INTERNATIONAL COMPUTER SCIENCE INSTITUTE

1947 Center St. • Suite 600 • Berkeley, California 94704-1198 • (510) 643-9153 • FAX (510) 643-7684

Interactive Proof Systems with Public Coin: Lower Space Bounds and Hierarchies of Complexity Classes

Maciej Liśkiewicz*

TR-96-047

November 1996

Abstract

This paper studies small space-bounded interactive proof systems (IPSs) using public coin tosses, respectively Turing machines with both nondeterministic and probabilistic states, that works with bounded number of rounds of interactions. For this model of computations new impossibility results are shown. As a consequence we prove that for sublogarithmic space bounds, IPSs working in k rounds are less powerful than systems of $2k^{k-1}$ rounds of interactions. It is well known that such a property does not hold for polynomial time bounds. Babai showed that in this case any constant number of rounds can be reduced to 2 rounds.

*On leave of Institute of Computer Science, University of Wrocław. The research was supported by KBN Grant 2 P301 034 07.

1 Introduction

Interactive proof systems (IPSs) working in constant space seems to be very powerful devices. Dwork and Stockmeyer ([DwSt92]) showed that any language recognized in deterministic exponential time has an IPS, where the verifier is a probabilistic finite automaton. The power of computation of constant-space-bounded IPSs becomes more realistic when the random moves of the verifier are known to the prover, as it is the case for Arthur-Merlin games, resp. for Turing machines with both nondeterministic and probabilistic states. Condon ([Co89]) proved that any language recognized by such system is in P even when the verifier works in logarithmic space. Hence the difference between the privet and the public coin tossing is significant. This property does not hold for the corresponding polynomial time classes (see [GoSi86]).

In their paper [DwSt92], Dwork and Stockmeyer showed some further separating results proving a number of strong lower bounds for probabilistic finite automata and for small space-bounded IPSs tossing public coin. They proved for example that the language

$$\texttt{CENTER} \;:=\; \{w0x \;:\; w,x \in \{0,1\}^* \quad \text{and} \quad |w| = |x|\}\;,$$

cannot be recognized by a sublogarithmic space-bounded probabilistic Turing machine with any error probability $\epsilon < 1/2$. On the other hand, there exists a constant-space-bounded interactive proof system (with public coin flips) for CENTER: in [DwSt92] a protocol for this language is given. Thus at least in case of sublogarithmic space bounds interactive proof systems enable the verifier to accept something it cannot accept on its own. It seems, however, that to accept CENTER in constant space, a huge number of rounds of interactions is crucial. The protocol of [DwSt92], for example, works in exponential number of rounds and it is unknown if the language can be recognized in smaller number of interactions. Hence, the lower and upper bound results of [DwSt92] prove that 1 probabilistic round is less powerful than exponential number of rounds and it is open whether there is also a difference in power of computations between IPSs of R and resp. R' interactions for any function R and R', with $1 \le R < R' \le \exp$. In the paper we will prove that for space bounds in $\mathcal{SUBLOG} := \Omega(\mathrm{llog}) \cap o(\log)$ the problem has a positive answer for any function $R(n) \le k$ and $R'(n) \ge 2k^{k-1}$, where k is an arbitrary constant. Here llog denotes the logarithmic function log iterated twice.

We refer to TMs which have both probabilistic and nondeterministic states as stochastic Turing machines (STMs). Denote by $AMSpace(S)$ the set of languages that can be accepted by STMs, equivalently by public coin IPSs, with space S. Let $MA_kSpace(S)$, resp. $AM_kSpace(S)$, denote the set of languages that can be accepted by such machines making at most $k-1$ alternations between nondeterministic and probabilistic configurations, equivalently making at most $k-1$ interaction, and starting in nondeterministic, resp. probabilistic, mode (for such a machine we also say that it works in k rounds). $BPSpace(S)$ denotes the class of languages accepted by S-space-bounded probabilistic TMs with bounded error. Then language CENTER yields the separations

$$BPSpace(S) = AM_1Space(S) \;\subset\; AMSpace(S)$$

for any $S \in o(\log)$. In this paper we strengthen these result for $S \in \mathcal{SUBLOG}$ showing an infinite hierarchy of $AM_kSpace(S)$ complexity classes between $AM_1Space(S)$ and $AMSpace(S)$.

Let $\texttt{BIN}(m) := \mathrm{bin}(0)\#\,\mathrm{bin}(1)\#\,\mathrm{bin}(2)\#\ldots\#\,\mathrm{bin}(m)$, where $bin(i)$ is the binary representation of the number i. We define for any positive integer k

$$\begin{aligned}\texttt{PATTERN}_k \;:=\; \{&W_1\#\ldots\#W_k\#u\#\texttt{BIN}(2^d) :\; W_1,\ldots,W_k,u \in \{0,1,*\}^+,\; |u| = d \\ &\text{for some } d \in \mathbb{N}, \text{ and } u \text{ is a substring of } W_i, \text{ for } i = 1,\ldots,k\;\}\;.\end{aligned}$$

One of the main results of this paper says that for any integer $k \ge 2$

$$\texttt{PATTERN}_{k^{k-1}} \;\notin\; AM_kSpace(o(\log)) \;\cup\; MA_kSpace(o(\log))\;. \tag{1}$$

To prove these lower bounds we show that any stochastic Turing machine M accepting an input $W_1\#\ldots\#W_{k^{k-1}}\#u\#\mathtt{BIN}(2^d) \in \mathtt{PATTERN}_{k^{k-1}}$ in k rounds, alternates between a nondeterministic and probabilistic move on some block W_i with very small probability. Using methods of [DwSt92] we choose the string W_i in such a way that M cannot distinguish in one round this specific string from some "wrong" string. Hence, if one replaces W_i by the wrong string, M to detect this difference has to alternate on this string; otherwise it behaves as previously. But because M alternates on the i-th block with very small probability hence it can detect the difference with small probability, too.

From (1) it follows that $\mathtt{PATTERN}_2$ cannot be accepted by 2-round STMs. In [LiRe96] we showed a little bit more for such machines. Namely, we proved that any $o(\log)$-space-bounded STM cannot recognize even $\mathtt{PATTERN}_1$ in 2 rounds when starting in probabilistic mode. On the other hand this language is in $MA_2Space(\mathrm{llog})$. Hence an optimal lower bound on the number of rounds has been founded in this case. Using a method of Freivalds ([Fr79]), we generalize in this paper the upper bound for recognizing $\mathtt{PATTERN}_1$ as follows:

$$\mathtt{PATTERN}_k \ \in \ MA_{2k}Space(\mathrm{llog}) \ , \tag{2}$$

for any $k \geq 2$. Note that the above upper bound does not match our lower bound (1). In fact the gap is rather large.

Therefore, we obtain the following separations:

Theorem 1 *For any integer $k \geq 2$ it holds that*

$$MA_{2k^{k-1}}Space(\mathrm{llog}) \ \not\subseteq \ AM_kSpace(o(\log)) \ \cup \ MA_kSpace(o(\log) \ .$$

This implies that round/alternation hierarchy for sublogarithmic space-bounded AM_k machines is infinite, similar as for standard alternating TMs (see e.g. [LiRe96a]):

Corollary 1 *For any function $S \in \mathcal{SUBLOG}$ and any integer $k \geq 2$*

$$AM_kSpace(S) \ \cup \ MA_kSpace(S) \ \subset \ MA_{2k^{k-1}}Space(S) \ .$$

This property does not hold for the corresponding polynomial time classes $AM_kTime(\mathcal{POL})$. Here, $\mathcal{POL}$ denotes the set of all polynomials. In [Ba85] Babai has shown that for polynomial time any constant number of rounds can be reduced to two rounds, that is

$$AM_2Time(\mathcal{POL}) \ = \ AM_kTime(\mathcal{POL}) \ ,$$

for any integer $k \geq 2$.

In the paper it is also shown an astonishing lower bound on space and on the number of rounds for recognizing the complement of $\mathtt{PATTERN}_1$. Though the language is in $MA_2Space(\mathrm{llog})$, we prove that that for any integer function $R \in O(\log \,/\, \mathrm{llog})$, its complement

$$\overline{\mathtt{PATTERN}_1} \ \notin \ AM_RSpace(o(\log)) \ . \tag{3}$$

Obviously, this result implies the following

Theorem 2 *For any $S \in \mathcal{SUBLOG}$, and for any integer function R, with $3 \leq R \in O(\log \,/\, \mathrm{llog})$ the classes $AM_RSpace(S)$ and $MA_RSpace(S)$ are not closed under complement.*

The remainder of this paper is organized as follows. In Section 2 some definitions and notions are introduced. Section 3 contains the proofs of our lower bound results (1) and (3). In Section 4 a machine for $\mathtt{PATTERN}_k$ is described what proves (2).

2 Definitions

A computation of a stochastic machine M on an input X can be described by a computation tree. In a probabilistic state a stochastic machine M chooses among the successor configurations with equal probability. To define acceptance of X, for each nondeterministic configuration one chooses a successor that maximizes the probability of reaching an accepting leaf. The acceptance probability of X is then given by the acceptance probability of the starting configuration in this truncated tree. M accepts a language L in space S if

- for all $X \in L$, the probability that M accepts X is more than $3/4$,
- every $X \notin L$ is accepted with probability less than $1/4$, and
- M never uses more than $S(|X|)$ space.

We say that M accepts the language L in R rounds if with any input X M makes at most $R(|X|)-1$ alternations between nondeterministic and probabilistic configurations. The above machines are equivalent the to S-space-bounded IPSs using public coin tosses and working with R rounds of interactions between verifier and prover (see [DwSt92] for a formal definition of S-space-bounded IPSs).

Let M be an STM. We will assume that M is equipped with a two-way read-only input tape and a single read-write work tape. A *memory state* of M is an ordered triple $\alpha = (q, u, i)$, where q is a state of M, u a string over the work tape alphabet, and i a position in u (the location of the work tape head). By $|\alpha|$ we denote the length of the string u of the memory state α. A *configuration* of M on an input X is a pair (α, j) consisting of a memory state α and a position j with $0 \leq j \leq |X|+1$ of the input head. $j = 0$ or $j = |X|+1$ means that this head scans the left, resp. the right end-marker. Let $h(\alpha, j) \stackrel{\text{def}}{=} j$ be the function describing the input head position for an configuration (α, j). We say that a configuration (α, j), with $\alpha = (q, u, i)$, is nondeterministic, probabilistic (or random), accepting or rejecting, according to q.

We call a phase of computation of M a *probabilistic* (or *random*) *round* if M starts the phase in a probabilistic configuration and makes only probabilistic steps during the phase and finally reaches a non-probabilistic configuration. Analogously we call a phase of computation a *nondeterministic round* if M starts in a nondeterministic configuration and performing only nondeterministic steps during the phase reaches probabilistic, accepting or rejecting state. Let for a probabilistic configuration c and a nondeterministic, accepting or rejecting configuration configuration c'

$$\mathcal{R}[c, c', X]$$

denote the probability that M with X on its input tape and starting in c reaches the configuration c' in a probabilistic round. Let for a nondeterministic configuration c

$$\mathcal{N}(c, c', X)$$

be 1 if M starting in c on the input X reaches the configuration c' in a nondeterministic round; otherwise $\mathcal{N}(c, c', X) = 0$. Denote by

$$\mathcal{A}_k[c, X]$$

the probability that M accepts the input X in k or less rounds starting in configuration c. Formally let for the accepting configuration c, $\mathcal{A}_0[c, X] := 1$ and for the rejecting c, $\mathcal{A}_0[c, X] := 0$. Then for any $k \geq 1$ if c is probabilistic, then

$$\mathcal{A}_k[c, X] \;:=\; \sum_{\substack{c'-\text{nondet.} \\ \text{or accept}}} \mathcal{R}[c, c', X] \cdot \mathcal{A}_{k-1}[c', X] \;,$$

and if c is nondeterministic, then

$$\mathcal{A}_k[c,X] \;:=\; \max_{\substack{c'-\text{random}\\ \text{or accept}}} \{\mathcal{A}_{k-1}[c',X] \;:\; \mathcal{N}(c,c',X)\;\}.$$

Let $\mathcal{A}_k[X] := \mathcal{A}_k[c_0,X]$, where c_0 be the initial configuration of M.

3 Lower Bounds

In this section we give proofs of our impossibility results:

Theorem 3 *For any $S \in o(\log)$, an S-space-bounded STM cannot recognize*

(1) the language $\texttt{PATTERN}_{k^{k-1}}$ *in k rounds, for any integer $k \geq 2$, and*

(2) the complement of $\texttt{PATTERN}_1$ *in R rounds, for any $R \in O(\log / \operatorname{llog})$.*

We start with definitions and technical preliminaries that were originally be showed in [DwSt92] for probabilistic Turing machines. Here we extend them for STMs.

The *word probabilities* of M on a word Z over the input alphabet of M is defined as follows. A starting condition for the word probability is a pair $\langle \alpha, h\rangle$ where α is a probabilistic memory state of M and $h \in \{\text{Left}, \text{Right}\}$ what means that M starts according to the value of h on the leftmost or on the rightmost symbol of Z in memory state α. A stopping condition for the word probability is either:

- a pair $\langle \alpha, h\rangle$ as above meaning that in a probabilistic round the input head falls off according to h the leftmost, resp. the right most symbol of Z with M in memory state α,
- "Accept" meaning that M in a probabilistic round halts in the accepting state before the input head falls off either end of Z,
- "Reject" meaning that M in a probabilistic round halts in the rejecting state before the input head falls off either end of Z,
- "Alter" meaning that within Z M alternates from a probabilistic to a nondeterministic round, or
- "Loop" meaning that within Z the probabilistic computation of M loops forever.

For each starting condition σ and each stopping condition τ, let $p(Z,\sigma,\tau)$ be the probability that stopping condition occurs given that M started in starting condition σ on Z.

Computations of a probabilistic round of M are modeled by Markov chains with finite state space, say $1, 2, \ldots, s$ for some s. A particular Markov chain is completely defined by its matrix $R = \{r_{ij}\}_{1\leq i,j\leq s}$ of transition probabilities. If the Markov chain is in state i, then it next moves to state j with probability r_{ij}. The chains we consider have the designated starting state, say, state 1, and some set T_R of trapping states, so $r_{tt} = 1$ for all $t \in T_R$. For $t \in T_R$, let $p^*[t,R]$ denote the probability that Markov chain R is trapped in state t when started in state 1.

Let $\beta \geq 1$. Say that two numbers r and r' are β-close if either $r = r' = 0$, or $r > 0$, $r' > 0$, and $\beta^{-1} \leq r/r' \leq \beta$. Two Markov chains $R = \{r_{ij}\}_{1\leq i,j\leq s}$ and $R' = \{r'_{ij}\}_{1\leq i,j\leq s}$ are β-close if r_{ij} and r'_{ij} are β-close for all pairs i, j.

Lemma 1 ([DwSt92]) *Let R and R' be two s-state Markov chains which are β-close, and let t be a trapping state of both R and R'. Then $p^*[t,R]$ and $p^*[t,R']$ are β^z-close where $z = 2s$.*

We characterize a word Z according to a nondeterministic round of M on Z by *word transitions*. As previously, a starting condition for the word transition is a pair $\langle \alpha, h \rangle$ where α is a nondeterministic memory state of M and $h \in \{\text{Left}, \text{Right}\}$. A stopping condition for the word transition is either a pair $\langle \alpha, h \rangle$ as above, "Accept" meaning that M in a nondeterministic round halts in the accepting state before the input head falls off either end of Z, or "Reject" meaning that M in a nondeterministic round halts in the rejecting state before the input head falls off either end of Z. For each starting condition σ and each stopping condition τ, the word transition $t(Z, \sigma, \tau)$ equals to 1 if M starting in σ can reach τ on Z during a nondeterministic round; otherwise it is 0.

3.1 Constant Number of Rounds

In this section a proof for Theorem 3(1) will be given. Let $k \geq 2$ be arbitrary integer and assume that M is a STM, of space complexity $S \in o(\log)$ that works in k rounds. Moreover, let n be sufficiently large integer of the form 2^d. In the section we will consider the input words of the form

$$w_{1,1} * w_{1,2} * \ldots * w_{1,n} \# \ldots \# w_{k^{k-1},1} * \ldots * w_{k^{k-1},n} \# u \# \mathtt{BIN}(n) \ , \tag{i}$$

with $w_{i,j}, u \in \{0,1\}^d$. It will be proved that if M accepts with high probability an input of this form that belongs to $\mathtt{PATTERN}_{k^{k-1}}$ than M has to accept with probability exceeding $1/4$ an input which does not belong to the language.

Denote by N the length of considered inputs, i.e. let

$$N \ := \ k^{k-1}(nd+n) + d + 1 + |\mathtt{BIN}(n)| \, ,$$

and let $\mathrm{Vol}(N)$ be the number of possible memory states of the machine M on input words of length N. Note that $\mathrm{Vol}(N) \leq 2^{O(S(N))}$. We will consider the word probabilities and the word transitions restricting the starting and the stopping conditions generated by memory states α to states with $|\alpha| \leq S(N)$. Let us fix some order of the pairs (σ, τ) of starting and stopping conditions for word probabilities as well as some order of the pairs (σ, τ) for word transitions. Let $\vec{p}(Z)$ be the vector of the word probabilities and let $\vec{t}(Z)$ be the vector of the word transitions according to these orderings. Define

$$\mu \ := \ 2^{-\sqrt{n}} \ . \tag{ii}$$

Lemma 2 *There exist two words*

$$W \ := \ w_1 * w_2 * \ldots * w_n \ \# \qquad \text{and} \qquad \overline{W} \ := \ \overline{w}_1 * \overline{w}_2 * \ldots * \overline{w}_n \ \# \ ,$$

with $w_j, \overline{w}_j \in \{0,1\}^d$, *for* $j = 1, \ldots, n$, *and* $\{w_1, w_2, \ldots, w_n\} \setminus \{\overline{w}_1, \overline{w}_2, \ldots, \overline{w}_n\} \neq \emptyset$ *such that* $\vec{t}(W) = \vec{t}(\overline{W})$ *and* $\vec{p}(W)$ *and* $\vec{p}(\overline{W})$ *are componentwise* 2^{μ}*-close.*

To prove the lemma one can adapt a counting argument of [DwSt92]. Now let us fix two words W and $\overline{W}$ as in the lemma above. Because word transitions of W and $\overline{W}$ are the same it means that for any string X and Y, with $|XWY| = N$, M cannot distinguish W from $\overline{W}$ in one nondeterministic round when starting on the prefix X or on the suffix Y. For a probabilistic round an analogous property holds for γ defined as follows

$$\gamma \ := \ 2^{2\mu(10\mathrm{Vol}(N)+20)} \ . \tag{iii}$$

Lemma 3 *Let* XWY *be an input string of the length* N, *and let* c *and* c' *be configurations such that* $h(c), h(c') \in [0..|X|] \cup [|XW|+1..N+1]$. *Then it holds that*

(a) $\mathcal{N}(c, c', XWY) = \mathcal{N}(c, c', X\overline{W}Y)$ *and* $\mathcal{A}_1[c, XWY] = \mathcal{A}_1[c, X\overline{W}Y]$, *if* c *is nondeterministic, and*

(b) the probabilities $\mathcal{R}[c, c', XWY]$ *and* $\mathcal{R}[c, c', X\overline{W}Y]$*, resp.,* $\mathcal{A}_1[c, XWY]$ *and* $\mathcal{A}_1[c, X\overline{W}Y]$*, are* γ*-close if* c *is probabilistic.*

Proof. A proof of (a) is straightforward and we will omit it here at all. For (b) we will sketch only a proof for configurations with $h(c) \leq |X|$ and $h(c') \geq |XW| + 1$. The other cases can be showed in a very similar way.

We will describe Markov chains R_W and $R_{\overline{W}}$ which model the probabilistic round of the machine M on inputs XWY and $X\overline{W}Y$, respectively, when M starts in configuration c. This configuration will correspond to the starting state of this Markov chains and configuration c' will correspond to their trapping state.

Let us denote the prefix of X of the length $h(c)$ by X_1 and let the remaining part of X will be denoted by X_2. Similarly, let Y_1 be the prefix of Y such that M in configuration c' reads the last symbol of Y_1 and let Y_2 be a suffix such that $Y = Y_1\ Y_2$. Each Markov chain we describe has $s = 10 \cdot \mathrm{Vol}(N) + 20$ states. The first $10 \cdot \mathrm{Vol}(N)$ states have the form $\langle \alpha, h \rangle$, where α is a memory state of the length bounded by $S(N)$ and h is a position of the first or the last symbol of words $\$X_1,\ X_2,\ W,\ Y_1,\ Y_2\$$ on the input tape containing the string $X_1X_2WY_1Y_2$ (remember that the left end-marker has position 0, and the last one – the position $N + 1$). An intuitive meaning of a state $\langle \alpha, h \rangle$ of the chain R_W is: start M in configuration (α, h) on the input $X_1X_2WY_1Y_2$. The meaning of state $\langle \alpha, h \rangle$ for the chain $R_{\overline{W}}$ is analogous. The next twenty states are the following: $\mathrm{Accept}_j, \mathrm{Reject}_j, \mathrm{Alter}_j$ and Loop_j, for $j = 1, \ldots, 5$. For chain R_W they mean that the probabilistic computation of M accepts, rejects, alternate or loops forever within $\$X_1,\ X_2,\ W,\ Y_1,\ Y_2\$$, respectively. For $R_{\overline{W}}$ the meaning is analogous.

The transition probabilities r_{ij} of R_W for non-trapping states i are obtained from word probabilities of M on the substrings: $\$X_1,\ X_2,\ W,\ Y_1,\ Y_2\$$. More precisely, the transitions r_{ij} such that states i, j are applied both to the same substring are equal to an appropriate word probabilities of M on this substring. E.g. if $i = \langle \alpha, |X_1X_2W| \rangle$ and $j = \langle \beta, |X_1X_2| + 1 \rangle$ then $r_{ij} = p(W, \langle \alpha, \mathrm{Right} \rangle, \langle \beta, \mathrm{Left} \rangle)$ since the position $|X_1X_2W|$ and $|X_1X_2| + 1$ means the rightmost resp., the leftmost symbol of the substring W. Remaining values r_{ij}, i.e. transitions for states i, j connecting with two different substrings are defined as follows. If $i = \langle \alpha_1, h_1 \rangle$ and $j = \langle \alpha_2, h_2 \rangle$ then the transition equals to the probability that M reaches in one step the configuration (α_2, h_2) starting in (α_1, h_1). Otherwise $r_{ij} = 0$. The transition probabilities of $R_{\overline{W}}$ are obtained analogously.

The states $\mathrm{Accept}_j, \mathrm{Reject}_j, \mathrm{Alter}_j, \mathrm{Loop}_j$, for $j = 1, \ldots, 5$ as well as all states $\langle \alpha, h \rangle$, with non-probabilistic α, are defined to be trapping states for both R_W and $R_{\overline{W}}$. For any trapping state t the transitions $r_{t,t}$ are defined to be 1.

Let the memory state of c (c') be α_c ($\alpha_{c'}$, resp.). Then the initial state of each chain is $\langle \alpha_c, h(c) \rangle$. Note that according to the definition of the trapping states, $\langle \alpha_{c'}, h(c') \rangle$ is a trapping state of both Markov chains.

W.l.o.g. let us assume that the first symbols of W and $\overline{W}$ are the same. Remember that the last symbols of the both words are equal, too. Hence a transition for any pair of states i, j connecting with two different substrings, is the same in considered chains. From this and from the fact that $\vec{p}(W)$ and $\vec{p}(\overline{W})$ are componentwise 2^{μ}-close we have that chains R_W and $R_{\overline{W}}$ are 2^{μ}-close. Now, using Lemma 1 we obtain that for the configuration c' the probabilities $p^*[\langle \alpha_{c'}, h(c') \rangle, R_W]$ and $p^*[\langle \alpha_{c'}, h(c') \rangle, R_{\overline{W}}]$ are $2^{2s\mu}$-close, what proves (b). □

Let for the strings $W = w_1 * w_2 * \ldots * w_n\ \#$ and $\overline{W} = \overline{w}_1 * \overline{w}_2 * \ldots * \overline{w}_n\ \#$, w be a word such that $w \in \{w_1, w_2, \ldots, w_n\} \setminus \{\overline{w}_1, \overline{w}_2, \ldots, \overline{w}_n\}$. Define, for short, $W^j := \underbrace{W\ W\ \ldots\ W\ W}_{j \text{ times}}$, for any integer $j \geq 0$ and $\hat{w} := \#w\#\mathtt{BIN}(n)$. Obviously $W^j\hat{w} \in \mathtt{PATTERN}_j$ but for any i, with $1 \leq i \leq j$, if in $W^j\hat{w}$ one replaces the i-th substring W by $\overline{W}$ then the new input string does not belong to $\mathtt{PATTERN}_j$ any more. Below we show that if M accepts in k rounds then there is an integer i such that the probability that M accepts the input $W^{i-1}\overline{W}W^{k^{k-1}-i}\hat{w}$ does not decrease drastically according to the probability that M accepts $W^{k^{k-1}}\hat{w}$.

Lemma 4 (Key) *There exists integer i, with $1 \leq i \leq k^{k-1}$, such that*

$$\mathcal{A}_k[\ W^{i-1}\overline{W}W^{k^{k-1}-i}\hat{w}\ ] \quad \geq \quad \gamma^{-\lceil k/2 \rceil} \cdot (1-1/k)^{k-1} \cdot \mathcal{A}_k[\ W^{k^{k-1}}\hat{w}\ ] \ .$$

Proof. Proceeding by induction on the number of rounds r we will show a more general fact. Let U and V be words over the input alphabet of M such that the string $UW^{k^{r-1}}V$ is of the form (i) and let A be a set of configurations such that

(♠) all non-accepting configuration of A are either probabilistic or nondeterministic and for any $c \in A$, machine M on the input string $UW^{k^{r-1}}V$ scans, with the input head position $h(c)$, the prefix U or the suffix V, i.e. $h(c) \leq |U|$ or $h(c) \geq |UW^{k^{r-1}}| + 1$.

Additionally, let us assign to each configuration $c \in A$ a non-negative real p_c. The only condition we will assume is that $\sum_{c \in A} p_c \ \leq \ 1$. The intuitive meaning of the number p_c is: start the machine M in configuration c with probability p_c.

Fact. *For any r, with $1 \leq r \leq k$, there exists integer i_r, with $1 \leq i_r \leq k^{r-1}$, such that for $Z := UW^{k^{r-1}}V$ and $Z_r := UW^{i_r-1}\overline{W}W^{k^{r-1}-i_r}V$*

$$\sum_{c \in A} p_c \cdot \mathcal{A}_r[c, Z_r] \quad \geq \quad \gamma^{-\tilde{r}} \cdot (1-1/k)^{r-1} \cdot \sum_{c \in A} p_c \cdot \mathcal{A}_r[c, Z] \ ,$$

where $\tilde{r} \ = \ \lfloor r/2 \rfloor$ if A has no probabilistic states and $\tilde{r} \ = \ \lceil r/2 \rceil$, otherwise.

For $r := k$, $A := \{c_0\}$, where c_0 is the initial configuration, $p_{c_0} := 1$ and for the empty word U and $V := \hat{w}$ this fact proves the lemma.

Proof of Fact. We proceed by induction on r. Assume first that $r = 1$ and A does not contain probabilistic states. Then by Lemma 3(a) we have that $\mathcal{A}_1[c, U\overline{W}V] = \mathcal{A}_1[c, UWV]$ for any $c \in A$. For $r = 1$ and the set A that does not contain nondeterministic states, by Lemma 3(b) we conclude that the probabilities $\mathcal{A}_1[c, U\overline{W}V]$ and $\mathcal{A}_1[c, UWV]$ are γ-close. This means that $\mathcal{A}_1[c, U\overline{W}V] \geq \gamma^{-1}\mathcal{A}_1[c, UWV]$ for any $c \in A$. Hence for $r = 1$ the fact holds. Below we will prove that it holds for any $r \geq 1$.

We will consider two cases. Assume first that A does not contain nondeterministic states. Let for the input Z, B be the set of all nondeterministic and accepting configurations. Define for any $c' \in B$ the real

$$p'_{c'} \ := \sum_{c \in A} p_c \cdot \mathcal{R}[c, c', Z] \ .$$

According to the definition of $\mathcal{A}$ it holds that

$$\sum_{c \in A} p_c \cdot \mathcal{A}_r[c, Z] \ = \ \sum_{c \in B} \ p'_c \cdot \mathcal{A}_{r-1}[c, Z] \ . \tag{iv}$$

Partition next the set B into $k+1$ subsets $B_0, B_1, \ldots, B_k$ as follows: let for $j = 1, \ldots, k$

$$B_j \ := \ \{c \in B : \ |UW^{(j-1)k^{r-2}}| \ < \ h(c) \ \leq \ |UW^{jk^{r-2}}| \ \} \ ,$$

and let $B_0 := \{c \in B : \ h(c) \ \leq \ |U| \ \text{ or } \ |UW^{k^{r-1}}| \ < \ h(c) \ \}$. Clearly, by the pigeon-hole-principle, there exists integer i, with $1 \leq i \leq k$ such that

$$\sum_{c \in B_i} p'_c \cdot \mathcal{A}_{r-1}[c, Z] \ \leq \ (1/k) \sum_{c \in B} p'_c \cdot \mathcal{A}_{r-1}[c, Z] \ . \tag{v}$$

Apply now the inductive hypothesis for: $r-1$, $U' := UW^{(i-1)k^{r-2}}$, $V' := W^{k^{r-1}-ik^{r-2}}$, and for the set of configurations $A' := B \setminus B_i$. Note that for A' and the input $U'W^{k^{r-2}}V'$ the assumption (♠) is fulfilled. Hence, by the hypothesis there exists integer i_{r-1} such that for $Z_{r-1} := U'W^{i_{r-1}-1}\overline{W}W^{k^{r-2}-i_{r-1}}V'$ it holds that

$$\sum_{c\in B\setminus B_i} p'_c \cdot \mathcal{A}_{r-1}[c, Z_{r-1}] \quad \geq \quad \gamma^{-\tilde{r}'} \cdot (1-1/k)^{r-2} \cdot \sum_{c\in B\setminus B_i} p'_c \cdot \mathcal{A}_{r-1}[c, Z] \ , \tag{vi}$$

where $\tilde{r}' = \lfloor (r-1)/2 \rfloor$ because $B \setminus B_i$ does not contain probabilistic states. Define $i_r := (i-1)\cdot k^{r-2} + i_{r-1}$. Obviously for this value i_r, words Z_{r-1} and Z_r are equal.

Now putting all of this together we conclude

$$\begin{aligned}
\sum_{c\in A} p_c \cdot \mathcal{A}_r[c, Z_r] &= \sum_{c\in A}\sum_{c'\in B} p_c \cdot \mathcal{R}[c, c', Z_r] \cdot \mathcal{A}_{r-1}[c', Z_r] && \text{by def. of } \mathcal{A} \\
&\geq \sum_{c'\in B\setminus B_i}\sum_{c\in A} p_c \cdot \mathcal{R}[c, c', Z_r] \cdot \mathcal{A}_{r-1}[c', Z_r] \\
&\geq \gamma^{-1} \sum_{c'\in B\setminus B_i}\sum_{c\in A} p_c \cdot \mathcal{R}[c, c', Z] \cdot \mathcal{A}_{r-1}[c', Z_r] && \text{by Lemma 3}(b) \\
&= \gamma^{-1} \sum_{c'\in B\setminus B_i} p'_{c'} \cdot \mathcal{A}_{r-1}[c', Z_r] && \text{by def. of } p' \\
&\geq \gamma^{-\tilde{r}'-1}(1-1/k)^{r-2} \sum_{c'\in B\setminus B_i} p'_{c'} \cdot \mathcal{A}_{r-1}[c', Z] && \text{by (vi)} \\
&\geq \gamma^{-\tilde{r}'-1}(1-1/k)^{r-1} \sum_{c'\in B} p'_{c'} \cdot \mathcal{A}_{r-1}[c', Z] && \text{by (v)} \\
&= \gamma^{-\tilde{r}'-1}(1-1/k)^{r-1} \sum_{c\in A} p_c \cdot \mathcal{A}_r[c, Z] && \text{by (iv)}
\end{aligned}$$

what proves the fact for the first case. Consider now the symmetric case, i.e. that A does not have probabilistic states. Then for any $c \in A$, define the configuration $\mathrm{Max}(c)$ such that $\mathrm{Max}(c)$ can be reached from c in one nondeterministic round and additionally for this configuration the probability $\mathcal{A}_{r-1}[\mathrm{Max}(c), Z] \geq \mathcal{A}_{r-1}[c', Z]$ for any c' which is reachable from c in one nondeterministic round. By the definition of $\mathcal{A}$ we have that $\mathcal{A}_r[c, Z] = \mathcal{A}_{r-1}[\mathrm{Max}(c), Z]$. Let

$$B := \{\mathrm{Max}(c) : \ c \in A\} \ ,$$

and let us define for any $c' \in B$ the real $p'_{c'} := \sum_{\substack{c\in A \text{ s.t.} \\ \mathrm{Max}(c)=c'}} p_c$. According to the definition of $\mathrm{Max}(c)$ we have

$$\sum_{c\in A} p_c \cdot \mathcal{A}_r[c, Z] \ = \ \sum_{c\in B} p'_c \cdot \mathcal{A}_{r-1}[c, Z] \ . \tag{vii}$$

Partition the set B into subsets $B_0, B_1, \ldots, B_k$ in the same way as previously. Let i, with $1 \leq i \leq k$, be an integer such that

$$\sum_{c\in B_i} p'_c \cdot \mathcal{A}_{r-1}[c, Z] \ \leq \ (1/k)\sum_{c\in B} p'_c \cdot \mathcal{A}_{r-1}[c, Z] \ . \tag{viii}$$

Then by the inductive hypothesis for $r-1$, $U' := UW^{(i-1)k^{r-2}}$, $V' := W^{k^{r-1}-ik^{r-2}}$, and for the set of configurations $A' := B \setminus B_i$, the inequality vi holds for $\tilde{r}' = \lceil (r-1)/2 \rceil$ because $B \setminus B_i$ does

not contain nondeterministic states. Define $i_r := (i-1)k^{r-2} + i_r$. Now putting the inequalities together we obtain

$$\begin{aligned}
\sum_{c\in A} p_c \cdot \mathcal{A}_r[c, Z_r] &\geq \sum_{\substack{c\in A \text{ s.t.} \\ \mathrm{Max}(c)\notin B_i}} p_c \cdot \mathcal{A}_r[c, Z_r] \\
&\geq \sum_{c'\in B\setminus B_i} p'_{c'} \cdot \mathcal{A}_{r-1}[c', Z_r] && \text{by def. of } p' \text{ and by Lemma 3}(a) \\
&\geq \gamma^{-\tilde{r}'}(1-1/k)^{r-2} \sum_{c'\in B\setminus B_i} p'_{c'} \cdot \mathcal{A}_{r-1}[c', Z] && \text{by (vi)} \\
&\geq \gamma^{-\tilde{r}'}(1-1/)^{r-1} \sum_{c'\in B} p'_{c'} \cdot \mathcal{A}_{r-1}[c', Z] && \text{by (viii)} \\
&= \gamma^{-\tilde{r}'}(1-1/k)^{r-1} \sum_{c\in A} p_c \cdot \mathcal{A}_r[c, Z] && \text{by (vii)}
\end{aligned}$$

what proves the lemma. □

Now we are ready to prove Theorem 3(1). Let us assume that M is a STM accepting $\mathtt{PATTERN}_{k^{k-1}}$ in $k \geq 2$ rounds and in sublogarithmic space S. Since $W^{k^{k-1}}\hat{w} \in \mathtt{PATTERN}_{k^{k-1}}$, hence M has to accept $W^{k^{k-1}}\hat{w}$ with probability greater or equal to 3/4, which means that $\mathcal{A}_k[\, W^{k^{k-1}}\hat{w}\,] \geq 3/4$. From the Key Lemma we conclude, however, that there exists integer i, with $1 \leq i \leq k^{k-1}$, such that

$$\mathcal{A}_k[\, W^{i-1}\overline{W}W^{k^{k-1}-i}\hat{w}\,] \;\geq\; \gamma^{-\lceil k/2\rceil}\cdot(1-1/k)^{k-1}\cdot 3/4 \;\geq\; \gamma^{-\lceil k/2\rceil}\cdot 3/8 \;>\; 1/4,$$

since $\gamma^{-\lceil k/2\rceil}$ tends to 1. But string $W^{i-1}\overline{W}W^{k^{k-1}-i}\hat{w}$ does not belong to $\mathtt{PATTERN}_{k^{k-1}}$ – a contradiction.

3.2 Proof of Theorem 3(2)

Let W and $\overline{W}$ be the words as defined in the previous subsection. Let Z and $\bar{Z}$ be the strings W and $\overline{W}$, resp., where the # symbols are removed, e.g. let $Z := w_1 * w_2 * \ldots * w_n$ and $\bar{Z} := \overline{w}_1 * \overline{w}_2 * \ldots * \overline{w}_n$.

Lemma 5 *Let $r(n)$ be an integer function, with $r(n) \in O(\log n/\operatorname{llog} n)$. Then for any sufficiently large integer n there exists integer i, with $1 \leq i \leq n$, such that*

$$\mathcal{A}_{r(n)}[\, \bar{Z}^{i-1}Z\bar{Z}^{n-i}\hat{w}\,] \quad\geq\quad \gamma^{-\lceil r(n)/2\rceil}\cdot e^{-1}\cdot\mathcal{A}_{r(n)}[\, \bar{Z}^n\hat{w}\,]\,.$$

Theorem 3(2) follows now straightforward from the above lemma.

4 Space Efficient Algorithm for PATTERN Languages

In this section we show for any integer $k \geq 1$ and for arbitrary small $\epsilon > 0$, a llog-space-bounded STM M that recognizes $\mathtt{PATTERN}_k$ with error probability ϵ. M works in $2k$ rounds starting in nondeterministic mode and in time bounded by a polynomial.

The machine M performs the following algorithm: Check deterministically at the beginning whether the input is of the form $W_1\#W_2\#\ldots\#W_k\#u\#\mathtt{BIN}(2^d)$, for some words $W_1,\ldots,W_k,u \in \{0,1,*\}^+$, with $|u| = d$. Reject and stop if this condition does not hold. Otherwise let $i := 1$ and go to step 1 below.

1. Nondeterministically guess a substring w_i of the length d in W_i.

2. Randomly choose a prime q_i with $2 \leq q_i \leq d^2$ and then compute $r_i := n_{w_i} \bmod q_i$, where n_{w_i} denotes an integer with the binary representation w_i.

3. Reject and stop if $r_i \neq n_u \bmod q_i$; otherwise if $i = k$ accept and stop else increase i by 1 and go to step 1.

If for any i, with $1 \leq i \leq k$, the strings w_i and u are equal then of course $n_{w_i} = n_u \bmod q_i$ for any value q_i and machine M accepts correctly in step 3. If $w_i \neq u$ for some i, with $1 \leq i \leq k$, than it could happen that $n_{w_i} = n_u \bmod q_i$ and M reaches in step 3 the accepting state that is wrong. This event happens, however, with probability that tends to 0. Indeed. Since $|n_{w_i} - n_u| \leq 2^d$ hence $n_{w_i} - n_u$ has at most d different prime divisors. On the other hand, M chooses from about $d^2/2 \ln d$ different primes at the beginning of step 2. So the probability that a wrong value q_i is chosen is at most $(2 \ln d)/d$.

Obviously, M uses $O(\mathrm{llog})$ space and works in $2k$ rounds.

5 Conclusions and Open Problems

In this paper separations were obtained for sublogarithmic $AM_k Space$ complexity classes. An interesting open problem is if our separations can be refined. Is it true that

$$AM_k Space(S) \subset AM_{k+1} Space(S) ,$$

for any integer k and sublogarithmic S?

How looks the hierarchy for at least logarithmic space bounds? Using a simple simulation of space-bounded NTMs by one-sided-error probabilistic TMs (see e.g. [Gil77] or the survey paper [Ma95]) one can easily show that $AM_2 Space(\log) = AM_1 Space(\log)$, that means the AM_2-class is quite weak in case of space bounds – contrary to time bounded classes. Is it also true that the class $AM_2 Space(\mathcal{SUBLOG})$ is equal to $AM_1 Space(\mathcal{SUBLOG})$?

What is the situation for space bounds S smaller than llog? The most interesting case seems to be space bounds restricted to constant functions. It is well known that $MA_1 Space(\mathcal{CON})$ coincides with the class of regular languages. This result, however, does not extend to the class $AM_1 Space(\mathcal{CON})$. Freivalds has shown the surprising result [Fr81] that

$$\texttt{COUNT} := \{1^n 0 1^m : n = m\},$$

can be accepted by a probabilistic TM in constant space with an arbitrarily small constant for the error probability. Is there a language that separates $AM_1 Space(\mathcal{CON})$ from $AM_k Space(\mathcal{CON})$ classes, for some $k > 1$? Dwork and Stockmeyer ([DwSt92]) showed that CENTER does not belong to $AM_1 Space(\mathcal{CON})$ and that there exists a constant space interactive proof system for this language. Can their protocol be improved to make only constant number of rounds?

References

[Ba85] L. Babai, *Trading group theory for randomness,* in Proceedings of the 17th ACM Symposium on Theory of Computing, ACM Press, 1985, 421-429.

[Co89] A. Condon, *Computational model of games,* MIT Press, 1989.

[DwSt92] S. Dwork and L. Stockmeyer, *Finite state verifiers I: the power of interaction,* Journal of the ACM, 39, 1992, 800-828.

[Fr79] R. Freivalds, *Fast probabilistic algorithms,* in Proceedings of the 8th International Symposium on Mathematical Foundations of Computer Science, Springer-Verlag, Heidelberg, 1979, 57-69.

[Fr81] R. Freivalds, *Probabilistic 2-way machines,* in Proceedings of the 10th International Symposium on Mathematical Foundations of Computer Science, Springer-Verlag, Heidelberg, 1981, 33-45.

[Gil77] J. Gill, *Computational complexity of probabilistic Turing machines,* SIAM Journal on Computing, 7, 1977, 675-695.

[GoSi86] S. Goldwasser and M. Sipser, *Private coins versus public coins in interactive prove systems,* in Proceedings of the 18th ACM Symposium on Theory of Computing, ACM Press, 1986, 59-68.

[LiRe96a] M. Liśkiewicz and R. Reischuk, *The sublogarithmic alternating space world,* SIAM Journal on Computing, 24, 1996, 828-861.

[LiRe96] M. Liśkiewicz and R. Reischuk, *Space Bounds for Interactive Proof Systems with Public Coins and Bounded Number of Rounds,* ICSI Technical Report No. TR-96-025, Berkeley, July 1996.

[Ma95] I. Macarie, *Space-bounded probabilistic computation: old and new stories,* SIGACT News, vol. 26(3), 1995, 2-12.