Scientific Annals of Computer Science vol. 33 (1), 2023, pp. 53–77

doi: 10.7561/SACS.2023.1.53

Maximal Existential and Universal Width[1]

Casey KEELER[2] and Kai SALOMAA[2]

Abstract

The tree width of an alternating finite automaton (AFA) measures the parallelism in all computations of the AFA on a given input. The maximal existential (respectively, universal) width of an AFA A on string w measures the maximal number of existential choices (respectively, of parallel universal branches) in one computation of A on w.

We give polynomial time algorithms deciding finiteness of an AFA's tree width and maximal universal width. Also we give a polynomial time algorithm that for an AFA A with finite maximal universal width decides whether or not the maximal existential width of A is finite. Finiteness of maximal existential width is decidable in the general case but the algorithm uses exponential time. Additionally, we establish necessary and sufficient conditions for an AFA to have exponential tree width growth rate, as well as sufficient conditions for an AFA to have exponential maximal existential width or exponential maximal universal width.

Keywords: finite automaton, alternation, universal and existential choices, decision problems

---

[1]An extended abstract of a preliminary version of this paper appeared in the proceedings of the 23rd International Conference Descriptional Complexity of Formal Systems, DCFS'21, Yonsei University, Republic of Korea, June 2021.

[2]School of Computing, Queen's University, Kingston, Ontario K7L 2N8, Canada, Email: `{ 1ck7, salomaa }@queensu.ca`

1 Introduction

An alternating finite automaton (AFA) extends a nondeterministic finite automaton by allowing both existential (i.e., nondeterministic) choices and universal choices [1, 13]. Alternating finite automata recognize only the regular languages but can be double exponentially more succinct than deterministic finite automata [1].

Kintala and Wotschke [12] initiated the study of measures of nondeterminism for finite automata. Limited nondeterminism and limited ambiguity of finite automata has received much attention since then [4, 8, 15, 17, 19], see also the surveys [3, 6].

This paper quantifies existential and universal parallelism in AFAs. Existential (respectively, maximal existential) width of an AFA A on string w counts the number (respectively, the maximal number) of existential choices not followed by A in one computation on w. The universal width and maximal universal width of an AFA are defined similarly. The measures for the number of existential and universal choices in computations of an AFA were previously considered by the authors [10].

Tree width (a.k.a. leaf size) [8, 11, 16] is a commonly used nondeterminism measure for nondeterministic finite automata (NFA) and the definition extends naturally for AFA. Roughly speaking, the tree width of an NFA measures the nondeterminism of all possible computations on a given input. On the other hand, existential width measures the number of existential (or nondeterministic) choices on a particular computation path and, in this sense, is similar to the guessing or branching measures [4], although the details of the definitions are not the same. Similarly, universal width measures the amount of parallelism in one existentially chosen computation of an AFA.

We consider the decision question whether the maximal existential or universal width of an AFA is bounded and want to develop efficient algorithms for this question. With this goal in mind we develop a widget characterization of AFAs with finite maximal universal width and a set of widgets that guarantee that the maximal existential width is finite. We give a polynomial time algorithm to decide (i) finiteness of maximal universal width, and, (ii) finiteness of maximal existential width under the assumption that maximal universal width is finite. We give an algorithm to decide finiteness of maximal existential width also in the general case but the algorithm requires exponential time.

We note the following concerning the AFA model used here. While the most general definition of alternating finite automata [1, 7, 14] allows general Boolean functions on the states, since we want to measure existential and universal choices separately in a computation we divide the states into existential states (corresponding to disjunction) and universal states (corresponding to conjunction). This alternating finite automaton model is used also e.g. by Geffert [2] when considering alternation hierarchies, and the original definition of alternating Turing machines [1] divides states into existential and universal states. In this paper we focus on the worst case variants of the existential and universal width, called maximal width measure. The “best case” existential (respectively, universal) width of an AFA A on string w is the existential width (universal width) of the “best” computation of A on w [10]. Decision problems for the “best” case existential and universal width measure are considered in [5].

2 Preliminaries

We assume that the reader is familiar with basics of finite automata. For more information see e.g. the textbook [18] or the survey [7]. In this section we fix notations and recall some basic definitions on alternating finite automata.

In the following Σ is always a finite alphabet, the set of strings over Σ is Σ^* and ε is the empty string. Let $f(\ell) : \mathbb{N} \to \mathbb{N}$ be a function. If $f(\ell) \in \Theta(\ell^d)$ (for some $d \in \mathbb{N}$), we say that $f(\ell)$ has *polynomial growth degree d*. If $f(\ell) \in 2^{\Theta(\ell)}$, we say that $f(\ell)$ has *exponential growth*.

An *alternating finite automaton* (AFA) is defined as an extension of a nondeterministic finite automaton (NFA) that allows also universal states. Formally, an AFA is a 6-tuple, $A = (Q_e, Q_u, \Sigma, \delta, q_0, F)$ where Q_e is a finite set of *existential states,* Q_u is a finite set of *universal states* ($Q_e \cap Q_u = \emptyset$), Σ is the input alphabet, $\delta : (Q_e \cup Q_u) \times \Sigma \to 2^{Q_e \cup Q_u}$ is the transition function, $q_0 \in Q_e \cup Q_u$ is the initial state, and $F \subseteq Q_e \cup Q_u$ is the set of final states.

For $q \in Q_e \cup Q_u$, we denote $A_q = (Q_e, Q_u, \Sigma, \delta, q, F)$, that is, A_q is obtained from A by changing the initial state to be q. For strings over Σ, membership in the language $L(A_q)$ is defined inductively as follows. First $\varepsilon \in L(A_q)$ if and only if $q \in F$. Consider $a \in \Sigma$ where $\delta(q, a) = \{p_1, \ldots, p_n\}$ for $n \geq 1$. For $w \in \Sigma^*$, define:

- If $q \in Q_u$, then $aw \in L(A_q)$ if and only if $w \in L(A_{p_i})$ for all $1 \leq i \leq n$.
- If $q \in Q_e$, then $aw \in L(A_q)$ if and only if $w \in L(A_{p_i})$ for some $1 \leq i \leq n$.

If $\delta(q,a)=\emptyset$, then $aw \notin L(A_q)$. The language of the AFA A is defined as $L(A)=L(A_{q_0})$.

In the following, $A=(Q_e,Q_u,\Sigma,\delta,q_0,F)$ is always an AFA. Unless otherwise mentioned, we assume that all states of A are reachable from the initial state q_0.

A *computation tree* of A is a tree structure whose internal nodes are labeled by pairs (p,a), for $p \in Q_e \cup Q_u$, $a \in \Sigma$ (that is, each internal node is labeled by a state and a character), and whose leaves are labeled by (p,ε) or the fail symbol $\perp$. We call a node of the computation tree T labeled by (p,a) a p-node of T, and the leaves of T labeled by (p,ε) are called *state leaves*.

The computation tree of A on a string $w \in \Sigma^*$ from state $q \in Q_e \cup Q_u$, $T_{A,q,w}$, is defined inductively as follows. As the base case, $T_{A,q,\varepsilon}$ is the singleton tree where the only node is labeled (q,ε). For $c \in \Sigma$ and $v \in \Sigma^*$, $T_{A,q,cv}$, is defined inductively as the tree where:

- the root is labeled by (q,c), and,
- the trees rooted at the children of (q,c) are
  - the computation trees $T_{A,p_1,v},\ldots,T_{A,p_n,v}$, if $\delta(q,c)=\{p_1,\ldots,p_n\}$
  - the root has a single child labeled by the failure symbol $\perp$ if $\delta(q,c)=\emptyset$.

The computation tree of A on w starting from q_0 (the initial state of A), $T_{A,q_0,w}$, is denoted simply as $T_{A,w}$.

If A is an NFA, the definition yields the computation trees as considered in [8, 16], since an NFA can be seen as an AFA with no universal states.

Next we define the *pruning operation* on computation trees. A pruned computation tree represents one particular computation of an AFA. For $q \in Q_e \cup Q_u$, the pruned computation tree of $T_{A,q,\varepsilon}$ is the singleton node (q,ε). For a string cv, where $c \in \Sigma$ and $v \in \Sigma^*$, a pruned computation tree of A on cv from $q \in Q_e \cup Q_u$ is obtained recursively from $T_{A,q,cv}$, where $\delta(q,c)=\{p_1,\ldots,p_k\}$, as follows:

i) If q is an existential state, then replace $k-1$ of the immediate subtrees by a singleton tree consisting of a node labeled by a new symbol ψ (representing a pruning of that branch), and the final child $T_{A,p_i,v}$, $1 \le i \le k$, by a pruned computation tree of $T_{A,p_i,v}$.

ii) If q is a universal state, then replace each immediate subtree $T_{A,p_i,v}$ by a pruned computation tree of $T_{A,p_i,v}$, for all $1 \le i \le k$.

We note that a pruned computation tree represents one specific "run" of an AFA. The number of existential ψ-leaves in a pruned computation tree measures the number of existential choices in that computation (strictly, speaking it will be the number of existential choices that are not followed).

For a computation tree T, we use $\text{stateLeaves}(T)$ to denote the multiset of all leaves of T labeled by a state-ε pair, and $\text{failLeaves}(T)$ to denote the multiset of leaves of T labeled by the fail symbol $\perp$. In a pruned tree we call leaves labeled by ψ "*cut leaves*", and use $\text{cutLeaves}(T)$ to denote the multiset of all cut leaves in a pruned tree T.

The set of all pruned trees of a computation tree T is denoted ✂(T). A pruned computation tree is *accepting* if all of its state-ε leaves are labeled by accepting states, and no leaves are labeled by the fail symbol $\perp$. We denote the set of all accepting pruned computations of a tree T as ✂$^{\text{acc}}(T)$. Directly from the definition of the language of an AFA and the definition of pruned computation trees, it follows that a string w is in $L(A)$ if and only if ✂$^{\text{acc}}(T_{A,w}) \neq \emptyset$.

By the *skeleton* of an AFA A we mean the NFA obtained from A by interpreting all states to be existential. Formally, for an AFA $A = (Q_e, Q_u, \Sigma, \delta, q_0, F)$, we define the skeleton of A as the NFA $A' = (Q_e \cup Q_u, \emptyset, \Sigma, \delta, q_0, F)$. That is, the skeleton of an AFA has the same state set and transition structure, except all of the states are existential. We note that the language of the skeleton of an AFA A is not usually the same as the language of A, however, for any string w, $T_{A,w} = T_{A',w}$, that is, the computation tree of A coincides with the computation tree of A'.

Let $q \in Q_e \cup Q_u$ and $b \in \Sigma$. If $\delta(q, b)$ is a singleton set, then for purposes of the alternating computation it does not make a difference whether the state q is existential or universal.

For purposes of the width measures we consider, often the relevant cases are when $\delta(q, b)$ has more than one element. For $q \in Q_e$ (respectively, $q \in Q_u$) and $b \in \Sigma$, we say that the transition from q on b is *properly existential* (respectively, *properly universal*) if $|\delta(q, b)| \geq 2$.

2.1 Width Measures of Alternating Finite Automata

The *tree width* of an AFA A on a string w, denoted $\text{tw}(A, w)$, is the number of state leaves and fail symbols in the computation tree $T_{A,w}$ [10]. If A is an NFA, then the definition coincides with the tree width (or leaf size) for NFAs [8, 16].

The tree width of A on string w measures, roughly speaking, the amount of the existential and universal choices in all branches of the computation of A on w. One computation of A corresponds to a pruned tree and we define universal and existential width in terms of a pruned tree.

Definition 1 *Consider a pruned computation tree T^p of an AFA. The* universal width *of T^p, denoted* $\mathrm{uw}(T^p)$*, is the number of state leaves and fail leaves in T^p.*

The leaves of a pruned tree may be labeled by states or the fail symbol. Universal width measures the amount of universal branching in a pruned tree and counts leaves labeled both by states and the fail symbol, that is,

$$\mathrm{uw}(T^p) = |\mathrm{stateLeaves}(T^p)| + |\mathrm{failLeaves}(T^p)|.$$

Next we extend the notion of universal width for strings.

Definition 2 *For an AFA A and a string $w \in \Sigma^*$, the* maximal universal width *of A on w, denoted* $\mathrm{uw}^{\max}(A, w)$*, is the greatest number of leaves in any pruned computation tree of $T_{A,w}$. Formally, this is*

$$\mathrm{uw}^{\max}(A, w) = \max\{\mathrm{uw}(T^p) \mid T^p \in ✂(T_{A,w})\}.$$

Intuitively, the maximal universal width of an AFA A on a string w measures the largest amount of parallelism that can occur in a pruned computation tree of A on w.

We measure the number of existential choices present in an alternating computation by counting the number of cut-leaves in a pruned tree.

Definition 3 *Consider a pruned computation tree T^p of an AFA. The* existential width *of T^p, denoted* $\mathrm{ew}(T^p)$*, is the number of leaves of T^p labeled by the cut-symbol ψ.*

Formally, for a pruned tree T^p, $\mathrm{ew}(T^p) = |\mathrm{cutLeaves}(T^p)|$. Again, we extend the measure for strings.

Definition 4 *For an AFA A and a string $w \in \Sigma^*$, the* maximal existential width *of A on w, denoted* $\mathrm{ew}^{\max}(A, w)$*, is the largest number of leaves labeled by the symbol ψ in a pruned computation tree of $T_{A,w}$. Formally, this is*

$$\mathrm{ew}^{\max}(A, w) = \max\{\mathrm{ew}(T^p) \mid T^p \in ✂(T_{A,w})\}.$$

The maximal existential width of an AFA on a string w measures, roughly speaking, the largest number of existential branches that are not followed in an alternating computation of A on w. According to this definition the existential width of a deterministic finite automaton is zero.

If an AFA does not have any existential branching, then no branches are removed during the pruning operation. This means that there will be exactly one pruned computation tree for any string, and the maximal universal width and tree width will coincide.

Lemma 1 *Let $A = (Q_e, Q_u, \Sigma, \delta, q_0, F)$ be an AFA such that $|\delta(q, a)| \leq 1$ for all $q \in Q_e$ and $a \in \Sigma$. Then for all $w \in \Sigma^*$, we have* ✂$(T_{A,w}) = \{T_{A,w}\}$*,* $\mathrm{uw}^{\max}(A, w) = \mathrm{tw}(A, w)$ *and* $\mathrm{ew}^{\max}(A, w) = 0$*.*

To consider the growth rates of the measures as a function of input length, we extend the tree width, maximal universal width, and maximal existential width functions as functions over the natural numbers in the normal manner. For $f \in \{\mathrm{tw}, \mathrm{uw}^{\max}, \mathrm{ew}^{\max}\}$:

$$f(A, \ell) = \max\{f(A, w) \mid w \in \Sigma^\ell\}, \text{ and } f(A) = \sup_{\ell \in \mathbb{N}}\{f(A, \ell)\}.$$

The value $\mathrm{ew}^{\max}(A)$ (respectively, $\mathrm{uw}^{\max}(A)$) is finite if and only if the function $\mathrm{ew}^{\max}(A, \ell)$ (respectively, $\mathrm{uw}^{\max}(A, \ell)$) is bounded.

3 Finite Universal and Existential Width

We want to characterize AFAs that have, respectively, finite tree width, finite maximal universal width or finite maximal existential width. In the following, we use the term widget to describe, roughly speaking, subgraphs of the state graph of an AFA. Similar terminology was used earlier in the study of ambiguity and tree width of NFAs [11, 19].

By a *widget* of an AFA $A = (Q_e, Q_u, \Sigma, \delta, q_0, F)$ we mean a subgraph of the state graph of A where the vertices are labeled by elements of $Q_e \cup Q_u$ and the directed edges are labeled by elements of Σ. As a short hand notation when drawing widgets we allow edges to be labeled by a string $w \in \Sigma^*$: this represents a path where the individual transitions spell out w. For example, in the widget represented by Figure 1, the state (q, u) is connected to itself by a cycle where the transitions are labeled by elements of the string av. A more detailed description of widgets can be found in [9].

In our figures, if a state q is universal (respectively, existential) then it is labeled (q, u) (respectively, (q, e)). If a state is labeled e/u, then it can be either universal or existential.

Since the tree width of an AFA is defined over *unpruned* computation trees, directly from the definitions it follows that $\mathrm{ew}^{\max}(A, w), \mathrm{uw}^{\max}(A, w) \leq \mathrm{tw}(A, w)$ for any string w.

Since only the properly universal transitions increase universal width, using analogous reasoning as the characterization of NFAs with infinite tree width [11, 16], we see that $\mathrm{uw}^{\max}(A)$ can be infinite only if a properly universal transition of A is involved in a cycle. Recalling that we assume all states of an AFA to be reachable, this gives the following characterization.

Theorem 1 *An AFA A has infinite maximal universal width if and only if A has a widget* (IUW)*, as shown in Figure 1.*

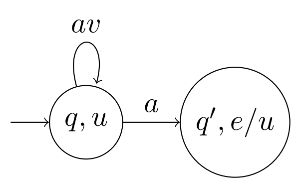


Figure 1: Widget (IUW), for $a \in \Sigma, v \in \Sigma^*$ [10]

Characterizing finite existential width is not equally simple because the number of cut-leaves can grow unboundedly due to certain structures on existential states *and/or* universal states.

Lemma 2 *Let $A = (Q_e, Q_u, \Sigma, \delta, q_0, F)$ be an AFA where $\mathrm{ew}^{\max}(A, \ell) \notin O(1)$. Then either*

i) A has a cycle containing a properly existential transition, or

ii) For some strings $u, v, w \in \Sigma^*$, characters $a, b \in \Sigma$, and states $q \in Q_u$, $q_1, q_2 \in Q_e \cup Q_u$, and $p \in Q_e$, we have: $q \in \delta(q_0, u)$, $\{q_1, q_2\} \subseteq \delta(q, a)$, $q \in \delta(q_1, v)$, $p \in \delta(q_2, w)$, $|\delta(p, b)| \geq 2$, and $\mathrm{ew}(T_{A,u(av)^i wb}) < \mathrm{ew}(T_{A,u(av)^{i+1} wb})$ for all $i \geq 1$.

Proof: Suppose that A has m states. Let x be the largest number of transition choices of any existential state, minus one. That is, $x =$

$\max_{q \in Q_e, a \in \Sigma} \{|\delta(q, a)|\} - 1$. Similarly, let y be the largest number of choices of any universal transition. That is, $y = \max_{q \in Q_u, a \in \Sigma} \{|\delta(q, a)|\}$.

If some computation tree of A contains a branch with at least $m + 1$ nodes with cut leaves, then A has a cycle containing a properly existential transition, and case i) holds.

In the following then, we assume that for any computation tree, each branch has at most m nodes having cut leaves as children. Since $\mathrm{ew}^{\max}(A, \ell)$ is unbounded, there must be a pruned tree T^p such that

$$\mathrm{ew}(T^p) > y^m \cdot x \cdot m. \tag{1}$$

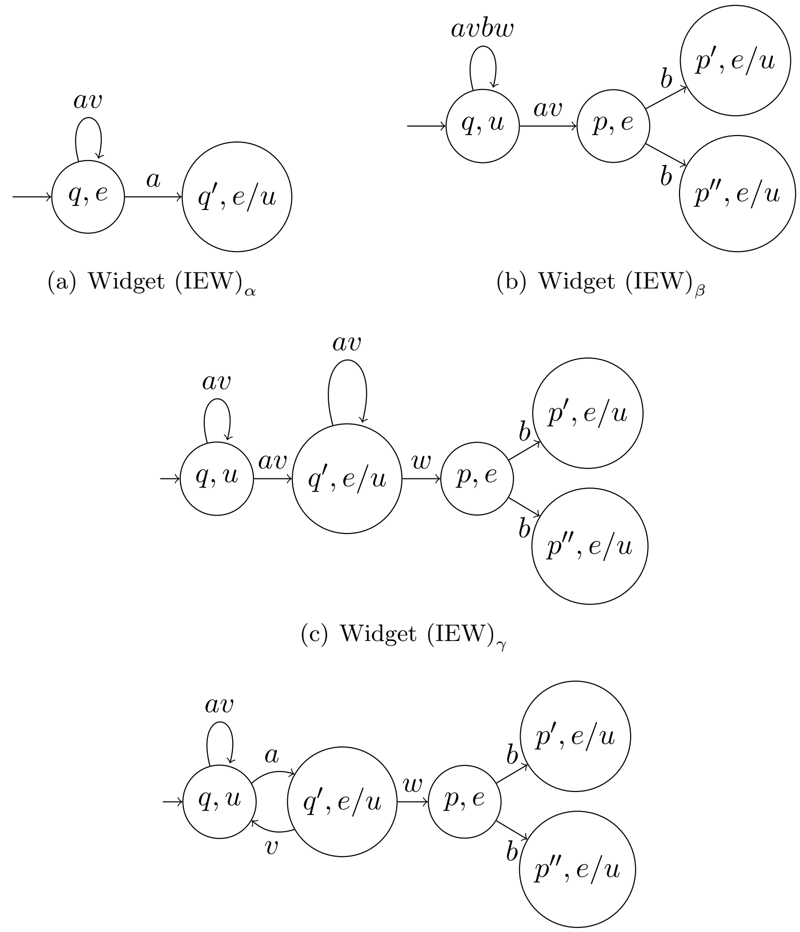
Since a single branch contains at most m nodes with cut leaves, at most $x \cdot m$ cut leaves can be connected to a single branch. By (1), T^p must contain more than y^m branches, where any two distinct branches contain cut leaves also after the two branches separate. Note that, if the cut leaves appeared above the universal branching, then $\mathrm{ew}(T^p)$ wouldn't be able to reach $y^m \cdot x \cdot m$ because at most $x \cdot m$ cut leaves can be connected to the same branch. To produce more than y^m universal branches that all eventually lead to a cut leaf, there must be a branch that has more than m universal branching points. That is, a universal node must repeat in some branch, and A must have a cycle containing a properly universal transition. Since we are only counting branches which eventually lead to a cut leaf, the universal choice has the property that *after exiting the cycle*, and possible further symbols, it leads to a cut leaf. □

In the converse direction, the following lemma gives a number of sufficient conditions to cause the maximal existential width of an AFA to be infinite.

Lemma 3 *Let $A = (Q_e, Q_u, \Sigma, \delta, q_0, F)$ be an AFA with at least one of the widgets from Figure 2. Then* $\mathrm{ew}^{\max}(A, \ell) \notin O(1)$.

Proof: If A has an $(\mathrm{IEW})_\alpha$ widget, then we have a cycle of the form (q, av, q), for some $q \in Q_e, a \in \Sigma, v \in \Sigma^*$. Each time we repeat the cycle (q, av, q), we are adding at least one cut leaf to the pruned computation tree. Since all states of A are reachable, there exists some prefix $u \in \Sigma^*$ such that $q \in \delta(q_0, u)$, and then $\mathrm{ew}^{\max}(A, u(av)^i) < \mathrm{ew}^{\max}(A, u(av)^{i+1})$ for all $i \geq 0$.

If we have an $(\mathrm{IEW})_\beta$ widget, then we have a cycle of the form $(q, avbw, q)$ for some $q \in Q_u, a, b \in \Sigma, v, w \in \Sigma^*$. Note that we do not



(a) Widget $(\mathrm{IEW})_\alpha$

(b) Widget $(\mathrm{IEW})_\beta$

(c) Widget $(\mathrm{IEW})_\gamma$

(d) Widget $(\mathrm{IEW})_\zeta$

Figure 2: Widgets causing infinite $\mathrm{ew}^{\max}(A)$, where $a, b \in \Sigma, v, w \in \Sigma^*$

necessarily need $a \neq b$, but in the case that state q has a self-loop on character a, then the simple cycle is (q, a, q), not (q, aa, q) or (q, ab, q). There is some prefix string $u \in \Sigma^*$ such that $q \in \delta(q_0, u)$. Each time we repeat the cycle $(q, avbw, q)$, we are adding at least one universal branch. Each of these universal branches leads to at least one additional cut leaf on the pruned computation tree. So then $\mathrm{ew}^{\max}(A, u(avbw)^i) < \mathrm{ew}^{\max}(A, u(avbw)^{i+1})$ for all $i \geq 1$. We note that avb must be a prefix of $avbw$ to have both universal branches continue at least until the b is read, but that there can be a suffix w

on the cycle because the existential branch on b on the other branch has already been passed. This means that the existence of an $(\mathrm{IEW})_\beta$ widget causes that $\mathrm{ew}^{\max}(A, \ell) \notin O(1)$.

If we have an $(\mathrm{IEW})_\gamma$ widget, then we have a cycle (q, av, q), a cycle (q', av, q'), where also $q' \in \delta(q, av)$, and there exists a state $p \in \delta(q', w)$ with an outgoing properly existential transition on $b \in \Sigma$. There is some prefix string $u \in \Sigma^*$ such that $q \in \delta(q_0, u)$. Each time we repeat the cycle (q, av, q), we are adding at least one universal branch. That is, a leaf node labeled by q will expand to at least two universal branches, one with a leaf node labeled by q and the other with a leaf node labeled by q'. So then $T_{A,u(av)^i}$ will have i universal branches with nodes labeled by q'. This means that $T_{A,u(av)^{i+1}w}$ will have at least one more universal branch compared to $T_{A,u(av)^i w}$. When reading the symbol b this universal branch encounters an existential choice and, thus, $\mathrm{ew}^{\max}(A, u(av)^i wb) < \mathrm{ew}^{\max}(A, u(av)^{i+1} wb)$ for all $i \geq 0$.

If we have an $(\mathrm{IEW})_\zeta$ widget, then we have a properly universal transition from state q such that there are two distinct cycles $C_1 = (q, av, q)$ and $C_2 = (q, av, q)$, for $a \in \Sigma, v \in \Sigma^*$. Furthermore, some $q' \in \delta(q, a)$ is able to reach a state p with a properly existential transition on $b \in \Sigma$. There is some prefix string $u \in \Sigma^*$ such that $q \in \delta(q_0, u)$. For a tree $T_{A,u}$, there is at least one universal branch with a leaf node labeled by q, and each repetition of av causes q to expand into two universal branches with leaf nodes labeled by q and q'. This means that $T_{A,u(av)^{i+1}}$ will have more universal branches with leaves labeled by q' compared to $T_{A,u(av)^i}$. When reading wb from q' the computation encounters an existential choice and therefore $\mathrm{ew}^{\max}(A, u(av)^i wb) < \mathrm{ew}^{\max}(A, u(av)^{i+1} wb)$ for all $i \geq 0$.

In every case, we get that $\mathrm{ew}^{\max}(A, \ell) \notin O(1)$. □

It remains open whether an AFA A having one of the widgets represented in Figure 2 is a necessary condition for $\mathrm{ew}^{\max}(A)$ to be infinite. In the next section we give an efficient algorithm to decide finiteness of $\mathrm{ew}^{\max}(A)$ in the case when maximal universal width of A is finite. Without assumptions on maximal universal width, we can decide finiteness of $\mathrm{ew}^{\max}(A)$ but the algorithm uses exponential time. If an AFA A having one of the widgets of Figure 2 is both necessary and sufficient for A to have infinite maximal existential width, this could yield a polynomial time algorithm to decide the property.

4 An Algorithm for Deciding Finiteness

The tree width of an NFA is unbounded if and only if there exists a nondeterministic transition involved in a cycle [8, 16]. This yields a low polynomial time algorithm to decide whether or not the tree width of an m-state NFA is finite [11]. Since the tree width of an AFA and the tree width of its skeleton automaton are the same, deciding finiteness of an AFA's tree width has the same complexity upper bound. Since the structure of (IUW) widgets matches closely to the structure of the widgets causing unbounded tree width in NFAs, we can also decide finiteness of an AFA's maximal universal width using the same algorithm [11].

Proposition 1 *Let A be an m-state AFA over a fixed alphabet Σ.*[3] *We can decide whether or not* $\mathrm{tw}(A,\ell) \in O(1)$ *and whether or not* $\mathrm{uw}^{\max}(A,\ell) \in O(1)$ *in time $O(m^4)$.*

On the other hand, unbounded maximal existential width in AFAs can result from a number of different structures (cf. Figure 2), and consequently we cannot use the same algorithm without modification.

The interrelationship of existential and universal width is illustrated also by considering bounds for the largest finite existential width of an AFA with a given number of states. We know that, if the maximal existential width of an m-state NFA is finite, then it is at most $\frac{(m-1)\cdot(m-2)}{2}$ [16]. However, in AFAs the addition of universal branching allows the finite maximal existential width to be exponential as a function of the number of states. More specifically, for $m \geq 6$ there exist AFAs A_m such that $\mathrm{ew}^{\max}(A_m) = 5 \cdot 2^{m-5}$ [10]. Figure 3 depicts a 7-state AFA meeting this lower bound. The construction requires at least 1 universal state at the beginning of the chain, followed by 5 existential states, where all states are maximally connected. Note that, strictly speaking, the final two states can be existential or universal, since they both have exactly one outgoing transition on a given symbol. We believe this to be the greatest finite maximal existential width among m-state AFAs but do not have a proof for the claim.

Conjecture 1 *Let A be an m-state AFA such that* $\mathrm{ew}^{\max}(A)$ *is finite. Then* $\mathrm{ew}^{\max}(A) \leq 5 \cdot 2^{m-5}$.

Now we continue with the question of deciding finiteness of maximal existential width of an AFA. The following lemma characterizes an AFA A

[3]That is, the decision algorithm considers the size of the alphabet to be a constant.

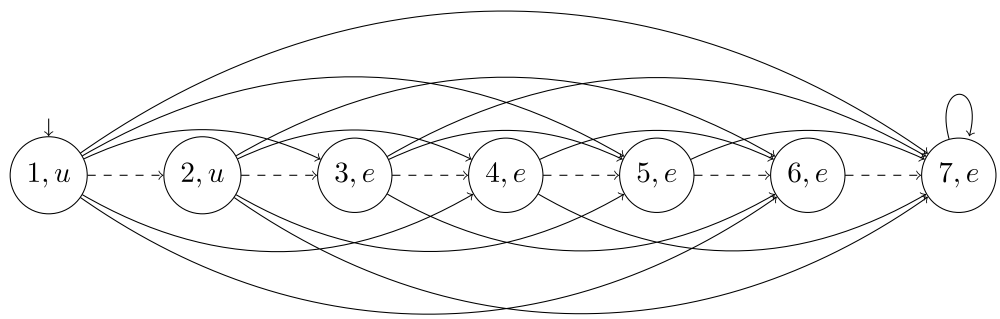


Figure 3: An AFA with 2 universal states, 5 existential states, and a maximal existential width of $5 \cdot 2^2$. Dashed edges are labeled by both a and b, and solid edges are labeled by a.

having infinite tree width in terms of A having infinite maximal existential width or infinite maximal universal width.

Lemma 4 *Let A be an AFA. The tree width of A is infinite if and only if* $\mathrm{ew}^{\max}(A)$ *is infinite or* $\mathrm{uw}^{\max}(A)$ *is infinite.*

Proof: If $\mathrm{tw}(A)$ is infinite then also the tree width of the skeleton automaton A' of A is infinite, and the NFA A' must have a nondeterministic transition from a state q occurring in a cycle [11]. If q is a universal state of A, then A has a widget (IUW) and has infinite maximal universal width. If q is an existential state in A, then A has a widget $(\mathrm{IEW})_\alpha$ and has infinite maximal existential width.

We note that since existential and universal width are defined in terms of pruned computation trees, for any string w, $\mathrm{ew}^{\max}(A, w) \leq \mathrm{tw}(A, w)$ and $\mathrm{uw}^{\max}(A, w) \leq \mathrm{tw}(A, w)$. This gives the converse implication. $\square$

Now we get a polynomial time algorithm for deciding finiteness of an AFA's maximal existential width, provided that the number of universal branches in a computation is guaranteed to be bounded.

Theorem 2 *Let A be an m-state AFA such that* $\mathrm{uw}^{\max}(A)$ *is finite. Then we can decide in time $O(m^4)$ whether or not* $\mathrm{ew}^{\max}(A)$ *is finite.*

Proof: Since maximal universal width of A is finite, by Lemma 4, $\mathrm{ew}^{\max}(A)$ is finite if and only if $\mathrm{tw}(A)$ is finite. Proposition 1 provides an algorithm with the required time bound to decide finiteness of $\mathrm{tw}(A)$. $\square$

4.1 Decidability of Finiteness of Maximal Existential Width

Since we don't know whether the widgets of Lemma 3 exactly characterize AFAs with infinite maximal existential width, we don't have an efficient algorithm to decide this property. In this subsection we show that finiteness of maximal existential width is at least decidable albeit the algorithm is based on subset construction and uses exponential time. The following definition provides in Lemma 5 below an "if and only if" condition for an AFA to have infinite maximal existential width.

Definition 5 *Let A be an AFA and consider a pruned computation tree T^p of A on input w. The ith level of T^p, $i = 0, 1, \ldots$ consists of nodes reached by reading a prefix of w of length i.*

We say that level i of T^p increases the existential width *if either*

(i) a transition on level i is properly existential, or,

(ii) level i contains a universal computation step where the subcomputations starting from at least two of the children both use a properly existential transition.

The correctness of the algorithm will be based on the following observation. Note that the number of levels that (according to Definition 5) increase the existential width of a pruned tree T^p does not determine the existential width of T^p and the following lemma just gives a necessary and sufficient condition for the existential width of an AFA to be infinite.

Lemma 5 *The maximal existential width of an AFA A is infinite if and only if for all $m \in \mathbb{N}$ there exists a pruned computation tree of A where at least m levels increase the existential width of the tree.*

Proof: Let $\max_A$ be the maximum number of existential or universal choices in one computation step of A. If in all pruned trees at most k_A levels increase the existential width, any pruned computation tree of A has existential width at most $(\mathrm{max_A})^{k_A}$. Conversely, if for all $m \in \mathbb{N}$ there exists a pruned computation tree where at least m levels increase the existential width, $\mathrm{ew}^{\max}(A)$ must be greater than m for all $m \in \mathbb{N}$. □

In the following, for an AFA A we construct an NFA B that keeps track of all the states of A occurring at the current level of a pruned computation tree of A and, when simulating a universal step of A, the

NFA B nondeterministically guesses which subcomputations will later use a properly existential transition. The acceptance conditions of B verify that the guesses have been correct.

More formally, consider an AFA $A = (Q_e, Q_u, \Sigma, \delta, q_0, F)$ and denote $Q = Q_e \cup Q_u$, and $Q^{\text{wait}} = \{q^{\text{wait}} \mid q \in Q\}$. ($Q^{\text{wait}}$ is a marked copy of Q.)

The state set of the NFA B is

$$P = 2^{Q \cup Q^{\text{wait}} \cup \{p_{\text{sink}}, p_{\text{fail}}\}}$$

and the set of final states of B is $F_B = \{X \in P \mid X \subseteq Q \cup \{p_{\text{sink}}\}\}$. Since maximal existential width of A is defined based on arbitrary (not necessarily accepting) pruned trees, the computation of B does not verify that the simulated computation of A accepts and the final states of B do not depend on final states of A. In fact, accepting states of B may contain also the symbol p_{sink} that represents failure of the computation of A. The purpose of the final states of B is just to verify that in a universal step that is guessed to increase the existential width in the sense of Definition 5 (ii), the appropriate subcomputations really contain an existential transition and the failure state p_{fail} indicates a violation of this condition.

The transition relation γ_B of B is defined for $b \in \Sigma$ and $X = \{x_1, \ldots, x_m\}$ belonging in P as follows. The set $\gamma(X, b)$ consists of all sets

$$\{\phi(x_1, b), \ldots, \phi(x_m, b)\}$$

where each $\phi(x_i, b)$ is a sequence of elements of $Q \cup Q^{\text{wait}} \cup \{p_{\text{sink}}, p_{\text{fail}}\}$ selected nondeterministically as follows:

(i) If $x_i \in Q_e$, then if $\delta(x_i, b) \neq \emptyset$, $\phi(x_i, b)$ is one of the elements of $\delta(x_i, b)$ and otherwise $\phi(x_i, b) = p_{\text{sink}}$.

(ii) If $x_i \in Q_u$ and $\delta(x_i, b) = \{q_1, \ldots, q_k\}$, $k \geq 1$, then $\phi(x_i, b)$ can be any sequence

$$q'_1, \ldots, q'_k, \quad \text{where } q'_j = q_j \text{ or } q'_j = q_j^{\text{wait}}, \;\; j = 1, \ldots, k.$$

If $\delta(x_i, b) = \emptyset$, then $\phi(x_i, b) = p_{\text{sink}}$.

(iii) If $x_i = q^{\text{wait}}$, $q \in Q_e$ and $\delta(q, b) = \{q_1, \ldots, q_k\}$ where $k \geq 2$, then $\phi(x_i, b)$ is one of the elements q_j, $1 \leq j \leq k$.

(iv) If $x_i = q^{\text{wait}}$, $q \in Q_e$ and $\delta(q, b) = \{q_1\}$, then $\phi(x_i, b) = q_1^{\text{wait}}$.

(v) If $x_i = q^{\text{wait}}$, $q \in Q_e \cup Q_u$ and $\delta(q, b) = \emptyset$, then $\phi(x_i, b) = p_{\text{fail}}$.

(vi) If $x_i = q^{\text{wait}}$, $q \in Q_u$ and $\delta(q, b) = \{p_1, \ldots, p_k\}$, $k \geq 1$, then $\phi(x_i, b)$ can be any sequence

$$q'_1, \ldots, q'_k, \quad \text{where } q'_j = q_j \text{ or } q'_j = q_j^{\text{wait}}, \;\; j = 1, \ldots, k, \text{ and } (\exists j) q'_j = q_j^{\text{wait}}.$$

(vii) If $x_i = p_{\text{sink}}$, then $\phi(x_i, b)$ is the empty sequence (that is, the transition of γ erases p_{sink} from the set of B).

(viii) If $x_i = p_{\text{fail}}$, then $\phi(x_i, b) = p_{\text{fail}}$ (that is, after p_{fail} appears in a state of B, the transitions of γ cannot remove it).

Roughly speaking, the NFA B operates as follows. States of B consist of sets of states of A where some states may be labeled as "wait". If B is in state X, and we ignore the wait-superscripts of the states of A in X, for each element of X that is existential in A, B simulates one existential choice of A and for a universal state of A, B simulates all choices. Assume B is in state $X \in P$ after reading a string w. If we ignore the "wait"-labels the elements of X are states of A that occur on the ith level, $i = |w|$, of a pruned computation tree of A. If the pruned tree contains the failure symbol $\perp$ on the ith level, this is represented in X by the element p_{sink}. Note that cut-symbols ψ are not stored in states of B.

The wait-superscripts introduced by transitions (ii) mark subcomputations that are required to simulate a properly existential transition of A before they can reach an accepting state in B. Note that only transitions (iii) remove the wait-superscript and "wait-elements" cannot occur in final states of F_B. If, according to (v), a state with wait-superscript encounters an undefined transition of A, the corresponding transition in B adds p_{fail} to the state set and, according to (viii), the element p_{fail} cannot be removed from a state of B.

The construction guarantees that a transition of B, occuring in an accepting computation of B, on a state $\{x_1, \ldots, x_m\}$ on input b simulates transitions of A that, according to Definition 5, increase the width of the pruned computation tree of A if and only if

INC(a) some $x_i \in Q_e$ and $|\delta(x_i, b)| \geq 2$, or,

INC(b) some $x_j \in Q_u$ and, according to (ii) in the definition of γ, at least two of the successor states are labeled "wait", or, INC(c) some $x_j \in Q_u^{\text{wait}}$ and, according to (vi) in the definition of γ, at least two of the successor states are labeled "wait".

Note that this "if and only if"-characterization requires that the resulting state of B eventually reaches a final state of B as the "wait"-superscripts can be erased only by simulating a properly existential transition of A.

Computations of the NFA B can have an unbounded number of transitions of types INC(a), INC(b) or INC(c) only if one of them occurs inside a cycle. Thus, by Lemma 5, to decide infiniteness of the existential width of A it is sufficent to check whether in B a transition of type INC(a), INC(b), or INC(c) occurs inside a cycle and, furthermore, that cycle is reachable from the start state and reaches a state of F_B. Roughly speaking, this corresponds to finding an occurrence of the widget $(\text{IEW})_\alpha$ in the NFA B and with the additional condition that the computation of B must eventually accept. This gives a decision algorithm for finiteness of existential width of A but the algorithm is not polynomial time because B is obtained from A by a modified subset construction.

Theorem 3 *For an AFA A it is decidable whether or not the existential width of A is finite.*

5 Growth Rate of the Measures

The tree width of an NFA is either finite, or it has polynomial or exponential growth rate [8]. The different NFA tree width growth rates can be characterized using widgets and this leads to efficient algorithms to decide the growth rate [11]. Since an AFA and its skeleton automaton have the same tree width, we can leverage existing NFA algorithms to decide the growth rate of an AFA's tree width.

Proposition 2 ([11]) *Let A be an m-state AFA over a fixed alphabet.*[4] *We can decide in $O(m^4)$ time whether $\text{tw}(A, \ell)$ is finite, or if it has polynomial or exponential growth.*

For AFAs with finite existential width, we can use the same algorithm, with slightly worse complexity, to decide the growth rate of maximal universal width. This is based on the following simulation result. In the following by an UFA we mean an AFA with only universal states.

[4] That is, all input AFAs for the algorithm have that same alphabet and when determining time complexity alphabet size is a constant.

Proposition 3 *Let A be an m-state AFA such that* $\mathrm{ew}^{\max}(A)$ *is finite. Then there exist $O(m^2)$ UFA's, $B_1, \ldots, B_z$, each having m states, such that $L(A)$ is the union of the languages $L(B_1), \ldots, L(B_z)$.*

Proof: Let $q_1, \ldots, q_k$ be the existential states of A. We define the "branching destinations" of each of these states as follows:

$$P_i = \bigcup_{a \in \Sigma, |\delta(q_i,a)|>1} \delta(q_i, a), \text{ for } 1 \leq i \leq k.$$

We note that $|P_i| \leq m - 1$ because A has finite maximal existential width and this means that no properly existential transition can be a self-loop. Let $\chi = P_1 \times P_2 \times \cdots \times P_k$, yielding all unique combinations by choosing one destination state from each properly existential transition. That is, χ is the set of all unique combinations of states obtained by choosing one state from each of $P_1, \ldots, P_k$. Since each P_i has at most $m - 1$ elements, $|\chi| \leq (m - 1) \cdot m$.

We can represent A's language as the union of $|\chi|$ languages, each recognized by an UFA with m states. We define UFAs $B_1, \ldots, B_{|\chi|}$, where each $B_j = (\emptyset, Q_e \cup Q_u, \Sigma, \delta_j, q_0, F)$, and the δ_j's are defined as follows: for each combination $1 \leq j \leq |\chi|$, which "keeps" destination states $\{p_{j,1}, \ldots, p_{j,k}\}$, and for each $a \in \Sigma$ we define

$$\delta_j(q, a) = \delta(q, a) \text{ if } q \notin \{q_1, \ldots, q_k\}, \text{or else}$$

$$\delta_j(q_i, a) = \begin{cases} \delta(q_i, a) & \text{if } |\delta(q_i, a)| = 1, \text{ or} \\ \{p_{j,i}\} & \text{if } p_{j,i} \in \delta(q_i, a), \text{ or} \\ \emptyset & \text{otherwise.} \end{cases}$$

That is, δ_j will have transitions leading from state q_i to state $p_{j,i}$ that simulate properly existential transitions from q_i, remove the original properly existential transitions from q_i, and δ_j has the original non-branching transitions leading out from state q_i.

Since all transitions in δ_j are present in the original transition function, then $L(B_j) \subseteq L(A)$, $j = 1, \ldots, k$. For all $w \in L(A)$, since each pruned computation tree of ✂$(T_{A,w})$ is represented across the B_j's, then at least one of the B_j's will accept w. So then we get the equality $L(A) = \bigcup_{j=1}^{|\chi|} L(B_j)$. □

Using Proposition 3 we get the following:

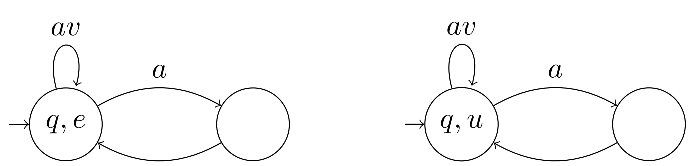
Theorem 4 *Let A be an m-state AFA such that* $\mathrm{ew}^{\max}(A)$ *is finite. Then we can decide whether the growth rate of* $\mathrm{uw}^{\max}(A, \ell)$ *is bounded, polynomial, or exponential in* $O(m^6)$ *time.*

Proof: According to Proposition 3 we can write $L(A)$ as the union of $O(m^2)$ languages recognized by m-state UFAs. For an UFA B and string w, $\mathrm{uw}^{\max}(B, w)$ is the same as tree width of B on w. According to Proposition 2 the tree width of B can be decided in time $O(m^4)$ and $\mathrm{uw}^{\max}(A, \ell)$ is bounded if all UFAs have bounded growth rate, $\mathrm{uw}^{\max}(A, \ell)$ grows exponentially if at least one of the UFAs has exponential growth rate and, otheriwse, $\mathrm{uw}^{\max}(A, \ell)$ grows polynomially. $\square$

5.1 Exponential Growth

The tree width of an NFA grows exponentially if and only if the NFA has a widget (ECOMP) [11]. Roughly speaking, this means that there exists a state involved in two cycles over the same string, see Figure 4(b).

Since the tree width of an AFA is the same as the tree width of it's skeleton automaton we can, essentially, characterize exponential growth of the tree width of an AFA using the same widget. However, in order to discuss existential and universal width separately, we define two new widgets, (EEW) and (EUW), as shown in Figure 4.



(a) Widget (EEW) (b) Widget (EUW)

Figure 4: Widgets for AFAs derived from (ECOMP), for $a \in \Sigma, v \in \Sigma^+$. In the figures the unlabeled states can be either existential or universal.

The following characterization is obtained by considering the skeleton automaton of the AFA A.

Theorem 5 *For an AFA A,* $\mathrm{tw}(A, \ell) \in 2^{\Theta(\ell)}$ *if and only if A has an* (EUW) *widget or A has an* (EEW) *widget.*

Since the maximal universal width counts the number of parallel branches, the presence of an (EUW) widget is sufficient to cause exponential growth.

Lemma 6 *Let A be an AFA with an* (EUW) *widget. Then*
$$\mathrm{uw}^{\max}(A, \ell) \in 2^{\Theta(\ell)}.$$

In addition to (EUW) widgets being sufficient to cause exponential maximal universal width, we also believe that they are necessary but do not have a complete proof for this conjecture. All AFAs with exponential universal width we have managed to construct have one of these widgets.

Conjecture 2 *Let A be an AFA such that $\mathrm{uw}^{\max}(A, \ell) \in 2^{\Theta(\ell)}$. Then A has an* (EUW) *widget.*

Even though (EEW) widgets are sufficient to cause exponential tree width, they do not cause exponential growth for the maximal existential width. Note that while tree width measures the parallelism of the entire computation tree, existential width measures choices only on one branch of the computation and thus, even for NFAs, tree width and existential width are very different measures. The following example considers the smallest AFA having widget (EEW).

Example 1 *Consider the AFA $A = (\{q_1, q_2\}, \emptyset, \{a\}, \delta, q_1, \{q_1\})$, where δ is defined as $\delta(q_1, a) = \{q_1, q_2\}$ and $\delta(q_2, a) = \{q_1\}$. In the computation tree T_{A,a^ℓ}, for any $\ell \in \mathbb{N}$, every non-leaf node labeled by q_1 will have two children, and every non-leaf node labeled by q_2 will have one child. Since there are no universal states in A, then any pruned tree $T^p \in$ ✂(T_{A,a^ℓ}) will have one branch. Since the branch consists of $\ell + 1$ nodes, and each node has at most 1 cut leaf attached to it, then $\mathrm{ew}(T_{A,a^\ell}) \leq \ell + 1$. This means that $\mathrm{ew}^{\max}(A, \ell) \in O(\ell)$.*

However, much like how $(\mathrm{IEW})_\alpha$ widgets are able to cause the maximal existential width to be unbounded (cf. Lemma 3), (EUW) widgets also have the potential to cause exponential growth for the maximal existential width, provided that the cycle has possibility for other existential choices. We recall the widgets of Figure 2, and note that the $(\mathrm{IEW})_\zeta$ widget has an (EUW) widget as part of its structure, followed by a properly existential transition. We use this observation to show that the existence of widgets $(\mathrm{IEW})_\zeta$ gives a sufficient condition for exponential growth rate.

Lemma 7 *Let A be an AFA with an* $(\mathrm{IEW})_\zeta$ *widget. Then*
$$\mathrm{ew}^{\max}(A,\ell) \in 2^{\Theta(\ell)}.$$

Proof: Suppose $A = (Q_e, Q_u, \Sigma, \delta, q_0, F)$ has an $(\mathrm{IEW})_\zeta$ widget. Using the notations of the widget from Figure 2, this means that there exists an (EUW) widget over a state $q \in Q_u$ and a string av, for $a \in \Sigma$ and $v \in \Sigma^+$. Furthermore, there exists a state $p \in Q_e$ such that $p \in \delta(q, aw)$ for some $w \in \Sigma^*$ and $|\delta(p,b)| \geq 2$ for some $b \in \Sigma$. Adding on repetitions of the cyclical string av will expand each leaf node of the tree labeled by q into two leaf nodes labeled by q. That is, $T_{A,u(av)^i}$ has at least 2^i universal branches, for $i \geq 1$, each with leaf nodes labeled by q. By extending the string from $u(av)^i$ to $u(av)^i awb$, it is clear that $T_{A,u(av)^i awb}$ has a pruned computation tree that after reading w reaches the state p and uses an existential transition when reading b. So then $\mathrm{ew}^{\max}(A, u(av)^i awb) \in 2^{\Theta(i)}$. □

To conclude this section we note the following. Recall that maximal existential width counts the cut-leaves in one pruned tree. This means that the number of cut-leaves can grow exponentially as a function of the height of the tree only if the number of universal branches grows exponentially as a function of the length of the input.

Corollary 1 *Let A be an AFA. If* $\mathrm{ew}^{\max}(A,\ell) \in 2^{\Theta(\ell)}$*, then*
$$\mathrm{uw}^{\max}(A,\ell) \in 2^{\Theta(\ell)}.$$

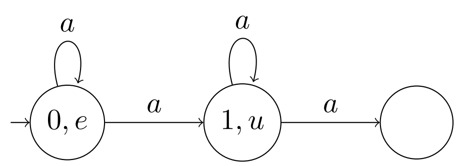
5.2 Polynomial Growth

By Theorem 1, the maximal universal width of an AFA is infinite if and only if the AFA has an (IUW) widget. Since the presence of an (IUW) widget forces at least linear growth, and any AFA with unbounded maximal universal width has an (IUW) widget, there can be no growth rates for an AFA's maximal universal width between finite and linear.

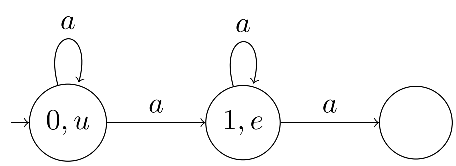
Corollary 2 *Let A be an AFA. If* $\mathrm{uw}^{\max}(A,\ell) \notin O(1)$*, then*
$$\mathrm{uw}^{\max}(A,\ell) \in \Omega(\ell).$$

If the tree width of an NFA is bounded by a polynomial, then we can decide what degree bounds the polynomial by, roughly speaking, determining how many successive $(\mathrm{IEW})_\alpha$ widgets over the same string can appear in a computation [11]. However, for AFAs this process is not so simple, as an (IUW) widget appearing before an $(\mathrm{IEW})_\alpha$ widget will increase the growth rate of the maximal existential width, but an $(\mathrm{IEW})_\alpha$ widget appearing before an (IUW) widget will *not* increase the growth rate of the maximal universal width. We demonstrate this difference in the following example.

Example 2 *In Figure 5, there are two AFAs, B_1 and B_2, each of which have an* (IUW) *widget and an* $(\mathrm{IEW})_\alpha$ *widget; the only difference is which one comes first. Figure 5(a) has linear growth rate for both the maximal universal width and the maximal existential width, whereas Figure 5(b)'s ordering of these widgets means that the number of pruned branches grows both because of the properly universal transition in the cycle* and *because of the properly existential transition.*



(a) $\mathrm{uw}^{\max}(B_1, \ell), \mathrm{ew}^{\max}(B_1, \ell) \in O(\ell)$



(b) $\mathrm{uw}^{\max}(B_2, \ell) \in O(\ell)$, $\mathrm{ew}^{\max}(B_2, \ell) \in O(\ell^2)$

Figure 5: Importance of widget ordering for polynomial growth. The unlabeled state can be either existential or universal.

It seems plausible that the maximal universal width of AFAs can have polynomial growth in a similar fashion to the polynomial tree width of NFAs. That is, an AFA with $1 \leq d \leq m$ consecutive universally branching cycles (and also no (EUW) widgets) would have at most $O(\ell^{d-1})$ universal branches in any pruned computation tree over strings of length ℓ.

Conjecture 3 *Let A be an m-state AFA with no* (EUW) *widgets. Then*

$$\mathrm{uw}^{\max}(A, \ell) \in O(\ell^{m-1}).$$

Unfortunately, it is not exactly clear how to determine the polynomial upper bound for maximal universal/existential width in AFAs with interleaved (IUW) and $(\mathrm{IEW})_\alpha$ widgets. However, since interleaved (IUW) and

$(\mathrm{IEW})_\alpha$ widgets require that the computation *alternates* between existential and universal states, it seems likely that the polynomial growth rates of maximal universal/existential width are related to the number of alternations an AFA is allowed to make, as studied by Geffert [2].

6 Conclusion

We have characterized necessary and sufficient conditions for an AFA to have unbounded tree width or exponential tree width by reducing the problem to the corresponding question for NFAs. A similar widget-condition characterizes AFAs with infinite maximal universal width and this yields an efficient algorithm to decide finiteness of maximal universal width. We have identified a set of widgets (Figure 2) that guarantee maximal existential width to be infinite but it remains open whether this is an exact characterization.

We have shown that when universal width is bounded, the finiteness of maximal existential width of an AFA can be decided in polynomial time but without the assumption the algorithm requires exponential time. The question of obtaining an efficient algorithm is related to getting a widget characterization of this property. In the last section we have initiated a study of possible growth rates of maximal universal and existential width. This topic has many questions for further research.

Acknowledgements

Research supported by NSERC grant OGP0147224.

References

[1] Ashok K. Chandra, Dexter Kozen, and Larry J. Stockmeyer. Alternation. *Journal of the ACM*, 28(1):114–133, 1981. doi:10.1145/322234.322243.

[2] Viliam Geffert. An alternating hierarchy for finite automata. *Theoretical Computer Science*, 445:1–24, 2012. doi:10.1016/j.tcs.2012.04.044.

[3] Jonathan Goldstine, Martin Kappes, Chandra M. R. Kintala, Hing Leung, Andreas Malcher, and Detlef Wotschke. Descriptional complexity of machines with limited resources. *Journal of Universal Computer Science*, 8(2):193–234, 2002. doi:10.3217/jucs-008-02-0193.

[4] Jonathan Goldstine, Chandra M. R. Kintala, and Detlef Wotschke. On measuring nondeterminism in regular languages. *Information and Computation*, 86:179–194, 1990. doi:10.1016/0890-5401(90)90053-K.

[5] Yo-Sub Han, Sungmin Kim, Sang-Ki Ko, and Kai Salomaa. Existential and universal width of alternating finite automata. In *25th International Conference on Descriptional Complexity of Formal Systems, DCFS 2023*, Lecture Notes in Computer Science. Springer, 2023 (to appear).

[6] Yo-Sub Han, Arto Salomaa, and Kai Salomaa. Ambiguity, nondeterminism and state complexity of finite automata. *Acta Cybernetica*, 23:141–157, 2017. doi:10.14232/actacyb.23.1.2017.9.

[7] Markus Holzer and Martin Kutrib. Descriptional and computational complexity of finite automata - a survey. *Information and Computation*, 209(3):456–470, 2011. doi:10.1016/j.ic.2010.11.013.

[8] Juraj Hromkovic, Sebastian Seibert, Juhani Karhumäki, Hartmut Klauck, and Georg Schnitger. Communication complexity method for measuring nondeterminism in finite automata. *Information and Computation*, 172(2):202–217, 2002. doi:10.1006/inco.2001.3069.

[9] Chris Keeler. *Finite Automata with Restricted Nondeterminism and Universality*. PhD thesis, Queen's University, 2021.

[10] Chris Keeler and Kai Salomaa. Combining limited parallelism and nondeterminism in alternating finite automata. In Galina Jirásková and Giovanni Pighizzini, editors, *22nd International Conference on Descriptional Complexity of Formal Systems, DCFS 2020*, volume 12442 of *Lecture Notes in Computer Science*, pages 91–103. Springer, 2020. doi:10.1007/978-3-030-62536-8_8.

[11] Chris Keeler and Kai Salomaa. Structural properties of NFAs and growth rates of nondeterminism measures. *Information and Computation*, 284:article 104690, 2022. doi:10.1016/j.ic.2021.104690.

[12] Chandra M. R. Kintala and Detlef Wotschke. Amounts of nondeterminism in finite automata. *Acta Informatica*, 13(2):199–204, 1980. doi:10.1007/BF00263994.

[13] Richard E. Ladner, Richard J. Lipton, and Larry J. Stockmeyer. Alternating pushdown and stack automata. *SIAM Journal of Computing*, 13(1):135–155, 1984. doi:10.1137/0213010.

[14] Ernst L. Leiss. Succinct representation of regular languages by Boolean automata. *Theoretical Computer Science*, 13:323–330, 1981. doi:10.1016/S0304-3975(81)80005-9.

[15] Hing Leung. Separating exponentially ambiguous finite automata from polynomially ambiguous finite automata. *SIAM Journal of Computing*, 27(4):1073–1082, 1998. doi:10.1137/S0097539793252092.

[16] Alexandros Palioudakis, Kai Salomaa, and Selim G. Akl. State complexity of finite tree width NFAs. *Journal of Automata, Languages and Combinatorics*, 17(2-4):245–264, 2012. doi:10.25596/jalc-2012-245.

[17] Bala Ravikumar and Oscar H. Ibarra. Relating the type of ambiguity of finite automata to the succinctness of their representation. *SIAM Journal of Computing*, 18(6):1263–1282, 1989. doi:10.1137/0218083.

[18] Jeffrey O. Shallit. *A Second Course in Formal Languages and Automata Theory*. Cambridge University Press, 2008. doi:10.1017/CBO9780511808876.

[19] Andreas Weber and Helmut Seidl. On the degree of ambiguity of finite automata. *Theoretical Computer Science*, 88(2):325–349, 1991. doi:10.1016/0304-3975(91)90381-B.