Two Lower Bound Arguments with "Inaccessible" Numbers

by

Martin Dietzfelbinger* and Wolfgang Maass*
Department of Mathematics, Statistics and Computer Science
University of Illinois at Chicago, Chicago, Illinois 60680

ABSTRACT

We present lower bound arguments for two general computational models: linear decision trees (LDT's) and random access machines (RAM's). Both proofs use (besides combinatorial and geometrical arguments) the method of constructing "hard" instances $(x_1,\ldots,x_n)$ of the considered problems, where the distances between some of the x_i are chosen so large that from the point of view of a fixed computational model the larger numbers are "inaccessible" from the smaller ones. In §2 we further refine this technique: there we have to satisfy at the same time equalities between certain sums of input numbers in order to allow a "fooling argument". The mentioned techniques allow us to derive sharper lower bounds for a variety of computational problems, including KNAPSACK, SHORTEST PATH and ELEMENT DISTINCTNESS.

§1. Introduction

A linear decision tree (LDT; often also called LSA: linear search algorithm) is a rooted tree where each internal node is labelled by a certain linear test $\sum_{i=1}^{n} a_i x_i : a_o$. $(x_1,\ldots,x_n)$ is the input for the LDT - consisting of n numbers x_i from $\mathbb{N}$ (or $\mathbb{Q}$, or $\mathbb{R}$, depending on the context). The edges from such node to its three sons are labelled by $<$, $=$, and $>$; and depending on whether $\sum_{i=1}^{n} a_i x_i < a_o$, $\sum_{i=1}^{n} a_i x_i = a_o$ or $\sum_{i=1}^{n} a_i x_i > a_o$ one follows the corresponding edge to the next node. Each leaf is labelled "accept" or "reject" (we consider here only decision problems, lower bound arguments are more interesting for this type of problems). The complexity of the LDT is measured in terms of n (the "dimension" of the problem instance $(x_1,\ldots,x_n)$), not in terms of the number of input bits.

Most lower bounds on the depth of LDT's T for decision problems P are "connectivity - arguments" (see [2],[3]), where one exploits that for each leaf ℓ of T the set of all inputs $(x_1,\ldots,x_n) \in \mathbb{R}_+^n$ that lead to leaf ℓ forms a connected subset of $\mathbb{R}^n$ (it is an intersection

*Written under partial support by NSF-grant DCR-8504247

of halfspaces and hyperplanes). Therefore the number of leafs in T must be at least as large as the number of connected components of the considered problem P respectively its complement $\mathbb{R}^n - P$. Unfortunately KNAPSACK and the other common NP - complete "number problems" have only $O(2^{O(n^2)})$ many connected components (see [12]) and we get in this way at best a lower bound that is quadratic in n. The (simplified) version of the KNAPSACK problem that one considers in the cited literature (and which we will study in this paper) is defined by

$$\text{KNAPSACK} = \bigcup_{n \in \mathbb{N}} K(n), \qquad \text{where}$$

$$K(n) = \{(x_1,\ldots,x_n) \in \mathbb{R}_+^n \mid \exists\, S \subseteq \{1,\ldots,n\} \left(\sum_{i \in S} x_i = 1\right)\}.$$

Actually one usually focuses instead on the discrete version of KNAPSACK, where $K(n)$ is restricted to $\mathbb{Q}_+^n$. This version of the problem is NP - complete.

In order to achieve a larger than quadratic lower bound for KNAPSACK one has to undertake a finer analysis of the mathematical structure of this problem. Dobkin, Lipton [4] and Ukkonen [16] made some progress in this direction: they exploited a geometrical property of the KNAPSACK - problem in order to prove an exponential lower bound for KNAPSACK on a very restricted class of LDT's (only linear tests $\sum_{i=1}^{n} \alpha_i x_i : \alpha_o$ with $\alpha_i \in \{0,1\}$ are allowed). Unfortunately their restriction is so severe that one is not even able to sort the n input numbers $x_1,\ldots,x_n$ on an LDT of this type. This entails that one gets on such a model also exponential lower bounds for a variety of problems that are in fact computationally trivial, but which require to compare the size of some of the input numbers x_i (e.g. for the problem of deciding whether $\sum_{i \in S} x_i \geq 1$ for some set $S \subseteq \{1,\ldots,n\}$ of size $n/2$).

In this paper we use nontrivial combinatorial and geometrical arguments in order to achieve a sharper lower bound for KNAPSACK on a quite general class of LDT's. We consider LDT's where the coefficients α_i in the linear tests $\sum_{i=1}^{n} \alpha_i x_i : \alpha_o$ may be arbitrary real numbers. We show in Theorem 1 that if $f(n) > |\{i \mid \alpha_i < o\}|$ for all linear tests $\sum_{i=1}^{n} \alpha_i x_i : \alpha_o$ in the tree then the depth of the tree is at least $2^{n/2f(n)}$. This implies an exponential lower bound for KNAPSACK on LDT's where the coefficients α_i in each linear test $\sum_{i=1}^{n} \alpha_i x_i : \alpha_o$ may be arbitrary real numbers provided that $|\{i \mid \alpha_i < o\}| = o(n/\log n)$ (it is known that this restriction on the number of negative coefficients can not be totally eliminated: without this restriction the upper bound on the depth of LDT's for KNAPSACK is known to be polynomial in n, see

[11]. In this way one gets an exponential lower bound for KNAPSACK on a computational model that is substantially more powerful than the restricted LDT's of Dobkin, Lipton [4] and Ukkonen [16]: There is no restriction anymore on the nonnegative coefficients. Furthermore linear tests with $o(n/\log n)$ negative coefficients do not only allow to sort the x_i (for a comparison of two input numbers x_i, x_j one only needs a single negative coefficient in the respective linear test), but also to sort sums $x_i + x_j$, $x_i + x_j + x_k$,... where up to $o(n/\log n)$ many of the input numbers x_i occur in a term. In fact, in spite of its restriction on the number of negative coefficients, this type of LDT appears to be one of the more powerful computational models on which superpolynomial lower bounds for NP-complete problems have been achieved.

The technique that we use in the proof of Theorem 1 provides a quite general new tool for the analysis of many algorithms that are based on linear tests. It allows to show for a variety of quite practical problems that these problems inherently require to compare sums of many input numbers. For example Theorem 2 exhibits an intrinsic difference between the computation of a minimal spanning tree (where the weights of the edges have to be compared, but no sums of several edge weights need be compared) and the decision problem associated with the shortest path problem, respectively the maximum weight matching problem, for which all familiar algorithms involve the comparison of sums of many edge weights. Theorem 2 shows that in fact there exist no polynomial time algorithms (based on linear tests) for the latter two problems where only sums of up to $o(n/\log n)$ many weights are compared. The argument of the proof of Theorem 2 yields in addition a number of not so obvious refinements of this negative result which may be of interest for the analysis of more practical algorithms. One can show that even algorithms that are only required to handle particularly "nice" types of problem instances in polynomial time (e.g. only graphs that are planar, or where the weights are given by the Euclidean distance of points in the plane) are forced to compare large sums of edge weights.

In §4 of this paper we present another new lower bound technique, that allows us to derive sharp lower bounds for random access machines (RAM's) with any "reasonable" bound on the number of registers that are used. This lower bound argument is quite different from the one that is used in §2 and §3, however it also employs the method of constructing "hard" problem instances where the individual input numbers x_i are mutually "inaccessible".

The model of a RAM that we consider in §4 has become the standard

model for the machine - independent analysis of the time and space requirements of concrete algorithms (see [1]). We consider the usual version of a RAM with unboundedly many registers $r_0, r_1, r_2, \ldots$. Each register is capable of holding an integer of arbitrary size. The RAM can address these registers both directly and indirectly, and it can perform addition, subtraction and comparison on the contents of registers. We apply the usual uniform cost criterion, where one unit of time is charged for each execution of an instruction, independently of the size and address of the operands. We assume that the first n registers hold initially the input numbers $x_1, \ldots, x_n$, and similarly as before we measure the length of the computation in terms of n (not in terms of the bit lenght of the input).

We analyze in §4 the length of RAM - computations on the following two well-known decision problems:

ELEMENT DISTINCTNESS = $\{(x_1, \ldots, x_n) \in \mathbb{N}^n \mid x_i \neq x_j \text{ for } i \neq j\}$ and

DISJOINT SETS = $\{((y_1, \ldots, y_n), (z_1, \ldots, z_n)) \in \mathbb{N}^{2n} \mid \{y_1, \ldots, y_n\} \cap \{z_1, \ldots, z_n\} = \emptyset\}$.

Obviously a RAM can decide both of these problems in $O(n)$ steps (write x_i into the register with address x_i). However this algorithm requires a very large number of registers and the question arises whether these problems can also be solved in linear time on a RAM with a "reasonable" memory size (e.g. polynomially in n many registers). For this case the best known upper bound is $O(n \log n)$ (via sorting).

We show in Theorem 3 that this upper bound of $O(n \log n)$ for ELEMENT DISTINCTNESS and DISJOINT SETS on space bounded RAM's is in fact optimal. Furthermore the lower bound of $\Omega(n \log n)$ does not only hold if the number of used registers is polynomial in n, but if the number of used registers is bounded by an <u>arbitrary</u> function in terms of the dimension n of the considered problem instance $(x_1, \ldots, x_n)$.

In this context we would like to mention that to our knowledge there are no other lower bound results which show that a superlinear algorithm for a natural decision problem on a RAM is optimal (however one has already shown that a number of superlinear algorithms for the computation of certain <u>functions</u> on a RAM are optimal, see [14]; in addition there are superlinear lower bounds for KNAPSACK on a RAM - but they are not believed to be optimal, see [6], [12]).

At the end of §4 we indicate in Theorem 4 an extension of Theorem 3, where it is shown that the lower bounds of Theorem 3 remain correct

even if the RAM is made more powerful by the addition of an arbitrary "oracle" $Q \subseteq \mathbb{R}^q$ (the RAM may ask the oracle Q repeatedly during the computation for arbitrary q-tupels of input numbers $(x_{i_1},\ldots,x_{i_q})$ whether $(x_{i_1},\ldots,x_{i_q}) \in Q$). The proof of Theorem 4 relies on the fact that the technique of making the input numbers x_i mutually "inaccessible" is compatible with the well known method from model theory where one chooses with the help of Ramsey's Theorem input numbers that are "order indiscernible" with respect to a given predicate Q. This application of Ramsey's Theorem is similar to that by Moran, Smir, and Manber [9], [10] for decision trees (both applications were found independently, see our preprint [7]). For more detailed proofs of the results in §4 we refer to Maass [8].

§2. A lower bound for KNAPSACK on linear decision trees.

THEOREM 1. Let T_n be a linear decision tree for inputs $\bar{x} \in \mathbb{R}^n$, for all $n \in \mathbb{N}$, and let $f : \mathbb{N} \to \mathbb{N}$ be a function such that every test $\sum a_i x_i : a_o$ in T_n ($a_i \in \mathbb{R}$; possible outcomes: $<,=,>$) satisfies $|\{i \geq 1 \mid a_i < o\}| < f(n)$. If T_n recognizes the Knapsack problem

$$K(n) := \{\bar{x} \in \mathbb{R}_+^n \mid \exists S \subseteq \{1,\ldots,n\} \ (\sum_{i \in S} x_i = 1)\},$$

then depth $(T_n) \geq 2^{\lfloor n/2f(n) \rfloor}$ for all $n \in \mathbb{N}$.

NOTE: This lower bound is superpolynomial if $f(n) = o\left(\frac{n}{\log n}\right)$.

REMARK: It will be seen from the proof that it suffices to assume that T_n finds the correct answer for inputs $\bar{x} \in \mathbb{Q}_+^n$.

PROOF OF THEOREM 1: Fix n and set $k := f(n)$ and $p := \frac{n}{2k}$. We show that depth $(T_n) \geq 2^p$.

Note here that we can assume w.l.o.g. that $2k$ divides n. If this is not the case, let $n_o := 2k \cdot \lfloor \frac{n}{2k} \rfloor$, and consider the LDT T' obtained from T_n by replacing all test $\sum_{i=1}^{n} a_i x_i : a_o$ by $\sum_{i=1}^{n_o} a_i x_i : a_o$. Then it is clear that T' recognizes $K(n_o)$ and that for all tests in T' $|\{i \geq 1 \mid i \leq n_o \text{ and } a_i \neq o\}| < k$. We have $2k \mid n_o$, hence by the special case depth $(T') \geq 2^{n_o/2k}$. Since depth $(T') =$ depth (T_n) and $\lfloor n/2k \rfloor = n_o/2k$, it follows that depth $(T_n) \geq 2^{\lfloor n/2k \rfloor}$.

We shall define a point $\bar{a} \in \mathbb{Q}_+^n - K(n)$, and distinct points $\bar{a}_I \in K(n)$, and distinct sets $S(I) \subseteq \{1,\ldots,n\}$, for $I \subseteq \{1,\ldots,p\}$,

such that the only "Knapsack hyperplane" $\{\bar{x} \mid \sum_{i \in S} x_i = 1\}$ (for some $S \subseteq \{1,\ldots,n\}$) on which $\bar{a}_I$ lies is $K_I := \{\bar{x} \mid \sum_{i \in \{I\}} x_i = 1\}$. Since T_n gives different outputs for $\bar{a}$ and $\bar{a}_I$, there is for each $I \subseteq \{1,\ldots,p\}$ a test $\sum_{i=1}^{n} \alpha_i x_i : \alpha_o$ on the path in T_n taken by $\bar{a}$ such that the corresponding "test hyperplane" $\{\bar{x} \mid \sum_{i=1}^{n} \alpha_i x_i = \alpha_o\}$ intersects L_I, the closed line segment starting at $\bar{a}$ and ending at $\bar{a}_I$. The choice of $\bar{a}$ and $\bar{a}_I$ will ensure that the only "test hyperplane" which intersects L_I, if any, is K_I itself. This implies that at least the 2^p tests corresponding to the Knapsack hyperplanes K_I are executed along the computation path for $\bar{a}$. (This is the desired lower bound.) The analogous task was quite easy in the models of [4], [16], since there the only "test hyperplanes" that were allowed in T_n were just the Knapsack hyperplanes (therefore in those models one could choose the components of $\bar{a}$ to be equal). In our case the definition of $\bar{a}$ is more involved: its coordinates will satisfy two kinds of "inaccessibility conditions", together with equalities between certain sums of coordinates.

To simplify notation later, we note that w.l.o.g. we can make the following assumption: if $\delta = \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 \neq o$, where $\gamma_i \in \{o\} \cup \{\gamma \mid \gamma$ or $-\gamma$ is a coefficient in $T_n\}$, then $|\delta| \geq 1$. (If this is not already the case, multiply all coefficients in T_n by C^{-1}, where $C := \min\{|\delta| \mid \delta \neq o, \delta = \sum_{i=1}^{4} \gamma_i$ for $\gamma_i \in \{o\} \cup \{\gamma \mid \gamma$ or $-\gamma$ is a coefficient in $T_n\}\}$.) This assumption allows us to prove the following lemma, which is the first step towards the definition of $\bar{a}$.

LEMMA 1. There exists a number $b \in \mathbb{N}$ such that for all $m \in \mathbb{N}$, all $\delta_i = \gamma_{i1} + \gamma_{i2} + \gamma_{i3} + \gamma_{i4}$, where $\gamma_i \in \{o\} \cup \{\gamma \mid \gamma$ or $-\gamma$ is a coefficient in $T_n\}$ $(i = 1,\ldots,m)$ and all ϵ with $|\epsilon| \leq b^{-m-1}$: if $\sum_{i=1}^{m} \delta_i b^{-i} + \epsilon = o$, then $\delta_1 = \ldots = \delta_m = \epsilon = o$. (The powers of b are mutually "inaccessible" w.r.t. linear combination with coefficients from T_n.)

PROOF: Choose $b \in \mathbb{N}$ so that $b > \max(n + 1, 5 \cdot D)$, where $D := \max\{|\gamma| \mid \gamma$ is a coefficient in $T_n\} \geq 1$. Let m be arbitrary, and assume that ϵ is such that $|\epsilon| \leq b^{-m-1}$, and that $\delta_1,\ldots,\delta_m$ are of the indicated form. Assume that $\sum_{i=1}^{m} \delta_i b^{-i} + \epsilon = o$. We show that $\delta_1 = o$. (Then the lemma follows by induction on m.) Suppose for a contradiction that $\delta_1 \neq o$. By the assumption preceding the lemma we even have $|\delta_1| \geq 1$. Hence $b^{-1} \leq |\delta_1 b^{-1}| = |\sum_{i=2}^{m} \delta_i b^{-i} + \epsilon| \leq \sum_{i=2}^{m+1} 4Db^{-i} \leq \frac{4}{5} \sum_{i=2}^{m+1} b \cdot b^{-i}$ $< \frac{4}{5} \cdot \frac{b^{-1}}{1 - b^{-1}} < b^{-1}$, a contradiction.

For the rest of the proof, we fix some $b \in \mathbb{N}$ as in Lemma 1. Now we define a scaling factor B by

$$B := \sum_{i=1}^{p} b^{-i},$$

and choose $\delta > 0$ so small that $2p\delta < b^{-p-(2p+3)k-2}$, $\delta \in \mathbb{Q}$. For $1 \le i \le p$, let

$$a_i := b_i := \frac{1}{B} \cdot b^{-i} + \delta$$

Note that the point $\frac{1}{B}(b^{-1},b^{-2},\ldots,b^{-p},b^{-1},b^{-2},\ldots,b^{-p})$ lies on the 2^p hyperplanes $\{(y_1,\ldots,y_p,z_1,\ldots,z_p) \in \mathbb{R}^{2p} \mid \sum_{i\in I} y_i + \sum_{i\notin I} z_i = 1\}$, for $I \subseteq \{1,\ldots,p\}$. It turns out that the point $(a_1,\ldots,a_p,b_1,\ldots,b_p)$ lies strictly inside a polytope which has all these 2^p hyperplanes as supporting hyperplanes. This arrangement already allows us to prove the lower bound for linear decision trees with arbitrary nonnegative coefficients: For each $I \subseteq \{1,\ldots,p\}$ we consider a vector $(a_1^I,\ldots,a_p^I, b_1^I,\ldots,b_p^I)$ where

$$a_i^I := \begin{cases} a_i - \delta = \frac{1}{B} \cdot b^{-i}, & \text{if } i \in I \\ a_i, & \text{if } i \notin I, \end{cases} \qquad b_i^I := \begin{cases} b_i, & \text{if } i \in I \\ b_i - \delta = \frac{1}{B} \cdot b^{-i}, & \text{if } i \notin I. \end{cases}$$

Obviously, $\sum_{i\in I} a_i^I + \sum_{i\notin I} b_i^I = 1$. In the next lemma we show that there is at most one hyperplane in $\mathbb{R}^{2p}$ definable with nonnegative coefficients from T_n which makes a difference between $(a_1,\ldots,a_p, b_1,\ldots,b_p)$ and $(a_1^I,\ldots,a_p^I, b_1^I,\ldots,b_p^I)$ in the sense that the two points do not both lie on the hyperplane or in the same of the two open halfspaces defined by it. This hyperplane is

$$\{(y_1,\ldots,y_p, z_1,\ldots,z_p) \in \mathbb{R}^{2p} \mid \sum_{i\in I} y_i + \sum_{i\notin I} z_i = 1\}.$$

LEMMA 2. (Use of inaccessibility of the "first kind".) Let $0 \le \eta \le \delta$. Let for $1 \le i \le p$, $\alpha_i,\beta_i \in \{\gamma \mid \gamma \text{ or } -\gamma \text{ is a coefficient in } T_n\}$, $\alpha_i,\beta_i \ge 0$, but not all α_i,β_i equal 0. Let $\gamma \in \mathbb{R}$ be such that γ or $-\gamma$ is a coefficient in T_n. Finally, let $I \subseteq \{1,\ldots,p\}$ be arbitrary. Assume that for

$$y_i^0 := \begin{cases} a_i - \eta, & \text{if } i \in I \\ a_i & \text{if } i \notin I \end{cases} \qquad z_i^0 := \begin{cases} b_i, & \text{if } i \in I \\ b_i - \eta, & \text{if } i \notin I \end{cases}$$

holds

$$\sum_{i=1}^{p} \alpha_i y_i^0 + \sum_{i=1}^{p} \beta_i z_i^0 = \gamma.$$

Then $\eta = \delta$, and $\forall i \in I : \alpha_i = \gamma \wedge \beta_i = o$, and $\forall i \notin I : \alpha_i = o \wedge \beta_i = \gamma$, i.e. the hyperplane $\{(y_1,\ldots,y_p, z_1,\ldots,z_p) \in \mathbb{R}^{2p} \mid \sum_{i=1}^{p} \alpha_i y_i + \sum_{i=1}^{p} \beta_i z_i = \gamma\}$ equals $\{(y_1,\ldots,y_p, z_1,\ldots,z_p) \in \mathbb{R}^{2p} \mid \sum_{i\in I} y_i + \sum_{i\notin I} z_i = 1\}$.

PROOF: The assumption $\sum_{i=1}^{p} \alpha_i y_i^o + \sum_{i=1}^{p} \beta_i z_i^o = \gamma$ means, by the definitions,

$$\sum_{i=1}^{p} \alpha_i\left(\frac{1}{B} \cdot b^{-i} + \delta\right) - \sum_{i\in I} \alpha_i \eta + \sum_{i=1}^{p} \beta_i\left(\frac{1}{B} b^{-i} + \delta\right) - \sum_{i\notin I} \beta_i \eta = \gamma.$$

Multiplying by $B = \sum_{i=1}^{p} b^{-i}$ and collecting summands with the same power of b yields

$$\sum_{i=1}^{p} (\alpha_i + \beta_i - \gamma) b^{-i} + B \cdot \left[\sum_{i\in I} (\alpha_i(\delta - \eta) + \beta_i \delta) + \sum_{i\notin I} (\alpha_i \delta + \beta_i(\delta - \eta))\right] = o.$$

Since $o \le \delta - \eta \le \delta$ and $\alpha_i + \beta_i \le b$, the second summand is

$$\le B \cdot p \cdot \frac{2b}{5} \cdot \delta < 2 \cdot b^{-1} \cdot p \cdot \frac{b}{2} \cdot \delta = p \cdot \delta < b^{-p-1},$$

by choice of b and δ. By lemma 1 we get $\alpha_i + \beta_i - \gamma = o$, i.e. $\alpha_i + \beta_i = \gamma$, for $1 \le i \le p$ and

$$\sum_{i\in I} (\alpha_i(\delta - \eta) + \beta_i \delta) + \sum_{i\notin I} (\alpha_i \delta + \beta_i(\delta - \eta)) = o.$$

Since the α_i, β_i are $\ge o$ and are not all $= o$, the last equality can hold only if

$$\begin{aligned} &\forall i \in I : \beta_i = o \quad \text{(hence } \alpha_i = \gamma\text{)}, \\ &\forall i \notin I : \alpha_i = o \quad \text{(hence } \beta_i = \gamma\text{), and} \\ &\delta = \eta. \end{aligned}$$

Thus the equation $\sum_{i=1}^{p} \alpha_i y_i + \sum_{i=1}^{p} \beta_i z_i = \gamma$ is in fact the same as $\sum_{i\in I} \gamma y_i + \sum_{i\notin I} \gamma z_i = \gamma$. This proves the claim.

An additional effort is needed if the tree T_n uses questions with both positive and negative coefficients. Clearly, tests like "$x_i - x_j : o$" can distinguish $(a_1,\ldots,a_p, b_1,\ldots,b_p)$ from $(a_1^I,\ldots,a_p^I, b_1^I,\ldots,b_p^I)$, so lemma 2 does not apply any more directly. To accommodate for negative coefficients, we are forced to use another "level" of inaccessible numbers (inaccessibility of the "second kind"): The numbers a_i and b_i $(i = 1,\ldots,p)$ are split into k parts each (e.g. $a_i = a_{i1} + \ldots + a_{ik}$) so that all the $2pk$ parts we obtain are mutually "inaccessible" (with regard to the coefficients which occur in T_n). The vector with all these a_{ij}'s and b_{ij}'s as components will be

the vector $\bar{a} \in \mathbb{R}^n$, with the properties indicated at the beginning of the proof: $\bar{a}$ does not lie on any Knapsack hyperplane in $\mathbb{R}^n$, but for each $I \subseteq \{1,\ldots,p\}$ one can reach from it on a straight line a Knapsack-hyperplane K_I without intersecting any "test-hyperplanes" other than K_I.

NOTATION: For the following, it is convenient to rename the components of vectors $(x_1,\ldots,x_n) \in \mathbb{R}^n$. They are split into two groups and given double indices:

$$\bar{x} = (x_1,\ldots,x_n) = (y_{ij}, z_{ij} \mid 1 \le i \le p,\ 1 \le j \le k).$$

We write $(y_{ij},z_{ij})_{i,j}$ for such vectors in $\mathbb{R}^n$.

DEFINITION: For $i = 1,\ldots,p$ let

$$a_{ij} := \frac{1}{B} \cdot b^{-p-2ik-j}, \quad b_{ij} := \frac{1}{B} \cdot b^{-p-(2i+1)k-j} \quad (2 \le j \le k),$$

$$a_{i1} := \frac{1}{B} \cdot b^{-i} - \sum_{j=2}^{k} a_{ij} + \delta = a_i - \sum_{j=2}^{k} a_{ij}$$

$$b_{i1} := \frac{1}{B} \cdot b^{-i} - \sum_{j=2}^{k} b_{ij} + \delta = b_i - \sum_{j=2}^{k} b_{ij}.$$

$$\bar{a} := (a_{ij},b_{ij})_{i,j}.$$

For $I \subseteq \{1,\ldots,p\}$ let

$$K_I := \{(y_{ij},z_{ij})_{i,j} \mid \sum_{i\in I} \sum_{j=1}^{k} y_{ij} + \sum_{i\notin I} \sum_{j=1}^{k} z_{ij} = 1\}.$$

(K_I is a Knapsack-hyperplane close to $\bar{a}$.) "Characteristic vectors" $\bar{c}_I = (c^I_{ij},d^I_{ij})$, defined by

$$c^I_{ij} := \begin{cases} 1, & \text{if } i \in I \text{ and } j = 1 \\ 0, & \text{otherwise} \end{cases}, \quad d^I_{ij} := \begin{cases} 1, & \text{if } i \notin I \text{ and } j = 1 \\ 0, & \text{otherwise,} \end{cases}$$

are needed to define the line segments L_I from $\bar{a}$ to points $\bar{a}_I \in K_I$:

$$L_I := \{\bar{a} - \eta\bar{c}_I \mid 0 \le \eta \le \delta\}$$
$$\bar{a}_I := \bar{a} - \delta\bar{c}_I.$$

The following three lemmata verify that $\bar{a}$ and the L_I, K_I, $\bar{a}_I$ ($I \subseteq \{1,\ldots,p\}$) have the desired properties:

- $\bar{a} \notin K(n)$ (Corollary to Lemma 5)
- $\bar{a}_I \in K_I \subseteq K(n)$ (Lemma 3)
- if L_I intersects a test hyperplane, then this test hyperplane equals K_I (Corollary to Lemma 5).

As we have argued at the beginning of the proof, these properties together with the obvious fact that the K_I are all different from each other imply that the path in the tree T_n which is taken by $\bar{a}$ contains $\geq 2^p$ tests, which is what we wanted to show.

LEMMA 3: For all $I \subseteq \{1,\dots,p\}$: $\bar{a}_I \in K_I \subseteq K(n)$.

PROOF: Straightforward computation, using the facts that

$$a_{i1} + \sum_{j=2}^{k} a_{ij} - \sum_{j=1}^{k} \delta c^I_{ij} = \frac{1}{B} \cdot b^{-i} \quad \text{for } i \in I \quad \text{and}$$

$$b_{i1} + \sum_{j=2}^{k} b_{ij} - \sum_{j=1}^{k} \delta d^I_{ij} = \frac{1}{B} \cdot b^{-i} \quad \text{for } i \notin I, \text{ and } \frac{1}{B} \cdot \sum_{i=1}^{p} b^{-i} = 1.$$

LEMMA 4: (Use of inaccessibility of the "second kind") Let $0 \leq \eta \leq \delta$. Let α_{ij}, β_{ij} be real numbers which occur as coefficients in T_n, for $1 \leq i \leq p$, $1 \leq j \leq k$. Let $\gamma \in \mathbb{R}$ be such that γ or $-\gamma$ is a coefficient in T_n. Let $I \subseteq \{1,\dots,n\}$ be arbitrary. Assume that $\bar{x} = (y_{ij}, z_{ij})_{i,j} = \bar{a} - \eta \bar{c}_I \in L_I$ satisfies

$$(*) \qquad \sum_{i,j} \alpha_{ij} y_{ij} + \sum_{i,j} \beta_{ij} z_{ij} = \gamma.$$

Then $\alpha_{ij} = \alpha_{i1}$ and $\beta_{ij} = \beta_{i1}$ for $1 \leq i \leq p$, $2 \leq j \leq k$.

PROOF: We rewrite (*) according to the definitions:

$$\sum_{i,j} \alpha_{ij}(a_{ij} - \eta c^I_{ij}) + \sum_{i,j} \beta_{ij}(b_{ij} - \eta d^I_{ij}) = \gamma, \quad \text{i.e.}$$

$$\sum_{i=1}^{p} \alpha_{i1}\Big(\frac{1}{B} \cdot b^{-i} - \sum_{j=2}^{k} a_{ij} + \delta - \eta c^I_{i1}\Big) + \sum_{i=1}^{p}\sum_{j=2}^{k} \alpha_{ij} a_{ij} +$$

$$+ \sum_{i=1}^{p} \beta_{i1}\Big(\frac{1}{B} \cdot b^{-i} - \sum_{j=2}^{k} b_{ij} + \delta - \eta d^I_{i1}\Big) + \sum_{i=1}^{p} \sum_{j=2}^{k} \beta_{ij} b_{ij} = \gamma.$$

Multiply both sides by $B = \sum_{i=1}^{p} b^{-i}$, and recall that $Ba_{ij} = b^{-p-2ik-j}$, $Bb_{ij} = b^{-p-(2i+1)k-j}$, for $1 \leq i \leq p$, $2 \leq j \leq k$; then collect summands containing the same power of b:

$$\sum_{i=1}^{p} (\alpha_{i1} + \beta_{i1} - \gamma)b^{-i} + \sum_{i=1}^{p}\sum_{j=2}^{k} (\alpha_{ij} - \alpha_{i1})b^{-p-2ik-j} +$$

$$+ \sum_{i=1}^{p}\sum_{j=2}^{k} (\beta_{ij} - \beta_{i1})b^{-p-(2i+1)k-j} +$$

$$+ B \cdot \Big(\sum_{i\in I}(\alpha_{i1}(\delta - \eta) + \beta_{i1}\delta) + \sum_{i\notin I}(\alpha_{i1}\delta + \beta_{i1}(\delta - \eta))\Big) = o.$$

The absolute value of the last summand is $\leq B \cdot p \cdot 2 \cdot \frac{b}{5} \cdot \delta < b^{-p-(2p+3)k-1}$, by the choice of δ. We apply Lemma 1 to obtain $\alpha_{ij} - \alpha_{i1} = \beta_{ij} - \beta_{i1} = o$ for $1 \leq i \leq p$, $2 \leq j \leq k$.

LEMMA 5: Let η, α_{ij}, β_{ij} $(1 \leq i \leq p,\ 1 \leq j \leq k)$, γ, I be as in Lemma 4, such that the α_{ij}, β_{ij} are not all o. Assume that $\bar{x} = (y_{ij}, z_{ij})_{i,j} = \bar{a} - \eta\bar{c}_I$ satisfies (*), and that

$$|\{(i,j) \mid \alpha_{ij} < o\}| + |\{(i,j) \mid \beta_{ij} < o\}| < k.$$

Then the hyperplane defined by (*) equals K_I, and $\eta = \delta$, i.e. $\bar{x} = \bar{a}_I$.

COROLLARY: i) If $\sum_{i,j} \alpha_{ij}y_{ij} + \sum_{i,j} \beta_{ij}z_{ij} : \gamma$ is a test in T_n, and L_I has a point in common with $\{(y_{ij}, z_{ij})_{i,j} \mid \sum_{i,j} \alpha_{ij}y_{ij} + \sum_{i,j} \beta_{ij}z_{ij} = \gamma\}$, then this hyperplane equals K_I. ii) $\bar{a} \notin K(n)$.

PROOF OF LEMMA 5: Applying Lemma 4 yields $\alpha_{ij} = \alpha_{i1}$, $\beta_{ij} = \beta_{i1}$, for $1 \leq i \leq p$, $2 \leq i \leq k$. Hence none of the coefficients can be negative (otherwise $\geq k$ of them would be negative, contradicting the assumption). We now collect summands with the same coefficients in (*) and obtain

$$\sum_{i=1}^{p} \alpha_{i1} \cdot \sum_{j=1}^{k} (a_{ij} - \eta c_{ij}^I) + \sum_{i=1}^{p} \beta_{i1} \cdot \sum_{j=1}^{k} (b_{ij} - \eta d_{ij}^I) = \gamma.$$

By the definition of the a_{ij}, b_{ij}, c_{ij}^I, d_{ij}^I, a_i, b_i, this is the same as

$$\sum_{i=1}^{p} \alpha_{i1} \cdot (a_i - \eta \cdot c_{i1}^I) + \sum_{i=1}^{p} \beta_{i1}(b_i - \eta d_{i1}^I) = \gamma.$$

To this equation we apply Lemma 2, and we get

$\eta = \delta$, $\forall i \in I : \alpha_{i1} = \gamma \wedge \beta_{i1} = o$, and $\forall i \notin I : \alpha_{i1} = o \wedge \beta_{i1} = \gamma$.

γ can not be o, since some of the α_{ij}, β_{ij} were assumed to be $\neq o$. By multiplying (*) by γ^{-1} we finally get that (*) is equivalent to

$$\sum_{i\in I}\sum_{j=1}^{k} y_{ij} + \sum_{i\notin I}\sum_{j=1}^{k} z_{ij} = 1,$$

which is the equation defining K_I.

PROOF OF COROLLARY: i) If $\sum_{i,j} \alpha_{ij}y_{ij} + \sum_{i,j} \beta_{ij}z_{ij} : \gamma$ is a test in T_n, then w.l.o.g. not all coefficients are o, and the number of negative coefficients among the α_{ij},β_{ij} is less than k by the assumption we made about T_n. So the lemma applies directly. ii) Let K be an arbitrary Knapsack hyperplane. In our notation, K has the form $\{(y_{ij},z_{ij})_{i,j} \mid \sum_{i,j} \alpha_{ij}y_{ij} + \sum_{i,j} \beta_{ij}z_{ij} = 1\}$ for certain $\alpha_{ij},\beta_{ij} \in \{o,1\}$, not all o. Suppose that $\bar{a} - \eta \cdot \bar{c}_I \in K$ for some η, $o \leq \eta \leq \delta$, and some $I \subseteq \{1,\ldots,p\}$. Choose an arbitrary $\gamma > o$ such that γ or $-\gamma$ is a coefficient in T_n. Then $\bar{x} = (y_{ij},z_{ij})_{i,j} = \bar{a} - \eta \cdot \bar{c}_I$ satisfies

$$\sum_{i,j} (\alpha_{ij}\gamma)y_{ij} + \sum_{i,j} (\beta_{ij}\gamma)z_{ij} = \gamma.$$

Applying Lemma 5 to this situation yields $\eta = \delta$, i.e. $\bar{x} = \bar{a}_I \notin \bar{a}$. In particular, $a \notin K$. This holds for all Knapsack hyperplanes K, hence $\bar{a} \notin K(n)$.

This finishes the proof of Theorem 1.

§3. Lower bounds for graph problems on linear decision trees.

In this section the method of §2 is applied to some languages defined in terms of graphs with weighted edges: the shortest path problem, the minimum perfect matching problem, and the traveling salesperson problem. The main result (Theorem 2) says essentially that a linear decision tree can not solve these problems fast, i.e. recognize the corresponding languages fast, unless it can compare sums of many input numbers to each other. Thus the comparisons of lengths of paths in any of the standard polynomial time algorithms for the shortest path problem or the minimal perfect matching problem are essential. This observation pinpoints a difference between these problems and, say, the minimum spanning tree problem which can be solved in polynomial time by linear decision trees which use only comparisons of single edges.

An equivalent formulation of weighted graph problems as recognition problems in $\mathbb{R}^n$ is obtained as follows. For $m \in \mathbb{N}$ consider the complete graph K_m on m vertices $v_1,\ldots,v_m$. Fix a numbering $e_1,\ldots,e_n$ of the $n = \frac{1}{2}m(m-1)$ edges of K_m. Then there is a one-one correspon-

dence between vectors $(x_1,\ldots,x_n) \in \mathbb{R}^n$ and weight functions $w : \{e_1,\ldots,e_n\} \to \mathbb{R}$ which assign a weight $w(e_i)$ to every edge e_i, the correspondence being given by $w(e_i) = x_i$, for $1 \leq i \leq n$.

The problem SHORTEST PATH as a decision problem, can be formulated as follows: is there a path from v_1 to v_2 of total weight ≤ 1? This problem corresponds to the language

$$SP(n) := \{\bar{x} \in \mathbb{R}^n \mid \exists S \subseteq \{1,\ldots,n\}\ (\{e_i \mid i \in S\} \text{ forms a path in } K_m \text{ between } v_1 \text{ and } v_2 \text{ and } \sum_{i \in S} x_i \leq 1)\}.$$

Similarly, the problem MINIMUM PERFECT MATCHING gives rise to the following recognition problem:

$$PM(n) := \{\bar{x} \in \mathbb{R}^n \mid \exists S \subseteq \{1,\ldots,n\}\ (\{e_i \mid i \in S\} \text{ forms a perfect matching in } K_m \text{ and } \sum_{i \in S} x_i \leq 1)\}.$$

Finally, the TRAVELLING SALESPERSON problem, i.e. the problem to decide whether there is a Hamiltonian cycle in K_m of total weight ≤ 1, gives rise to the language

$$TSP(n) := \{\bar{x} \in \mathbb{R}^n \mid \exists S \subseteq \{1,\ldots,n\}(\{e_i \mid i \in S\} \text{ forms a Hamiltonian cycle in } K_m \text{ and } \sum_{i \in S} x_i \leq 1)\}.$$

There is a geometrical difference between the languages $K(n)$ of §2 and the languages just defined. $K(n)$ consists of a union of hyperplanes in $\mathbb{R}^n$, whereas $SP(n)$, $PM(n)$, $TSP(n)$ are unions of closed halfspaces in $\mathbb{R}^n$. The following variation of Theorem 1 adapts the results of §2 to this situation.

COROLLARY TO THEOREM 1. Let, for $k, p \in \mathbb{N}$

$$L(p,k) := \{(y_{ij},z_{ij})_{1\leq i\leq p,\,1\leq j\leq k} \in \mathbb{R}^{2pk} \mid \exists I \subseteq \{1,\ldots,p\}(\sum_{i \in I} \sum_{j=1}^{k} y_{ij} + \sum_{i \notin I} \sum_{j=1}^{k} z_{ij} \leq 1)\}.$$

Let $T_{p,k}$ be a linear decision tree for inputs from $\mathbb{R}^{2pk}$ which recognizes $L(p,k)$, such that all tests in $T_{p,k}$ contain less than k negative coefficients. Then depth $(T_{p,k}) \geq 2^p$.

PROOF: Consider the points $\bar{a}$ and $\bar{a}_I$ ($I \subseteq \{1,\dots,p\}$) in $\mathbb{R}^{2pk}$ as in the proof of Theorem 1. We observe that $\bar{a} \notin L(p,k)$:

$$\sum_{i\in I}\sum_{j=1}^{k} a_{ij} + \sum_{i\notin I}\sum_{j=1}^{k} b_{ij} = \sum_{i\in I}\left(\frac{1}{B}\cdot b^{-i} + \delta\right) + \sum_{i\notin I}\left(\frac{1}{B}\cdot b^{-i} + \delta\right) = 1 + p\delta > 1,$$

for all $I \subseteq \{1,\dots,p\}$. But $\bar{a}_I \in L(p,k)$ for all $I \subseteq \{1,\dots,p\}$, since

$$\sum_{i\in I}\sum_{j=1}^{k}(a_{ij} - \delta c^I_{ij}) + \sum_{i\notin I}\sum_{j=1}^{k}(b_{ij} - \delta d^I_{ij}) = \sum_{i\in I}\left(\frac{1}{B}\cdot b^{-i} + \delta - \delta d^I_{il}\right) + \sum_{i\in I}\left(\frac{1}{B}\cdot b^{-i} + \delta - \delta d^I_{il}\right) = 1.$$

Hence on the path in $T_{p,k}$ which is taken by $\bar{a}$ there must be a test for each $I \subseteq \{1,..,p\}$ which can distinguish $\bar{a}$ from $\bar{a}_I$. By the corollary to Lemma 5 in §2 the hyperplane corresponding to such a test is K_I. Hence the path in $T_{p,k}$ taken by $\bar{a}$ has length $\geq 2^p$.

The languages $L_{p,k}$ will be "reduced" to the graph problems we consider here. Thus we get lower bounds in the following manner: from a linear decision tree T which solves the graph problem, using few negative coefficients in its tests, we obtain an LDT of the same structure which recognizes $L(p,k)$, for certain $p, k \in \mathbb{N}$. This implies that depth $(T) \geq 2^p$, by the above corollary.

THEOREM 2. Let $(T_n)_{n\geq 1}$ be a sequence of LDT's, $f : \mathbb{N} \to \mathbb{N}$ a function such that each test in T_n uses less than $f(n)$ negative coefficients. If T_n recognizes one of the languages SP(n), PM(n), TSP(n), then depth $(T_n) \geq 2^{\lfloor\sqrt{n}/2f(n)\rfloor}$, for n of the form $\frac{1}{2}m(m-1)$, $m \in \mathbb{N}$.

NOTE: This lower bound is superpolynomial if $f(n) = o\left(\frac{\sqrt{n}}{\log n}\right)$.

PROOF: In each of the three cases, we obtain from T_n an LDT $T_{p,k}$ of the same depth as T_n which recognizes $L(p,k)$, where $k := f(n)$ and $p := \frac{m}{2f(n)}$, and $T_{p,k}$ uses $< k$ negative coefficients in its tests. Then, by the above corollary,

$$\text{depth } (T_n) = \text{depth } (T_{p,k}) \geq 2^p = 2^{m/2f(n)} \geq 2^{\lfloor\sqrt{n}/2f(n)\rfloor}.$$

a) Suppose T_n recognizes SP(n). We restrict our attention to a fixed subgraph G_1 of K_m as sketched in Figure 1. G_1 uses $(2k-1)p+1 < m$ vertices (among them v_1 and v_2) and $2kp$ edges. The variables x_i corresponding to the edges of this subgraph are

renamed y_{ij} resp. z_{ij} $(1 \le i \le p,\ 1 \le j \le k)$, as indicated in Figure 1. We consider only input vectors $\bar{x} = (x_1,\ldots,x_n) \in \mathbb{R}_+^n$ which give weight 2 to all edges not in G_1. Then it is clear that such edges can not occur in a path of length ≤ 1. We change T_n by fixing the values of the x_i's corresponding to such edges to be 2. The result is an LDT T' for inputs $(y_{ij},z_{ij})_{1\le i\le p,\,1\le j\le k}$ which accepts

$\{(y_{ij},z_{ij})_{i,j} \mid$ one of the 2^p possible paths from v_1 to v_2 in G_1 has weight $\le 1\}$.

This language obviously equals $L(p,k)$. Furthermore, no test in T' uses $\ge f(n) = k$ negative coefficients, since this was true in T_n.

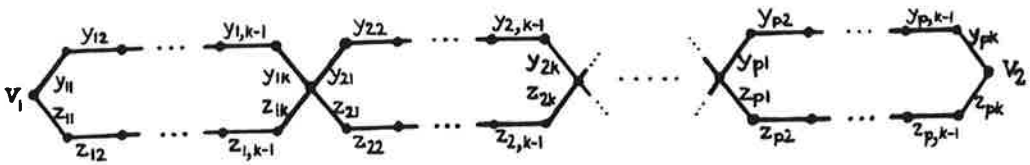


Figure 1

b) Suppose T_n recognizes PM(n). This time we consider only a fixed subgraph G_2 of K_m of the form given in Figure 2. G_2 has $m = 2pk$ vertices and $2pk$ edges. The variables x_i corresponding to the edges of this subgraph are renamed y_{ij} resp. z_{ij} $(1 \le i \le p,\ 1 \le j \le k)$, as indicated in Figure 2. We obtain a new LDT T" from T_n by fixing the values of all other x_i to be 2. Clearly, a perfect matching made up from edges in G_2 either contains all edges corresponding to $y_{i1},\ldots,y_{ik}$ or all edges corresponding to $z_{i1},\ldots,z_{ik}$, for $1 \le i \le p$. Hence the language (which is recognized by T")

$\{(y_{ij},z_{ij})_{i,j} \mid$ there is a perfect matching in G_2 of weight $\le 1\}$

equals $L(p,k)$.

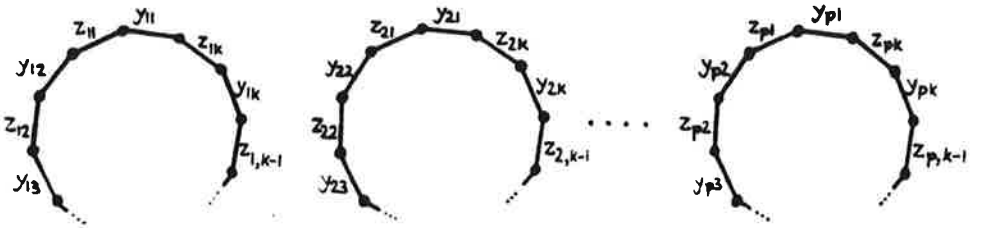


Figure 2

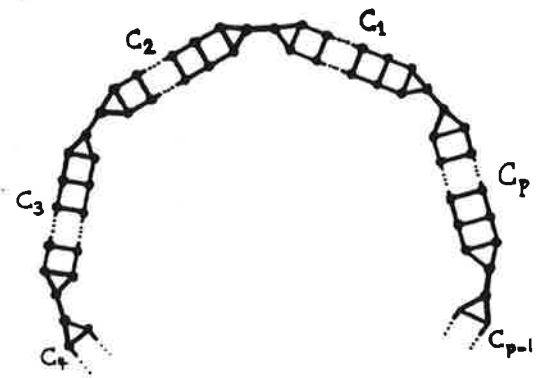
c) Suppose T_n recognizes TSP(n). We consider a fixed subgraph G_3 of K_m of the form given in Figure 3. Give constant weight 2 to all edges not contained in G_3, and weight $\frac{1}{2} \cdot \frac{1}{kp}$ to the edges corresponding to variables u_{ij} $(1 \leq i \leq p,\ 1 \leq j \leq k)$. Consider the tree T''' obtained from T_n by fixing these input variables and renaming the other ones to y_{ij}, z_{ij} as indicated in Figure 3. Then a Hamiltonian cycle of weight ≤ 1 obviously uses in component C_i either the edges $y_{i1}, u_{i1}, y_{i2}, \ldots, y_{ik}, u_{ik}$ or the edges $z_{i1}, u_{i1}, z_{i2}, \ldots, z_{ik}, u_{ik}$ (for $1 \leq i \leq k$). Hence the language

$$\{(y_{ij}, z_{ij})_{i,j} \mid \text{there is a Hamiltonian cycle of length} \leq 1 \text{ in } G_3\}$$

equals

$$\{(y_{ij}, z_{ij})_{i,j} \mid \exists I \subseteq \{1, \ldots, p\} \left(\sum_{i \in I} \sum_{j=1}^{k} y_{ij} + \sum_{i \notin I} \sum_{j=1}^{k} z_{ij} \leq \frac{1}{2} \right)\},$$

a language so similar to $L(p,k)$ that it obviously implies the lower bound 2^p for the depth of T'''.



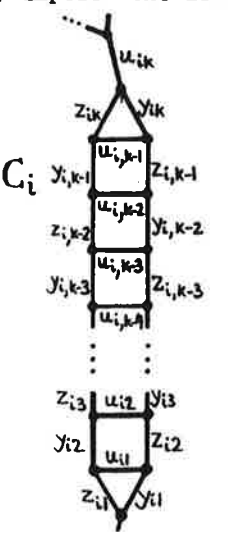


Figure 3

REMARK: Application to geometrical instances

The lower bound of Theorem 2 stays valid if the LDT T_n is only required to solve the respective graph problem for the following restricted class of "geometrical instances": incomplete graphs on m vertices which can be drawn in the Euclidean plane in such a way that the edges are straight lines, no two edges cross, and the weight of each edge equals its length. (We assume that T_n uses an additional type of test which allows it to find out whether an edge is present in the input graph or not.) To construct "difficult inputs" for an LDT T_n which can decide the graph problem only for instances from this restricted class, we can use the same subgraphs of K_m as in Figures 1-3. But we can not use the same weight sets as in the proof of Theorem 2, since there the weights were vastly different from each other (e.g. a_{11} is very much larger than a_{12}), and it is not clear if one can draw graphs in the required way with these weights as edge lengths. If, however, the edge weights y_{ij}, z_{ij} $(1 \le i \le p,\ 1 \le j \le k)$ of the graphs depicted in Figures 1-3 are all equal, these graphs can be drawn in this manner. We will exploit in the following that the same is true if the edge weights are nearly the same, say if $\frac{1}{pk}(1-\epsilon) \le y_{ij}, z_{ij} \le \frac{1}{pk}$ for all i,j, for a sufficiently small $\epsilon = \epsilon(p,k)$. There is an easy way to obtain such "geometrical" weight sets from arbitrary ones: add a large constant to all edge weights, then scale down by a constant factor. More precisely, if any weight set $(\tilde{y}_{ij}, \tilde{z}_{ij})_{i,j}$ with $0 \le \tilde{y}_{ij}, \tilde{z}_{ij} \le 1$ is given, the new weight set $(y_{ij}, z_{ij})_{i,j}$ defined by

$$y_{ij} := \epsilon \cdot \tilde{y}_{ij} + \frac{1-\epsilon}{pk}, \qquad z_{ij} := \epsilon \cdot \tilde{z}_{ij} + \frac{1-\epsilon}{pk}$$

belongs to a graph which can be drawn in the required manner. This observation leads to the following "reduction procedure", described here for the case of SHORTEST PATH. Suppose an LDT T_n is given which accepts weight sets for subgraphs of K_m which admit a path of length ≤ 1 between v_1 and v_2, but T_n does so only for input vectors which arise from "geometrical instances" in the way described above. We define k and p as before, and restrict our attention to the subgraph G_1 of Figure 1, renaming variables as in the proof of Theorem 1. Modify T_n as follows: replace tests

$$\sum_{i,j} \alpha_{ij} y_{ij} + \sum_{i,j} \beta_{ij} z_{ij} : \gamma \qquad \text{by}$$

$$\sum_{i,j} \alpha_{ij} \tilde{y}_{ij} + \sum_{i,j} \beta_{ij} \tilde{z}_{ij} : \frac{1}{\epsilon} \cdot \left[\gamma - \sum_{i,j} (\alpha_{ij} + \beta_{ij}) \cdot \frac{1-\epsilon}{pk}\right],$$

and call the tree so obtained $\tilde{T}_n$. Then it is easily checked that for $o \le \tilde{y}_{ij}, \tilde{z}_{ij} \le 1$

$\tilde{T}_n$ accepts $(\tilde{y}_{ij},\tilde{z}_{ij})_{i,j}$ iff (by definition of $\tilde{T}_n$)

T_n accepts $(\epsilon\tilde{y}_{ij} + \frac{1-\epsilon}{pk}, \epsilon\tilde{z}_{ij} + \frac{1-\epsilon}{pk})_{i,j}$ iff (by the structure of G_1)

$\exists I \subseteq \{1,\dots,p\}$ $\left(\sum_{i\in I} \sum_{j=1}^{k} (\epsilon\tilde{y}_{ij} + \frac{1-\epsilon}{pk}) + \sum_{i\notin I} \sum_{j=1}^{k} (\epsilon\tilde{z}_{ij} + \frac{1-\epsilon}{pk}) \le 1 \right)$ iff

$\exists I \subseteq \{1,..,p\}$ $\left(\sum_{i\in I} \sum_{j=1}^{k} \tilde{y}_{ij} + \sum_{i\notin I} \sum_{j=1}^{k} \tilde{z}_{ij} \le 1 \right)$.

Thus $\tilde{T}_n$ recognizes $L(p,k)$ for inputs from $[o,1]^{2pk}$ (the language $L(p,k)$ was defined in the corollary at the beginning of §3). The lower bound proof of §2 uses only inputs in $[o,1]^{2pk}$, hence $\tilde{T}_n$ has depth $\ge 2^p$, and so does T_n.

Similar constructions yield the same result for PM(n) and TSP(n).

§4. A lower bound on the time for spacebounded random access machines

We refer to §1 for the definition of the here considered problems and machine models.

THEOREM 3. Let R be a random access machine (RAM) that recognizes ELEMENT DISTINCTNESS (respectively DISJOINT SETS). Assume that R uses for inputs from $\mathbb{N}^n$ only its first $2^{f(n)}$ registers, where $f : \mathbb{N} \to \mathbb{N}$ is an arbitrary function. Then R uses $\Omega(n \log n)$ computation steps.

SKETCH OF THE PROOF (see [8] for details): Fix a RAM R with space bound $2^{f(n)}$ that recognizes ELEMENT DISTINCTNESS (the argument for DISJOINT SETS is similar). Obviously R can generate in t steps, starting from numbers $\le b$, only numbers of size $\le 2^t \cdot b$. Therefore for computations of length $< n \log n$ the numbers in the set

$$IN = \{\, 2^{f(n)} + i \cdot n \log n \mid 1 \le i \le n \,\}$$

are mutually "inaccessible" from the point of view of R. We consider the "test set" $T \subseteq$ ELEMENT DISTINCTNESS that consists of all $n!$ permutations of IN and we show that R uses $\ge \frac{n \log n}{4}$ steps for some input $I \in T$. More specifically, we will show that under the

assumption that R applies $< \frac{n \log n}{2}$ arithmetical operations to each input from T, R defines for each input I from T a different binary sequence P(I) that codes the outcomes of all comparison steps in that computation on input I.

The proof makes use of the following simple fact, which follows from the inaccessibility of the numbers in IN: every register content that occurs during the computation of R on input $(x_1,\ldots,x_n) \in T$ can be written uniquely in the normal form: $s_o + \sum_{i=1}^{n} s_i \cdot x_i$, where $s_i \in \mathbb{Z}$ and $|s_i| \leq 2^{n \log n}$. Assume for a contradiction that there are two different inputs $I = (x_1,\ldots,x_n)$ and $I' = (x'_1,\ldots,x'_n)$ in T with $P(I) = P(I')$. Let π be the permutation of $\{1,\ldots,n\}$ so that $x_{\pi(1)} < x_{\pi(2)} <\ldots< x_{\pi(n)}$. Choose ℓ minimal so that $x'_{\pi(\ell+1)} < x'_{\pi(\ell)}$. Consider the variation $\tilde{I} = (\tilde{x}_1,\ldots,\tilde{x}_n)$ of input I where the $\pi(\ell+1)$-th component of I has been replaced by another copy of $x_{\pi(\ell)}$ (thus $\tilde{x}_{\pi(\ell+1)} = \tilde{x}_{\pi(\ell)} = x_{\pi(\ell)}$ and $\tilde{I} \notin$ ELEMENT DISTINCTNESS). We will show that since R did not "notice" that the relative order of the $\pi(\ell)$-th and $\pi(\ell+1)$-th component is different in I and I' ($x_{\pi(\ell)} < x_{\pi(\ell+1)}$, but $x'_{\pi(\ell)} > x'_{\pi(\ell+1)}$, R will also not notice that $\tilde{x}_{\pi(\ell)} = x'_{\pi(\ell+1)}$) in $\tilde{I}$ (and therefore R will accept $\tilde{I}$ just as it accepted I and I'). More precisely one shows by induction on t that for all $t \leq n \log n$ R executes for all three inputs I, I' and $\tilde{I}$ the same instruction at step t, and that if some register r_a has at the end of step t for input I the content $s_o + \sum_{i=1}^{n} s_i x_i$ (with $|s_i| \leq 2^{n \log n}$) then the same register r_a holds at the end of step t for input I' the number $s_o + \sum_{i=1}^{n} s_i x'_i$, and for input $\tilde{I}$ the number $s_o + \sum_{i=1}^{n} s_i \tilde{x}_i$. Note that indirect addressing causes no problem in this argument since $o \leq s_o + \sum_{i=1}^{n} s_i x_i < 2^{f(n)}$ implies that $s_i = o$ for $i \geq 1$. The only nontrivial step in the inductive argument occurs when R compares at step t the content of register r_o with o. One has to show that the outcome is the same for I, I' and $\tilde{I}$. Let $s_o + \sum_{i=1}^{n} s_i x_i$ be the content of r_o at step t for input I. By the induction hypothesis we know that $s_o + \sum_{i=1}^{n} s_i x'_i$ ($s_o + \sum_{i=1}^{n} s_i \tilde{x}$) are the corresponding contents of r_o at step t for input I'($\tilde{I}$). Consider the case where $s_o + \sum_{i=1}^{n} s_i x_i > o$. By assumption we have $P(I) = P(I')$ and therefore $s_o + \sum_{i=1}^{n} s_i x'_i > o$. We have to show that $s_o + \sum_{i=1}^{n} s_i \tilde{x}_i > o$. Let j be maximal so that $s_{\pi(j)} \neq o$ (in the considered case this implies that $s_{\pi(j)} > o$). If $j > \ell + 1$ this immediately implies that $s_o + \sum_{i=1}^{n} s_i \tilde{x}_i > o$. If $j < \ell + 1$ then $s_o + \sum_{i=1}^{n} s_i x_i = s_o + \sum_{i=1}^{n} s_i \tilde{x}_i$. Thus the only nontrivial case occurs when $j = \ell + 1$ and $s_{\pi(\ell)} < o$. However in this case we get a contradicition to $s_o + \sum_{i=1}^{n} s_i x'_i > o$ since by the

choice of ℓ we have $x'_{\pi(1)} < \ldots < x'_{\pi(\ell)}$ and $x'_{\pi(\ell+1)} < x'_{\pi(\ell)}$ (further $s_{\pi(i)} = 0$ for $i > \ell + 1$ by the choice of j). All other cases are handled analogously.

With the help of Ramsey's Theorem (see [5], [15]) one can derive the following generalization (here one selects out of a very large pool IN of mutually "inaccessible" numbers n special numbers that are "indiscernible" for the oracle Q — except for their order).

THEOREM 4: The lower bound of Theorem 3 remains valid if one allows the RAM R to use an arbitrary oracle $Q \subseteq \mathbb{R}^q$ (for any constant $q \in \mathbb{N}$) that answers questions about arbitrary q-tupels of input numbers.

ACKNOWLEDGEMENTS: We thank Friedhelm Meyer auf der Heide and György Turan for helpful discussions.

REFERENCES:

[1] A.V. Aho, J.E. Hopcroft, J.D. Ullman, The Design and Analysis of Computer Algorithms, Reading, Mass., 1974.

[2] M. Ben-Or, Lower bounds for algebraic computation trees, Proc. 15th STOC (1983), 80-86.

[3] D.P. Dobkin and R.J. Lipton, A lower bound of $n^2/2$ on linear search programs for the knapsack problem, J. Comput. System Sci. 16(1978), 413-417.

[4] D.P. Dobkin and R.J. Lipton, On the complexity of computations under varying sets of primitives, J. Comput. System Sci. 18(1979), 86-91.

[5] R.L. Graham, B.L. Rothschild, J.H. Spencer, Ramsey Theory (Wiley, New York 1980).

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

[7] W. Maass, An optimal quadratic lower bound for random access machines and other applications of Ramsey's theorem, Preliminary Report (Berkeley, 1984).

[8] W. Maass, On the use of inaccessible numbers and order indiscernibles in lower bound arguments for random access machines, U. of Ill. at Chicago, Res. Report in C. Sc. No.4 (to appear in the Journal of Symbolic Logic)

[9] S. Moran, M. Snir and U. Manber, Applications of Ramsey's theorem to decision tree complexity, Proc. of 25th FOCS 1984, 332-337.

[10] S. Moran, M. Snir and U. Manber, Applications of Ramsey's theorem to decision tree complexity, J. of the ACM 32(1985), 938-949.

[11] F. Meyer auf der Heide, A polynomial linear search algorithm for the n-dimensional knapsack problem, Proc. 15th Ann. ACM Symp. on Theory of Computing (1983), 70-79.

[12] F. Meyer auf der Heide, Lower bounds for solving linear diophantine equations on random access machines, Tech. Report 6/84 (Universitaet Frankfurt).

[13] F. Meyer auf der Heide, Fast algorithms for n-dimensional restrictions of hard problems, Proc. 17th ACM STOC (1985), 413-420.

[14] W.J. Paul and J. Simon, Decision trees and random access machines, Proc. Symp. Logik und Algorithmik (Zuerich, 1980), 331-339.

[15] G.E. Sacks, Saturated model theory (Benjamin, Reading 1972).

[16] E. Ukkonen, Exponential lower bounds for some NP-complete problems in a restricted linear decision tree model, B.I.T. 23(1983), 181-192.