NONDETERMINISTIC VERSUS PROBABILISTIC LINEAR SEARCH ALGORITHMS

Friedhelm Meyer auf der Heide

IBM Research Laboratory, San Jose, CA 95193 [1]

Abstract : The "component counting lower bound" known for deterministic linear search algorithms (LSA's) also holds for their probabilistic versions (PLSA's) for many problems, even if two-sided error is allowed, and if one does not charge for probabilistic choice. This implies lower bounds on PLSA's for e.g. the element distinctness problem ($n \log n$) or the knapsack problem (n^2). These results yield the first separations between probabilistic and non-deterministic LSA's, because the above problems are non-deterministically much easier. Previous lower bounds for PLSA's either only worked for one-sided error "on the nice side", i.e. on the side where the problems are even non-deterministically hard, or only for probabilistic comparison trees. The proof of the lower bound differs fundamentally from all known lower bounds for LSA's or PLSA's, because it does not reduce the problem to a combinatorial one but argues extensively about e.g. a non-discrete measure for similarity of sets in R^n. This lower bound result solves an open problem posed by Manber and Tompa as well as by Snir.

Furthermore, a PLSA for n input variables with two-sided error and expected runtime T can be simulated by a (deterministic) LSA in $T^2 n$ steps. This proves that the gaps between probabilistic and deterministic LSA's shown by Snir cannot be too large. As this simulation even holds for algebraic computation trees we show that probabilistic and deterministic versions of this model are polynomially related. This is a weaker version of a result due to the author which shows that in case of LSA's, even the non-deterministic and deterministic versions are polynomially related.

I. Introduction

Linear search algorithms (LSA's) and algebraic computation trees (ACT's) are abstractions of random access machines (RAM's) with operations {+,-} or {+,-,*}. They have turned out to be a convenient computation model for proving lower bounds for many interesting problems (see [*DL*], [*R*], or [*M1*] for LSA's, and [*B*] for ACT's). As these lower bounds for LSA's can be carried over to RAM's (see [*KM*], [*M1*]), they even hold for a very realistic computational model.

Recently some effort was done to understand the power of probabilistic versions of LSA's (PLSA's). Manber and Tompa in [*MT*] and Snir in [*S*] proved lower bounds for PLSA's with one- or two-sided error. A PLSA with runtime T recognizes a language $L \subset R^n$ with two-sided error α, if, for each input, it computes the wrong output with probability $\leq \alpha$ and in expected time at most T. The error is one-sided, if for inputs from L, the output is always correct. One has to assume that $\alpha < \frac{1}{2}$, because otherwise every language can be recognized by the PLSA which just flips a coin to determine the output.

Manber and Tompa showed $n \log n$ lower bounds on PLSA's for comparison problems as the element distinctness problem, if only comparisons are allowed as queries and one charges for probabilistic choice. They allowed two-sided error.

In [*DL*], Dobkin and Lipton proved the "component counting lower bound" which says that an LSA which recognizes a language with s connected components has depth at $\Omega(\log s)$. Snir showed how to carry over the "component counting lower bound" for LSA's due to Dobkin and Lipton from [*DL*] to PLSA's with one-sided error. He does not charge for probabilistic choice. But the important examples for his bound all have the property that they are even non-deterministically hard, as shown in [*MT*]. Snir's method does not work for the (non-deterministically easy) complements of these languages.

Thus, as pointed out by Manber and Tompa as well as by Snir in the papers mentioned above, the remaining open question is

[1] The author is now at Johann Wolfgang Goethe Universitat, Fb Informatik, 6000 Frankfurt a.M., W.-Germany

0272-5428/85/0000/0065$01.00 © 1985 IEEE

For which languages does the component counting lower bound also hold on PLSA's with one-sided error on the "hard side", or, more generally, on PLSA's with two-sided error.

As shown by Snir in [*S*], we may not expect that all lower bounds for LSA's also hold for PLSA's. He showed counter examples, even for randomized LSA's where no errors are allowed.

We will show the following result.

Let L be a union of m hyperplanes in R^n with $s = m^{\Omega(n)}$ connected components. (As shown in [KM], L then has almost as many connected components as possible.) Then each PLSA with two-sided error which recognizes L has runtime $\Omega(\log s)$.

These bounds hold even if we do not charge for probabilistic choice and allow this choice to be very powerful : the algorithm may pick randomly *real* numbers from *real* intervals.

This implies e.g. the $n \log n$ lower bound for the element distinctness problem or the n^2 lower bound for the knapsack problem previously known for LSA's. As the complements of these languages are non-deterministically easy ($O(\log n)$ and $O(n)$, resp.), we establish the first separation result between non-deterministic and probabilistic LSA's.

The second result of this paper deals with ACT's. We show for their probabilistic version (PACT's) with two-sided error the following result, if one charges for probabilistic choice. Assume that we consider inputs from R^n.

Each PACT or PLSA with two-sided error and expected runtime T can be simulated by an ACT or LSA with depth $O(T^2 n)$.

This result for LSA's shows that the gaps between deterministic and probabilistic LSA's shown by Snir cannot become too big.

The result for ACT's is important in the spirit of the papers [*M2*] and [*M3*]. There it is shown that the property of LSA's only to handle inputs consisting of a fixed number of variables, i.e. only to deal with n-dimensional restrictions of problems, makes them powerful enough to compute n-dimensional restrictions of some NP-complete problems as the knapsack problem or the traveling salesman problem in polynomial time. Also ACT's only deal with n-dimensional restrictions of problems. Thus the question arises whether this property makes also ACT's very strong. Our result shows that they are at least so strong that their probabilistic and deterministic versions are polynomially related.

The paper is organized as follows. In section II we define PLSA's, give some geometrical definitions and state our main theorem, namely the lower bound. This will still be slightly different from the component counting lower bound mentioned above.

In section III we conclude the component counting lower bound from the main theorem and show applications.

The sections IV and V contain the proof of the main theorem. In section IV we state the main lemma. It is easy to see that the inputs arriving at a leaf of an LSA D form a convex polytope. The faces of these polytopes are called the faces of D. Let $L \subset R^n$ be a the union of hyperplanes, which intersect in exactly one point. Let $L' \subset R^n$ be some other language. The main lemma says :

If L' is similar to L, i.e. $L' \cap L$ contains "a large part" of L, and $\bar{L}' \cap \bar{L}$ contains "a large part" of $\bar{L}$, then D has a face of "small" dimension.

The proof of this lemma differs fundamentally from all known lower bound proofs for LSA's or PLSA's, because it does not seem to be reducible to a combinatorial argumentation by explicitly defining a finite set of inputs, for which the bound is shown. In our case, the measure for similarity of languages mentioned above is a non-discrete measure, and the proof of the main lemma does not seem to have a discrete analogy.

In section V we conclude the main theorem from the main lemma by showing the existence of an LSA, constructed from the PLSA by fixing the probabilistic choices suitably, which has many different faces of small dimension. Now usual combiatorial arguments apply and the lower bound follows.

Section VI contains the simulation. The proof of this result is based on a technique to simulate probabilistic computations with two-sided error by deterministic computations, if only finitely many inputs are allowed. Such a technique is introduced by Gill in [*G*] in connection with probabilistic Turing machines.

In order to obtain simulations for our computation models (in which the input set, R^n, is infinite), we have to consider the structure of functions computed in ACT's and LSA's. Here we again, as in [*B*], make use of Milnors bound from [*Mi*] for the number of connected components R^n can be subdivided into by the set of roots of a polynomial with given degree.

II. Definitions and the main theorem

A linear search algorithm (LSA) D is a finite, binary tree whose inner nodes v are labelled with queries of the form "$f(\bar{x}) > 0$", where $f: R^n \to R$ is an affine function, i.e. $f(\bar{x}) = \bar{a}\bar{x} - b$ for some $\bar{a} \in R^n$, $b \in R$. $H = \{\bar{x} \in R^n, f(\bar{x}) = 0\}$ is the hyperplane defined by v. The leaves are accepting or rejecting.

Started with some input $\bar{x} \in R^n$, the tree is traversed from the root to a leaf always following the left or right branch of a node according to whether its query is fulfilled or not. The language $L \subset R^n$ recognized by D is the set of inputs arriving at accepting leaves. Let $T_D(\bar{x})$ denote the length of the path followed by $\bar{x}$ in D.

A probabilistic LSA (PLSA) D is a collection $\{D_c, c \in [0,1]\}$ of LSA's D_c. Started with some input $\bar{x} \in R^n$, D first chooses randomly, with uniform distribution, a $c \in [0,1]$ and then starts D_c with $\bar{x}$. D accepts $L \subset R^n$ with error probability α, i.e. D is an α-PLSA for L, if for each $\bar{x} \in R^n$ the following holds.

-If $\bar{x} \in L$ then $Prob(D\ rejects\ \bar{x}) \leq \alpha$, and
-if $\bar{x} \notin L$ then $Prob(D\ accepts\ \bar{x}) \leq \alpha$.

The expected runtime of D started with $\bar{x}$ is $T(\bar{x}) = E(T_{D_c}(\bar{x}))$. The runtime of D is $\max\{T(\bar{x}), \bar{x} \in R^n\}$.

In order to make sure that the above probabilities and the (expected) runtime exist, we assume that the function which maps each $c \in [0,1]$ to D_c (described by the vector of the the coefficients of the queries at the nodes of D_c) is measurable.

The fact that the probabilistic choice is only done in the beginning of a computation does not weaken the model, because we can simulate the model in which probabilistic choices are done step by step in the obvious way. Note that we do not charge for probabilistic choice.

Now, in order to describe our main result, we first need some geometrical definitions. Let L be defined by m hyperplanes $H_1, \ldots, H_m$ in R^n, i.e. $L = \bigcup_{i=1}^{m} H_i$. $\bar{L}$ denotes the complement of L in R^n. For $I \subset \{1, \ldots, m\}$, an affine subspace $B = \bigcap_{i \in I} H_i$ is a face of L. The 0-dimensional faces of L are the vertices of L.

The set $c(v)$ of inputs arriving at a leaf v of an LSA D is defined by the queries on the path to v. As they are defined by affine functions, c(v) is a (convex) polytope. The faces of c(v) are the closures of the faces of D. They are (perhaps lower-dimensional) polytopes.

Now let $L \subset R^n$ be defined by m hyperplanes $H_1, \ldots, H_m$. Let D be an α-PLSA for L with runtime T.

Remark : We may w.l.o.g. assume the following.
1) Each D_c has depth T.
2) There is $\varepsilon > 0$ with the following property. Let H be defined by a node in D, x be a vertex of L, $\bar{x} \notin H$. Then the distance $d(H, \bar{x})$ between $\bar{x}$ and H is greater than ε.

Proof : (1) is shown in [*MT*]. To see (2) let r(c)=min$\{d(\bar{x}, H)$, *$\bar{x}$ is vertex of L, H is defined by a node of D_c, $\bar{x} \notin H$*$\}$. r(c)>0 holds, because D_c is a finite tree. For $\varepsilon > 0$ let $A_\varepsilon = \{c \in [0,1], r(c) > \varepsilon\}$. A_ε is measurable, and its volume tends to 1, if $\varepsilon \to 0$. Now replace each D_c with $c \notin A_\varepsilon$ by the trivial LSA with one node, which accepts everything. The resulting PLSA is a α'-PLSA for L, and $\alpha' > \alpha$ can be chosen arbitrarily closed to α by choosing ε sufficiently small. q.e.d.

We will assume that the remark holds for all PLSA's considered in this paper.

Now we are ready to state the main result of this paper.

Main Theorem : Let $L = \bigcup_{i=1}^{m} H_i$ have s vertices. Let D be an α-PLSA for L with runtime T. Then $2^{2T} = \Omega(\frac{s}{m^{3\alpha n}})$.

In the next section we will derive concrete lower bounds from this result.

III. Applications of the main theorem

In this section we shall relate our main theorem to the known lower bounds for (deterministic) LSA's based on the "component counting argument" due to Dobkin and Lipton from [*DL*]. This argument proves that an LSA needs $\Omega(\log s)$ steps to recognize a language with s connected components.

Theorem 1 : Let $L = \bigcup_{i=1}^{m} H_i$ such that $\bar{L}$ has $s \geq m^{\beta n}$ connected components for some $\beta > 0$.Then for $\alpha < \frac{1}{2}$, each α-PLSA for L has runtime $\Omega(\log s)$.

First we note that the restriction "$\alpha < \frac{1}{2}$" is necessary, because, for larger α, L can be recognized in one step just by flipping a coin to determine the output.

For the proof we first state a lemma based on an idea due to Bennett and Gill from [*BG*]. (See also the idea of the proof of lemma 6.) It shows that lower bounds on PLSA's are independent of the error probability, as long as this probability is smaller than $\frac{1}{2}$.

Lemma 1 : Let $\alpha < \frac{1}{2}$ and D be an α-PLSA for some L with runtime T. Then, for each $\alpha' < \frac{1}{2}$, there is an α'-PLSA for L with runtime O(T).

Now we relate the number of connected components of $\bar{L}$ to the number of faces of L.

Lemma 2 : Let L have l faces, and let $\bar{L}$ have s connected components. Then $l \geq s - 1$.

Proof : We proof this lemma by induction on m and n. If n=1, then obviously $l = m, s = m + 1$, and the lemma holds. If $m = 1$, then $s = 2, l = 1$, and the lemma also holds. Now let $m,n > 1$. Let $L_1 = L - H_1$ and $L_2 = L_1 \cap H_1$. Let l_1,l_2,s_1,s_2 be the respective parameters for L_1 and L_2. Then we get $s = s_1 + s_2$ and $l = l_1 + l_2 + 1$. By induction hypothesis, $l_1 \geq s_1 - 1$ and $l_2 \geq s_2 - 1$. Thus $l = l_1 + l_2 + 1 \geq s_1 + s_2 - 1 = s - 1$. q.e.d.

Proof of theorem 1 : Let l_p denote the number of p-dimensional faces of L, $l = \sum_{p=1}^{n-1} l_p$. By lemma 2, $l \geq s - 1$. Therefore there is p such that $l_p \geq \frac{s-1}{n} \geq \frac{m^{\beta n} - 1}{n} \geq m^{\beta' n}$ for some $\beta' < \beta$. Let n'=(n-p) and B be an n'-dimensional affine subspace of R^n which intersects each p-dimensional face of L in one point. (The existence of B follows from elementary linear algebra.) We now only consider inputs from B and view D as a α-PLSA D' for $L' \subset R^{n'}, L' \cong L \cap B$. By construction, L' has $s' \geq m^{\beta' n}$ vertices.

We now want to apply the main theorem to D'. Let $\alpha = \frac{1}{6}\beta'$. Then the main theorem yields $2^{2T} = \Omega(\frac{m^{\beta' n}}{m^{3\alpha n'}}) = \Omega(\frac{m^{\beta' n}}{m^{3\alpha n}}) = \Omega(m^{\frac{1}{2}\beta' n})$. Thus $T = \Omega(n \log m) = \Omega(\log s)$. By lemma 1, this bound holds for arbitrary $\alpha < \frac{1}{2}$. q.e.d.

The following examples generalize lower bounds for (deterministic) LSA's due to Reingold, Dobkin and Lipton, and Meyer auf der Heide from [*R*], [*DL*], and [*M1*].

Examples : If $\alpha < \frac{1}{2}$, then the following lower bounds hold for the runtime of α-PLSA's :

$\Omega(n \log n)$ lower bounds hold for
1) *element non-distinctness* :
recognize $\{\bar{x} \varepsilon R^n \mid x_i = x_j \mathit{ for\ some\ } i \neq j\}$.
2) *set inclusion* :
recognize $\{\bar{x} \varepsilon R^{2n} \mid \{x_1,\ldots,x_n\} \subset \{x_{n+1},\ldots,x_{2n}\}\}$.
3) *set non-disjointness* :
recognize $\{\bar{x} \varepsilon R^n \mid \{x_1,\ldots,x_n\} \cap \{x_{n+1},\ldots,x_{2n}\} \neq \emptyset\}$.
An $\Omega(n^2 \log(k+1))$ holds for
4) *Integer programming with solutions* $\leq k$:
recognize $\{(a_1,\ldots,a_n,b) \in R^{n+1} \mid \exists y_1,\ldots,y_n \in \{0,\ldots,k\}$ s.t. $\sum_{i=1}^{n} a_i y_i = b\}$.
(For k=1 this implies an $\Omega(n^2)$ lower bound for the knapsack problem.)

Proof : The first three languages are defined by $n^{O(1)}$ hyperplanes, and their complements have $O(n!^{\Omega(1)}) = O(n^{\Omega(n)})$ connected components as shown by Reingold in [*R*]. Thus these lower bounds follow from theorem 1.

The fourth language is defined by $(k+1)^n$ hyperplanes. As shown by Meyer auf der Heide in [*M1*], generalizing the respective bound for the knapsack problem due to Dobkin and Lipton from [*DL*], its complement has $(k+1)^{\Omega(n^2)}$ connected components. Therefore this lower bound follows also from theorem 1. q.e.d.

IV. The main lemma

In this chapter we state and prove the main lemma . For this purpose we define a measure for the "similarity" of languages. Let for this purpose $L = \bigcup_{i=1}^{m} H_i$ be as in the last section. Let $L' \subset R^n$ be another language. Informally, $D_n(L',L)$ denotes $\frac{1}{2}$*(ratio between the (n-dimensional) volumes of* $\bar{L} \cap \bar{L}'$ *and* $\bar{L}$ *)* + *(ratio between the ((n-1)-dimensional) volumes of* $L \cap L'$ *and L).* As the (n-1)-dimensional volume of e.g. L is infinite, we have to be a bit more careful with a formal definition.

A set $A \subset R^n$ is p-dimensional, if the lowest dimensional affine subspace of R^n containing A is p-dimensional. Let now A be at most p-dimensional, bounded, and measurable. (All sets we deal with in the sequel are obviously measurable, we shall no longer point that out in this paper.) $V_p(A)$ denotes the p-dimensional volume of A. Now let B be a p-dimensional subspace of R^n, $A \subset B$. Then the "ratio between A and B " is $I_p(A,B) = \overline{\lim_{\epsilon \to \infty}} \frac{V_p(A \cap U_\epsilon(\bar{0}))}{V_p(B \cap U_\epsilon(\bar{0}))}$, where $U_\epsilon(\bar{0})$ denotes the ball around $\bar{0}$ with radius ϵ in R^n. The "ratio between the volumes of $L' \cap L$ and L " can be measured by $R^a(L',L) = \frac{1}{m} \sum_{i=1}^{m} I_{n-1}(L' \cap H_i, H_i)$, and for the complements, $R^r(L',L) = I_n(\bar{L}', R^n)$. In this case we do not have to care about in L, because $V_n(L) = 0$.

Now, we can define the "similarity of L' to L " to be $R_n(L',L) = \frac{1}{2}(R^a(L',L) + R^r(L',L))$.

We now state some elementary properties of $R_n(L',L)$. For this purpose we make the convention that, for a hyperplane H, $R_{n-1}(L' \cap H, L \cap H)$ is meant to be defined relative to H ($\cong R^{n-1}$).

Lemma 3 (Properties of $R_n(L',L)$) :
a) $0 \le R_n(L',L) \le 1$.
b) $R_n(L',L) \le R_n(L', L - H_i) + \frac{1}{2m}$.
c) If H is not parallel to any H_i, then there is H' parallel To H such that $R_n(L',L) \le R_{n-1}(L' \cap H', L \cap H')$.
d) If H is parallel to H_i, then there is H' parallel to H, H'$\neq H_i$, such that $R_n(L', L - H_i) \le R_{n-1}(L' \cap H', L \cap H')$.

These results follow directly from the definition of $R_n(L',L)$ and elementary measure theory.

In the sequel let $L = \bigcup_{i=1}^{m} H_i$, D an α-PLSA for L, and L_c the language recognized by D_c. The following lemma will justify the above definition. All over this paper $E(...)$ denotes the expectation of some random variable defined on all $c \in [0,1]$. For $I \subset \{1,...,m\}$ let $L_I = \bigcup_{i \in I} H_i$.

Lemma 4 : For each $I \subset \{1,...,m\}$, $E(R_n(L_c,L_I)) \ge 1 - \alpha$.

Again, this lemma follows by the definitions and elementary measure theory.

We are now ready to state our main lemma. Let L be as above. If $m = n$ and L has (exactly) one vertex, then L is called an *n-star*.

Main Lemma : Let L be an n-star and $L' \subset R^n$ such that $R_n(L',L) = \gamma$. Let D be an LSA recognizing L'. Then D has a face of dimension $\le \min\{n, (\frac{1}{\gamma} - 1)n\}$.

Proof : For an LSA D let $B(D)$ denote the smallest dimension of a face of D, and $B(n,\gamma) = \max\ \{B(D) \mid$ *D recognizes an* $L' \subset R^n$ *with* $R_n(L',L) \ge \gamma$ *for some n-star* $L\}$. The lemma is implied in the following proposition.

Proposition : (i) $B(n,\gamma) \leq n - 1$ if $\gamma > \frac{1}{2}$

(ii) $B(n,\gamma) \leq B(n - 1,\gamma - \frac{1}{2n})$.

The lemma can be concluded as follows. By (ii), $B(n,\gamma) \leq B(n - i,\gamma - \frac{1}{2n} - \frac{1}{2(n-1)} - \dots - \frac{1}{2(n-i+1)}) \leq B(n - i,\gamma - \frac{i}{2(n-i+1)})$, because $B(n,\gamma)$ increases when γ decreases. If $\gamma \leq \frac{1}{2}$ the main lemma is trivial. If $\gamma > \frac{1}{2}$, we may choose $i = (2 - \frac{1}{\gamma})n - 1$ and get that $\gamma - \frac{i}{2(n-i+1)} > \frac{1}{2}$. Thus, by (i), $B(n,\gamma) \leq B(n - i,\gamma - \frac{i}{2(n-i+1)}) \leq n - i - 1 = (\frac{1}{\gamma} - 1)n$. As $B(n,\gamma) \leq n$ always holds, the main lemma follows.

Proof of the proposition : It is easy to see that $R_n(R^n,L) = R_n(\emptyset,L) = \frac{1}{2}$. Therefore (i) holds, because an LSA without an (n-1)-dimensional face has no query and therefore accepts either R^n or nothing.

To prove (ii) let D be an LSA as described in the main lemma. Let H be the hyperplane defined by the first query in D.

Claim 1 : Let H' be parallel to H, $H' \notin \{H_1,\dots,H_n\}$. If D restricted to H' has a p-face, i.e. if there is a leaf v of D such that $c(v) \cap H'$ has a p-face, then D has a p-face, too.

Proof : Let B be the p-face of $c(v) \cap H'$, and B^* the face of D with $B^* \cap H' = B$. If $B^* \subset H'$, then $B = B^*$ and we are done. Otherwise B^* has dimension (p+1). Let $A(B^*)$ denote the (p+1)-dimensional subspace containing B^*. As H is defined by the first query in D, each set c(v) is contained in one halfspace of H. Thus $B^* \neq A(B^*)$ and therefore B^* must have a p-dimensional face. As this is a face of D, the claim is proved. q.e.d.

Now we are ready to prove (ii) of the proposition. Recall that H is defined by the first query of D.

If H is not parallel to any H_i, then we apply lemma 3 (c) and get some H' parallel to H such that $R_{n-1}(L' \cap H',L \cap H') \geq R_n(L',L) = \gamma$. But $L \cap H'$ is no (n-1)-star. Therefore we remove a suitable H_j from L to obtain that $L^* = (L - H_j) \cap H'$ is an (n-1)-star (on H'). By lemma 3 (b), $R_{n-1}(L' \cap H',L^*) \geq R_{n-1}(L' \cap H',L \cap H') - \frac{1}{2n} \geq \gamma - \frac{1}{2n}$ by the above. Thus the proposition follows by claim 1.

If H is parallel to H_i, say, then choose H' parallel to H as in lemma 3 (d). In this case $L \cap H'$ is an n-star, and, by lemma 3 (d), $R_{n-1}(L' \cap H',L \cap H') \geq \gamma - \frac{1}{2n}$. Thus, the proposition follows again from claim 1. q.e.d.

Corollary : Let $L = \bigcup_{i=1}^{m} H_i$ have at least one vertex. Let D be an α-PLSA for L. If $d(c)$ denotes the smallest dimension of a face of D_c, then $E(d(c)) \leq 2\alpha n$.

Proof : As L has a vertex, there is an n-star $\tilde{L} \subset L$. Let L_c be recognized by D_c. By lemma 4, $E(R_n(L_c,\tilde{L})) \geq 1 - \alpha$. By the main lemma, $d(c) \leq (\min\{1,\frac{1}{\gamma_c} - 1\})n$ with $\gamma_c = (R_n(L_c,\tilde{L}))$. As for $x \in [0,1]$ it holds that $\min\{1,\frac{1}{x} - 1\} \leq 2(1 - x)$, we get $E(d(c)) \leq E((\min\{1,(\frac{1}{\gamma_c} - 1)\})n) \leq E(2(1 - \gamma_c)n) = 2(1 - E(\gamma_c))n \leq 2\alpha n$. q.e.d.

V. Proof of the theorem

Let in this section $L = \bigcup_{i=1}^{m} H_i$ be arbitrary, and let D be an α-PLSA for L. Let $\bar{x}_1,\dots,\bar{x}_r$ be the vertices of L. Let h(c,j) denote the smallest dimension of a face of D_c containing $\bar{x}_j$.

Lemma 5 : For each j, $E(h(c,j)) \leq 2\alpha n$.

Proof : Let $L^j \subset L$ be the n star with vertex $\bar{x}_j$ consisting of all those H_i's which contain $\bar{x}_j$. By the remark from section II there is $\epsilon > 0$ such that all hyperplanes H defined by nodes of D with $d(H, \bar{x}_j) < \epsilon$ contain $\bar{x}_j$. Now let U be the ball with radius ϵ around $\bar{x}_j$. We construct the PLSA D' from D by removing all branches from each D_c which are not chosen by some input from U.

Claim 2 : D' has the following properties.
1) Each hyperplane defined by a node of D' contains $\bar{x}_j$.
2) For inputs from U, D and D' do the same, and for sufficiently small ε, $L \cap U = L^j \cap U$.
3) Each face of each D'_c is contained in a face of D_c with the same dimension.
4) D' is an α-PLSA for L^j.

Proof : (1), (2), and (3) are clear by the definitions of D' and L^j. To prove (4) let $\bar{x} \in L^j$. We have to show that Prob(D' rejects $\bar{x}$)$\leq\alpha$. (For $\bar{x} \notin L^j$ the proof for "Prob(D' accepts $\bar{x}$)$\leq\alpha$" is analogous.)

Let g be the straight line between $\bar{x}_j$ and $\bar{x}$ excluding $\bar{x}_j$, and $\bar{x}' \in g \cap U$. As by (1), all hyperplanes defined by nodes of D contain $\bar{x}_j$, $\bar{x}$ and $\bar{x}'$ reach the same leaf in each D_c. Thus Prob(D' rejects $\bar{x}$) = Prob(D' rejects $\bar{x}'$). As by (2), $\bar{x} \varepsilon L^j$ holds if and only if $\bar{x}' \varepsilon L^j$, and Prob(D' rejects $\bar{x}'$) = Prob(D rejects $\bar{x}'$)$\leq\alpha$, we get that Prob(D' rejects $\bar{x}$)$\leq\alpha$. q.e.d.

Now, in order to finish the proof of lemma 5, we apply the corollary from the last chapter to D'. This yields, that the expected smallest dimension of a face of some D'_c is $\leq 2\alpha n$. But this bound even holds for the expected smallest dimension of a face of some *D'_c containing $\bar{x}_j$*, because all faces of each D'_c contain $\bar{x}_j$ by (2). As each face of D'_c is contained in a face of D_c with the same dimension by (3), the lemma follows. q.e.d.

Now we are ready to prove the main theorem. Let $\mathrm{h(c)} = \sum_{j=1}^{s} \mathrm{h(c,j)}$. Then $\mathrm{E(h(c))} \leq 2\alpha ns$ by lemma 3. Thus there is $\tilde{c}$ such that $\mathrm{h}(\tilde{c}) \leq 2\alpha ns$. Let D* denote the LSA $D_{\tilde{c}}$. Let p be the number of vertices of L which belong to a $3\alpha n$-dimensional face of D*. Then, as $\mathrm{h}(\tilde{c}) \geq (s-p)3\alpha n$, we get $(s-p)3\alpha n \leq 2\alpha ns$, which implies $p \geq \frac{1}{3}s$, i.e. at least $\frac{1}{3}s$ vertices of L lie on a $3\alpha n$ -dimensional face of D*. As, by elementary linear algebra, at most $\binom{m}{3\alpha n}$ vertices lie on the same such face, we have shown

Claim 3 : D* has at least $\dfrac{\frac{1}{3}s}{\binom{m}{3\alpha n}} \geq \dfrac{\frac{1}{3}s}{m^{3\alpha n}}$ different $3\alpha n$-dimensional faces.

Now, because of the remark from section II, D* has depth T, where T denotes the runtime of D. Therefore D* has at most 2^T leaves. By elementary arguments from linear algebra, each set c(v) associated to a leaf v has at most $\binom{T}{3\alpha n} \leq 2^T$ many $3\alpha n$-dimensional faces.

Thus, by claim 3, $2^{2T} \geq \dfrac{\frac{1}{3}s}{m^{3\alpha n}}$, which proves the main theorem. q.e.d.

VI. The simulation

In this section we assume that probabilistic choices are fair coin flips, and each coin flip adds one step to the runtime of the algorithm. We first consider a very general type of computation trees.

Let A be a set, and let F be a family of functions $f:A \rightarrow R$. A probabilistic computation tree (PCT) D with queries defined by F is a binary computation tree which takes inputs from A. An inner node v of D is either a probabilistic node or a query node. At a probabilistic node, a coin is flipped to determine which branch to follow. At a query node a query f(x)'c'0 is asked to determine which branch to follow. Here fεF, xεA is the input, and cε{<,>,=}. Each leaf is either accepting or rejecting. The complexity of D is the maximum over the expected runtimes of D started with x for all xεA. An α-PCT for some $L \subset A$ is defined as for PLSA's. D is deterministic, i.e. a CT, if it contains no probabilistic nodes. A computation of D is a sequence of functions from F used for queries on some path of D. Let in the sequel D be an α-PCT with queries defined by F and with complexity T which recognizes $L \subset A$. By the first part of the remark from section II we may assume that D has depth T.

We say, a CT strongly simulates D, if it recognizes L and if its computations are concatenations of computations of D.

Next we show how to simulate D by a CT if A is finite. Similar forms of the following lemma are already implicitly used in [*BG*] and [*Re*].

Lemma 6 : If A is finite, then D can be strongly simulated by a CT with depth O(T log($|A|$)).

The idea of the proof of this lemma is to let D run s times started with input x and to accept if the majority of the runs was accepting. The error probability decreases exponentially with s. Thus, for s=$O(\log T)$, the error probability β is smaller than $\frac{1}{|A|}$. Now we suitably fix the results of the coinflips and obtain a CT with depth O($T\log|A|$) which treats at least $(1-\beta)|A| > |A| - 1$, i.e. all inputs correctly. This is the desired CT.

Lemma 6 shows an efficient simulation of probabilistic by deterministic computations, if the input set is finite. But for the computational models we are interested in the input set is infinite, namely R^n. On the other hand, in lemma 2 we have not used any properties of the set F of functions defining the queries. We now shall see how to take into account the 'structure' of F in order to get results similar to lemma 2 when the input set is infinite.

Let F={f_1,...,f_m}, and let $\bar{c} = (c_1,...,c_m)\varepsilon\{<,>,=\}^m$. Then $I_{\bar{c}}:=\{x\varepsilon A\,, f_i(x)'c_i'0 \text{ for } i=1,...,m\}$.

Lemma 7 : Let k= $|\{\bar{c}\varepsilon\{<,>,=\}^m, I_{\bar{c}} \neq \emptyset\}|$. Then D can be strongly simulated by a CT with depth O(T log(k)).

Proof : For $\bar{c}\varepsilon\{<,>,=\}^m$ with $I_{\bar{c}} \neq \emptyset$ let $x_{\bar{c}}\varepsilon I_{\bar{c}}$. By lemma 2 we know that there is a CT D' with depth 0(T log(k)) which strongly simulates D if we only allow inputs from $\{x_{\bar{c}}, I_{\bar{c}} \neq \emptyset\}$. The following claim will prove lemma 7.

Claim 4 : D' recognizes L (for all inputs from A).

Proof : Let xεA. As the non-empty $I_{\bar{c}}$'s partition A, there is a unique $\bar{c}$ with x $\varepsilon I_{\bar{c}}$. By the definition of $I_{\bar{c}}$, no query defined by a function from F can distinguish between x and $x_{\bar{c}}$. Thus both in D and D' x and $x_{\bar{c}}$ follow the same computation paths. Therefore, D accepts x if and only if it accepts $x_{\bar{c}}$, i.e. xεL if and only if $x_{\bar{c}}\varepsilon$L. By the same argument we get that D' accepts x if and only if it accepts $x_{\bar{c}}$. Thus D' accepts L. q.e.d.

Now we apply lemma 7 to get a simulation of probabilistic by deterministic algebraic computation trees and linear search algorithms. A probabilistic algebraic computation tree (PACT) D is a tree with degree 0, 1, or 2. To each node v with degree 1 a function $f_v{:}R^n \to R$ is attached. f_v is either a projection on one of the input variables $x_1,...,x_n$, or a constant, or $f_v = f_{v'}\,\$\,f_{v''}$ for some nodes v', v'' on the path to v, and for $\$\varepsilon\{+,-,*,/\}$. A node v with degree 2 is either a probabilistic node or a query node. Probabilistic nodes work as in PCT's. At a query node v, a query $f_{v'}(\bar{x})\,'c'\,0$ is asked to determine which branch to follow, where v' is a node on the path to v and $c\varepsilon\{<,>,=\}$. The leaves are accepting or rejecting. α-PACT's, the recognized language, and the complexity, as well as the deterministic version (ACT) are defined as for PCT's.

We now are ready to state the result of this section.

Theorem 2 : Let $\alpha < \frac{1}{2}$, and let D be an α-PACT or α-PLSA with complexity T (where we charge for coinflips) recognizing $L \subset R^n$. Then there is an ACT or LSA recognizing L in O(T^2n) steps.

Proof : Let D be as in theorem 2. We again may assume that has depth T. Let F=$\{f_1,...,f_m\}$ be the set of functions computed at the nodes of D. Then $m \leq 2^T$, because D is a binary tree. Furthermore, because of the arithmetic operations allowed in D, each f_i is a rational function, $f_i = \frac{r_i}{q_i}$, where r_i and q_i are polynomials of degree at most 2^T.

For $\bar{c}\varepsilon\{<,>,=\}^m$ let $I_{\bar{c}}$ be defined as in the previous section.

Claim 5 : Let k=$|\{\bar{c}\varepsilon\{<,>,=\}^m, I_{\bar{c}} \neq \emptyset\}|$. Then there is an ACT recognizing L in O(T log(k)) steps.

Proof : By lemma 7, D can be strongly simulated by a CT D' of depth O(T log (k)). The definition of 'strongly simulating' guarantees that D' is an ACT. q.e.d.

Now it remains to bound k. For this purpose we first note that for $c\varepsilon\{<,>,=\}$ and for $\bar{x}\varepsilon R^n$ such that $f_i(\bar{x})$ is defined, $f_i(\bar{x})'c'0$ holds if and only if $p_i := r_i \bullet q_i'c'0$. Thus, for $\bar{c}\varepsilon\{<,>,=\}^m$, $I_{\bar{c}} = \{\bar{x}\varepsilon R^n, f_i(\bar{x})'c_i'0$ for $i = 1,\ldots,m\} = \{\bar{x}\varepsilon R^n, p_i(\bar{x})'c_i'0$ for $i = 1,\ldots,m\}$.

Lemma 8 : $k = |\{\bar{c}\varepsilon\{<,>,=\}^m, I_{\bar{c}} \neq \emptyset\}| \leq (2d+1)(2d+2)^{n-1}$, where d is the degree of $\prod_{i=1}^{m} p_i$.

Before we prove the lemma we conclude theorem 2 from it. As the p_i's have degree at most 2^{T+1} and as $m \leq 2^T$, $d \leq 2^{2T+1}$. Thus $k = 2^{O(Tn)}$. Inserting this in claim 5 yields theorem 2 for ACT's. For LSA's, a simplified version of this proof already yields the result.

Proof of lemma 8 : This proof is based on a theorem due to Milnor from [*Mi*] which is previously already used in [*B*].

Theorem (Milnor): Let $p: R^n \to R$ be a polynomial with degree d'. Then c(p):$=\{\bar{x}\varepsilon R^n, p(\bar{x}) \neq 0\}$ has at most $(d'+2)(d'+1)^{n-1}$ connected components.

Now, in order to prove lemma 8, let A'$\subset R^n$ contain exactly one element of each non-empty $I_{\bar{c}}$. Then |A'| =k. Let $\delta > 0$ be chosen such that $\delta < \min\{|p_i(\bar{x})|, i = 1,\ldots,m, \bar{x}\varepsilon A', p_i(\bar{x}) \neq 0\}$. Let $p' := \prod_{i=1}^{m} (p_i + \delta)(p_i - \delta)$.

Claim 6 : Each connected component of c(p') contains at most one element from A'.

Proof : Let $\bar{x},\bar{y}\varepsilon A'$. Then, as $\bar{x}$ and $\bar{y}$ belong to different $I_{\bar{c}}$'s, there is p_i such that, w.l.o.g. $p_i(\bar{x}) > 0$ and $p_i(\bar{y}) \leq O$. By the definition of δ, we therefore get that $p_i(\bar{x}) - \delta > 0$ and $p_i(\bar{y}) - \delta < 0$. As $p_i - \delta$ is continuous, each continuous path from $\bar{x}$ to $\bar{y}$ contains a root of $p_i - \delta$, and therefore of p'. Thus $\bar{x}$ and $\bar{y}$ belong to different connected components of c(p'). q.e.d.

We know that d'$\leq$2d (recall that d is the degree of p). Thus, by Milnor's theorem, c(p') has at most $(2d+1)(2d+2)^{n-1}$ connected components. As by claim 6, k=|A'| $\leq$ (number of connected components of c(p')), lemma 8 follows. q.e.d.

References

[*B*] *M. Ben Or :* Lower bounds for algebraic computation trees, 15th ACM STOC, 80-86, 1983.

[*BG*] *C. H. Bennett, J. Gill :* Relative to a random oracle A, $P^A \neq NP^A \neq co\text{-}NP^A$ with probability 1, SIAM J. Comp. 10, 96-113, 1981.

[*DL*] *D. Dobkin, R. J. Lipton :* A lower bound of $\frac{1}{2}n^2$ on linear search algorithms for the knapsack problem, J.C.S.S. 16, 413-416, 1978.

[*KM*] *P. Klein, F. Meyer auf der Heide :* A lower time bound for the knapsack problem on random access machines, Acta Informatica 19, 385-395, 1983.

[*M1*] *F. Meyer auf der Heide :* Lower bounds for solving Diophantine equations on random access machines, to appear in J. ACM.

[*M2*] ________ : A polynomial linear search algorithm for the n-dimensional knapsack problem, J. ACM 31(3), 668-676, 1984.

[*M3*] ________ : Fast algorithms for n-dimensional restrictions of hard problems, to appear in ACM STOC, 1985.

[*Mi*] *J. Milnor :* Singular points of complex hypersurfaces, Princeton Univ. Press, 1968.

[*MT*] *U. Manber, M. Tompa :* Probabilistic, nondeterministic, and alternating decision trees, 14th ACM STOC, 234-244, 1982.

[*R*]*E. Reingold :* On the optimality of some set algorithms, J. ACM 19, 649-659, 1972.

[*S*] *M. Snir :* Lower bounds for probabilistic linear decision trees, Research Report 83-6, Dept. of Computer Science, the Hebrew University of Jerusalem, Israel, 1983.