On Computational Power of Partially Blind Automata

Pavol Ďuriš [1]

Department of Informatics
Faculty of Mathematics, Physics and Informatics
Comenius University
Mlynská dolina
842 48 Bratislava
Slovakia

Abstract In this paper we deal with 1-way multihead finite automata, in which the symbol under only one head (called *read* head) controls its move and other heads cannot distinguish the input symbols, they can only distinguish the end-marker from the other input symbols and they are called the *blind* head. We call such automaton a partially blind multihead automaton. We prove that partially blind $k + 1$-head finite automata are more powerful than such k-head finite automata. We show also that nondeterministic partially blind k-head finite automata languages are not closed under iteration and intersection for any $k \geq 2$, and moreover, deterministic partially blind k head finite automata languages are not closed under intersection, union, complementation and reversal for any $k \geq 2$. Finally we prove that deterministic partially blind k-head finite automata with endmarker are more powerful that such automata without endmarker for each $k \geq 4$.

Keywords: computational power, one-way multihead automata, blind head

1   Introduction

Ibarra and Ravikumar in [6] investigate one-way multihead finite automata (deterministic and also nondeterministic), in which all the heads except one are unable to distinguish the input symbols; they can only distinguish the end-marker from the other input symbols and they are called the *blind* head. The head that can distinguish the input symbol is called the *read* head. This model will be called as a partially multihead blind finite automaton.

The authors in [6] present several basic results concerning the partially blind multihead finite automata (FA). More particularly, they show the equivalence and the containment problems are undecidable for the nondeterministic model while they are decidable for the deterministic model. Moreover, they show how to simulate a counter machine by a partially blind multihead FA and vice-versa (with respect to the number of counters and heads needed). They also present some results concerning the simulation of partially blind multihead FA by a 2-way probabilistic finite automaton.

The basic results on the related models of multihead automata (in which all the heads can recognize the input symbols) and on the multicounter machines are presented in [3, 8, 1, 5, 2, 4].

Our main aim of this paper is to bring more results on partially blind multihead finite automata. We prove here that partially blind $(k+1)$-head finite automata are more powerful than such k-head finite automata. Similar result was shown by Yao and Rivest in [9] for one-way read-only multihead finite automata. Ibarra and Ravikunar present in [6] results on some basic closure properties (intersection, union and complementation) of family of languages recognizable by partially blind multihead finite automata (deterministic and nondeterministic as well). We extend these results by showing (among others) that the family of languages recognizable by partially blind nondeterministic automata with k heads are not closed under iteration for any k, and the family of languages corresponding to such deterministic automata are not closed under reversal. We also show that there is a NSPACE($\log n$)-complete problem that cannot be accepted by any nondeterministic partially blind multihead finite automaton. Analogical result was presented in [6] but only for automata with two heads. The problem whether a partially blind multihead finite automaton can accept a language that is complete for NSPACE($\log n$) was addressed also in [6]. Finally we show that deterministic partially blind k-head finite automata with endmarker are more powerful

1

than such automata without endmarker for $k \geq 4$. Ibarra and Ravikunar [6] showed that endmarker does not increase the computational power of nondeterministic partially blind k-head finite automata for any $k \geq 1$.

2   Preliminaries

By $|S|$ we denote cardinality of a set S and by $|x|$ we denote the length of a word x. In this section we also introduce the basic computational models from [6].

For $k \geq 1$ let k-NFA [let k-DFA] denote nondeterministic [deterministic] finite automaton with k one-way read-only input heads. We assume that the input is augmented with the endmarker \$.

Let M be a k-NFA where only one head (called the *read* head) is capable to distinguish the input symbols, the other $k - 1$ heads (called the *blind* heads) cannot differentiate the input symbols, they can differentiate only an input symbol from the endmarker. A configuration of M is $(k + 1)$-tuple $(p_1, p_2, \ldots, p_k, q)$, where p_i is the position of the i-th head from the left-end of the input tape, and q denotes the current state. We assume that the acceptance is by accepting state with all heads scan the endmarker. The language accepted by M is denoted by $L(M)$. We call M a *partially blind k-NFA* (shortly, k-NPBFA). The deterministic version is called a *partially blind k-DFA* (shortly, k-DPBFA). A *partially blind multihead*-NFA [DFA] is k-NPBFA [is k-DPBFA] for some $k \geq 1$.

By $\mathcal{L}(k\text{-NPBFA})$ [by $\mathcal{L}(k\text{-DPBFA})$] we denote the family of languages accepted by k-NPBFAs [by k-DPBFAs].

3   Results

3.1   k+1 Heads versus k Heads

Yao and Rivest showed in [9] that for every $k \geq 1$ there is a language R_{k+1} that can be recognized by a $(k + 1)$-DFA but cannot be recognized by any k-NFA. Consequently, $k + 1$ heads are more powerful than k heads for one-way deterministic as well nondeterministic finite automata. Here we prove an analogical result for partially blind multihead finite automata.

For every $k \geq 2$, let

$$J_k = \{w | w = a_1^{l_1} a_2^{l_2} \ldots a_{k+1}^{l_{k+1}} \# a_1^{m_1} a_2^{m_2} \ldots a_{k+1}^{m_{k+1}},$$
$$2k(l_i + m_i) = |w| - 1 \text{ for } 1 \leq i \leq k\}.$$

Theorem 1. *Let* $k \geq 2$. *Then* J_k *can be recognized by a* $(k+1)$-*DPBFA* A, *but cannot be recognized by any* k-*NPBFA.*

Proof. A recognizes $w = a_1^{l_1} \ldots a_{k+1}^{l_{k+1}} \# a_1^{m_1} \ldots a_{k+1}^{m_{k+1}}$ as follows. At the beginning, A moves all its blind heads one input tape cell (for $\#$). Then the read head traverses w and whenever it enters a tape cell containing a_i for some $1 \leq i \leq k$, then the read head stays at this tape call for $2k$ steps during which the i-th blind head traverses $2k$ tape cells. If the read head scans a_{k+1}, then no blind head moves. One can easily observe that $w \in J_k$ iff all blind heads scan \$ when the read head enters \$.

Now suppose to the contrary, that there would be a k-NPBFA B recognizing J_k. Let Q denote the set of states of B. Choose a positive integer n large enough and let

$$J_k^n = \{w | w \in J_k, |w| = n, w = a_1^{l_1} \ldots a_{k+1}^{l_{k+1}} \# a_1^{m_1} \ldots a_{k+1}^{m_{k+1}}, 2\Sigma_{i=1}^{k+1} l_i = n - 1\}.$$

Thus, the symbol $\#$ is in the middle of each $w \in J_k^n$. For each $w \in J_k^n$ choose any accepting computation of B on w and let $C(w)$ be the configuration at which the read head of B enters $\#$ during that computation. It holds

$$((n-1)/(2k))^k \leq |J_k^n|, \tag{1}$$

since for each sequence of integers $l_1, l_2, \ldots, l_k$, where $0 \leq l_i \leq (n-1)/(2k)$ for $1 \leq i \leq k$, there is a word $w \in J_k^n$ with such values $l_1, l_2, \ldots, l_k$. On the other hand, $|\{C(w) | w \in J_k^n\}| \leq |Q|(n+1)^{k-1})$, where $|w| = n$, since the position of each blind head of B is between 1 and $|w| + 1 = n + 1$, and the read head of B reads $\#$ in the middle of w, when B is in any configuration $C(w)$. Clearly, $|Q|(n+1)^{k-1} < ((n-1)/(2k))^k$ for n large enough, and hence, by(1), there are two different words $w = w_1 \# w_2$ and $w' = w_1' \# w_2'$ in J_k^n with $C(w) = C(w')$. It means that B has to accept also $\tilde{w} = w_1 \# w_2'$, since B can behave by the same way on w and on $\tilde{w}$ during the computation from the initial configuration to the configuration $C(w)$;

note that during such computation the read head traverses in both cases the same string w_1. Similarly, B can behave by the same way on w' and on $\tilde{w}$ during the computation from the configuration $C(w') = C(w)$ to an accepting configuration; note that during such computation the read head traverses in both cases the same string w'_2. But $\tilde{w} \notin J_k$, since $w_1 \neq w'_1$ or $w_2 \neq w'_2$ (recall $w \neq w'$), and moreover, one can easily observe that for each word v_1 [for each word v_2] there is at most one word v_2 [there is at most one word v_1] width $|v_1| = |v_2|$ such that $v = v_1 \# v_2 \in J_k$. Hence B cannot accept J_k - a contradiction. $\square$

Theorem 2. *For every $k \geq 1$ the following holds.*
(a) $\mathcal{L}((k+1)\text{-DPBFA}) \subset \mathcal{L}(k\text{-DPBFA})$,
(b) $\mathcal{L}((k+1)\text{-NPBFA}) \subset \mathcal{L}(k\text{-NPBFA})$.

Proof. Theorem 1 yields the proof. $\square$

3.2   Closure Properties

The following two Theorems were proved in [6].

Theorem 3 [6]. *The family of languages accepted by partially blind multihead NFAs is closed under union and intersection, but not closed under complementation.*

Theorem 4 [6]. *The family of languages accepted by partially blind multihead DFAs is closed under union, intersection and complementation.*

We prove here the following related results.

Theorem 5. *Neither $\mathcal{L}(k\text{-NPBFA})$ nor $\mathcal{L}(k\text{-DPBFA})$ is closed under intersection for any $k \geq 2$.*

Proof. Let us prove Theorem only for $\mathcal{L}(k\text{-NPBFA})$. (The proof for $\mathcal{L}(k\text{-DPBFA})$ is similar.) Suppose to the contrary that $\mathcal{L}(k\text{-NPBFA})$ is closed under intersection for some $k \geq 2$. For every j, $1 \leq j \leq k$, let

$$P_{j,k} = \{w \mid w = a_1^{l_1} a_2^{l_2} \ldots a_{k+1}^{l_{k+1}} \# a_1^{m_1} a_2^{m_2} \ldots a_{k+1}^{m_{k+1}}, 2k(l_j + m_j) = |w| - 1\}.$$

Clearly, $P_{j,k} \in \mathcal{L}(2\text{-DPBFA}) \subseteq \mathcal{L}(k\text{-DPBFA}) \subseteq \mathcal{L}(k\text{-NPBFA})$ for every $1 \leq j \leq k$, (see the proof of Theorem 1). By the assumption above, $\mathcal{L}(k\text{-NPBFA})$

4

is closed under intersection. Hence, $\bigcap_{j=1}^{k} P_{j,k}$ should belong in $\mathcal{L}(k\text{-NPBFA})$. But it contradicts Theorem 1, since $\bigcap_{j=1}^{k} P_{j,k} = J_k$. $\square$

Theorem 6. $\mathcal{L}(k\text{-DPBFA})$ *is not closed under union for any* $k \geq 2$.

Proof. One can easy observe that $\mathcal{L}(k\text{-DPBFA})$ is closed under complementation. If $\mathcal{L}(k\text{-DPBFA})$ were closed under union, then it should be closed also under intersection (because of De Morgan's laws), but this contradicts Theorem 5. $\square$

Theorem 7. $\mathcal{L}(k\text{-NPBFA})$ *is not closed under iteration for any* $k \geq 2$.

Proof. Clearly, the language $L = \{a^l b^l | l \geq 0\}$ can be recognized by a 2-DPBFA A. Suppose to the contrary that there would be a k-NPBFA B for some k with the set Q of states recognizing L^*. Let us consider a word $w \in L^*$ of the form $w = u_1 u_2 \ldots u_k$, where $u_i = a^m b^m \ldots a^m b^m = (a^m b^m)^d$ for $1 \leq i \leq k$,

$$d = 2m(2(k-1)(2km+1))^{k-1}, \tag{2}$$

and $m = |Q| + 1$. Let us choose any accepting computation α of B on w. Clearly, there is an index j such that no blind head enters \$ while the read head traverses the j-th substring u_j during α, since B has only $k-1$ blind heads but w has k u_i's. Now mark each substring a^m of u_j having the following property: No blind head of B traverses more than $2(k-1)(2km+1)$ input tape cells while the read head of B traverses (during α) the marked substring a^m. The number of all unmarked substrings a^m of u_j is at most $d/2$, since $k-1$ blind heads cannot traverse altogether more than $(k-1)|w\$| = (k-1)(2kmd+1) < (2(k-1)(2km+1))d/2$ input tape cells during α over the unmarked substrings a^m of u_j. Thus, the number of marked strings a^m of u_j is at least $d/2$.

    Let $h_0(t)$ [let $h_i(t)$] be the position of the read [of the i-th blind] head of B at the step t of α for $1 \leq i \leq k-1$. For each marked substring a^m of u_j, let us choose any two steps $t_1 < t_2$ of α such that the read head of B still scans that marked substring a^m at t_1 and also at t_2, $h_0(t_1) < h_0(t_2)$, and moreover, B is in the same state at these two steps t_1 and t_2. (Since $m = |Q| + 1$, see above, there are such t_1, t_2.) Clearly, $1 \leq h_0(t_2) - h_0(t_1) \leq m - 1$ and $0 \leq h_i(t_2) - h_i(t_1) \leq 2(k-1)(2km+1) - 1$ for $1 \leq i \leq k-1$. This means together with (2) that there are two marked substrings a^m of u_j - one with chosen two steps $\tau_1 < \tau_2$ and the second one with chosen two

5

steps $\tau_1' < \tau_2'$ such that $1 \leq h_0(\tau_2) - h_0(\tau_1) = h_0(\tau_2') - h_0(\tau_1') \leq m - 1$ and $0 \leq h_i(\tau_2) - h_i(\tau_1) = h_i(\tau_2') - h_i(\tau_1') \leq 2(k-1)(2km+1) - 1$ for $1 \leq i \leq k - 1$. Now let us modify the input $w \in L^*$ mentioned above as follows. Replace the marked substring a^m with the chosen two steps τ_1 and τ_2 by the string $a^{m-(h_0(\tau_2)-h_0(\tau_1))}$, and similarly, replace the marked substring a^m with the chosen two steps τ_1' and τ_2' by the string $a^{m+(h_0(\tau_2')-h_0(\tau_1'))}$. The resulting string denote by w'. Clearly, $w' \notin L^*$, but w' is accepted by B via the accepting computation α' obtained from α by deleting the computation corresponding to the steps $\tau_1 + 1, \tau_1 + 2 \ldots, \tau_2$, and inserting to α (between τ_2' and $\tau_2' + 1$) the computation corresponding to the steps $\tau_1' + 1, \tau_1' + 2, \ldots, \tau_2'$. $\square$

Let

$$L_1 = \{b0^j1^l \# x | x \in \{0,1\}^*, 1 \leq j \leq |x|, b \in \{0,1\},$$
$$l \geq 1, (|x| - j + 1)\text{-st bit of } x \text{ is } b\}.$$

Lemma 1. L_1 can be recognized by a 2-DPBFA A.

Proof. First, A stores the bit b in its finite memory. Then, while the read head of A traverses 0^j, the blind head of A moves two times faster, and after the read head enters the leftmost symbol 1 (between b and $\#$), both heads moves one input tape cell per one step until the blind head enters the endmarker $\$$. Clearly, at this time the read head scans the $(|x| - j + 1)$-st symbol of x and thus, A can compare it to the stored value b. Hence, A can recognize L_1. $\square$

Lemma 2. Reverse of L_1 cannot be recognized by any k-DPBFA for any k.

Proof. Suppose to the contrary that reverse of L_1 can be recognized by a k-DPBFA B for some k. Let Q denote the set of states of B. Choose any positive integer m such that

$$|Q|(2m+3)^{k-1} < 2^m. \tag{3}$$

Let us consider the language

$$S_{rev} = \{y\#1^{m-1}0b | y \in \{0,1\}^m, b \in \{0,1\}, \text{ the leftmost bit of } y \text{ is } b\}.$$

B has to accept each $w \in S_{rev}$, since S_{rev} is a subset of reverse of L_1. For each word $w \in S_{rev}$, let $C(w)$ be the configuration at which the read head of

B enters $\#$ during the accepting computation of B on w. Since $|w| = 2m+2$ for each $w \in S_{rev}$, the position of each blind head of B is between 1 and $|w| + 1 = 2m + 3$ when the reading head enters $\#$ of any word $w \in S_{rev}$. Hence,

$$|\{C(w)|w \in S_{rev}\}| \leq |Q|(2m + 3)^{k-1}. \tag{4}$$

By (3), (4) and by the fact that $2^m = |S_{rev}|$, we have that there are two different words $w = y\#1^{m-1}0b$ and $w' = y'\#1^{m-1}0b'$ in S_{rev} with $C(w) = C(w')$.

Since $y, y' \in \{0,1\}^m$ and $y \neq y'$, there is an index i such that the i-th symbol of y and the i-th symbol of y' are different. W.l.o.g. we assume that the i-th symbol of y is 0 and the i-th symbol of y' is 1. Let $\bar{w} = y\#1^{m-i}0^i b$ and $\hat{w} = y'\#1^{m-i}0^i b$, where $b = 0 = i$-th symbol of y. One can easily observe that $\bar{w}$ is in reverse of L_1, but $\hat{w}$ is not in reverse of L_1.

Let α [let α'] denote the computation of B on w [on w'] from the initial configuration C_0 to the configuration $C(w)$ [to the configuration $C(w')$], i.e. when the reading head of B enters the symbol $\#$ of w [of w'], (see above). Since w and $\bar{w}$ have the same prefix $y\#$, B can perform the computation α also on $\bar{w}$. By the same reason, B can perform the computation α' also on $\hat{w}$. Let β denote the computation of B on $\bar{w}$ from the configuration $C(w)$ to an accepting configuration C_{acc}; there is such computation, since B has to accept $\bar{w}$ in reverse of L_1, (see above). Since $\bar{w}$ and $\hat{w}$ have the same suffix $\#1^{m-i}0^i$, B can perform the computation β also on $\hat{w}$. Thus, by joining the computations α' and β, (recall $C(w) = C(w')$, see above), we get an accepting computation of B on $\hat{w}$, - a contradiction, since $\hat{w}$ is not in reverse of L_1, (see above). Hence, B cannot accept reverse of L_1. $\square$

Theorem 8. $\mathcal{L}$(k-DPBFA) is not closed under reversal for any $k \geq 2$.

Proof. The proof follows from Lemmas 1 and 2. $\square$

3.3 $NSPACE(\log n)$- Complete Problems

The following interesting question is mentioned in [6]: Is there a language that is complete for NSPACE($\log n$) that can be accepted by a partially blind

multihead-NFA? It was shown in [7] that

$$L_2 = \{x\#w_1\#w_2\#\ldots\#w_k| \text{ there exist indices } i_1 < i_2 < \cdots < i_r$$
$$\text{such that } x = w_{i_1} w_{i_2} \ldots w_{i_r}\}$$

is complete for NSPACE($\log n$). Moreover, it was proved in [6] that the language L_2 cannot be accepted by any partially blind 2-NFA. We prove here the following more general result.

Theorem 9. *The language L_2 cannot be accepted by any k-NPBFA for any $k \geq 1$.*

Proof. Assume to the contrary that there is a k-NPBFA A accepting L_2 for some $k \geq 1$. Let Q denote the set of states of A. Choose any positive integer n such that

$$|Q|(2n+2)^{k-1} < 2^n. \tag{5}$$

Consider the language $S_n = \{w|w = v\#v, v \in \{0,1\}^n\}$. Since A accepts L_2, and clearly, $S_n \subseteq L_2$, then A has to accept also each $w \in S_n$. For each $w \in S_n$ choose any accepting computation of A on w, and let $C(w)$ be the configuration at which the read head of A enters $\#$ during that accepting computation. Since $|w| = 2n + 1$ for each $w \in S_n$, then $|\{C(w)|w \in S_n\}| \leq |Q|(2n+2)^{k-1} < 2^n = |S_n|$, by (5). Thus, there are two different words $w = v\#v$ and $w' = v'\#v'$ in S_n with $C(w) = C(w')$. It means that A has to accept also $\bar{w} = v\#v'$ since A can behave by the same way on w and on $\bar{w}$ during the computation from the initial configuration to the configuration $C(w)$; note that during such computation the read head traverses in both cases the same string $v\#$. Similarly, A can behave by the same way on w' and on $\bar{w}$ during the computation from the configuration $C(w') = C(w)$ to an accepting configuration; note that during such computation the read head traverses in both cases the same string $\#v'$. But $\bar{w} \notin L_2$, since clearly, $v \neq v'$. Hence, A cannot accept L_2 - a contradiction. $\square$

3.4 Power of Endmarker

In this section, we consider k-DPBFA *without endmarker*. This model is very similar to k-DPBFA mentioned above with the exception that the input is not augmented with the endmarker. Thus, we assume in such a case

that acceptance is by accepting state with all heads falling off the input tape. More particularly, an input w is accepted by a k-DPBFA M without enmarker, if there is an computation of M on w from the initial configuration $(1, 1, \ldots, 1, q_0)$ to an accepting configuration $(n+1, n+1, \ldots, n+1, f)$, where $n = |w|$, q_0 is the start state and f is an accepting state. Similarly, we can consider also the nondeterministic version of the model without enmarker.

It was shown in [6] that partially blind k-NFAs with endmarker are equivalent to partially blind k-NFAs without endmarker for every $k \geq 1$. Here we prove that partially blind k-DFAs with endmarker are more powerful than partially blind k-DFAs without endmarker for every $k \geq 4$. To do so we need the following two Lemmas.

Let

$$L_3 = \{x \# y | x, y \in \{0, 1\}^*, j = \lceil |x \# y|/5 \rceil, |x| \geq j, |y| \geq j,$$
$$j\text{-th bit of } x \text{ is } j\text{-th bit of } y\}.$$

Lemma 3. L_3 can be recognized by a 4-DPBFA A with the endmarker.

Proof. The read head of A can find the j-th bit of x as follows. While the read head of A traverses x, the first blind head of A moves five times faster until it reaches the end marker \$ - at this time the read head scans the j-th bit of x, and A stores it in its finite memory. During this process the second blind head of A moves by the same way as the read head and the third blind head of A does not move. Thus, when the read head scans the j-th bit of x, the difference between the position of the second and the third blind head is exactly $j - 1$. Then the second blind head moves to the end marker \$ and the third blind head moves by the same way as the second blind head. Hence, when the second blind head enters \$, then the distance of the third blind head from \$ is $j - 1$. When the second blind head enters \$, the read head moves just after \#. Next, the read head and the third blind head move by the same way until the third blind head enters \$. At this time the read head scans the j-th bit of y, and thus, A can compare it to the stored j-th bit of x, (see above). Hence, A can recognize L_3. $\square$

Lemma 4. L_3 cannot be recognized by any k-DPBFA without the endmarker for any k.

Proof. Suppose to the contrary that L_3 can be recognized by a k-DPBFA B without enmarker for some k. Let Q denote the set of states of B. Choose

any positive integer m such that

$$|Q|(5m+2)^{k-1} < 2^m. \tag{6}$$

For each $z \in \{0,1\}^m$ let us construct a word $w \in L_3$ of the form $w = 0^m z \# v^{3m}$, where v is the leftmost bit of z. Thus, $j = \lceil |0^m z \# v^{3m}|/5 \rceil = \lceil (5m+1)/5 \rceil = m+1$, and hence $w = 0^m z \# v^{3m} \in L_3$. For each constructed word w, let $C(w)$ be the configuration at which the read head of B enters $\#$ during the accepting computation of B on w. Since $|w| = 5m+1$ for each constructed word w, then the position of each blind head of B is between 1 and $|w| + 1 = 5m+2$ when the reading head enters $\#$ of any constructed word w. Hence,

$$|\{C(w)|w \text{ is constructed for some } z \in \{0,1\}^m\}| \leq |Q|(5m+2)^{k-1}. \tag{7}$$

By (6), (7) and by the fact that $2^m = |\{0,1\}^m|$, we have that there are two different constructed words $w = 0^m z \# v^{3m}$ and $w' = 0^m z' \# v'^{3m}$ in L_3 with $C(w) = C(w')$. Since $z, z' \in \{0,1\}^m$ and $z \neq z'$, there is an index i such that the i-th symbol of z and the i-th symbol of z' are different. W.l.o.g. we assume that the i-th symbol of z is 0 and the i-th symbol of z' is 1. Let $\bar{w} = 0^m z \# 0^h$ and $\hat{w} = 0^m z' \# 0^h$, where $m + i = \lceil |0^m z \# 0^h|/5 \rceil$ for some h. One can easily observe (using the fact that $m + 1 \leq m + i \leq 2m$) that $3m \leq h \leq 8m - 1$. Thus, $\bar{w} \in L_3$ (since $(m+i)$-th symbol of the word $0^m z$ and of the word 0^h is 0), but $\hat{w} \notin L_3$ (since $(m+i)$-th symbol of the word $0^m z'$ is 1). Moreover, $|w'| = |w| \leq |\bar{w}| = |\hat{w}|$, since $3m \leq h$, see above.

Let α [let α'] denote the computation of B on w [on w'] from the initial configuration C_0 to the configuration $C(w)$ [to the configuration $C(w')$], i.e. when the read head of B enters the symbol $\#$ of w [of w'], (see above). Since w and $\bar{w}$ have the same prefix $0^m z \#$, and the read head of B traverses that prefix during α, and $|w| \leq |\bar{w}|$, (see above), then B can perform the computation α also on $\bar{w}$. Similarly, B can perform the computation α' also on $\hat{w}$. Let β denote the computation of B on $\bar{w}$ from the configuration $C(w)$ to an accepting configuration C_{acc}; there is such computation, since B has to accept $\bar{w} \in L_3$, (see above). Since $\bar{w}$ and $\hat{w}$ have the same suffix $\# 0^h$, B can perform the computation β also on $\hat{w}$. Thus, by joining the computations α' and β, (recall $C(w) = C(w')$, see above), we get an accepting computation of B on $\hat{w}$, - a contradiction, since $\hat{w} \notin L_3$, (see above). Hence, B cannot accept L_3. $\square$

Theorem 10. *Partially blind k-DFAs with endmarker are more powerful than partially blind k-DFAs without endmarker for every $k \geq 4$.*

Proof. The proof follows from Lemmas 3 and 4. □

References

[1] Baker, B., Book, R.: Reversal-bounded multipushdown machines. *Journal of Computer and System Sciences*, 8(1974), 315-332.

[2] Greibach, S.: Remarks on the complexity of nondeterministic counter languages, *Theoretical Computer Science*, 1(1976), 269-288.

[3] Harrison, M., Ibarra, O.: Multi-tape and multi-heads pushdown automata. *Information and Control*, 13(1968), 433-470.

[4] Hopcroft, J., Ullman, J.: Introduction to Automata, Formal Languages and Theory of Computation, Addision-Wesley, Inc. Reading, MA (1979).

[5] Ibarra, O.: Reversal -bounded counter machines and their decision problems, *Journal of the ACM*, 25(1), (1978).

[6] Ibarra, O., Ravikumar, B.: On partially blind multihead finite automata, *Theoretical Computer Science*, 365(1), (2006), 190-199.

[7] Jenner, B.: Knapsack problems for NL, *Information Processing Letters*, 54(3), (1995), 169-174.

[8] Sudborough, I.: A note on tape-bounded complexity clases and linear contex-free languages, *Journal of the ACM*, 22(4), (1975), 499-500.

[9] Yao, A., Rivest, R.: k+1 heads are better than k, *Journal of the ACM*, 25(2), (1978), 337-340.