Determinism and Nondeterminism in Finite Automata with Advice

Pavol Ďuriš[1], Rafael Korbaš[1], Rastislav Královič[1](✉), and Richard Královič[2]

[1] Faculty of Mathematics, Physics, and Informatics,
Comenius University, Bratislava, Slovakia
{duris,kralovic}@dcs.fmph.uniba.sk
[2] Google, Inc., Zürich, Switzerland
ri.kralovic@gmail.com

Abstract. We consider the model of finite automata with advice introduced by Küçük et al. We show that there are languages, in particular the language of palindromes, that cannot be recognized by DFA regardless of the size of advice. Also, we show that a DFA cannot utilize more than exponential advice. We initiate the study of NFA with advice: we show that, unlike the DFA, NFA can recognize all languages with advice of exponential size. On the other side of the spectrum, with constant advice, DFA are as powerful as NFA. We show that for any growing function f, there are languages that can be recognized by NFA with advice $f(n)$, but cannot be recognized by DFA regardless of advice size. We also ask what languages can be recognized with polynomial advice. For NFA we show that this class is not closed under complementation, and that it contains all bounded languages. Bounded languages over one-letter words can even be recognized by DFA with polynomial advice. We also give examples of languages that cannot be recognized by NFA with polynomial advice. Finally, we show that increasing advice helps for NFA, and for any advice of size $f(n) \leq n$ we show that there is a language that can be recognized by a DFA with advice $f(n)$, but cannot be recognized by an NFA with advice $o(f(n))$.

1 Introduction

Uniformity, i.e., the fact that a single, finitely-described, device is used to process instances of arbitrary size, is a central property shared by all computation models deemed feasible. Understanding the role this restriction plays in the inherent limitations of feasible computation models is one of the fundamental directions in theoretical computer science. Models that are naturally defined as non-uniform (like circuits), usually come with uniformity as an add-on requirement, and the uniform and non-uniform versions can be compared. Turing machines are naturally uniform, and their non-uniform version was introduced in the seminal paper of Karp and Lipton [5] in the form of advice machines. The machine,

Research partially supported by grant VEGA 2/0165/16.

H.-J. Böckenhauer et al. (Eds.): Hromkovič Festschrift, LNCS 11011, pp. 3–16, 2018.
https://doi.org/10.1007/978-3-319-98355-4_1

together with the input word x of length n, is also provided an advice string $\alpha(n)$, that does not need to be computable, but remains the same for all words of length n. It is well known how fruitful the line of research investigating this notion has been in understanding the fundamentals of computation. However, since the main questions concerning Turing machines still remain unsolved, it is natural to focus on their restricted versions for gaining better insight. While there had been previous attempts to study non-uniform versions of automata (e.g., Ibarra and Ravikumar [4] considered 2-way automata with growing sets of states), Damm and Holzer [1] proposed the first model of finite automata with advice along the lines of [5]: a one-tape finite automaton for which each input word is prefixed by the advice string (see Definition 1). Since the advice is on the same tape as the input, the automaton can use only constant advice. The class of languages recognized by these devices is denoted REG$/k$ where k is the length of advice string, extending the notation REG for the class regular languages. Even REG/1 can recognize some non-recursive languages (e.g., unary languages), and there is a strict hierarchy $\mathrm{REG}/(k-1) \subsetneq \mathrm{REG}/k$. In order to overcome the limitation to constant amount of advice, Tadaki et al. [7] consider advice of length n written on separate track (see Definition 2). We denote the class of languages recognized by these automata $\widehat{\mathrm{REG}}/n$ to distinguish them from the previous model. In [7] it is shown that $\widehat{\mathrm{REG}}/n = 1\mathrm{DLIN}/O(n)$, i.e., the class of languages recognized by a linear-time 1-tape deterministic Turing machine with advice written on a separate track. Hence, the power of the Turing machine to write information to the tape does not help in this case. The advice written on a separate track overcomes the shortcomings of the model from [1], but it does not allow to study other than linear size of advice. Freivalds [3], with the motivation to study the amount of nonconstructiveness in non-constructive arguments, proposes a model of finite automata that use separate tapes to store the advice. In his model (see Definition 3), the advice may be split into several tapes. However, the advice string of length m must be valid for all words of lengths up to m. He considers deterministic automata with two-way input and advice tapes. We denote the class of these languages $\mathscr{F}(\mathrm{DFA})/f(n)$. Freivalds shows that $\mathscr{F}(\mathrm{DFA})/o(\log n) = \mathrm{REG}$, but there are some non-recursive languages that can be recognized with polylogarithmic advice. On the other hand $\mathscr{F}(\mathrm{DFA})/(n2^n)$ contains all languages, and there are languages that cannot be recognized with advice $o(2^n)$. We adopt the model by Küçük et al. [6] (see Definition 4) that combines the models of Freivalds, and Tadaki et al. Denoted by $\mathscr{L}(\mathrm{DFA})/f(n)$, the advice of length $f(n)$ for a one-way deterministic FA is written on separate tapes (in our results we consider only a single advice tape), and the advice is specific for the inputs of given length. Küçük et al. showed that $\mathscr{L}(\mathrm{DFA})/\mathit{exp}(\texttt{2w-input})$ contains all languages, a hierarchy $\mathscr{L}(\mathrm{DFA})/(n^k) \subsetneq \mathscr{L}(\mathrm{DFA})/(n^{k+1})$, and a separation $\mathscr{L}(\mathrm{DFA})/\mathit{poly} \subsetneq \mathscr{L}(\mathrm{DFA})/\mathit{poly}(\texttt{2w-input})$. They also showed that the language of palindromes, $L_{\mathsf{PAL}} \notin \mathscr{L}(\mathrm{DFA})/\mathit{poly}$. They asked a question whether exponential advice allows to recognize all languages (with one-way input), and, in particular, whether $L_{\mathsf{PAL}} \in \mathscr{L}(\mathrm{DFA})/\mathit{exp}$.

Our Contribution

We answer the question from [6], and show that L_{PAL} cannot be recognized by a DFA regardless of the advice size, i.e., $L_{\mathsf{PAL}} \notin \mathscr{L}(\text{DFA})/\star$ (Corollary 1). Moreover, we show that DFA cannot utilize more than exponential advice (Theorem 3). Then we extend the model from [6] to nondeterministic FA, and show that $\mathscr{L}(\text{NFA})/\mathit{exp}$ contains all languages (Theorem 4). We also show that for constant advice the nondeterminism doesn't help, since $\mathscr{L}(\text{NFA})/k = \text{REG}/k$ (Theorem 5). Since NFA can recognize any language with exponential advice, it is natural to ask which languages are in $\mathscr{L}(\text{NFA})/\mathit{poly}$. We show that $L_{\mathsf{PAL}} \notin \mathscr{L}(\text{NFA})/\mathit{poly}$ (Corollary 2) whereas $\mathit{co}L_{\mathsf{PAL}} \in \mathscr{L}(\text{NFA})/\mathit{poly}$ (Theorem 8), so $\mathscr{L}(\text{NFA})/\mathit{poly}$ is not closed under complement. Moreover, since $\mathscr{L}(\text{DFA})/\star$ is obviously closed under complement, $\mathit{co}L_{\mathsf{PAL}}$ is an example of a language that can be recognized nondeterministically with polynomial advice, but cannot be recognized deterministically regardless of the advice size. We extend this observation to show that for any growing function f, there is a language that can be recognized by a NFA with advice $O(f(n))$, but cannot be recognized by DFA regardless of advice (Theorem 6). Further, we show that any bounded language can be recognized by NFA with polynomial advice (Theorem 9), and if the language is of the form $L \subseteq a_1^\star \cdots a_k^\star$, it can even be recognized deterministically with polynomial advice (Theorem 10). Finally, we show a hierarchy of advice lengths for NFA (Theorem 11), and even stronger result for sublinear advice stating that for any advice of size $f(n) \le n$ there is a language that can be recognized by a DFA with advice $f(n)$, but cannot be recognized by an NFA with advice $o(f(n))$ (Theorem 12).

2 Notation and Definitions

Let kDFA (resp. kNFA) denote a k-tape one-way deterministic (resp. nondeterministic) finite automaton. We use the standard model of multi-tape automata (see e.g., [2]) where the input words on each tape are delimited by special symbols, and in each step, the automaton decides, based on the symbols on the tapes, and the current state, which heads to advance and how to change the state. We say that a tuple $(w_1, \ldots, w_k) \in (\Sigma^\star)^k$ is the input of the kDFA automaton A if each w_i is written on the respective tape. For an automaton A, let $\mathscr{L}(A)$ be the language recognized by A. Let $\mathscr{L}(k\text{DFA}) \subseteq (\Sigma^\star)^k$ (resp. $\mathscr{L}(k\text{NFA})$) be the family of languages recognized by the respective automata. The symbol Σ denotes a finite alphabet, $\Sigma_n := \{0, 1, \ldots, n-1\}$. Unless stated otherwise, our automata will be 1-way. For technical clarity, we recall the definitions of the various approaches mentioned in the introduction. Damm and Holzer introduced the advice string as a prefix of the input word:

Definition 1 (Damm and Holzer [1]). *Let Σ be a finite alphabet, and let $\alpha \colon \mathbb{N} \to \Sigma^\star$ such that $\forall n, |\alpha(n)| \le f(n)$. For a language L, let $\alpha L = \{\alpha(|w|)\#w \mid w \in L\}$. Then*

$$\text{REG}/f(n) := \{L \subseteq \Sigma^\star \mid \exists \alpha, \textit{ and a } \text{DFA } A \colon \alpha L = \mathscr{L}(A) \cap \alpha\Sigma^\star\}.$$

Tadaki et al. considered the advice written on a separate track:

Definition 2 (Tadaki et al. [7]). *For two words $u = u_1 \dots u_n \in \Sigma_1, v = v_1 \dots v_n \in \Sigma_2$ let $\left[\begin{smallmatrix} u \\ v \end{smallmatrix}\right] = (u_1, v_1)(u_2, v_2) \dots (u_n, v_n) \in \Sigma_1 \times \Sigma_2$. Let Σ be a finite alphabet, and let $\alpha \colon \mathbb{N} \to \Sigma^\star$ such that for each $n \in \mathbb{N}$, $|\alpha(n)| = n$. For a language L, let $\left[\begin{smallmatrix} \alpha \\ L \end{smallmatrix}\right] = \left\{ \left[\begin{smallmatrix} \alpha(|w|) \\ w \end{smallmatrix}\right] \mid w \in L \right\}$. Then*

$$\widehat{\mathrm{REG}}/n := \left\{ L \subseteq \Sigma^\star \mid \exists \alpha, \textit{and a } \mathrm{DFA}\ A \colon \begin{bmatrix} \alpha \\ L \end{bmatrix} = \mathscr{L}(A) \cap \left(\begin{bmatrix} \alpha \\ \Sigma^\star \end{bmatrix} \right) \right\}.$$

Freivalds considered 2-way multitape machines with advice on several tapes, such that the advice $\alpha(n)$ is valid for all words of length at most n:

Definition 3 (Freivalds [3]). *Let, for each $1 \le i \le k$, $\alpha_i \colon \mathbb{N} \to \Sigma^\star$ such that $\forall n$, $\sum_{i=1}^{k} |\alpha_i(n)| \le f(n)$. Then*

$$\begin{aligned} \mathscr{F}(\mathrm{DFA})/f(n) := \{L \subseteq \Sigma^\star \mid \exists k, \alpha_1, \dots, \alpha_k, \ &\textit{and a 2-way } (k+1)\mathrm{DFA}\ A, \\ w \in L \Rightarrow \forall m \ge |w| \colon \ &(w, \alpha_1(m), \dots, \alpha_k(m)) \in \mathscr{L}(A) \textit{ and} \\ w \notin L \Rightarrow \forall m \ge |w| \colon \ &(w, \alpha_1(m), \dots, \alpha_k(m)) \notin \mathscr{L}(A)\}. \end{aligned}$$

We adopt the approach from Küçük et al., where the advice is on separate tapes (in general, we allow multiple advice tapes) and is specific for words of given length:

Definition 4. *Let, for each $1 \le i \le k$, $\alpha_i \colon \mathbb{N} \to \Sigma^\star$ such that $\forall n, |\alpha_i(n)| = f(n)$. For a language L, let $L_\alpha = \{(w, \alpha_1(n), \dots, \alpha_k(n)) \mid w \in L, n = |w|\} \subseteq (\Sigma^\star)^{k+1}$. Then*

$$\begin{aligned} \mathscr{L}(\mathrm{DFA})/f(n)_k := \{L \subseteq \Sigma^\star \mid \exists \alpha_1, \dots, \alpha_k, \ &\textit{and a } (k+1)\mathrm{DFA}\ A \colon \\ &L_\alpha = \mathscr{L}(A) \cap \Sigma^\star_\alpha\}. \end{aligned}$$

We say that a $(k+1)$DFA A recognizes language L with advice α if $L_\alpha = \mathscr{L}(A) \cap \Sigma^\star_\alpha$. We can leave out k if $k = 1$. The class $\mathscr{L}(\mathrm{NFA})/f(n)_k$ is defined in a similar way.

We write $\mathscr{L}(\mathrm{DFA})/\star_k$ if the size of the advice is unlimited, $\mathscr{L}(\mathrm{DFA})/exp_k$ if it is at most exponential, and $\mathscr{L}(\mathrm{DFA})/poly_k$ if it is at most polynomial in the input length. We can further modify the automaton by giving specifications of the form `rt-tape` meaning that the tape is realtime, i.e., the head moves in every step to the right, and the automaton stops after the head reaches the end of the tape, or `2w-tape` meaning that the head is 2-way. So, e.g., $\mathscr{L}(\mathrm{DFA})/o(n)$(`rt-input`, `2w-advice`) describes deterministic automata with realtime input tape and 2-way advice tape of sublinear size. Note that the requirement $|\alpha(n)| = f(n)$ comes with no loss of generality, since the advice can always be padded with some new symbol. Also note, that we do not specify the cardinality of the advice alphabet. While this may be of importance in some cases, e.g., when studying advice of constant size, it has no effect on our results.

3 Results

The model quickly becomes extremely powerful: with two advice tapes, or exponential advice and 2-way input, all languages can be recognized. On the other hand, a DFA cannot recognize some very simple languages even with fairly large advice, and there is a hierarchy showing that additional advice size increases the power. The following statements have been proven in [6]:

Theorem 1 (Küçük et al. [6]).

1. $\mathscr{L}(\mathrm{DFA})/exp_2 = \mathscr{L}(\mathrm{DFA})/exp(\texttt{2w-input}, \texttt{rt-advice}) = 2^{\Sigma^\star}$.
2. *For all* k, $\mathscr{L}(\mathrm{DFA})/(n^k) \subsetneq \mathscr{L}(\mathrm{DFA})/(n^{k+1})$.
3. $L_{\mathsf{PAL}} \notin \mathscr{L}(\mathrm{DFA})/poly \cup \mathscr{L}(\mathrm{DFA})/\star(\texttt{rt-input})$ *where* L_{PAL} *is the language of palindromes* $L_{\mathsf{PAL}} = \{ww^R \mid w \in \Sigma_2^\star\}$.

In this paper, we focus our attention on machines with one advice tape, and 1-way input tape. In [6], the authors asked if $L_{\mathsf{PAL}} \in \mathscr{L}(\mathrm{DFA})/\star$. We show that it is not the case. In fact, our proof applies not only to L_{PAL}, but to a slightly more general class of languages described in the following definition. Informally, the words consist of two parts of fixed lengths: an arbitrary request string, and a response. There may be several responses to a given request. The required property is that for any two requests there is a string that is a valid response for exactly one of them.

Definition 5. *Let* $\{R_n\}_{n=1}^\infty$ *be a family of relations* $R_n \subseteq \Sigma_2^n \times \Sigma^{f(n)}$ *for some* $f\colon \mathbb{N} \to \mathbb{N}$*, such that* $\forall x_0, x_1 \in \Sigma_2^n, x_0 \neq x_1$*, there is a* $y \in \Sigma^{f(n)}$ *such that* $R_n(x_i, y)$ *and* $\neg R_n(x_{1-i}, y)$ *for some* $i \in \{0,1\}$*. Let* L_R *be the language*

$$L_R := \{xy \mid x \in \Sigma_2^\star,\ |y| = f(|x|),\ R_{|x|}(x,y)\}.$$

We call L_R *a* prefix-sensitive language *for relation* R*.*

Example 1. Examples of prefix-sensitive languages are some well studied languages like

- $L_{\mathsf{PAL}} = \{ww^{\mathsf{R}} \mid w \in \Sigma^\star\}$,
- $NUM_{\leq} := \{x\#y \mid x, y \in \Sigma_2^\star, |x| = |y|, [x]_2 \leq [y]_2\}$ where $[x]_2 \in \mathbb{N}$ denotes the number with binary representation x,
- $NUM_{<} := \{x\#y \mid x, y \in \Sigma_2^\star, |x| = |y|, [x]_2 < [y]_2\}$,
- $\{ww \mid w \in \Sigma^\star\}$.

Theorem 2. *Let* L *be a prefix-sensitive language. Then* $L \notin \mathscr{L}(\mathrm{DFA})/\star$.

Before proving the theorem, let us introduce some notation. Let A be any 2-tape DFA with the set of states Q, where $s := |Q|$, and advice α. For a fixed n we shall consider words of length $n + f(n)$, and denote $m := |\alpha(n + f(n))|$. For an $i \in \{1, \ldots, m\}$ and a $q \in Q$, we say that A is in *internal configuration* (i, q) if the advice head of A is reading the i-th advice symbol and A is in state q. We define the *internal configuration graph* G *of* A as the graph whose vertices are (i, q) for all such i, q. For each vertex (i, q), there are two outgoing directed edges, labelled by symbols 0 and 1, respectively. Each of the edges may be additionally labelled by $+$. These edges describe the behavior of A: An edge labelled by $x \in \{0, 1\}$ leads to (i', q') such that A moves to internal configuration (i', q') within one computation step when the input head reads x. The edge is additionally labelled by $+$ if and only if A moves its input head in this step.

The internal configuration graph is completely defined by the transition function of A and the advice for input length $n + f(n)$. Also, the behavior of A on inputs of length $n + f(n)$ is determined by the internal configuration graph. When A is in internal configuration $z = (i, q)$ and reads symbol x, it follows a path in G induced by edges labelled by x which ends with an edge labelled by $x+$, leading to internal configuration z' and we say that x *leads from* z *to* z'. The definition of *leads to* can be naturally extended to words from $\Sigma_2^\star$.

Lemma 1. *Let $z = (i, q) \in G$ be arbitrary internal configuration of A. There exist two words $u, v \in \Sigma_2^\star$ such that*

1. $u \neq v$,
2. neither u is prefix of v nor v is prefix of u,
3. $1 \leq |u| \leq 2s + 2, 1 \leq |v| \leq 2s + 2$,
4. both u and v lead from z to the same internal configuration z'.

Proof. Consider all words over Σ_2 of length exactly $2s + 1$. Each of these words leads from z to some $(j, \_)$. Let w be such word where j is minimal. Consider each proper prefix p of w, including the empty word ε; there are $2s + 1$ of them. Thus, we have $w = pxg$ for some $x \in \Sigma_2, g \in \Sigma_2^*$ (see Fig. 1). Since p is prefix of w, it leads from (i, q) into (i', q'), where $i \leq i' \leq j$. Let $x \neq x' \in \Sigma_2$. The word $px'g$ leads from (i, j) via (i', q') to $(j', \_)$, where $j' \geq j$. Thus, some edge outgoing from some $(j, \_)$ is used by A when reading $px'g0$. Let w' be the longest prefix of $px'g0$ such that an edge like this is used when reading the last symbol of w'. It holds that px' is a prefix of w'. This ensures that any two w' constructed for different prefixes p are not a prefix of each other.

In this way, we have constructed $2s + 1$ different words $w'_1, \ldots, w'_{2s+1}$ and any two of them satisfy conditions 1, 2, and 3. Since there are only $2s$ edges outgoing from $(j, \_)$, we can apply the pigeonhole principle to find w'_a and w'_b which use the same outgoing edge. This implies that w'_a and w'_b lead from z to the same internal configuration z'. □

Proof (of Theorem 2). Let $n := 4(s + 1)^2$. We prove that there are two input words of length $n + f(n)$, one in L and one not in L, which are both either accepted or rejected by A.

Automaton A starts in internal configuration $(1, q_0)$. We construct a sequence of internal configurations $c_0 = (1, q_0), c_1, \ldots, c_{2s+2}$ by invoking the claim to get the next configuration from the previous one. In this way, we obtain, for each i, some u_i and v_i satisfying all conditions from the claim that both lead from c_{i-1} to c_i. We now have $2s + 2$ pairs u_i, v_i. For each pair, $||u_i| - |v_i|| \leq 2s + 1$. Our goal is to construct two different words of length at most $4(s + 1)^2$ that lead

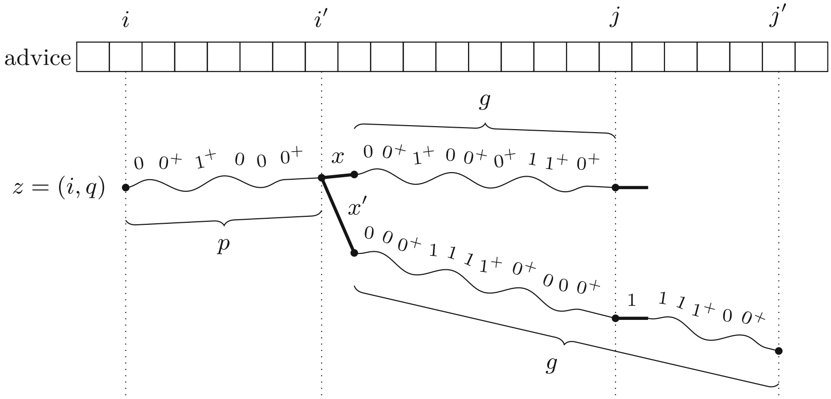

Fig. 1. Situation in the proof of Lemma 1: the word $w = pxg$ leads to some configuration $(j, \_)$. The word $px'g$ leads to some $(j', \_)$ for $j' \geq j$. Hence some prefix of $px'g$ uses an edge outgoing from some $(j, \_)$. In the proof we use the words $px'g0$ to cover the case $j' = j$.

from $(1, q_0)$ to the same internal configuration. We consider two cases. First, if $|u_i| = |v_i|$ for some i, we can take $u_1 u_2 \dots u_i$ and $u_1 u_2 \dots u_{i-1} v_i$. Both such words have equal length, lead from $(1, q_0)$ to c_i, and their length is at most $4(s+1)^2$. In the second case, $|u_i| \neq |v_i|$ for all $i \in \{1, \dots, 2s+2\}$. By the pigeonhole principle, there are two pairs such that $||u_i| - |v_i|| = ||u_j| - |v_j||$. Without loss of generality, let $|u_i| > |v_i|$ and $|u_j| > |v_j|$. Then the words $u_1 \dots u_{j-1} v_j$ and $u_1 \dots u_{i-1} v_i u_{i+1} \dots u_j$ satisfy our condition.

Thus, we have some two different words of equal length, no longer than $n = 4(s+1)^2$, that lead from $(1, q_0)$ to the same internal configuration. We can arbitrary pad both words to have length exactly n to obtain $u \neq v$ such that $|u| = |v| = n$ that lead from the initial internal configuration to the same internal configuration. Since L is prefix-sensitive, there is a $y \in \Sigma^{f(n)}$ such that $uy \in L$ and $vy \notin L$, or $vy \in L$ and $uy \notin L$. However, uy and vy are either both accepted or both rejected by A. □

Using Theorem 2, we can show that several languages cannot be recognized by DFA, regardless of the advice size. In particular:

Corollary 1. *$L_{\mathsf{PAL}} \notin \mathscr{L}(\mathrm{DFA})/\star$.*

An interesting question to ask is what is the maximal advice that a DFA is able to use. We can show that advice above $2^{O(n)}$ cannot be utilized:

Theorem 3. *$\mathscr{L}(\mathrm{DFA})/\star = \mathscr{L}(\mathrm{DFA})/2^{O(n)}$.*

Proof. Let A be an s-state DFA with advice α recognizing some language L with advice using alphabet Σ_k for some k, and let $\ell = s^{ks} + 1$. We show that L can be recognized by A using advice of length less than $\ell(nk^n + 1) = 2^{O(n)}$.

Suppose, for the sake of contradiction, that $|\alpha(n)| \geq \ell(nk^n + 1)$ for some n. We construct a modified advice function α' such that $\alpha'(n') = \alpha(n')$ for $n' \neq n$, and $|\alpha'(n)| < |\alpha(n)|$, and show that A recognizes the same language with advice α, and α'.

Suppose, without loss of generality, that A always moves both its heads to the ends of the respective tapes. Also, suppose that A in each step moves at least one head. Partition the advice string $\alpha(n)$ into blocks of size ℓ. For each of the k^n possible input words w of length n consider the computation of A on w with advice $\alpha(n)$. If the head on the input tape moves at least once while the head on the advice tape scans block B, mark B as *relevant*. Since each word w can mark at most n blocks, there are at most nk^n relevant blocks overall. Since $|\alpha(n)| > \ell nk^n$, there is some block B that was not marked relevant by any word.

For each word w, the automaton A enters block B in state q_w, reading symbol a_w on the input tape. Then for the consecutive ℓ steps it moves only the advice head, going through a sequence of states $q_w = q_w^{(1)}, q_w^{(2)}, \ldots, q_w^{(\ell)}$. Since the input head does not move, this sequence is fully determined by (a_w, q_w), so there are only ks distinct sequences. For an index i, consider the vector $\eta_i = \left(q_{w_1}^{(i)}, q_{w_2}^{(i)}, \ldots, q_{w_{k^n}}^{(i)}\right)$ where $\{w_1, \ldots, w_{k^n}\} = \Sigma_k^n$. Since there are ks distinct sequences, there are at most s^{ks} possible values of η_i, and because $\ell > s^{ks}$, there are two indices i, j, such that for each $w \in \Sigma_k^n$ it holds $q_w^{(i)} = q_w^{(j)}$. This means that if the advice string $\alpha(n)$ is shortened by leaving out the part of block B starting from index $i+1$, and continuing until (and including) index j, the automaton A will behave exactly the same way. □

It remains open whether advice of exponential size can actually be utilized by a DFA. From Theorem 1 we know that for any polynomial n^k, there is a language that needs advice $O(n^k)$, but we are not aware of any example of a language that would require more than polynomial advice.

3.1 Nondeterministic Automata

Next we turn our attention to nondeterministic automata, which have not been studied with respect to advice before. With no restriction on the advice size, it is easy to see that NFA can recognize all languages:

Theorem 4. *$\mathscr{L}(\mathrm{NFA})/f(n) = 2^{\Sigma^\star}$, where $f(n) = (n+1)|\Sigma|^n$. In particular, any language L can be recognized by an* NFA *with advice of size $(n+1)|L \cap \Sigma^n|$.*

Proof. Let $L \subseteq \Sigma^\star$ be any language. The advice function

$$\alpha(n) = \#w_1\#w_2\#\cdots\#w_z$$

where $\{w_1, \ldots, w_z\} = L \cap \Sigma^n$. The 2-tape NFA automaton just scans the advice tape, stops nondeterministically on some symbol #, and checks the input tape with the advice. □

In [6] it has been proven that $\mathscr{L}(\text{DFA})/k(\texttt{2w-input}) = \text{REG}/k$. Again, it is easy to observe that to constant advice, the nondeterminism is not more powerful than determinism:

Theorem 5. $\mathscr{L}(\text{NFA})/k = \text{REG}/k$.

Proof. Obviously, $\mathscr{L}(\text{NFA})/k \supseteq \text{REG}/k$. The other inclusion is easy to see, too. Any 2-tape NFA with advice tape of constant size can be transformed into a normal form where it first (deterministically) reads the content of the tape, stores it in the state, and continues to work (nondeterministically) on the input tape. We can use standard subset construction to turn this automaton into a deterministic one that first reads the advice for either the prefix of the input or a separate advice tape. □

Unlike DFA, NFA can recognize all languages with exponential advice, but are not more powerful than DFA's when equipped with constant advice. One may ask where is the threshold when NFA become more powerful. We show that it is just above the constant:

Theorem 6. *Let $f(n) = \omega(1)$. There is a language $L \in \mathscr{L}(\text{NFA})/o(f(n))$ such that $L \notin \mathscr{L}(\text{DFA})/\star$.*

Proof. Choose a function $g(n)$ such that $g(n)2^{g(n)} = o(f(n))$, e.g., $g(n) := \log\log f(n)$. Let $L := \{v \mid v = ww^{\mathsf{R}}\#^i, |w| = g(|v|)\}$. From Theorem 4 it follows that L_{PAL} can be recognized by a NFA with advice α such that $|\alpha(n)| = O(n2^{\frac{n}{2}})$. Moreover, to recognize L, one can utilize the advice $\alpha(2g(n))$, which is of size $o(f(n))$.

On the other hand, L is a prefix-sensitive language in terms of Definition 5, so due to Theorem 2, $L \notin \mathscr{L}(\text{DFA})/\star$. □

Interesting classes of languages are those that can be recognized by NFA (or DFA) with polynomial advice. While we don't know whether DFA can utilize more than polynomial advice, we show that NFA can. In particular, NFA can recognize all languages with exponential advice, but cannot recognize with polynomial advice (even with two-way advice tape) many prefix-sensitive languages. To state the next theorem, we use a subclass of prefix-sensitive languages:

Definition 6. *A prefix-sensitive language L is called* strictly prefix-sensitive, *if there is a function $d\colon \Sigma_2^\star \to \Sigma^\star$, such that for each $x \in \Sigma_2^\star$, it holds $R_{|x|}(x, d(x))$ (i.e., $|d(x)| = f(|x|)$, $xd(x) \in L$), and for any two $x_0, x_1 \in \Sigma_2^n, x_0 \neq x_1$ it holds $R_n(x_i, d(x_i))$, and $\neg R_n(x_{1-i}, d(x_i))$ for some $i \in \{0, 1\}$.*

Note that all the languages from Example 1 are strictly prefix-sensitive: for $NUM_{\geq}$, consider the function $d(x) = \#x$. Similarly, for $NUM_{<}$ the function $d(x) = \#y$ such that $[y]_2 = [x]_2 + 1$ fulfills the previous definition.

Theorem 7. *Let L be a strictly prefix-sensitive language. Then*

$$L \notin \mathscr{L}(\text{NFA})/\mathit{poly}(\texttt{2w-advice}).$$

Proof. Consider a 2-tape NFA A with polynomial advice α. For a fixed n, consider all words $w \in \Sigma^{n+f(n)}$, such that $w = xd(x)$, $x \in \Sigma_2^n$, $w \in L$, and select one accepting computation γ_w of A on w (using the advice $\alpha(n + f(n))$). After reading x in γ_w, let A be in state q_w, and the advice head be on the i_w-th symbol. Since there are only polynomially many pairs (q_w, i_w), there are two words $x_1, x_2 \in \Sigma_2^n$ such that A is in the same state with the same position on the advice tape after reading x_1 and x_2 in the respective accepting computations $\gamma_{w_1}, \gamma_{w_2}$. Since L is strictly prefix-sensitive, without loss of generality $x_1 d(x_1) \in L$, $x_2 d(x_1) \notin L$. Since the automaton is 1-way, and the advice is fixed, there is an accepting computation on $x_2 d(x_1)$ with the same prefix as γ_{w_2}. Thus, $x_2 d(x_1) \in L$ – a contradiction. □

Corollary 2. $L_{\mathsf{PAL}} \notin \mathscr{L}(\mathrm{NFA})/\mathit{poly}$.

On the other hand,

Theorem 8. $coL_{\mathsf{PAL}} = \{a, b\}^\star - L_{\mathsf{PAL}} \in \mathscr{L}(\mathrm{NFA})/O(n^2)$.

Proof. We construct a nondeterministic automaton A with quadratic advice α that recognizes coL_{PAL}. For odd n, let the advice $\alpha(n) = \$$. A immediately accepts in this case.

For even n, let the advice

$$\alpha(n) = \#w_1\#w_2\#\cdots\#w_{\frac{n}{2}}\#$$

where $w_i = 0^{i-1}10^{n-2i}10^{i-1}$. Note that $|w_i| = n$ so the advice is of length $O(n^2)$. The automaton A nondeterministically selects one word w_i, and uses it to check the input symbols on the positions indicated by ones in w_i. If they differ, A accepts. □

One class of languages that can be accepted by NFA with polynomial advice is the class of bounded languages:

Theorem 9. *Let* $w_1, \ldots, w_k \in \Sigma^\star$. *Let* L *be any bounded language* $L \subseteq w_1^\star \cdots w_k^\star$. *Then* $L \in \mathscr{L}(\mathrm{NFA})/\mathit{poly}$.

Proof. It is easy to see that a bounded language contains at most $(n+1)^k = O(n^k)$ words of length n. The result follows from Theorem 4. □

We don't know whether bounded languages can be recognized with polynomial advice by DFA, but an important subclass of bounded languages can:

Theorem 10. *There is a* DFA A_k *such that for any language*

$$L \subseteq 0^\star 1^\star \cdots (k-1)^\star \subseteq \Sigma_k^\star$$

there is an advice α*, such that* $|\alpha(n)| \le c_k n^{k-1}$ *for some* c_k*, and* A_k *recognizes* L *with advice* α.

Proof. The proof is by induction on k. For $k = 1$, any language over unary alphabet contains at most one word for each n, so advice of size 1 is sufficient. Let $L \subseteq 0^\star 1^\star \cdots (k-1)^\star$. For a word w, and a language L, let $wL := \{wu \mid u \in L\}$. Denote L_i the language $L_i \subseteq 1^\star 2^\star \cdots (k-1)^\star$ such that $0^i L_i = L \cap 0^i\{1, \dots, k-1\}^\star$. Obviously, $L = \cup_{i=0}^{\infty} L_i$, and each L_i is a bounded language over alphabet Σ_{k-1} (under a renaming of the symbols). By induction, each language L_i can be recognized by some DFA A_{k-1} with advices α_i such that $|\alpha_i(n)| \leq c_{k-1}n^{k-2}$ for some c_{k-1}. Construct the advice function α such that

$$\alpha(n) = 0\alpha_0(n)0\alpha_1(n-1)0\alpha_2(n-2)0 \cdots 0\alpha_{n-1}(1)0\alpha_n(0).$$

Note that $|\alpha(n)| \leq (n+1)c_{k-2}n^{k-2} + n + 1 \leq c_k n^{k-1}$ for some c_k. The DFA A_k recognizing L with advice α works as follows: while the input symbol is 0, it scans the advice tape for the next occurrence of 0. If the input symbol is not 0, it simulates the automaton A_{k-1} with the current advice. Note that the transition function of A does not depend on the language L, so it fulfills the statement of the theorem. □

3.2 Hierarchies

In [6] it was shown that for all k, $\mathscr{L}(\text{DFA})/(n^k) \subsetneq \mathscr{L}(\text{DFA})/(n^{k+1})$. We show a similar hierarchy for NFA:

Theorem 11. *Let $f, g \colon \mathbb{N} \to \mathbb{N}$ be such that $f(n)\log(f(n)) = o(g(n))$ and $g(n) \leq n2^{\frac{n}{2}}$. Then $\mathscr{L}(\text{NFA})/f(n)(\texttt{2w-advice}) \subsetneq \mathscr{L}(\text{NFA})/g(n)$.*

Proof. We repeat the ideas from Theorems 6 and 7 in a more specific way. Fix two functions f, g from the statement of the theorem. Note that $\lim_{n\to\infty} g(n) = \infty$. Let h be a function such that

$$h(n) := \max\{x \mid (2x+1)2^x \leq g(n)\}.$$

Since $(2h(n)+1)2^{h(n)} \leq n2^{\frac{n}{2}}$, it holds $2h(n) \leq n$. Consider the language

$$L = \{ww^{\mathsf{R}}\#^{n-2h(n)} \mid w \in \Sigma_2^{h(n)}, n \in \mathbb{N}\},$$

where $\#^0$ is defined as empty string. First, we show that

$$L \notin \mathscr{L}(\text{NFA})/f(n)(\texttt{2w-advice}).$$

Let us suppose, for the sake of contradiction, that L is recognized by some NFA A with advice $|\alpha(n)| \leq f(n)$. Let s be the number of states of A. We show that for all large enough n it holds

$$sf(n) < 2^{h(n)}. \tag{1}$$

Assume, for the sake of contradiction, that $sf(n) \geq 2^{h(n)}$ for arbitrary large n. Since $sf(n) \geq 2^{h(n)}$, it holds $\log s + \log f(n) \geq h(n)$, so $\log f(n) \geq h(n) - \log s$.

Also recall that $f(n) \geq 2^{h(n)}/s$. We have

$$\begin{aligned} f(n)\log f(n) &\geq f(n)(h(n)-\log s) \geq \frac{2^{h(n)}}{s}h(n) - \frac{2^{h(n)}}{s}\log s \\ &= \frac{2^{h(n)+1}4h(n)}{8s} - \frac{2^{h(n)}\log s}{s} > \frac{g(n)}{8s} - \frac{2^{h(n)}\log s}{s} \end{aligned}$$

where the last inequality comes from the fact that $4h(n) \geq 2(h(n)+1)+1$, which is satisfied for $h(n) \geq 2/3$, and $2^{h(n)+1}(2(h(n)+1)+1)$, which follows directly from the definition of $h(n)$. Thus, we get

$$f(n)\log f(n) > \frac{g(n)}{16s} + \left(\frac{g(n)}{16s} - \frac{2^{h(n)}\log s}{s}\right) \geq \frac{g(n)}{16s} \tag{2}$$

where the last inequality holds since for large enough n we have $2^{h(n)}16\log s \leq 2^{h(n)}(2h(n)+1) \leq g(n)$. However, $f(n)\log(f(n)) = o(g(n))$, so (2) cannot hold arbitrary large n – thus we have proven (1).

Now fix a large enough n. $|L \cap \Sigma_2^n| = 2^{h(n)}$. For each $w \in \Sigma_2^{h(n)}$ choose one accepting computation γ_w of A on $ww^{\mathsf{R}}\#^{n-2h(n)}$. Let A be in state q_w after reading the prefix w in γ_w, and its advice head is on position i_w. Since there are $sf(n)$ pairs (q_w, i_w), and $sf(n) < 2^{h(n)}$ there must be two words $w \neq w'$ with the same pair $(q_w, i_w) = (q_{w'}, i_{w'})$, which means there is also an accepting computation for $ww'^{\mathsf{R}}\#^{n-2h(n)}$.

On the other hand, it is easy to observe that $L \in \mathscr{L}(\text{NFA})/g(n)$: an advice of length $2^{h(n)}(2h(n)+1)$ is enough to describe all palindromes ww^{R}, $w \in \Sigma_2^{h(n)}$. □

For sublinear advice we give a stronger result:

Theorem 12. *Let $f, g\colon \mathbb{N} \to \mathbb{N}$ be such that $f(n) \leq n$, $g(n) = o(f(n))$. Then there is a language $L_f \subseteq \{a,b\}^\star$ that does not depend on g such that $L_f \in \mathscr{L}(\text{DFA})/f(n)$ and $L_f \notin \mathscr{L}(\text{NFA})/g(n)$.*

Proof. We shall present a language $L_f \subseteq \Sigma_2^\star$, such that for each n, L_f contains exactly one word of length n, called w_n. Moreover, this word is of the form $w_n \in \Sigma_2^{f(n)}0^{n-f(n)}$. This immediately implies that $L_f \in \mathscr{L}(\text{DFA})/f(n)$. In the rest of the proof we show how to specify w_n such that $L_f \notin \mathscr{L}(\text{NFA})/g(n)$ for any $g(n) = o(f(n))$.

It is sufficient to prove the claim for NFA's with binary advice alphabet: If there exists an NFA with advice alphabet of size k that accepts L_f with advice $g(n) = o(f(n))$, there also exists a NFA with binary advice alphabet that accepts L_f with advice $g(n)\log k = o(f(n))$.

Consider a fixed enumeration $A_1, A_2, \ldots$ of all 2-tape NFA's with binary advice alphabets. For a given 2-tape NFA A, we call a word $v \in \Sigma_2^\star$ a *n-singleton word* for A, if A when equipped with v as advice, accepts exactly one word u of length n (among all words of length n), and u is of the form $\Sigma_2^{f(n)}0^{n-f(n)}$.

Now we describe how to select w_n for a given n. Let d_n be the maximum integer such that $2d_n + 1 < f(n)$, i.e., $d_n := \lfloor (f(n) - 1)/2 \rfloor$. Consider the first 2^{d_n} automata in the fixed enumeration. Since there are at most 2^{d_n+1} binary words of length at most d_n, there are at most $2^{d_n+d_n+1}$ possible pairs (A, v) such that v is n-singleton for A, $|v| \leq d_n$. Since $d_n + d_n + 1 < f(n)$, there is a word $w_n \in \Sigma_2^{f(n)}$ such that w_n is never accepted as a single n-letter word by any of the first 2^{d_n} automata, with advice of size at most d_n.

Finally, we show that language L_f defined as above is not in $\mathscr{L}(\text{NFA})/g(n)$. Assume, for the sake of contradiction, that there is an automaton $A = A_j$ that recognizes L_f with some advice α of size $|\alpha(n)| = g(n)$. Obviously, $\alpha(n)$ is n-singleton for A, for any n. Since $g(n) = o(f(n))$, there is some large enough $\bar{n}$ such that $g(\bar{n}) \leq d_{\bar{n}}$ and $d_{\bar{n}} \geq \log j$. Hence, A with advice $\alpha(\bar{n})$ accepts $w_{\bar{n}}$ as the single $\bar{n}$-letter word. However $d_{\bar{n}} \geq \log j$ means $w_{\bar{n}}$ is never accepted as a single $\bar{n}$-letter word by any of the first j automata with any advice of size at most $d_{\bar{n}}$, thus it is not accepted with any advice of size $g(\bar{n})$. □

4 Conclusion and Open Questions

We showed that there are languages that cannot be recognized by DFA, regardless of advice size. Moreover, we showed that DFA cannot utilize more than exponential advice. However, we don't know any example of a language, where advice of exponential size is needed. Indeed, it may be the case that any language that can be recognized by DFA with advice, can be recognized also with polynomial advice. In particular, it would be interesting to know if all bounded languages can be recognized by DFA with polynomial advice.

We initiated the study of NFA with advice. We showed that there are languages that cannot be recognized with polynomial advice, but any language can be recognized with exponential advice. It is a natural task to characterize the languages in $\mathscr{L}(\text{NFA})/poly$.

Also, Küçük et al. showed in [6] that the language

$$\mathsf{EQUAL}_3 = \{w \in \Sigma_3 \mid \#_0(w) = \#_1(w) = \#_2(w)\}$$

cannot be recognized by a DFA with linear advice, but it can be recognized by a randomized FA with 1-sided bounded error with linear advice. It would be interesting to know whether randomization can help for larger advices: in particular, what languages can be recognized by randomized FA with polynomial advice.

Finally, one of the features of the model from [3] is that it is concerned only for the size of the advice, which can be split into several tapes. In our model, DFA with two advice tapes with exponential advice can recognize all languages, however, the power of multi-tape DFA's with limited advice is to be considered.

Dedication

The authors are very grateful to Juraj for his great support, many enlightening discussions and mountain adventures. He has always encouraged people to push their limits, and has been finding means to help them grow. His constant optimism has been a stable source of inspiration.

References

1. Damm, C., Holzer, M.: Automata that take advice. In: Wiedermann, J., Hájek, P. (eds.) MFCS 1995. LNCS, vol. 969, pp. 149–158. Springer, Heidelberg (1995). https://doi.org/10.1007/3-540-60246-1_121
2. Fischer, P.C., Rosenberg, A.L.: Multitape one-way nonwriting automata. J. Comput. Syst. Sci. **2**(1), 88–101 (1968)
3. Freivalds, R.: Amount of nonconstructivity in deterministic finite automata. Theor. Comput. Sci. **411**(38–39), 3436–3443 (2010)
4. Ibarra, O.H., Ravikumar, B.: Sublogarithmic-space turing machines, nonuniform space complexity, and closure properties. Math. Syst. Theory **21**(1), 1–17 (1988)
5. Karp, R.M., Lipton, R.J.: Turing machines that take advice. Enseign. Math. **28**(2), 191–209 (1982)
6. Küçük, U., Cem Say, A.C., Yakaryilmaz, A.: Finite automata with advice tapes. Int. J. Found. Comput. Sci. **25**(8), 987–1000 (2014)
7. Tadaki, K., Yamakami, T., Lin, J.C.H.: Theory of one-tape linear-time turing machines. Theor. Comput. Sci. **411**(1), 22–43 (2010)