma^* \setminus L$. □

By the argument justifying Lemma 4, the UFA for $\Sigma^* \setminus L$ in Theorem 3 is obtained either by swapping accepting and non-accepting states in the forward determinization or by swapping initial and non-initial states in the backward determinization. So we have actually proved that for any UFA with n states, either its forward determinization or its backward determinization has at most $\sqrt{n+1} \cdot 2^{n/2}$ states. This bound is optimal up to a factor of 2:

Proposition 11. *For all $n \geq 0$ there is a UFA with n states such that both its forward and its backward determinization have at least $\frac{1}{2}\sqrt{n+1} \cdot 2^{n/2}$ states.*

Proof. Let $n \geq 0$. By Theorem 7, there is a graph with n vertices that has at least $\frac{1}{2}\sqrt{n+1} \cdot 2^{n/2}$ cliques and at least $\frac{1}{2}\sqrt{n+1} \cdot 2^{n/2}$ cocliques. Now the statement follows from Lemma 6. □

We conclude that the UFA complementation construction due to Jirásek et al. [1] has a worst-case state complexity of $\Theta(\sqrt{n+1} \cdot 2^{n/2})$. This is currently the best upper bound on the state complexity of complementing UFAs. Obtaining it has been the main contribution of this article. It remains possible that other constructions lead to smaller, even sub-exponential, UFAs for the complement language.

References

[1] J. Jirásek Jr., G. Jirásková, J. Sebej, Operations on unambiguous finite automata, International Journal of Foundations of Computer Science 29 (5) (2018) 861–876. `doi:10.1142/S012905411842008X`.
URL `https://doi.org/10.1142/S012905411842008X`

[2] M. Holzer, M. Kutrib, Nondeterministic descriptional complexity of regular languages, International Journal of Foundations of Computer Science 14 (6) (2003) 1087–1102.

[3] M. Raskin, A superpolynomial lower bound for the size of non-deterministic complement of an unambiguous automaton, in: 45th International Colloquium on Automata, Languages, and Programming (ICALP 2018), Vol. 107 of Leibniz International Proceedings in Informatics (LIPIcs), 2018, pp. 138:1–138:11.

[4] T. Colcombet, Unambiguity in automata theory, in: 17th International Workshop on Descriptional Complexity of Formal Systems (DCFS), Vol. 9118 of Lecture Notes in Computer Science, Springer, 2015, pp. 3–18.