Composition of the Universal Relation

Johan Håstad[*] Avi Wigderson[†]

September 26, 1997

Abstract

We prove that the communication complexity of the k-fold composition of the universal relation on n bits is $(1-o(1))kn$ when $k = o(\sqrt{n/\log n})$.

1 Introduction

Does $NC^1 \neq P$? If so, as most people believe, what are the inherently sequential functions in P? One reasonable answer is functions who are complete for P under NC^1 reductions, such as the circuit value problem or maximum flow. However, completeness alone does not seem to provide a concrete direction in pursuing a depth lower bound for these functions.

In trying to develop a concrete approach, Karchmer, Raz and Wigderson [KRW] proposed a different class of functions that seems inherently sequential to compute. These functions are obtained from arbitrary functions by the composition operator. Given a function on (a vector of) n bits f define the function $f \circ f$, the composition of f with itself, of (an $n \times n$ matrix of) n^2 bits as follows: Simply apply f to each row of the input matrix, and then apply f again to the resulting vector. The k-fold composition of f, $f^{(k)}$, of (a k-dimensional matrix of) n^k bits is similarly defined by repeatedly applying f to the rows k times - each time the dimension decreases by one till we get a Boolean value.

If f requires circuit depth $d(f)$, then the definition of composition yields a circuit for $f^{(k)}$ of depth $kd(f)$. Is this sequential way of computing $f^{(k)}$ inherent? [KRW] conjecture that it is, i.e that $d(f^{(k)}) = \Omega(kd(f))$ for every function f and every k. More importantly, they show that this conjecture (and in fact, much weaker one in which $d(f)$ is replaces by anything asymptotically larger than its logarithm!) implies that $NC^1 \neq P$.

[KRW] presents some evidence in support of their conjecture. One piece of evidence is that the conjecture holds for some functions in the monotone circuit model. Another is a result

[*] Royal Institute of Technology

[†] Princeton University and Hebrew University

of Andreev [An] showing that in the nonmonotone model composing a random function with the parity function (obviously extending the definition of composition) requires depth that is substantially more than the depth of either of the two functions (for exact statement see e.g. [BS]). To further explore the intuition for this conjecture [KRW] suggest to understand it from the communication complexity point of view of [KW], and the notion of universal relations described below.

Karchmer and Wigderson [KW] proved that the depth $d(f)$ of a function f is exactly the communication complexity of the following search problem: One player is given an n bit vector x such that $f(x) = 1$ and the other a vector y with $f(y) = 0$. Their task is to find a coordinate in which x and y differ, i.e. an index $i \in [n]$ satisfying $x_i \neq y_i$.

The simultaneous computation of the communication problems above for all functions f on n bits is captured by the following universal relation U. Each player gets an n-bit vector, and their task is to find a coordinate where their input differ (if one exists) or answer 'bad' when $x = y$ (which is "illegal" as it never occurs for functions). The communication complexity of U is at most $n + 2$, which says that even though it is harder than the search problem for each particular f, it is not much harder than most of them.

In a similar way [KRW] abstract the communication problem U_2 associated with the composition of f with itself. One player gets a pair (M, x) where M is an $n \times n$ matrix and x is an n-vector. The other gets a similar pair (N, y). They should find a coordinate in which their matrices differ, or answer 'bad' for "illegal" inputs, namely either $x = y$ or for some index $i \in [n]$ $x_i \neq y_i$ but the corresponding rows M_i and N_i are equal. Again, a natural extension define the universal relation U_k for k-fold composition. [KRW] conjectured that the communication complexity $c(U_k)$ is $\Omega(kn)$. They noted that it is trivially implied by the stronger conjecture for functions above, and hence can be used to test it.

Very recently Edmonds et al. [EIRS] proved this conjecture in the strong form $c(U_k) \geq kn - O(k^2\sqrt{nlogn})$. Their proof uses beautiful information theoretic arguments that capture the progress made by a protocol on each iteration of the composition. Our main result is a new proof of a somewhat stronger (at least for small k) statement $c(U_k) \geq kn - O(k^3 logk)$. Our proof of the lower bound for U_2 is based on a careful analysis of the structure of inputs arriving to 'bad' leaves showing that a certain Neciporuk - like measure [Ne] on them has to be small. Then we use induction due to a relationship between 'bad' answers for U_k and 'good' answers for U_{k-1}. We stress that though our proof is completely different than that of [EIRS], we have obtained it after learning about their result and its proof.

Another result in [EIRS] is a lower bound on the composition of the address function (multiplexor) with itself that is obtained using the same information theoretic methods. We observe that the standard Neciporuk argument for formula size gives the same bound.

Can we infer something from the inherent sequentiality of composition of the universal relation about the same for functions? Both proofs, while enriching the arsenal of communication lower bound techniques, heavily rely on the fact that the input domain for both players considerably overlaps. This feature is not present in the problem for functions, or rather, it is present in a subtle form (when one looks at projections). It is not clear thus that this result or techniques directly apply to the real problem, but we believe that they motivate a serious confrontation with it.

In section 2 we define protocols and prove some technical lemmas regarding them. Section

3 contains the formal definitions and results. In section 4 we prove the lower bound on U_2. In section 5 we give the intuition behind the general lower bound on U_k, and in section 6 give a proof of this lower bound.

2 Communication Protocols and Subadditive Measures

Let $R \subseteq X \times Y \times Z$ be a relation. For subsets $A \subseteq X$ and $B \subseteq Y$ we say that $A \times B$ is a rectangle. We call it a monochromatic rectangle (of color $z \in Z$) if for all $(x, y) \in A \times B$ we have $(x, y, z) \in R$. A protocol for R is a binary tree in which every node v is labeled with a rectangle $R_v = A_v \times B_v$ such that
(1) the root is labeled $X \times Y$
(2) every leaf is labeled with a monochromatic rectangle
(3) the labels of a node v and its two children u and w satisfy either
(i) $A_v = A_u = A_w$ and $B_v = B_u \cup B_w$, or
(ii) $A_v = A_u \cup A_w$ and $B_v = B_u = B_w$.

Let the communication complexity of R, $c(R)$ be the depth of the shallowest protocol for R. Similarly, the (formula) size of R, $L(R)$ is the smallest number of leaves in any protocol for R. Clearly $c(R) \geq \log L(R)$.

A labeled tree for which only condition (3) is satisfied is called a protocol. It is convenient to actually consider more general protocols in which every node v is labeled with a vector of rectangles $\bar{R}_v = \{R_v^j \mid j \in J\}$ for some index set J. Here one of conditions (3i) or (3ii) applies simultaneously to all rectangles per node. Rectangles of the form $A \times B$ with A and B disjoint subsets of strings of fixed length m are called m-rectangles. A protocol in which all rectangles involved are m-rectangles is called a Boolean protocol. We proceed to define a measure on such vectors that will be subadditive on Boolean protocols.

Let $\bar{T} = \{A^j \times B^j \mid j \in J\}$ be a vector of m-rectangles. A family G of Boolean functions on m bits is said to cover $\bar{T}$ if for every $j \in J$ there is a function $g \in G$ such that $A^j \subseteq g^{-1}(0)$ and $B^j \subseteq g^{-1}(1)$. Let $\Gamma(\bar{T})$ denote the smallest number of nonconstant functions in any cover of $\bar{T}$, and $\gamma(\bar{T}) = \log(1 + \Gamma(\bar{T}))$. Now given a Boolean protocol in which every node v is labeled by an m-vector $\bar{R}_v$ we set $\gamma(v) = \gamma(\bar{R}_v)$.

Lemma 1 *Let v be a node in a Boolean protocol and u, w his two children. Then $\gamma(v) \leq \gamma(u) + \gamma(w)$.*

Proof: (Similar to [Zw]). Assume that condition (3i) is satisfied in v. Assume that the functions $0, 1$ and the nonconstant functions G_u cover $\bar{R}_u$ and similarly that $0, 1$ and G_w cover $\bar{R}_w$. Then $\bar{R}_v$ is covered by the set $0, 1, G_u, G_w$ and all functions of the form $g_u \wedge g_w$ for every pair $g_u \in G_u$ and $g_w \in G_w$. When condition (3ii) is satisfied, just replace $\wedge$ with $\vee$. ■

We now derive two corollaries to the above lemma. The first gives a slightly more general statement of Neciporuk lower bound on formula size (over the Boolean basis). The second gives the basic measure that we will use for the composition lower bound. Both arise from replacing rectangles labeling certain protocols by vectors of rectangles.

First construction: For a set of m-bit strings W, a set $S \subseteq [m]$ and $\sigma \in \{0,1\}^{m-|S|}$ let $W^\sigma \in \{0,1\}^S$ be the set of all continuations of σ in S. Now an m-rectangle $A \times B$ and a set $S \subset [m]$ define the vector of rectangles $\bar{P}(A,B;S) = \{A^\sigma \times B^\sigma | \sigma \in \{0,1\}^{m-|S|}\}$.

Corollary 1 *[Ne] For disjoint $X, Y \subseteq \{0,1\}^n$ let $R \subseteq X \times Y \times [n]$ be defined by $(x,y,i) \in R$ iff $x_i \neq y_i$. Further let $S_1, S_2, \cdots S_t$ be any collection or pairwise disjoint subsets of $[n]$. Then*

$$L(R) \geq \sum_{j=1}^{t} \gamma(\bar{P}(X,Y;S_j))$$

Proof: By the definition of $\bar{P}$ and the lemma this summation when applied to the rectangles at the nodes of a protocol is subadditive.

Now observe that this measure can take at most the value 1 at each leaf. To see this assume that the leaf is labeled with i and that $i \in S_{j_0}$. Then for $j \neq j_0$, $\gamma(\bar{P}(X,Y;S_j)) = 0$ since for each σ either X^σ or Y^σ is empty. On the other hand $\gamma(\bar{P}(X,Y;S_{j_0})) \leq 1$ since x_i (or $\bar{x}_i$) can be used to cover the rectangle. Finally:

$$L(R) = \sum_{\text{all leaves}} 1 \geq \sum_{\text{all leaves}} \text{weight at leaf} \geq$$

$$\text{weight at root} = \gamma(\bar{P}(X,Y;S_j)).$$

■

Remark: Let us compare this to the ordinary statement of the theorem by Neciporuk. In that case we are looking at a Boolean function and thus $X \bigcup Y$ is the entire set. This means that a function g cannot cover more than one set and $\gamma(\bar{P}(X,Y;S_j))$ is essentially the logarithm of the number of different functions we can get on S_j by fixing the bits in the complement of S_j.

Second construction: Assume A and B are arbitrary (not necessarily disjoint) sets of m-bit strings. Define the vector of m-rectangles $\bar{Q}(A,B)$ to be the set $\{(T \cap A) \times ((\{0,1\}^m - T) \cap B) | T \subset \{0,1\}^m\}$. In words, these are all ways (perhaps with repetitions) of partitioning $A \cup B$ into two disjoint sets, one which is a subset of A and one which is a subset of B. Then the lemma immediately gives:

Corollary 2 *The measure $\gamma(\bar{Q}(A,B))$ is subadditive on the nodes of any Boolean protocol.*

Another way of deducing the above corollary is to observe that this measure can be simply computed as follows. Remember that $\gamma(\bar{T}) = \log(1 + \Gamma(\bar{T}))$.

Lemma 2 $\Gamma(\bar{Q}(A,B))$ *take the following values:*

- $\Gamma(\bar{Q}(A,B)) = 2^{|A \cap B|}$ *if $A \not\subseteq B$ and $B \not\subseteq A$.*
- $\Gamma(\bar{Q}(A,B)) = 2^{|A \cap B|} - 1$ *if $A \subset B$ or $B \subset A$.*
- $\Gamma(\bar{Q}(A,B)) = 2^{|A \cap B|} - 2$ *if $A = B$.*

We conclude this section with a useful structural result about optimal protocols for relations with binary output. It states informally that the number of leaves of each type differ by at most a constant factor. For a protocol P denote $d(P)$ its depth and $L(P)$ its number of leaves. If P computes a relation $R \subseteq X \times Y \times Z$ with $Z = \{0, 1\}$ denote by $L_0(P)$ and $L_1(P)$ the number of leaves with answer 0 and 1 respectively.

Proposition 1 *Let P be any protocol for a relation with binary output. Then there exists another protocol P' for the same relation with*

- $d(P') \leq d(P)$
- $L(P') \leq L(P)$
- $L_0(P') \leq 3L_1(P')$ *(and* $L_1(P') \leq 3L_0(P')$*).*

Proof: Consider the protocol P, whose leaves are labeled from $\{0, 1\}$. Consider a node v and suppose it has a leaf as one child while its other child is an internal node. Consider the first descendant w of v which has two internal nodes as children or two leaves as children. It is easy that if all nodes on the path from v to w has one internal node and one leaf as children. If three consecutive nodes of these nodes have leaf-children with the same label, it is easy to see that one of these can be eliminated. In fact, two of them must have the same player acting and they can be merged. Thus if there are l leaf-children labeled 1 on this path then there at most $2l + 2$ leaf-children labeled 0.

If we now eliminate all leaves whose sibling is not a leaf, and bypass vertices with one child, the result is a full binary tree with leaves that come in sibling pairs in which exactly one is labeled 0 and the other 1. If there are k such pairs, and if m leaves labelled 1 were eliminated then we claim that at most $2k + 2m$ leaves labelled 0 were eliminated. To see this we argue as follows. Suppose that on the path corresponding to edge e in the final binary tree l_e leaves labelled 1 were eliminated. By the above argument at most $2l_e + 2$ leaves labelled 0 were eliminated. Since the number of edges is bounded by k and $\sum_e l_e = m$ the claim, and hence the proposition now follow.

■

3 Composition and Universal relations : Definitions and Results

For $j \geq 0$ define V_j to be the set of all binary j-dimensional arrays indexed by $[n]^j$. Thus V_0 is just one bit, V_1 is the set of all n bit strings, V_2 is all binary $n \times n$ matrices etc. For arrays $M_j \in V_j$, $M_{j+1} \in V_{j+1}$ and a position $\alpha \in [n]^j$ we denote by $M_j(\alpha)$ the bit value in that position, and by $M_{j+1}(\alpha)$ the row (n-bit vector) of M_{j+1} associated with this position.

Let $X_k = V_k \times V_{k-1} \times \cdots \times V_1 \times \{0\}$ and $Y_k = V_k \times V_{k-1} \times \cdots \times V_1 \times \{1\}$. X_k and Y_k are the input domains for the two players, which consists of one array of every dimension between 1 and k, and (for convenience) the bit (0-dimensional array) which is 0 for one player and 1 for the other.

We say that a pair of inputs $\bar{M} = (M_k, \cdots M_1, M_0) \in X_k$ and $\bar{N} = (N_k, \cdots N_1, N_0) \in Y_k$ is *bad* if for some $j < k$ we have $M_j(\alpha) \neq N_j(\alpha)$ but $M_{j+1}(\alpha) = N_{j+1}(\alpha)$. In this case we say that α is a witness to the badness the pair $(\bar{M}, \bar{N})$. Other pairs are called *good.* (*bad* pairs never occur in the composition of real functions). Note that since always $M_0 = 0$, $N_0 = 1$, *good* pairs always satisfy $M_k(\alpha) \neq N_k(\alpha)$ for some $\alpha \in [n]^k$, and that such α may exist for *bad* pairs too. Such indices α are called legal answers for $(\bar{M}, \bar{N})$.

Definition 1 *The universal relation for k-fold composition $U_k \subset X_k \times Y_k \times ([n]^k \cup \{bad\})$ can be defined as follows. $(\bar{M}, \bar{N}, z) \in U_k$ if either $z = \alpha$ which is a legal answer for $(\bar{M}, \bar{N})$ or $z = bad$ and this input pair is* bad.

In fact, the standard definition of U_k is slightly different, i.e. the answer is always a position, and it has to be legal only if the input pair is *good.* However, the communication complexity of the two problems differ by at most two bits, so lower bounds for the above definition suffice. We will now define a relative of the above universal relation, in which instead of a bad answer the players have to provide a witness to the badness of the input. This seems to be a harder problem to solve, and a lower bound for it will not suffice, but it will provide the intuition to our lower bound.

Definition 2 *The extended universal relation $W_k \subseteq X_k \times Y_k \times (\bigcup_{j=0}^{k}[n]^j)$ is defined as follows. $(\bar{M}, \bar{N}, \alpha) \in W_k$ with $\alpha \in [n]^j$ if either $j = k$ and α is a legal answer for the input pair, or $j < k$ and α witnesses the badness of the input pair.*

We can now state our main results. For $k = 1$ it is easy to see that whenever both players get the same string as input, the resulting (*bad*) answer must reach a distinct leaf. Therefore $L(U_2) \geq 2^n$ and hence $c(U_1) \geq n$. For $k = 2$ we have a somewhat stronger result than for general k, so we state it separately.

Theorem 1 $L(U_2) \geq (1 - o(1))2^{2n-1}$, *and so* $c(U_2) \geq 2n - 1$ *for sufficiently large* n.

Theorem 2 $L((U_k) \geq 2^{kn - O(k^3 \log k)}$, *and so* $c(U_k) \geq kn - O(k^3 \log k)$.

Remark: Our result becomes trivial when $k = \Omega(\sqrt{n/\log n})$ (which (by coincidence ?) is the same point of breakdown as [EIRS]). In particular both proofs succeed in proving an $\Omega(kn)$ lower bound only when the total number of positions, which is roughly n^k is much smaller than the number of values each row can take, which is 2^n. It will be interesting to extend the lower bound for all k, or exhibit a better protocol for some $k > n$.

Finally, we define the address function (also called the multiplexor) and its composition with itself, and derive a quadratic lower bound for its size from Corollary 1.

The address function has $2^m + m$ inputs. It has input (T, x) where T of size 2^m is interpreted as a function from $\{0,1\}^m$ to $\{0,1\}$ and an m-bit string x. The output of the function is $T(x)$. (This function also captures the simultaneous computation of all m-bit functions, but has exponential in m many variables). We will study a two level partial composition of this function. To be more precise each of the bits x_i will be the output of address functions while the bits T will remain as inputs.

Definition 3 *The two level address function (AF_2) is a function of $(m+1)2^m+m^2$ variables. These are interpreted as $m+1$ functions (called $(T_i)_{i=1}^m$ and T) from $\{0,1\}^m$ to $\{0,1\}$ and m input vectors (called $(x_i)_{i=1}^m$) each of length m. The output of the function is defined as $T(T_1(x_1), T_2(x_2), \ldots T_m(x_m))$.*

In [EIRS] it is proved that the function AF_2 requires depth $2m - O(\sqrt{m})$ (and their proof can be modified to give a bound on formula size too). We observe that one can directly use Neciporuk's theorem to obtain a somewhat better bound. If in Corollary 1 we take the disjoint sets $(S_i)_{i=1}^{2^m}$ be given of that for $i \leq 2^m$, S_i contains the i'th bit of the description of each T_j, $j \leq m$ then a straightforward calculation gives:

Theorem 3 *The formula size of AF_2 is at least 2^{2m} (and so the depth is at least $2m$).*

M. Karchmer pointed out the following. There is a natural way to extend the definition of the 2-level address function to define the k-level analog AF_k for any k, by performing the same "partial composition" only on the address variables x. Also, let $L_x(AF_k)$ denote the minimum number of leaves labeled by address variables x in any formula for AF_k (this number may be significantly smaller than the formula size). Now a lower bound of (say) $2^{\Omega(kn)}$ on $L_x(AF_k)$ for any k that grows with n will imply $NC^1 \neq P$. The proof is essentially the same as for standard composition of functions [KRW]. An easier task, which still seems difficult and very well motivated, is to strengthen the above theorem to a lower bound of (say) $2^{(1+\epsilon)m}$ on $L_x(AF_2)$.

4 Proof of the lower bound for U_2

Let P be a protocol for U_2, whose nodes are labeled by rectangles $A \times B$ with both A, B families of n^2+n-bit strings, which are pairs (M, x) with M and $n \times n$ matrix and x an n-bit vector. (Note that now we ignore the last bit.) For a fixed value of M denote by A_M (resp. B_M) the set of all x such that $(M, x) \in A$ (resp. $\in B$). Recall our second construction $\bar{Q}(A_M, B_M)$, and for simplicity denote $\gamma(A_M, B_M) = \gamma(\bar{Q}(A_M, B_M))$. We can now introduce the weight function on rectangles $A \times B$ as above.

$$\gamma(A,B) = \sum_{M \in \{0,1\}^{n^2}} \gamma(A_M, B_M)$$

Note that by Lemma 1 this measure is subadditive. Let us first give the key properties of $\gamma(A_M, B_M)$ we need for the proof. They all follow immediately from Lemma 2.

1. If $|A_M \bigcap B_M|$ is big then $\gamma(A_M, B_M) \approx |A_M \bigcap B_M|$.

2. If $A_M = B_m = \{x\}$ then $\gamma(A_M, B_M) = 0$

3. If $A_M = \emptyset$ or $B_M = \emptyset$ then $\gamma(A_M, B_M) = 0$.

4. If $A_M \bigcap B_M = \emptyset$ but both sets are nonempty then $\gamma(A_M, B_M) = 1$.

At the root, $A_M = B_M = \{0,1\}^n$, for every M. Therefore, by Lemma 2, the weight at the root is $2^{n^2} \log\left(2^{2^n} - 1\right) \approx 2^{n^2+n}$. Note that this approximation is off by at most 1 from the weight (for $n \geq 4$) so for notational convenience we can assume the root weight is exactly 2^{n^2+n}. It remains to upper bound the weight of the leaves. Whenever the leaf gives a legal answer, i.e. a position in the matrix, then for every M either A_M or B_M is empty, and thus, by property 3 above, the weight of such leaves is 0. It remains to bound the weight for *bad* leaves. The canonical types of bad leaves are given by the two possible bad answers of a protocol for the extended problem W_2:

- $j = 0$, which means that there is one fixed string x such that for every matrix M, $A_M = \{x\}$ or $A_M = \emptyset$ and $B_M = \{x\}$ or $B_M = \emptyset$. In this case, by properties 2 and 3, the leaf weight is 0.

- $j = 1$, which means that for some position $i \in [n]$ all matrices have their ith row fixed to some string r, but their extensions x in position i has opposite values in A and B. In this case there are at most 2^{n^2-n} matrices M, and each pair A_M, B_M can be covered by one nonconstant function (x_i or $\bar{x}_i$). Hence the weight of such a leaf is at most 2^{n^2-n}.

Indeed this argument gives a tight lower bound for W_2, namely

Theorem 4 $L(W_2) \geq 2^{2n}$

Proof: By the above analysis the weight at the root is 2^{n^2+n} while the maximal value of the weight of a leaf in W_2 is 2^{n^2-n}. This implies that we have 2^{2n} leaves. ■

The problem is that *bad* leaves in a protocol for U_2 may be more complicated, as the reason for badness is not necessarily pinned to a fixed position. An example of a heavier leaf than the above bound is the following. Let $\bar{0}$ and $\bar{1}$ denote respectively the all 0 and all 1 n-bit strings. Let A and B be subsets of all (M, x) with the first row of M and x chosen from $\{\bar{0}, \bar{1}\}$ such that in A they are equal and in B they are different. This is a bad rectangle, of weight 2^{n^2-n+1}. We will show that no *bad* rectangle can have weight bigger than the above example times $(1 + o(1))$ (in fact the factor is $(1 + O(n^4 2^{-n}))$). Even though the weight depends only on input pairs with the same matrix component, the proof that it is small will heavily rely in the interaction of input pairs with different matrix component.

Let $A \times B$ be a *bad* rectangle. We consider below only matrices M for which both A_M and B_M are nonempty, and they do not equal the same singleton set (as the other M's contribute 0 to the weight). We also ignore sets of matrices of size $o(2^{n^2-2n})$ (which we call small) as they contribute only $o(2^{n^2-n})$ (which we call negligible) to the weight. We shall need the following lemma.

Lemma 3 *The contribution of any set of matrices with two fixed rows to the weight is* $O(n 2^{n^2-2n})$

Proof: Assume without loss of generality that the first two rows are fixed. Note that there are at most 2^{n^2-2n} such matrices. Remove from the set all matrices M for which

$|A_M \bigcap B_M| \leq 5$ (as each contributes only $O(1)$ to the weight). Classify the remaining matrices into four classes $C_{00}, C_{01}, C_{10}, C_{11}$ by putting e.g. M in C_{00} if there are at least two strings with 00 in the first two coordinates in $A_M \cap B_M$ (if an M can be put in many classes choose one in some arbitrary way). As $A \times B$ is *bad*, every two members of one class have to agree on a different row than the first two. By taking a representative of each class we see that its size can be at most $n2^{n^2-3n}$. Since each matrix contributes weight at most 2^n we are done. ∎

Let us return to the proof of Theorem 1. Fix a matrix M, and classify all other (interesting) matrices into three types (0), (1) and (2), according to whether they have no row, exactly one row, or at least two rows, respectively, in common with M. By the above lemma the contribution of type (2) matrices is negligible (as there are only n^2 ways to pick the pair of fixed rows). We now bound the weight of the remaining matrices in the two cases corresponding to whether there is a matrix of type (0) or not. In the second case we may assume that each pair of interesting matrices have a common row, since otherwise we could have ended up in the first case by changing the choice of M.

Case 1: There is at least one matrix of type (0). Inspection shows this can happen only if for two different strings a, b $A_M = \{a\}$, $B_M = \{b\}$, and for every matrix N of type (0) $A_N = \{b\}$, $B_N = \{a\}$. If $a = b$ these matrices contributes 0 to the weight so assume $a \neq b$ which implies that each matrix of this type contributes 1. Now every pair of N's of type (0) must have a row in common. Suppose there are two matrices N_1 and N_2 which share exactly one row r (if there are no such matrices then the total number of matrices is $O(n^2 2^{n^2-2n})$). Now the remaining matrices go into two types, those that contain r and those that do not contain r. There are at most 2^{n^2-n} matrices of the first kind while there are at most $O(n^2 2^{n^2-2n})$ (since they have a different row in common with N_1 and N_2) of the second kind. We can conclude that the total contribution from type (0) matrices is at most $(1 + o(1))2^{n^2-n}$.

Now pick a fixed matrix of type (0) N. Every matrix of type (1) either has no row in common with N or has at least one. The contribution of the first kind is bounded by an argument identical to the above. The matrices of the second kind have a row in common with each of M and N (which are necessarily distinct), and hence by the lemma their contribution is negligible. We conclude that in this case the total weight is $(1 + o(1))2^{n^2-n+1}$.

Case 2: There are no matrices of type (0) and any two matrices have a row in common.

Fix one matrix N which have exactly one row in common with M. Note that if there is no such matrix we are already done by Lemma 3. Suppose without loss of generality that N and M have the first row in common and its value is r. All other matrices either has r as first row or they have separate rows in common with N and M. For the second type, again Lemma 3 applies and they have a negligible contribution. Thus all that is needed is to analyze the possible contribution of the set of matrices with first row r.

Let us forget the special choice of M and look at $|A_M \bigcap B_M|$ for a generic M. If this number is at most 1 for all our M then the total weight is at most $2^{n^2-n} \log 3$, and let us next suppose it is at least two for some M. There are three cases:

1. $A_M \bigcap B_M$ contains two element with first coordinate 1.
2. $A_M \bigcap B_M$ contains two element with first coordinate 0.

3. $A_M \bigcap B_M$ contains exactly one element with first coordinate 0 and exactly one with first coordinate 1.

Suppose M_1 satisfies the first condition, then for all other M, $A_M \bigcap B_M$ contains no element with first coordinate 1 or M has a second row in common with M_1. By Lemma 3 we can disregard the second set and disregarding those we can conclude that there is no other matrix than M_1 satisfying the first condition, and clearly we can disregard the contribution of a single matrix. Matrices satisfying the second condition can be treated similarly. Now let us consider M_1 and M_2 that satisfies the the third condition. If they have no other row in common then it is easy to see that we must have $A_{M_1} = B_{M_1} = A_{M_2} = B_{M_2} = \{v_1, v_2\}$ for two vectors v_1 and v_2. Thus again disregarding a set which contributes a negligible amount, each such matrix contributes at most weight $\log 3$. In particular, we have established that the weight in case 2 is at most $(\log 3 + o(1))2^{n^2-n}$.

Thus we have proved that the weight of any *bad* leaf is at most $(1 + o(1))2^{n^2+1-n}$ and since the weight at the root is 2^{n^2+n} we have established Theorem 1.

Remark: It is interesting to compare the present proof of the lower bound for U_2 with the lower bound of [EIRS]. Our measure seems to make a soft transition between the two stages of their proof.

5 Intuition for the lower bound on U_k

The basic idea is to use induction and to prove that it is almost true that the number of *bad* leaves for U_k is at least 2^n times the number of *good* leaves for U_{k-1}. This is argued as follows.

Given a protocol for extended universal relation W_k. Consider a leaf which answers that $\alpha \in [n]^{k-1}$ is a witness to the badness of the input. Label this leaf (r, α) where $r \in \{0,1\}^n$ is the value of the common row in position α. Each possible value of α may appear at many leaves and let r_α be the string that occurs least often in this position. We construct a matrix M_k^* by setting $M_k^*(\alpha) = r_\alpha$ for each α.

Consider the following protocol for U_{k-1}. Both players append M_k^* to their inputs and play the protocol for W_k. Clearly they end up at a *bad* leaf labeled $\alpha \in [n]^j$ for $j < k$ (since there are no possible legal answers). If $j = k-1$, we have obtained a *good* answer to the U_{k-1} problem otherwise we answer *bad*. By the choice of M_k^* it is easy to see that the number of *good* leaves for this protocol is at most 2^{-n} times the number of *bad* leaves of the W_k protocol. In view of Proposition 1 (with output 0 corresponding to *good* and output 1 corresponding to *bad*), this completes a proof that the number of *bad* leaves in a W_k protocol is at least $2^n/3$ of the number of *bad* leaves in a U_{k-1} protocol.

If we could replace W_k with U_k above we would be done by induction on k. Our problem is of course that the U_k protocol does not provide witnesses for badness, which allowed the construction of the matrix M_k^*. In the full proof we analyze *bad* leaves and show that we can always extract a "small" set of positions who are badness witnesses for "most" pairs arriving at that leaf. As we cannot get witnesses for all pairs, this will create the additional complication of having to deal with protocols that make errors. We prove the lower bound even for such protocols, by showing that choosing the matrix M_k^* at random will decrease the number of leaves by $O(2^{-n})$ as does the optimal choice above, and will not significantly deteriorate the quality of the new protocol for U_{k-1} in terms of errors.

6 Proof of the lower bound on U_k

A position $\alpha \in [n]^j$ will be any of the $m = n^{k-1} + n^{k-2} + \ldots + 1$ positions. It will denote both a row in M_{j+1} and the corresponding bit in M_j as before. We will often use the word density, which will be the fraction the set in question is of a suitable universe. We will denote density by μ.

We need to study bad leaves and let $A \times B$ be the set of inputs that arrive at a particular bad leaf, i.e. player 1 can have any input from A and player 2 any input from B. For any position α and row $r \in \{0,1\}^n$ let $Pr_B(\alpha, r)$ be the probability that a random element from B has r in position α. Now for any $\bar{M} \in A$ we have that

$$\sum_{\alpha} Pr_B(\alpha, \bar{M}(\alpha)) \geq 1, \quad (*)$$

where the sum ranges over all m possible positions. The inequality follows since in a *bad* leaf *every* element of A and *every* element of B have some row in common. Let l be a parameter. Say that a pair (α, r) is *large* if $Pr_B(\alpha, r) \geq \frac{1}{4l}$. The purpose of the next two lemmas is to show that unless A or B are extremely small (density less than 2^{-ln}) most inputs in this leaf will have a large pair as a witness for their badness.

Lemma 4 *If $\mu(B) \geq 2^{-ln}$, $l \leq n^2$ and n is sufficiently large then there are less than $16 \cdot l^2$ large pairs.*

Proof: Suppose there are at least $16 \cdot l^2$ large pairs. Let $((\alpha_i, r_i))_{i=1}^{16 \cdot l^2}$ be some of these. A random element from B will on the average contain $4l$ of these pairs. With probability at least $\frac{1}{8l}$ it will have at least $2l$ of these large pairs . The density of inputs that contain at least $2l$ out of any fixed set of $16 \cdot l^2$ rows is at most

$$\binom{16 \cdot l^2}{2l} 2^{-2ln} \leq 2^{2l(4+2\log l - n)}.$$

For $l \leq n^2$ and sufficiently large n the density of this set is much smaller than $\frac{1}{8l}2^{-ln}$ and thus the lemma follows ■

Next we have:

Lemma 5 *For $l \leq n^2$, $k \leq \sqrt{n}$ and sufficiently large n, the subset of A whose elements do not contain a large pair has density $\leq 2^{-ln}$.*

Proof: Say that a pair is *frequent* if $Pr_B(\alpha, r) \geq \frac{1}{4m}$. Since there are m positions we know that there are at most $4m^2$ frequent pairs. Furthermore, each element of A which does not contain a large pair must contain at least $2l$ frequent pairs (using inequality (*)). The fraction of inputs in A that contain at least $2l$ frequent pairs is at most

$$\binom{4m^2}{2l} 2^{-2ln} \leq 2^{2l(2+2\log m - n)}.$$

Using $\log m \leq k \log n$ and $k \leq \sqrt{n}$ the lemma follows. ■

Let us now see how to use these lemmas to prove the theorem as outlined in the previous section. The idea is to replace a *bad* leaf by large pair witnesses that exist by the lemmas. These large pair witnesses does not cover all possibilities and hence the procedure will result in protocols with a few errors.

Definition 4 *A protocol for U_k is ϵ-error if it is correct except that for a set of* good *inputs of density at most ϵ the answer* bad *is given.*

A stronger version of our main result (Theorem 2) is:

Theorem 5 *Suppose $k \leq \sqrt{n}$, let $r_k = 2(k+1)^2$, $l_k = k + 2k^2$, $c_k = 16^{k-1}\prod_{i=2}^{k}(66 \cdot l_i^2)^{l_i}$. Then for sufficiently large n a protocol for U_k which is 2^{-nr_k}-error will have at least $c_k^{-1}2^{kn}$ leaves.*

This gives us

Corollary 3 *For sufficiently large n, a protocol for U_k requires $2^{kn-O(k^3\log k)}$ leaves.*

We prove the theorem by induction over k. For $k = 1$ we allow the protocol to make errors for at most a fraction 2^{-8n} of the inputs. Since there are only 2^{2n} possible inputs this implies that there are no errors and hence any standard proof will apply. For the induction step we want to use Lemma 4 and Lemma 5 with $l = l_k$. We assume for contradiction that there is an 2^{-nr_k}-error protocol for U_k with fewer than $c_k^{-1}2^{kn}$ leaves, and obtain a better protocol for U_{k-1} than the induction hypothesis allows.

There is some problem with using these lemmas, as *bad* leaves potentially have good inputs reaching them. However, we can use the lemmas nevertheless! To start with Lemma 4 remains true since it had nothing to do with the badness of the leaf. Secondly Lemma 5 remains true as long as we disregard inputs $\bar{M} \in A$ who violate

$$\sum_{\alpha} Pr_B(\alpha, M(\alpha)) \geq 3/4.$$

(we will handle these inputs separately).

We are ready to describe how to change the given protocol. First the protocol for U_k is changed into a larger protocol for U_k in which the leaves are one of the following types (to be defined): *small*, *incorrect*, *witnessed*, or *good*. For every *bad* leaf $A \times B$ in the protocol for U_k, do the following. First, player 1 sends "incorrect" for every input $\bar{M} \in A$ for which $\sum_\alpha Pr_B(\alpha, \bar{M}(\alpha)) < 3/4$ (calling the resulting leaf *incorrect*) and "correct" otherwise. Each correct leaf is further split as follows.

If either $\mu(A)$ or $\mu(B)$ is smaller than 2^{-ln} call it *small* and leave it as is. Otherwise, by Lemma 4, there are at most $16l^2$ large pairs. If $\bar{M} \in A$ contains one of these large pairs, then player 1 sends the pair's name. If there is no such pair player 1 lets this be known. By Lemma 5, the density of the inputs in A, for which the latter is the case, is smaller than 2^{-ln}. Therefore, in this case the protocol will terminate in a *small* leaf. In the former case player 2 responds whether this pair is a witness for badness (in which case the conversation ends in a leaf we call *witnessed*) or not, in which case we consider this a new *bad* leaf and repeat the whole process above again.

Observe that we can repeat this at most l times, since when giving a row, player 1 reduces its density by 2^{-n}, so it will become *small.* As each repetition replaces a leaf with at most $2 + 2 \cdot 2 \cdot 16 \cdot l^2 \leq 66 \cdot l^2$ leaves we have the following upper bound on the total blow-up:

Lemma 6 *Each original bad leaf gives birth to at most* $(66 \cdot l^2)^l$ *new leaves.*

Using Lemma 6 and the values of c_{k-1}, c_k we see that the resulting protocol has at most $(16c_{k-1})^{-1}2^{kn}$ leaves. Each leaf has a type which can be one of following.

1. A *small* leaf. A leaf with either $\mu(A)$ or $\mu(B)$ less than 2^{-ln}.

2. An *incorrect* leaf as described above.

3. A *witnessed* leaf (with a pair witnessing its badness).

4. A *good* leaf (these are the original *good* leaves of the given protocol, and were not touched by the process above).

Now we show how to use this new protocol for U_k to construct a protocol for U_{k-1} that will violate the inductive assumption. Let (M_k, u) be a generic input to U_k, where $M_k \in V_k$ and u stands for the sequence of matrices of dimension less than k. The intuition in the previous section is made precise in the following lemma.

Lemma 7 *There exists a matrix* $M_k^* \in V_k$ *satisfying all three properties below with respect to the new protocol.*

1. *The fraction of inputs of form* $(M_k^*, v), (M_k^*, w)$ *that arrive at a small leaf is at most* $\frac{1}{2}2^{-r_{k-1}n}$.

2. *The fraction of inputs of type* $(M_k^*, v), (M_k^*, w)$ *that arrive at an incorrect leaf is at most* $\frac{1}{2}2^{-r_{k-1}n}$.

3. *The number of witnessed leaves with witness in* M_k^* *is at most* $(4c_{k-1})^{-1}2^{(k-1)n}$.

Before proving the lemma we observe that it finishes the induction since we can construct the following protocol for U_{k-1}. Both players start by appending M_k^* to their input. They now play a according to the new protocol for U_k. Since they have the same k-dimensional matrix, they never reach a *good* leaf. If they end up at a *small* or *incorrect* node they answer *bad.* If they end up at a *witnessed* leaf with witness (α, r), they do the following. If $\alpha = k - 1$ they supply this position (where they must differ) as a *good* leaf. If $\alpha < k_1$ they answer *bad,* (and in fact the witness still witnesses this badness).

The fraction of errors in this protocol is at most $2^{-r_{k-1}n}$ by the first two properties of M_k^*. The number of *good* leaves of this protocol is bounded by $(4c_{k-1})^{-1}2^{(k-1)n}$ by the third property. Since we can bound the total number leaves in term of the number of *good* leaves using Proposition 1 we are done.

Proof of Lemma 7

We prove that there is such an M_k^* by proving that the probability that a random M fails any of the three properties is at most 1/4.

Proof for property 1: Suppose that there are S_A leaves with A small and S_B leaves with B small (and not A). The overall fraction of inputs to player 1 such there is some possibility of ending up at a *small* leaf is $S_A 2^{-(l+1)n}$. Thus the probability for a random M that at least a fraction $8S_A 2^{-(l+1)n}$ of inputs of the form (M, v) is small is at most 1/8. A similar argument for player 2 and using that $S_A + S_B \leq 2^{kn}$ (since also this new protocol for U_k has $\leq 2^{kn}$ leaves) and $l_k - k = r_{k-1}$ finishes this case.

Proof for property 2: Say that a matrix M is overrepresented at a leaf $A \times B$ if the fraction of A of the form (M, v) is more than $2^{2kn}\mu(A)2^{-|M|}$ or the corresponding statement is true for B. Clearly only a fraction 2^{1-2kn} of all matrices can be overrepresented at a given leaf. This implies that only a fraction 2^{1-kn} of all matrices is overrepresented at some leaf. Now look at a matrix M which is not overrepresented anywhere. The fraction of inputs of the type $(M, v), (M, w)$ which arrive at any incorrect leaf is at most 2^{4kn} times the fraction of inputs overall that arrive at an incorrect leaf. However since at least 1/4 of all inputs at an incorrect leaf is a good pair with a bad answer this overall fraction is at most $2^{2-r_k n}$. Using $r_k - 2 - 4k = r_{k-1}$ finishes this case.

Proof for property 3: The total number of *witnessed* leaves is bounded by $(16c_{k-1})^{-1}2^{kn}$ and each witness is in M with probability at most 2^{-n} as every row of M is random. The property now follows.

Acknowledgments: We thank Russel Impagliazzo and Mauricio Karchmer for illuminating discussions of this problem. We also thank an anonymous referee for a very careful reading of the manuscript and pointing out some errors in the reasoning. The first author wishes to thank Andy Yao and DIMACS for the possibility to visit Princeton.

References

[KW] M. Karchmer, A. Wigderson, "Monotone Circuits for Connectivity Require Super-Logarithmic Depth", *Proceedings of the* 20^{th} *STOC*, pp. 539-550 (1988).

[KRW] M. Karchmer, R. Raz, A. Wigderson, "On Proving Super-Logarithmic Depth Lower Bounds via the Direct Sum in Communication Complexity", *Structures in Complexity Theory '91*, pp. 299-304 (1991).

[An] A. E. Andreev, "On a Method for obtaining more than Quadratic Effective Lower Bounds on the Complexity of π-Schemes", *Vestnik Moskovskogo Universiteta, Matematika*, 42:1 pp. 70-73 (in Russian). English translation in *Moscow University Mathematics Bulletin* 42:1, pp. 63-66 (1987).

[BS] R. B. Boppana, M. Sipser, "The Complexity of Finite Functions", in *Handbook of Theoretical Computer Science*, van Leeuwen Ed., MIT Press, pp. 757-804 (1990).

[EIRS] J. Edmonds, R. Impagliazzo, S. Rudich, J. Sgall, "Communication complexity towards lower bounds on circuit depth", *FOCS '91*

[Ne] E. I. Neciporuk, “A Boolean Function”, *Doklady Akademii Nauk SSSR* 169:4, pp. 765-766 (in Russian). English translation in *Soviet Mathematics Doklady* 7:4, pp. 999-1000 (1966).

[Zw] U. Zwick, “Optimizing Neciporuk Theorem”, PhD Thesis, Department of Computer Science, Tel Aviv University (1987).