

# The Communication Complexity of Enumeration, Elimination, and Selection

Andris Ambainis

*Dept. of C.S., Univ. of CA at Berkeley, Berkeley, CA 94720, U.S.A Supported in part by  
Berkeley Fellowship for Graduate Studies and in part NSF grant CCR-98-00024.  
ambainis@cs.berkeley.edu*

and

Harry Buhrman

*CWI, P.O. Box 94709, Amsterdam, The Netherlands. Supported in part by the EU fifth  
framework program projects QAIP IST-1999-11234. buhrman@cwi.nl*

and

William Gasarch

*Dept. of C.S. and Inst. for Adv. Comp. Stud., Univ. of MD., College Park, MD 20742,  
U.S.A. Supported in part by NSF grant CCR-9732692. gasarch@cs.umd.edu*

and

Bala Kalyanasundaram

*Dept. of C.S. Georgetown Univ., Washington, DC 20057, U.S.A. Supported in part by NSF  
Grant CCR-9734927. kalyan@cs.georgetown.edu*

and

Leen Torenvliet

*Dept. of C.S., Univ. of Amsterdam, 24 Plantage Muidergracht, Amsterdam, The  
Netherlands, leen@wins.uva.nl.*

---

Let  $k, n \in \mathbb{N}$  and  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Assume Alice has  $x_1, \dots, x_k \in \{0, 1\}^n$ , Bob has  $y_1, \dots, y_k \in \{0, 1\}^n$ , and they want to compute  $f^k(x_1 x_2 \cdots x_k, y_1 y_2 \cdots y_k) = (f(x_1, y_1), \dots, f(x_k, y_k))$  (henceforth  $f(x_1, y_1) \cdots f(x_k, y_k)$ ) communicating as few bits as possible. The *Direct Sum Conjecture* (henceforth DSC) of Karchmer, Raz, and Wigderson, states that the obvious way to compute it (computing  $f(x_1, y_1)$ , then  $f(x_2, y_2)$ , etc.) is, roughly speaking, the best. This conjecture arose in the study of circuits since a variant of it implies  $\text{NC}^1 \neq \text{NC}^2$ . We consider two related problems.

*Enumeration:* Alice and Bob output  $e \leq 2^k - 1$  elements of  $\{0, 1\}^k$ , one of which is  $f(x_1, y_1) \cdots f(x_k, y_k)$ .

*Elimination:* Alice and Bob output  $\vec{b}$  such that  $\vec{b} \neq f(x_1, y_1) \cdots f(x_k, y_k)$ .

*Selection:* ( $k = 2$ ) Alice and Bob output  $i \in \{1, 2\}$  such that if  $f(x_1, y_1) = 1 \vee f(x_2, y_2) = 1$  then  $f(x_i, y_i) = 1$ .

a) We devise the *Enumeration Conjecture* (henceforth ENC) and the *Elimination Conjecture* (henceforth ELC) which roughly state that the obvious ways to compute Enumeration and Elimination are the best. We use these conjectures to formulate an attack on DSC.

b) For several natural functions  $f$ , any deterministic protocol for the elimination problem for  $f^k$  requires  $\Omega(n)$  bits. This establishes a weak form of ELC for these functions.

c) For several graph properties  $f$  we show that any deterministic protocol for the elimination problem for  $f^k$  requires  $\Omega(|V|)$  bits. To accomplish this we establish some very general theorems about the communication complexity of graph properties which are of independent interest.

d) For several natural functions  $f$ , any randomized protocol for the elimination problem for  $f^k$  requires  $\Omega(\frac{n}{(\log \log(n))(\log(n))})$  bits. This establishes a weak randomized version of ELC for these functions.

e) Under a reasonable (but unproven) assumption, the elimination problem for  $f^2$  requires  $\Omega(D(f))$  bits, where  $D(f)$  is the deterministic complexity of  $f$ . This links a weak version of ELC to other assumptions.

---

## 0. INTRODUCTION

Let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Assume Alice has  $x \in \{0, 1\}^n$ , Bob has  $y \in \{0, 1\}^n$ , and both have unlimited computational power. They want to compute  $f(x, y)$  transmitting as few bits as possible. Both need the correct answer at the end of the protocol. Let  $D(f)$  be the minimum number of bits they need to transmit to compute  $f$ .  $D(f) \leq n + 1$  since Alice can transmit  $x$  to Bob, Bob can compute  $f(x, y)$  and transmit it to Alice. Communication complexity investigates  $D(f)$  and variants thereof [34, 37, 54].

Let  $k \in \mathbb{N}$  and let  $f^k(x_1 \cdots x_k, y_1 \cdots y_k) = f(x_1, y_1) \cdots f(x_k, y_k)$  (where  $|x_i| = |y_i| = n$ ). Now Alice has  $x_1, \dots, x_k$ , Bob has  $y_1, \dots, y_k$ , and they want to compute  $f^k(x_1 \cdots x_k, y_1 \cdots y_k)$ . Clearly  $D(f^k) \leq kD(f)$ . Does  $D(f^k) = kD(f)$ ? There is a counterexample: For  $x \in \{0, 1\}^n$  let  $|x|_1$  be the number of 1's in  $x$ . Let  $f(x, y) = 1$  iff  $|x|_1 + |y|_1 \geq n$ . Let  $n = 2^m$ . One can show  $D(f) = m + 2$ . (The  $2^{m+1} + 1$  inputs in  $\{(1^i 0^{2^m-i}, 1^{2^m-i} 0^i) \mid 0 \leq i \leq 2^m\} \cup \{(1^i 0^{2^m-i}, 1^{2^m-i-1} 0^{i+1}) \mid 0 \leq i \leq 2^m - 1\}$  all go to different leaves, hence there is some branch of length  $\lceil \log(2^{m+1} + 1) \rceil = m + 2$ .) For  $f^k$  consider that Bob need only transmit to Alice  $k$  numbers that are between 0 and  $n = 2^m$  (which takes  $\lceil \log(2^m + 1)^k \rceil = \lceil k \log(2^m + 1) \rceil$ ) and Alice then has to transmit back the answers (using  $k$  bits). Hence  $D(f^k) \leq \lceil k \log(2^m + 1) \rceil + k$ . For  $m$  large enough,  $\log(2^m + 1) \leq m + \frac{1}{k}$ , hence we get  $D(f^k) \leq km + k + 1$ . However  $kD(f) = km + 2k$ , so  $kD(f) - D(f^k) \geq k - 1$ .

Despite the counterexample there is a general notion that  $D(f^k)$  should be close to  $kD(f)$ . This notion is referred to as the *Direct Sum Conjecture* (henceforth DSC), however the literature does not seem to have a formal statement. Before making a formal statement we need to adapt some conventions.

**Convention:** A function  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  is actually a family of functions, one for each  $n$ . We think of  $n$  as growing.

We take the following formal statement which is implicit in [29] to be DSC:

**Direct Sum Conjecture (DSC):** If  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  then  $D(f^k) = k(D(f) - O(1))$ . (Formally  $(\exists N)(\exists K)(\exists c)(\forall n \geq N)(\forall k \geq K)[D(f^k) \geq k(D(f) - c)]$ .)

DSC is interesting for two reasons. (1) It is quite natural to compare solving  $k$  problems separately to solving them together. The complexity of doing  $k$  instances of a problem has been looked at in a variety of fields including decision trees [9, 40], computability [7, 22], complexity [2, 10, 11, 31], straightline programs [15, 14, 21, 52], and circuits [43]. (2) This conjecture arose in the study of circuits since a variant of it implies  $\text{NC}^1 \neq \text{NC}^2$  (see [29, 28] for connections to circuits, and see [34, Pages 42-48] for a more recent discussion). The reasons for the form  $D(f^k) = k(D(f) - O(1))$  are (a) the counterexample above still satisfies  $D(f^k) \geq k(D(f) - O(1))$ , and (b) the variant needed for  $\text{NC}^1 \neq \text{NC}^2$  allows for an additive constant. While there are no counterexamples to this conjecture there is some evidence against it [20].

What if Alice and Bob scale down their goals? We consider three such downscalings.

**Notation:** The notation  $x \in \{\{0, 1\}^n\}^k$  is used to emphasize that  $x$  is thought of as a concatenation of  $k$  strings of length  $n$ . The notation  $x = x_1 x_2 \dots x_k$  is

understood to imply that  $|x_1| = |x_2| = \dots = |x_k| = n$ . Similar conventions hold for  $\{\{0, 1\}^n\}^i$ ,  $\{\{0, 1\}^{n-1}\}^i$ , and  $\{\{0, 1\}^n\}^{k-i}$ .

**DEFINITION 0.1.** Let  $e, k, n, t \geq 1$ . Let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Let  $\mathcal{E}$  be the set of nonempty subsets of  $\{0, 1\}^k$  of size  $\leq e$ .

1. *Enumeration:* Alice and Bob output  $e \leq 2^k - 1$  candidates, one of which is the answer. Formally let  $\text{ENUM}(e, f^k) \subseteq \{\{0, 1\}^n\}^k \times \{\{0, 1\}^n\}^k \times \mathcal{E}$  be defined by  $(x, y, E) \in \text{ENUM}(e, f^k)$  iff  $f^k(x, y) \in E$ .

2. *Elimination:* Alice and Bob output a vector that is *not* the answer. Formally let  $\text{ELIM}(f^k) \subseteq \{\{0, 1\}^n\}^k \times \{\{0, 1\}^n\}^k \times \{0, 1\}^k$  be defined by  $(x, y, b) \in \text{ELIM}(f^k)$  iff  $f^k(x, y) \neq b$ . Note that this is the same as  $\text{ENUM}(2^k - 1, f^k)$ .

3. *Selection:* ( $k = 2$ ) Alice and Bob output  $i \in \{1, 2\}$  such that if  $f(x_1, y_1) = 1 \vee f(x_2, y_2) = 1$  then  $f(x_i, y_i) = 1$ . Formally let  $\text{SELECT}(f^2) \subseteq \{\{0, 1\}^n\}^2 \times \{\{0, 1\}^n\}^2 \times \{1, 2\}$  be defined by  $(x_1x_2, y_1y_2, i) \in \text{SELECT}(f^2)$  iff  $(f(x_1, y_1) = 1 \vee f(x_2, y_2) = 1) \Rightarrow f(x_i, y_i) = 1$ . Selection is equivalent to elimination where you are forced to eliminate one of  $\{01, 10\}$ .

The complexity of enumeration, elimination, and selection has been studied in the context of both polynomial time [1, 2, 10, 16, 17, 25, 30, 48, 50, 49, 51] and computability theory [8, 7, 22, 26, 32].

Let  $i \leq k$ . Clearly  $D(\text{ENUM}(2^{k-i} - 1, f^k)) \leq iD(f)$ : Alice and Bob can transmit  $iD(f)$  bits to compute  $b_1b_2 \dots b_i = f^i(x_1x_2 \dots x_i, y_1y_2 \dots y_i)$  and output the set of strings  $b_1b_2 \dots b_i \{0, 1\}^{k-i}$  as candidates. We state (for the first time) the following conjectures.

Let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  and  $i \leq k$ .

**1. Enumeration Conjecture (ENC):**

$$D(\text{ENUM}(2^{k-i} - 1, f^k)) = (i + 1)(D(f) - O(1)).$$

**2. Weak Enumeration Conjecture (WENC):**

$$D(\text{ENUM}(2^{k-i} - 1, f^k)) \geq \Omega\left(\frac{(i + 1)D(f)}{\log(D(f))}\right).$$

**3. Elimination Conjecture (ELC):**

$$D(\text{ELIM}(f^k)) = D(f) - O(1).$$

**4. Weak Elimination Conjecture (WELC):**

$$D(\text{ELIM}(f^k)) \geq \Omega\left(\frac{D(f)}{\log(D(f))}\right).$$

DSC is the special case of ENC when  $i = k - 1$ . ELC is the  $i = 0$  case of ENC. An approach to DSC would be to prove ENC, perhaps by induction on  $i$ . In this case ELC would be the base case. Although DSC is a special case of ENC, it is

sometimes easier to prove a stronger theorem (e.g., strengthening the induction hypothesis).

## 1. DEFINITIONS AND LEMMAS

In the following definition a protocol is a decision tree where, at each node, one of the players uses the knowledge of the string he has and the bits he has seen to transmit a bit string to the other player. We consider nondeterministic and randomized protocols for relations. These concept are not well studied; hence we define our terms carefully.

**DEFINITION 1.1.** Let  $S$  be a relation on  $X \times Y \times Z$  such that,  $(\forall x \in X)(\forall y \in Y)(\exists z \in Z)[S(x, y, z)]$ . We think of Alice as having  $x$  and Bob as having  $y$ .

1.  $D(S) \leq t$  if there is a  $t$ -bit deterministic protocol that will, on input  $(x, y)$ , output some  $z$  such that  $S(x, y, z)$ . Formally this means that there is a decision tree such that the following hold.

(i) The top node is labeled either ALICE or BOB. If a non-leaf node is labeled ALICE (BOB) then its children are labeled BOB (ALICE).

(ii) If  $v$  is a non-leaf BOB-node then there are  $2^n$  children of  $v$ , indexed by the input  $x$  that BOB sees. That is, for each  $x \in \{0, 1\}^n$  there is a child of  $v$  labeled  $(x, w)$  where  $w \in Z$ . The label  $(x, w)$  is interpreted as saying that if Bob has  $x$  then he sends Alice  $w$ . Note that node  $v$  describes what Bob has seen up to this point. If  $v$  is a non-leaf ALICE-node then its labelled in the exact same way, and interpreted as Alice sending Bob  $w$ .

(iii) If  $v$  is a leaf then  $v$  is labeled with an element of  $Z$ .

(iv) Let  $x \in \{0, 1\}^n$  and  $y \in \{0, 1\}^n$ . If the decision tree is executed on  $(x, y)$  in the obvious way then (1) the sums of the lengths of all the messages is  $\leq t$ , and (2) the leaf arrived at will be labeled  $z$  where  $S(x, y, z)$ .

This definition is equivalent to saying that there exist sets  $X_1, \dots, X_{2^t} \subseteq X$ , and  $Y_1, \dots, Y_{2^t} \subseteq Y$ , and  $z_1, \dots, z_{2^t} \in Z$  such that (1)  $X \times Y = \bigcup_{i=1}^{2^t} X_i \times Y_i$ , (2)  $(\forall i)(\forall x \in X_i)(\forall y \in Y_i)[S(x, y, z_i)]$ , (3) The sets  $X_i \times Y_i$  are all disjoint. The collection  $X_1 \times Y_1, \dots, X_{2^t} \times Y_{2^t}$  is called a *partition*. The equivalence follows from the fact that in any deterministic protocol every leaf corresponds to a set of the form  $A \times B$  (See [34].)

2.  $N(S) \leq t$  if there is a  $t$ -bit non-deterministic protocol such that on input  $(x, y)$  some leaf outputs a  $z$  such that  $S(x, y, z)$ . Formally this means that there is a decision tree such that the following hold.

(i) The top node is labeled either ALICE or BOB. If a non-leaf node is labeled ALICE (BOB) then its children are labeled BOB (ALICE).

(ii) If  $v$  is a non-leaf BOB-node then there are  $2^n$  sets of children of  $v$ , indexed by the input  $x$  that BOB sees. That is, for each  $x \in \{0, 1\}^n$  there is a set of children of  $v$  labeled  $(x, W)$  where  $W \subseteq Z$ . The label  $(x, W)$  is interpreted as saying that if Bob has  $x$  then he nondeterministically sends Alice some  $w \in W$ . Note that node  $v$  describes what Bob has seen up to this point. If  $v$  is a non-leaf ALICE-node then its labelled in the exact same way, and interpreted as Alice sending Bob  $w$ . We

count each nondeterministic choice as a bit of communication, hence if  $w \in W$  is chosen then we count the length of the messages sent as  $|w| + \lceil \log_2(|W|) \rceil$ .

(iii) If  $v$  is a leaf then  $v$  is labeled with an element of  $Z$ .

(iv) Let  $x \in \{0, 1\}^n$  and  $y \in \{0, 1\}^n$ . If the decision tree is executed on  $(x, y)$  in the obvious way then (1) the sums of the lengths of all the messages is  $\leq t$  (using the convention of counting lengths mentioned above), and (2) the leaf arrived at will be labeled  $z$  where  $S(x, y, z)$ .

(v) If  $v$  is a leaf then  $v$  is labeled with an element of  $Z$  or with the phrase “I DON’T KNOW!”

(vi) Let  $x \in \{0, 1\}^n$  and  $y \in \{0, 1\}^n$ . If the decision tree is executed on  $(x, y)$  in the obvious way then (1) the sum of the messages and the choice nodes encountered is  $\leq t$ , (2) all the leaves that the computation can arrive at are either labeled “I DON’T KNOW” or with a  $z$  such that  $S(x, y, z)$ , (3) at least one of the leaves the computation can arrive at is labeled  $z$  where  $S(x, y, z)$ .

This definition is equivalent to saying that there exists sets  $X_1, \dots, X_{2^t} \subseteq X$ , and  $Y_1, \dots, Y_{2^t} \subseteq Y$ , and  $z_1, \dots, z_{2^t} \in Z$  such that (1)  $X \times Y \subseteq \bigcup_{i=1}^{2^t} X_i \times Y_i$ , and (2)  $(\forall i)(\forall x \in X_i)(\forall y \in Y_i)[S(x, y, z_i)]$ . Note that, in contrast to the deterministic case, the  $X_i \times Y_i$  sets need not be disjoint. The collection  $X_1 \times Y_1, \dots, X_{2^t} \times Y_{2^t}$  is called a *covering*. The equivalence follows from the fact that in any nondeterministic protocol every leaf corresponds to a set of the form  $A \times B$  (See [34].)

The definition of a nondeterministic protocol to compute a function is *not* obtained by applying the definition for a relation. Hence we define it below.

DEFINITION 1.2. Let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ .

1.  $D(f) \leq t$  is defined by viewing  $f$  as a relation and using Definition 1.1.1.

2.  $N(f) \leq t$  if there is a  $t$ -bit non-deterministic protocol for  $f$ . Formally this is similar to the definition of  $N(S) \leq t$  except that, if  $f(x, y) = 0$ , we do not require that some leaf output 0.

3.  $\text{coN}(f) \leq t$  if  $N(\bar{f}) \leq t$ .

DEFINITION 1.3. Let  $S$  be a relation on  $X \times Y \times Z$  such that,  $(\forall x \in X)(y \in Y)(\exists z \in Z)[S(x, y, z)]$ . Let  $0 < \epsilon < \frac{1}{2}$ . We think of Alice as having  $x$  and Bob as having  $y$ .

1.  $R_\epsilon^{\text{pub}}(S) \leq t$  if there is a  $t$ -bit randomized protocol such that (1) Alice and Bob get to observe the coin flips of a referee without being charged any bits for the privilege (the ‘pub’ stands for ‘public’ in that the coins are flipped publicly not privately), (2) for any  $x \in X$  and  $y \in Y$ , the probability that the protocol outputs some  $z$  with  $S(x, y, z)$  is at least  $1 - \epsilon$ . Formally this means that there is a set of deterministic  $t$ -bit protocols  $T$  such that the following hold.

(i) All of them are labeled as in the definition of  $D(S) \leq t$ .

(ii) Fix  $x \in \{0, 1\}^n$  and  $y \in \{0, 1\}^n$ . Consider the following probabilistic experiment: pick a protocol from  $T$  at random and execute it on  $(x, y)$ . The

probability that the leaf arrived at is a  $z$  such that  $S(x, y, z)$  is greater than  $1 - \epsilon$ . (The probability of error is  $< \epsilon$ .)

2.  $R_\epsilon^{\text{pub}}N(S) \leq t$  if there is a  $t$ -bit randomized nondeterministic protocol such that (1) Alice and Bob get to observe the coin flips of a referee without being charged any bits for the privilege, and (2) for any  $x \in X$  and  $y \in Y$ , the probability that the protocol has some path that outputs some  $z$  with  $S(x, y, z)$  is at least  $1 - \epsilon$ . Formally this means that there is a set of nondeterministic  $t$ -bit protocols  $T$  such that the following hold.

(i) All of them are labeled as in the definition of  $N(S) \leq t$ .

(ii) Fix  $x \in \{0, 1\}^n$  and  $y \in \{0, 1\}^n$ . Consider the following probabilistic experiment: pick a protocol from  $T$  at random and execute it on  $(x, y)$ . The probability that there is some computation path leaf arrived at is a  $z$  such  $S(x, y, z)$  is greater than  $1 - \epsilon$ .

**Note:** The class of Boolean functions  $f$  such that  $R_\epsilon^{\text{pub}}(f) \leq t$  looks similar to randomized polynomial time; however, there is one important difference. With randomized polynomial time, an error of  $\frac{1}{4}$  can be made into  $\frac{1}{2^n}$  by repeating the procedure  $O(n)$  times and taking a majority. This works because multiplying a polynomial by  $n$  is not a large increase in the polynomial setting. However, the same trick would multiply the communication complexity by  $n$ , which is quite large in the communication complexity setting. Hence  $R_{1/4}^{\text{pub}}(f) \leq t$  does not imply  $R_{1/2^n}^{\text{pub}}(f) \leq t$ . However, using standard techniques, some amplification (at some cost) can be achieved. We state this rigorously in Lemma 6.1.

LEMMA 1.1. *Let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Let  $C \subseteq \{\{0, 1\}^n\}^k \times \{\{0, 1\}^n\}^k$ . If  $N(\text{ELIM}(f^k)) \leq t$  then there is  $A \subseteq \{\{0, 1\}^n\}^k$  and  $B \subseteq \{\{0, 1\}^n\}^k$  such that*

1.  $|C \cap (A \times B)| \geq |C|/2^t$ , and
2.  $(\exists b \in \{0, 1\}^k)(\forall x \in A)(\forall y \in B)[f^k(x, y) \neq b]$ .

*Proof.* Since  $N(\text{ELIM}(f^k)) \leq t$  we can, using Definition 1.1.2, cover  $\{\{0, 1\}^n\}^k \times \{\{0, 1\}^n\}^k$  with a set of  $2^t$  sets of the form  $A \times B$  (which may overlap). These sets also cover  $C$  (and of course may also cover points outside of  $C$ ). Since every element of  $C$  is covered, some set must cover  $|C|/2^t$  elements of  $C$ . ■

LEMMA 1.2. *Let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ , let  $g = 1 - f$ , and let  $k \in \mathbb{N}$ . Then  $D(\text{ELIM}(f^k)) = D(\text{ELIM}(g^k))$ .*

*Proof.* If  $P$  is a deterministic protocol for  $\text{ELIM}(f^k)$  then let  $P'$  be the protocol that runs  $P$  and if the output is  $b_1 b_2 \cdots b_k$ , outputs  $(1 - b_1)(1 - b_2) \cdots (1 - b_k)$ .  $P'$  is a deterministic protocol for  $g$ . ■

## 2. SUMMARY OF RESULTS

We state a subset of our results, in a weak form, for better readability. Assume throughout that Alice and Bob both get a  $k$ -tuple of strings of length  $n$ . We need the following definitions to state our results.

DEFINITION 2.1.

1. EQ :  $\{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  is defined by  $\text{EQ}(x, y) = \begin{cases} 1 & \text{if } x = y; \\ 0 & \text{if } x \neq y. \end{cases}$
  2. NE :  $\{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  is defined by  $\text{NE}(x, y) = 1 - \text{EQ}(x, y)$ .
  3. IP :  $\{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  is defined by  $\text{IP}(x_1 x_2 \cdots x_n, y_1 y_2 \cdots y_n) = \sum_{i=1}^n x_i y_i \pmod{2}$ . (“IP” stands for “Inner Product.”)
  4. We can view  $x \in \{0, 1\}^n$  as a bit vector representation of a subset of  $\{1, \dots, n\}$ .
- With this in mind  $\text{DISJ}(x, y) = \begin{cases} 1 & \text{if } x \cap y = \emptyset; \\ 0 & \text{if } x \cap y \neq \emptyset. \end{cases}$
5.  $\text{INTER}(x, y) = 1 - \text{DISJ}(x, y)$ .

**Note:** For  $f = \text{EQ}, \text{NE}, \text{IP}, \text{DISJ}$  and  $\text{INTER}$  it is known that  $D(f) = n + 1$  (see [34]).

**Note** For  $f = \text{INTER}, \text{IP}$  it is known that  $R_\epsilon^{\text{pub}}(f) = \Omega(n)$  (see [34]).

### Results about Particular Functions

In the statement of results below the implicit constant in the  $O()$  does not depend on  $k$ .

1.  $D(\text{ELIM}(\text{EQ}^k)) \geq n$ ,  $D(\text{ELIM}(\text{NE}^k)) \geq n$ , and  $D(\text{ELIM}(\text{IP}^k)) \geq n$  (Theorem 3.1, Corollary 3.1, Theorem 5.1). Hence, by Note 2, ELC holds for EQ, NE, and IP.

2.  $D(\text{ELIM}(\text{DISJ}^k)) \geq n - O(\log n)$  and  $D(\text{ELIM}(\text{INTER}^k)) \geq n - O(\log n)$  (Theorem 3.2 and Corollary 3.2). Hence, by Note 2, WELC holds for DISJ and INTER.

3. For many graph properties  $f$ ,  $D(f) \leq O(|V| \log |V|)$  and  $D(\text{ELIM}(f^k)) \geq \Omega(|V|)$ , hence  $D(f^k) \geq \Omega(\frac{D(f)}{\log D(f)})$  (Theorems 4.1). Therefore WELC holds for these graph properties. For another large class of graph properties we obtain  $D(\text{ELIM}(f^k)) \geq \Omega(|V|)$ , hence  $D(f^k) \geq \Omega(\sqrt{D(f)})$ . To prove these results we established some very general theorems about the communication complexity of graph properties. These theorems are of independent interest.

4. If  $k$  is constant and  $\epsilon < \frac{1}{2^k}$  then  $R_\epsilon^{\text{pub}}(\text{ELIM}(\text{IP}^k)) \geq \Omega(\frac{n}{(\log \log(n))(\log(n))})$  and  $R_\epsilon^{\text{pub}}(\text{ELIM}(\text{INTER}^k)) \geq \Omega(\frac{n}{(\log \log(n))(\log(n))})$  (Theorems 6.1, 6.2). Hence, by Note 2, a randomized weak version of ELC holds for IP and DISJ.

**Note:** The lower bounds on  $\text{EQ}^k$ ,  $\text{DISJ}^k$ ,  $\text{IP}^k$ , and some of the graph properties also hold for nondeterministic computation.

### Results about General Functions

1. Assume that computing  $f^m$  but allowing one mistake requires  $\frac{m}{2}D(f)$  bits for some (even)  $m$ . Then  $D(\text{ELIM}(f^2))$  requires  $\Omega(D(f))$  bits. (Corollary 7.2)

2.  $N(\text{SELECT}(f^2)) \geq N(f) - \log(n) - 1$  where  $N(f)$  is the nondeterministic communication complexity of  $f$ . (Theorem 10.2)

3. If DSC is true then  $D(\text{SELECT}(f^2)) \geq \frac{D(f)}{3} - O(1)$ . (Corollary 10.1)

4. If DSC is true then  $D(\text{ENUM}(k, f^k)) \geq D(f) - O(1)$ .

**Note** These results link ELC (and variants) to other conjectures that seem reasonable, and thus also provides evidence for its truth.



### 3. THE COMPLEXITY OF ELIM(EQ<sup>k</sup>) AND ELIM(DISJ<sup>K</sup>)

We show that  $N(\text{ELIM}(\text{EQ}^k)) \geq n$  and use this to show  $N(\text{ELIM}(\text{DISJ}^k)) \geq n - O(\log n)$ . This will establish ELC for EQ, NE and WELC for DISJ, INTER.

LEMMA 3.1. *Let  $i, n \in \mathbb{N}$ . Let  $A, B \subseteq \{0, 1\}^n$  be such that*

$$(\forall x_1 x_2 \cdots x_i \in A)(\forall y_1 y_2 \cdots y_i \in B)(\exists j)[\text{EQ}(x_j, y_j) = 1].$$

*Then  $|A||B| \leq 2^{2n(i-1)}$ .*

*Proof.* We use induction on  $n$ . The base case of  $n = 1$  is nontrivial, hence we present it. Assume  $A, B \subseteq \{0, 1\}^i$  and the hypothesis holds. Assume, by way of contradiction, that  $|A||B| > 2^{2(i-1)}$ . We can assume, without loss of generality, that  $|A| > 2^{i-1}$ . Let  $|A| = 2^{i-1} + a$  where  $a > 0$ . Note that for every  $x \in A$ ,  $\bar{x} \notin B$ . (Recall that  $\bar{z}$  means take  $z$  and replace the 0's with 1's and the 1's with 0's.) Hence  $|B| \leq 2^i - (2^{i-1} + a) = 2^{i-1} - a$ . Therefore  $|A||B| \leq (2^{i-1} + a)(2^{i-1} - a) = 2^{2(i-1)} - a^2 < 2^{2(i-1)}$ . This is a contradiction.

Assume the lemma is true for all  $n' < n$  and that  $n \geq 2$ . Let  $A, B$  satisfy the hypothesis with parameter  $n$ . Let

$$\begin{aligned} A_1 &= \{z_1 z_2 \cdots z_i \in \{0, 1\}^{n-1} : (\exists b_1 b_2 \cdots b_i \in \{0, 1\}^i)[z_1 b_1 z_2 b_2 \cdots z_i b_i \in A]\} \\ B_1 &= \{z_1 z_2 \cdots z_i \in \{0, 1\}^{n-1} : (\exists b_1 b_2 \cdots b_i \in \{0, 1\}^i)[z_1 b_1 z_2 b_2 \cdots z_i b_i \in B]\} \\ A_2 &= \{b_1 b_2 \cdots b_i \in \{0, 1\}^i : (\exists z_1 z_2 \cdots z_i \in \{0, 1\}^{n-1})[z_1 b_1 z_2 b_2 \cdots z_i b_i \in A]\} \\ B_2 &= \{b_1 b_2 \cdots b_i \in \{0, 1\}^i : (\exists z_1 z_2 \cdots z_i \in \{0, 1\}^{n-1})[z_1 b_1 z_2 b_2 \cdots z_i b_i \in B]\} \end{aligned}$$

Note that  $A_1, B_1$  satisfies the premise with parameter  $n - 1$ ,  $A_2, B_2$  satisfies the premise with parameter  $1 < n$ . Also note that  $|A| \leq |A_1||A_2|$ , and  $|B| \leq |B_1||B_2|$ . By the induction hypothesis  $|A_1||B_1| \leq 2^{2(n-1)(i-1)}$ , and  $|A_2||B_2| \leq 2^{2(i-1)}$ . Hence  $|A||B| \leq |A_1||A_2||B_1||B_2| \leq |A||B| \leq |A_1||B_1||A_2||B_2| \leq 2^{2(n-1)(i-1)} \times 2^{2(i-1)} = 2^{2n(i-1)}$ . ■

LEMMA 3.2. *Let  $k, n \in \mathbb{N}$ . If  $D \subseteq \{0, 1\}^n$  and  $|D| > 2^{(k-1)n}$  then  $(\forall b \in \{0, 1\}^k)(\exists x, y \in D)[\text{EQ}^k(x, y) = b]$ .*

*Proof.* By reordering the components of both  $b$  and the strings in  $D$  we need only consider  $b = 1^{k-i}0^i$  for  $0 \leq i \leq k$ . Fix such an  $i$ , and hence such a  $b$ .

For each  $z \in \{0, 1\}^{n-k+i}$  let  $D_z = z\{0, 1\}^i \cap D$ . Since  $|D| > 2^{(k-1)n}$  and the  $D_z$ 's partition  $D$  into at most  $2^{(k-i)n}$  parts, there exists  $z$  such that  $|D_z| > 2^{(i-1)n}$ . Let  $A = \{w \in \{0, 1\}^n : zw \in D\}$ . Note that  $|A| = |D_z| > 2^{(i-1)n}$ . By (the contrapositive of) Lemma 3.1  $(\exists x', y' \in A)(\forall j)[\text{EQ}(x'_j, y'_j) = 0]$ . Clearly  $\text{EQ}^k(zx', zy') = 1^{k-i}0^i$ . ■

THEOREM 3.1. *For all  $k, n \in \mathbb{N}$ ,  $N(\text{ELIM}(\text{EQ}^k)) \geq n$ .*

*Proof.* Assume, by way of contradiction, that  $N(\text{ELIM}(\text{EQ}^k)) = t < n$  via protocol  $P$ .

Let  $C = \{(x, x) \mid x \in \{0, 1\}^n\}$ . By Lemma 1.1 there exists  $A \subseteq \{0, 1\}^n$  and  $B \subseteq \{0, 1\}^n$  such that (1)  $|C \cap (A \times B)| \geq 2^{-t}|C| = 2^{kn-t}$  and (2) there is a real leaf  $L$  (i.e., a leaf that does not say I DON'T KNOW) such that for all  $(x, y) \in A \times B$  there is a nondeterministic computation path of  $P(x, y)$  that terminates at  $L$ . Let the label of  $L$  be  $b \in \{0, 1\}^k$ . Hence we know that  $(\forall x \in A)(\forall y \in B)[\text{EQ}^k(x, y) \neq b]$ .

Let  $D = A \cap B$ . Note that  $|D| = |C \cap (D \times D)| = |C \cap (A \times B)| \geq 2^{kn-t} > 2^{kn-n} = 2^{n(k-1)}$ . We can now apply Lemma 3.2 to obtain that  $(\exists x, y \in D)[\text{EQ}^k(x, y) = b]$ . This is a contradiction. ■

**COROLLARY 3.1.** *For all  $k, n \in \mathbb{N}$ ,  $D(\text{ELIM}(\text{NE}^k)) \geq n$ .*

*Proof.* This follows from Theorem 3.1 and Lemma 1.2. ■

**THEOREM 3.2.** *For all  $k, n \in \mathbb{N}$ ,  $N(\text{ELIM}(\text{DISJ}^k)) \geq n - O(\log n)$ .*

*Proof.* We take  $n$  to be even. The proof for  $n$  odd is similar but is notationally harder. Let  $L = \lceil \log_2 \binom{n}{n/2} \rceil \sim n - O(\log n)$ . Let  $\text{ELIM}(\text{EQ}_L^k)$  be  $\text{ELIM}(\text{EQ}^k)$  on  $k$ -tuples of  $\{0, 1\}^L$ . By Theorem 3.1  $N(\text{ELIM}(\text{EQ}_L^k)) \geq L$ . We show that  $N(\text{ELIM}(\text{EQ}_L^k)) \leq N(\text{ELIM}(\text{DISJ}^k))$ .

There are  $\binom{n}{n/2}$  subsets of  $\{1, \dots, n\}$  of size  $\frac{n}{2}$ . Each one can be represented as a string in  $\{0, 1\}^L$ . Let  $F$  map  $\{0, 1\}^L$  to  $\{0, 1\}^n$  by mapping a representation of an  $\frac{n}{2}$ -sized subset of  $\{1, \dots, n\}$  to its bit vector form. Let  $G(x)$  be the complement of  $F(x)$ . If  $\text{EQ}(x, y)$  then  $F(x)$  and  $G(y) = \overline{F(x)}$  are complements, hence  $\text{DISJ}(F(x), G(y))$ . (Recall that  $\bar{z}$  means take  $z$  and replace the 0's with 1's and the 1's with 0's.) If  $\neg \text{EQ}(x, y)$  then  $F(x)$  and  $G(y)$  are not complements of each other. Since both are sets of exactly  $\frac{n}{2}$  elements they must intersect, hence  $\neg \text{DISJ}(F(x), G(y))$ . Hence  $\text{EQ}(x, y)$  iff  $\text{DISJ}(F(x), G(y))$ . Hence  $\text{EQ}^k(x_1 \cdots x_k, y_1 \cdots y_k) \neq b$  iff  $\text{DISJ}^k(F(x_1) \cdots F(x_k), G(y_1) \cdots G(y_k)) \neq b$ .

The following nondeterministic protocol for  $\text{ELIM}(\text{EQ}_L^k)$  transmits  $N(\text{ELIM}(\text{DISJ}^k))$  bits, thus showing  $N(\text{ELIM}(\text{EQ}_L^k)) \leq N(\text{ELIM}(\text{DISJ}^k))$ . Alice gets  $x_1 x_2 \cdots x_k \in \{\{0, 1\}^L\}^k$  and Bob gets  $y_1 y_2 \cdots y_k \in \{\{0, 1\}^L\}^k$ . Alice and Bob run the optimal nondeterministic protocol for  $\text{ELIM}(\text{DISJ}^k)$  on  $(F(x_1) \cdots F(x_k), G(y_1) \cdots G(y_k))$ . ■

**COROLLARY 3.2.** *For all  $k, n \in \mathbb{N}$ ,  $D(\text{ELIM}(\text{INTER}^k)) \geq n - O(\log n)$ .*

*Proof.* This follows from Theorem 3.2 and Lemma 1.2. ■

**Note:** Babai et al. [3] defined reductions between problems in communication complexity. The proof of Theorem 3.2 actually showed  $\text{EQ} \leq_{\text{cc}} \text{DISJ}$ , which enabled us to transfer our lower bound for  $\text{ELIM}(\text{EQ}^k)$  to a lower bound for  $\text{ELIM}(\text{DISJ}^k)$ . Babai et al. [3] also defined  $\text{P}^{\text{cc}}$  and  $\text{NP}^{\text{cc}}$ , analogs of  $\text{P}$  and  $\text{NP}$ . Since we have  $D(\text{ELIM}(\text{NE}^k)) \geq n$  and  $D(\text{ELIM}(\text{EQ}^k)) \geq n$ , and  $\text{NE} \in \text{NP}^{\text{cc}}$ ,  $\text{EQ} \in \text{co-NP}^{\text{cc}}$ , we can get lower bounds for any  $\text{NP}$ -hard or  $\text{coNP}$ -hard problem in communication complexity. (We do this for graph properties in Section 4.) Since the reductions in [3] allow size  $n$  inputs to map to size  $2^{\text{polylog } n}$  the results will not be as good as those in Theorem 3.2.

#### 4. GRAPH PROPERTIES

In this section we prove some general theorems about the communication complexity of graph properties. We then apply them to obtain  $\text{WELC}$  for many graph properties.

Alice and Bob try to compute a graph property  $f$ . Each of them is given a graph on  $\{1, \dots, n\}$  and they need to compute whether the union of the graphs has the property. Formally Alice and Bob will both be given graphs on  $\{1, \dots, n\}$  and they

will try to determine if some property holds of the union of the two graphs. Hence it is possible that (say) they both find out that  $(1, 8)$  is an edge, though neither one knows that the other knows. This model of the communication complexity of graph properties is due to [23]. Other models have also been studied [35]. The notion of the communication complexity of graph properties has been generalized in [38].

**Notation:** In this section  $n$  is *not* the length of the input. Instead it is the number of vertices.

DEFINITION 4.1. If  $H$  and  $G$  are graphs then  $H$  is a *minor* of  $G$  if one can obtain  $H$  from  $G$  by removing vertices, removing edges, or contracting an edge (removing the edge and merging the two endpoints). We denote this by  $H \preceq G$ .

DEFINITION 4.2. A property  $f$  of graphs is *closed under minors* if, for all  $G$ , if  $f(G) = 1$  and  $H \preceq G$  then  $f(H) = 1$ .

The Graph Minor Theorem states that the set of graphs with the ordering  $\preceq$  forms a well quasi ordering (see [45] for a proof or [19] if all you want are definitions). The following is an easy corollary of the Graph Minor Theorem that ([45], see also [19]).

LEMMA 4.1. *Let  $f$  be a property of graphs closed under minors. There exist graphs  $H_1, \dots, H_k$  such that  $f(G) = 0$  iff  $(\exists i)[H_i \preceq G]$ . (The set of graphs  $\{H_1, \dots, H_k\}$  is called an obstruction set. Intuitively a graph  $G$  has the property unless there is a good reason, in the form of one of the  $H_i$ , that it does not.)*

EXAMPLE 4.1. Here are three examples of sets of graphs closed under minor ( $g$  and  $k$  are constants).

$$\begin{aligned} \text{PLANAR} &= \{G \mid G \text{ is Planar}\} \\ \text{GENUS}_g &= \{G \mid G \text{ has genus } g\} \\ \text{VC}_k &= \{G \mid G \text{ has a vertex cover of size } k\} \end{aligned}$$

1. For PLANAR it is known that the obstruction set is  $\{K_5, K_{3,3}\}$  (this is *not* Kuratowski's theorem [13, 33], that a graph is nonplanar iff it does not have  $K_5$  or  $K_{3,3}$  as a homeomorphic subgraph, but is easily derivable from it). For the other sets in the example the only proof that there is an obstruction set comes from the Lemma 4.1.

2. Let  $H$  be a fixed graph. It is known [46] that testing if  $H \preceq G$  can be done in  $O(|V|^3)$  steps. Using this and Lemma 4.1 one can obtain  $O(|V|^3)$  algorithms for all graph properties closed under minor. The case of  $\text{VC}_k$  is particularly interesting since it would seem that  $O(|V|^{k+1})$  is needed. The  $O(|V|^3)$  algorithm for  $\text{VC}_k$  is not very useful (big constants and nonconstructive), however, it inspired far more useful algorithms which run in time  $O(kn + g(k))$  for a variety of exponential  $g$ . See [19] for details.

DEFINITION 4.3. Let  $\text{TRIV}_{a,b}$  be the graph that is  $a$  isolated vertices unioned with  $b$  disjoint edges.

We will show that graph properties are hard by using reductions. We first need to define reductions formally.

**DEFINITION 4.4.** [3] Let  $f_n : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  and  $g_n : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  be infinite families of functions.  $f \leq_{cc} g$  means that there are functions  $T_1, T_2$  and  $L$  such that  $L : \mathbb{N} \rightarrow \mathbb{N}$ ,  $L(n) \leq 2^{\text{polylog } n}$ , and  $T_1, T_2 : \{0, 1\}^n \rightarrow \{0, 1\}^{L(n)}$  such that  $f(x, y) = 1$  iff  $g(T_1(x), T_2(y)) = 1$ . If  $L(n) = O(n)$  then we say that  $f \leq_{cc} g$  via a linear reduction.

**Note:** In Definition 4.4 we first bound  $L(n)$  by  $2^{\text{polylog } n}$  but then, for our purposes, bound it by  $O(n)$ . The reason for this is historical. When reductions were first defined in [3] they were making an analog between  $D(f) \leq 2^{\text{polylog } n}$  and P. Hence they needed reduction to not care about mere polylog factors.

We leave the following lemma to the reader.

**LEMMA 4.2.** *If  $f \leq_{cc} g$  by a linear reduction then (1)  $D(g) = \Omega(D(f))$ , (2)  $N(g) = \Omega(N(f))$ , (3)  $D(\text{ELIM}(g^k)) = \Omega(D(\text{ELIM}(f^k)))$ , and (4)  $N(\text{ELIM}(g^k)) = \Omega(N(\text{ELIM}(f^k)))$ .*

**Notation:** Let  $V(G)$  be the set of vertices in  $G$  and  $E(G)$  be the set of edges in  $G$ .

The following lemma was first shown by Mader [39]; however the interested reader may also see [12, Chapter 7, Theorem 1.16]).

**LEMMA 4.3.** *Let  $p \in \mathbb{N}$ . There exists a number  $c_p$  such that for any graph  $G = (V, E)$ , if  $|E| \geq c_p |V|$  then  $K_p \preceq G$ .*

**Note:** It is known that  $c_p \leq 8(p-2) \lceil \log(p-2) \rceil$ . In Theorem 1.14 of Chapter 7 of [12] is an easy proof of the weaker result that  $c_p \leq 2^{p-3}$ . There is some evidence that  $c_p = p-2$  or at least  $c_p = O(p)$ ; however, this is still open. See [12, Page 378].

**LEMMA 4.4.** *If  $f$  is a property of graphs that is closed under minors then, for all  $G = (V, E)$  such that  $f(G) = 1$ ,  $|E| = O(|V|)$ .*

*Proof.* By Lemma 4.1 there exist graphs  $H_1, \dots, H_k$  such that  $f(G) = 0$  iff  $(\exists i)[H_i \preceq G]$ . Let  $p = \min\{|V(H_1)|, \dots, |V(H_k)|\}$ . Let  $c_p$  be as in Lemma 4.3. For any  $G$ , if  $|E(G)| > c_p |V(G)|$ , then  $G$  has  $K_p$  as a minor; however, this implies that some  $H_i$  is a minor, hence  $f(G) = 0$ . By the contrapositive, if  $f(G) = 1$  then  $|E(G)| \leq c_p |V(G)| = O(|V(G)|)$ . ■

**THEOREM 4.1.** *Let  $f$  be a property of graphs closed under minors such that  $(\forall a, b)[f(\text{TRIV}_{a,b}) = 1]$ . Let  $g = 1 - f$ . Then the following occur.*

1.  $D(f) \leq O(n \log n)$ .
2.  $\text{DISJ} \leq_{cc} f$  by a linear reduction.
3.  $N(f) \geq \Omega(n)$ .
4.  $N(\text{ELIM}(f^k)) \geq \Omega(n)$ .
5.  $D(g) \leq O(n \log n)$ .
6.  $\text{INTER} \leq_{cc} g$  by a linear reduction.

- 7.  $D(g) \geq \Omega(n)$ .
- 8.  $D(\text{ELIM}(g^k)) \geq \Omega(n)$ .
- 9.  $D(\text{ELIM}(f^k)) \geq \Omega\left(\frac{D(f)}{\log(D(f))}\right)$ .
- 10.  $D(\text{ELIM}(g^k)) \geq \Omega\left(\frac{D(g)}{\log(D(g))}\right)$ .

*Proof.*

We prove items 1 and 2. We then easily derive items 3,4,5,6,7,8,9,10 from items 1,2 and prior results.

Proof of 1: We show  $D(f) \leq O(n \log n)$ . By Lemma 4.4 there exists a constant  $c$  such that any graph with  $f(G) = 1$  has  $\leq cn$  edges.

Here is the protocol: Alice looks at how many edges she has. If she has more than  $cn$  edges then she sends Bob a 0, and they both know  $f(G) = 0$ . If not she sends Bob a 1 and then sends him a list of the edges she has. Since each edge takes  $2 \log n$  bits to send and there are only  $cn$  edges, this takes  $2cn \log n = O(n \log n)$  bits.

Proof of 2: We show that  $\text{DISJ} \leq_{cc} f$  by a reduction that maps a pair of  $n$ -bit strings to an  $O(n)$ -node graph. By the Graph Minor Theorem [45] there exist graphs  $H_1, \dots, H_k$  such that  $f(G) = 0$  iff  $(\exists i)[H_i \preceq G]$ . Note that the  $H_i$ 's could be disconnected; however, none of the  $H_i$ 's can be  $\text{TRIV}_{a,b}$ .

Let  $H_1$  be the graph that has the smallest largest connected component, where we measure size by number of edges. We view  $H_1$  as being in two parts:  $\text{TRIV}_{a,b} \cup A$  where  $A$  does not share any edges or vertices with  $\text{TRIV}_{a,b}$ . It is possible  $a = 0$  or  $b = 0$  or both. The graph  $A$  must have a connected component with  $\geq 2$  edges in it. Break up the edge set of  $A$  into two disjoint sets such that every connected component of  $A$  with  $\geq 2$  edges is broken up. Call these two parts  $A_1$  and  $A_2$ .

We define the reduction  $T_1$  (respectively  $T_2$ ). On input  $(x_1 \cdots x_n)$  (respectively  $(y_1 \cdots y_n)$ )  $T_1$  does the following.

1. Put  $\text{TRIV}_{a,b}$  on the first  $a + 2b$  vertices. (Same with  $T_2$ ). Break up the remaining vertices into  $n$  groups of  $|V(A)|$  vertices each. (Same with  $T_2$ .)
2. For all  $i \in \{1, \dots, n\}$  do the following. If  $x_i = 1$  then put  $A_1$  on the  $i$ th group of vertices. If  $x_i = 0$  then do not put those edges in. (If  $y_i = 1$  then put  $A_2$  on the  $i$ th group of vertices. If  $y_i = 0$  then do not put those edges in.)

If  $\text{DISJ}(x_1 \cdots x_n, y_1 \cdots y_n) = 0$  then there exists  $i$  such that  $x_i = y_i = 1$ . Hence  $G$  will have  $\text{TRIV}_{a,b} \cup A = H_1$  as a minor so  $f(G) = 0$ .

If  $\text{DISJ}(x_1 \cdots x_n, y_1 \cdots y_n) = 1$  then there is no such  $i$ .  $G$  will be  $\text{TRIV}_{a,b}$  unioned with graphs all of whose connected components are smaller than the smallest largest connected component of a forbidden minor. Hence  $G$  cannot have any of  $H_1, \dots, H_k$  as minors, so  $f(G) = 1$ .

Proof of 3,4,5,6,7,8,9,10: Items 3 and 4 follow from item 2, Theorem 3.2, and Lemma 4.2. Items 5,6, and 7 are easy consequences of items 1,2,3. Item 8 follows from item 4, Corollary 3.2, and Lemma 1.2. (Alternatively, item 8 follows from item 6 and Lemma 4.2.) Item 9 follows from items 1,4. Item 10 follows from items 5,8. ■

**Note:** Theorem 4.1 raises the question of whether some non-trivial graph property  $f$  closed under minors has  $D(f) \ll n$ . The answer is yes: If  $f(G)$  returns yes

iff  $G$  has fewer than 64 edges then  $D(f) = O(1)$ . Theorem 4.1 also raises the question of whether some non-trivial graph property  $f$  closed under minors such that  $(\exists a)(\exists b)[f(\text{TRIV}_{a,b}) = 0]$  has  $D(f) = \Omega(n)$ . The answer is yes: Let  $f(G) = 1$  iff  $G$  does not have  $\text{TRIV}_{0,2}$  as a minor. Note that  $f$  is closed under minors and  $f(\text{TRIV}_{0,2}) = 0$ . We show that  $\overline{\text{INTER}} \leq_{\text{cc}} f$  by a linear reduction. Given  $(x_1 \dots x_n, y_1 \dots y_n)$  Alice constructs the graph that places an edge between  $a_i$  and  $b_i$  iff  $x_i = 1$ . Bob constructs the graph that places an edge between  $b_i$  and  $c_i$  iff  $y_i = 1$ . Let this graph be  $G$ . Clearly  $\overline{\text{INTER}}(x_1 \dots x_n, y_1 \dots y_n) = 1$  iff  $\text{TRIV}_{0,2}$  is a minor of  $G$ . Since  $D(\overline{\text{INTER}}) \geq n$  and  $\overline{\text{INTER}} \geq n$  we have  $D(f) \geq \Omega(n)$ . The question of which properties closed under minors have sublinear communication complexity looks like it will not have a clean answer.

Theorem 4.1 covers many graph properties; however, there are some properties that are not covered. One example is Hamiltonicity. Hence we look at another type of graph property.

**DEFINITION 4.5.** A property of graphs  $f$  is *delicate* if, for almost all  $n$ , there exists a graph  $G_n = (V_n, E_n)$  such that (1)  $|V_n| = \Theta(n)$ , (2)  $|E_n| \geq n$ , (3)  $f(G_n) = 1$ , and (4) for every nonempty  $E' \subseteq E_n$ ,  $f((V_n, E_n - E')) = 0$ .

**EXAMPLE 4.2.** The following graph properties are delicate.

$$\begin{aligned} \text{HAM} &= \{G \mid G \text{ has a Hamiltonian Cycle}\} \\ \text{HAMP} &= \{G \mid G \text{ has a Hamiltonian Path}\} \\ \text{EULER} &= \{G \mid G \text{ has an Eulerian Cycle}\} \\ \text{EULERP} &= \{G \mid G \text{ has an Eulerian Path}\} \\ \text{NOTCOL}_2 &= \{G \mid G \text{ is not 2-colorable}\} \\ \text{NOTCOL}_k &= \{G \mid G \text{ is not } k\text{-colorable}\} \\ \text{CONN} &= \{G \mid G \text{ is connected}\} \end{aligned}$$

For  $\text{HAM}$  and  $\text{EULER}$  take  $G_n$  to be the cycle on  $n$  vertices. For  $\text{HAMP}$ ,  $\text{EULERP}$ , and  $\text{CONN}$  take  $G_n$  to be the path on  $n$  vertices. For  $\text{NOTCOL}_2$  take the cycle on  $n$  vertices if  $n$  is odd, and the cycle on  $n + 1$  if  $n$  is even. For  $\text{NOTCOL}_k$  do the following. Let  $x, y \in \mathbb{N}$  be such that  $k + 1 = 3x + 2y$  and  $y \in \{0, 1, 2\}$ . Let  $n_x$  be the element of  $\{\lceil n/x \rceil, \lceil n/x \rceil + 1\}$  which is odd. Let  $G_n = (V_n, E_n)$  be the graph formed by taking  $x$  cycles of length  $n_x$ ,  $y$  edges, and for all pairs of vertices  $a, b$  where  $a$  and  $b$  come from different parts of the graph, put in the edge  $(a, b)$ . Since odd length cycles require 3 colors,  $G_n$  is  $3x + 2y = k + 1$ -colorable. It is easy to see that if you remove any (nonempty) set of edges then the resulting graph is  $k$ -colorable. Note that  $|V_n| = xn_x + 2y = \Theta(n)$ ,  $|E_n| = xn_x + y + \binom{x}{2}n_x^2 + 4\binom{y}{2} \geq n$ ,

Note that these properties and their complements are not closed under minor, hence Theorem 4.1 would not apply to them.

**THEOREM 4.2.** *Let  $f$  be a delicate property of graphs. Let  $g = 1 - f$ .*

1.  $\text{DISJ} \leq_{\text{cc}} f$  by a linear reduction.
2.  $N(f) \geq \Omega(n)$ .
3.  $N(\text{ELIM}(f^k)) \geq \Omega(n)$ .
4.  $\text{INTER} \leq_{\text{cc}} g$  by a linear reduction.

- 5.  $D(g) \geq \Omega(n)$ .
- 6.  $D(\text{ELIM}(g^k)) \geq \Omega(n)$ .
- 7.  $D(\text{ELIM}(f^k)) \geq \Omega(\sqrt{D(f)})$ .
- 8.  $D(\text{ELIM}(g^k)) \geq \Omega(\sqrt{D(g)})$ .

*Proof.* We will prove item 1. Items 2,3,4,5,6,7,8 will follow from item 1 and prior results.

Proof of part 1: Let  $n \in \mathbb{N}$  and let  $G_n = (V_n, E_n)$  be as in Definition 4.5. Let  $E_n = \{e_1, \dots, e_n, \dots, e_{|E_n|}\}$ . We show that  $\text{DISJ} \leq_{\text{cc}} f$  by a linear reduction. Map  $(x, y) \in \{0, 1\}^n \times \{0, 1\}^n$  to the pair of graphs  $((V_1, E_1), (V_2, E_2))$  where  $V_1 = V_2 = V_n$ ,  $E_1 = \{e_i \mid x_i = 0 \vee i \geq n+1\}$ , and  $E_2 = \{e_i \mid y_i = 0 \vee i \geq n+1\}$ . If  $\text{DISJ}(x, y) = 1$  then  $(\forall i)[x_i = 0 \vee y_i = 0]$  so  $(V_1 \cup V_2, E_1 \cup E_2) = (V_n, E_n) = G_n$  hence  $f((V_1 \cup V_2, E_1 \cup E_2)) = 1$ . If  $\text{DISJ}(x, y) = 0$  then  $(\exists i)[x_i = y_i = 1]$  so  $(V_1 \cup V_2, E_1 \cup E_2) = (V_n, E_n - E')$  where  $E' \neq \emptyset$  since  $e_i \in E'$ , so (by the nature of  $G_n$ )  $f((V_1 \cup V_2, E_1 \cup E_2)) = 0$ .

Proof of parts 2,3,4,5,6,7,8: Items 2 and 3 follow from item 1, Theorem 3.2, and Lemma 4.2. Items 4 and 5 follow from items 1,2. Item 6 follows from item 3, Corollary 3.2, and Lemma 1.2. Items 7,8 follow from items 2,5, and the fact that for any graph property  $h$   $D(h) \leq O(|V|^2)$ . ■

**Note:** It is known that  $D(\text{CONN}) = \Theta(n \log n)$  [23]. Hence, by Theorem 4.2,  $D(\text{ELIM}(\text{CONN}^k)) = \Omega\left(\frac{D(f)}{\log(D(f))}\right)$

## 5. THE COMPLEXITY OF ELIM(IP<sup>k</sup>)

We show that  $N(\text{ELIM}(\text{IP}^k)) \geq n$ , hence ELC holds for IP. For this we need a lemma. We state the lemma, then from it prove the theorem, then return to proving the lemma.

**LEMMA 5.1.** *Let  $A, B \subseteq \{\{0, 1\}^n - 0^n\}^k$  be such that  $|A||B| > pH^{2k}$  where  $p = \frac{1}{2^n - 4}$  and  $H = 2^n - 1$ . Then, for any  $z \in \{0, 1\}^k$ , there are  $x \in A$ ,  $y \in B$  such that  $\text{IP}^k(x, y) = z$ .*

**THEOREM 5.1.** *For all  $k$ , for all  $n \geq 4$ ,  $N(\text{ELIM}(\text{IP}^k)) \geq n$ .*

*Proof.* Let  $p$  and  $H$  be as in Lemma 5.1. Assume that  $N(\text{ELIM}(\text{IP}^k)) = t$ . Let  $C = \{\{0, 1\}^n - 0^n\}^k \times \{\{0, 1\}^n - 0^n\}^k$ . Note that  $|C| = H^{2k}$ . By Lemma 1.1 there is an  $A \subseteq \{\{0, 1\}^n\}^k$ , a  $B \subseteq \{\{0, 1\}^n\}^k$ , and a vector  $b \in \{0, 1\}^k$ , such that  $|C \cap (A \times B)| \geq |H|^{2k}/2^t$  and  $(\forall x \in A)(\forall y \in B)[\text{IP}^k(x, y) \neq b]$ . By the nature of  $C$  we can assume  $A, B \subseteq \{\{0, 1\}^n - 0^n\}^k$ . By Lemma 5.1 if  $|A||B| > pH^{2k}$  then  $(\exists x \in A)(\exists y \in B)[\text{IP}^k(x, y) = b]$ . Since  $b$  is eliminated from being  $\text{IP}^k(x, y)$  we have  $|A||B| \leq pH^{2k}$ . Therefore  $\frac{H^{2k}}{2^t} \leq pH^{2k}$ ,  $\frac{1}{p} \leq 2^t$ , and  $2^n - 4 \leq 2^t$ . Since  $n \geq 4$  we have  $t \geq n$ . ■

**Note:** Theorem 5.1 is proven for  $n \geq 4$ . For  $n = 1, 2$  the theorem is true and easy. The case of  $n = 3$  is open, though we suspect it holds there as well.

We first prove the lemma for the  $k = 1$  case.

**LEMMA 5.2.** *Let  $A, B \subseteq \{0, 1\}^n - 0^n$  and let  $i \in \{0, \dots, n\}$ . If  $|A| \geq 2^i$  and  $|B| \geq 2^{n-i-1}$  then  $(\exists x \in A)(\exists y \in B)[\text{IP}(x, y) = 1]$ .*

*Proof.* Let  $A'$  be the linear subspace of  $\{0,1\}^n$  spanned by  $A$ . Then,  $|A'| \geq |A| + 1 \geq 2^i + 1$  because  $A \subseteq A'$  and  $0^n \in A' - A$ . Therefore, the dimension of  $A'$  is at least  $i + 1$ . This means that the dimension of  $(A')^\perp$  (the set of all vectors perpendicular to all vectors in  $A'$ ) is at most  $n - i - 1$  and  $|(A')^\perp - 0^n| \leq 2^{n-i-1} - 1$ . Hence, there is an  $x \in B$  and  $y_1, \dots, y_k \in A$  such that  $x$  and  $\sum_{i=1}^k y_i \in A'$  are not perpendicular. Hence there must be an  $i$  such that  $\text{IP}(x, y_i) = 1$ . ■

LEMMA 5.3. *Let  $A, B \subseteq \{0,1\}^n - 0^n$  and let  $i \in \{1, \dots, n+1\}$ . If  $|A| \geq 2^{i-2} + 1$  and  $|B| \geq 2^{n-i} + 1$  then  $(\exists x \in A)(\exists y \in B)[\text{IP}(x, y) = 0]$ .*

*Proof.* Assume, by way of contradiction, that for every  $x \in A$  and  $y \in B$  we have  $\text{IP}(x, y) = 1$ . Fix  $x_0 \in A$  and  $y_0 \in B$ . Let  $A' = \{x - x_0 \mid x \in A\}$  and  $B' = \{y - y_0 \mid y \in B\}$ . For every  $y \in B$ ,  $\text{IP}(x - x_0, y) = \text{IP}(x, y) - \text{IP}(x_0, y) = 1 - 1 = 0$  and  $\text{IP}(x - x_0, y - y_0) = \text{IP}(x - x_0, y) - \text{IP}(x - x_0, y_0) = 0$ . Therefore,  $A'$  and  $B'' = B \cup B'$  are perpendicular. Moreover, the subspaces spanned by  $A'$  and  $B''$  are perpendicular.

The sets  $B$  and  $B'$  do not overlap: if  $y \in B$  and  $y - y_0 \in B$  then  $\text{IP}(x_0, y - y_0) = 1$ , so  $\text{IP}(x_0, y) - \text{IP}(x_0, y_0) = 1$ , and since  $\text{IP}(x_0, y_0) = 1$  we get  $\text{IP}(x_0, y) = 0$ . The sets  $B$  and  $B'$  are the same size since the function  $y \rightarrow y - y_0$  is a bijection between them.

The dimension of the subspace spanned by  $A'$  is at least  $i - 1$  because  $|A'| = |A| \geq 2^{i-2} + 1$ . The dimension of the subspace spanned by  $B''$  is at least  $n - i + 2$  because  $|B''| = |B| + |B'| = 2|B| = 2^{n-i+1} + 2$ . The sum of these two dimensions is at least  $(i - 1) + (n - i + 2) = n + 1$ . However, if two subspaces are perpendicular, the sum of their dimensions is at most  $n$ . This is a contradiction. ■

We now restate and prove the lemma.

LEMMA 5.4. *Let  $A, B \subseteq \{\{0,1\}^n - 0^n\}^k$  be such that  $|A||B| > pH^{2k}$  where  $p = \frac{1}{2^{n-4}}$  and  $H = 2^n - 1$ . Then, for any  $z \in \{0,1\}^k$ , there are  $x \in A$ ,  $y \in B$  such that  $\text{IP}^k(x, y) = z$ .*

*Proof.* By induction. The base case is  $k = 1$ :  $A, B \subseteq \{\{0,1\}^n - 0^n\}$ . and  $|A||B| > pH^2 \geq 2^n$ . By Lemmas 5.2 and 5.3, this implies that there are  $x_1, x_2 \in A$ ,  $y_1, y_2 \in B$  with  $\text{IP}(x_1, y_1) = 0$  and  $\text{IP}(x_2, y_2) = 1$ .

For the induction step there are two cases:  $z_k = 0$  and  $z_k = 1$ . We prove the  $z_k = 0$  case in detail, and then sketch the  $z_k = 1$  case which is similar.

### I) What if $z_k = 0$ ?

Assume  $k > 1$ . Let

$$A_1 = \{x_1 \cdots x_{k-1} \mid x_1 \cdots x_k \in A \text{ for at least one } x_k\}.$$

For  $i \in \{2, \dots, n+1\}$  let

$$A_i = \{x_1 \cdots x_{k-1} \mid x_1 \cdots x_k \in A \text{ for at least } 2^{i-2} + 1 \text{ } x_k\}.$$

The sets  $B_i$  for  $i \in \{1, \dots, n+1\}$  are defined similarly.

We consider two cases:

**Case 1:**  $|A_i||B_{n+2-i}| > pH^{2(k-1)}$  for some  $i \in \{1, \dots, n+1\}$ .

Then, by inductive assumption, there are  $x_1 \cdots x_{k-1} \in A_i$  and  $y_1 \cdots y_{k-1} \in B_{n-i}$  such that  $\text{IP}(x_1, y_1) = z_1, \dots, \text{IP}(x_{k-1}, y_{k-1}) = z_{k-1}$ . We fix  $x_1, y_1, \dots, x_{k-1}, y_{k-1}$  with this property. Let  $C = \{x_k \mid x_1 \cdots x_k \in A_i\}$ ,  $D = \{y_k \mid y_1 \cdots y_k \in B_{n-i}\}$ .



Then,  $|C| \geq 2^{i-2} + 1$  and  $|D| \geq 2^{n-i} + 1$ . By Lemma 5.3, this means that there are  $x_k \in C, y_k \in D$  such that  $\text{IP}(x_k, y_k) = 0 = z_k$ .

**Case 2:** For all  $i \in \{1, \dots, n+1\}$ ,  $|A_i||B_{n+2-i}| \leq pH^{2(k-1)}$ . We will show that this implies  $|A||B| \leq pH^{2k}$ , and hence cannot occur.

Note that  $A_1 \supseteq A_2 \supseteq \dots \supseteq A_{n+1}$ . For every  $x_1 \dots x_k \in A$  we know that  $x_1 \dots x_{k-1}$  is either in  $A_1 - A_2$  or  $A_2 - A_3$  or  $\dots$  or  $A_n - A_{n+1}$  or  $A_{n+1}$ . For  $1 \leq i \leq n$ , for every  $x_1 \dots x_{k-1} \in A_i - A_{i+1}$  there are at most  $2^{i-1}$  extensions of it that are in  $A$  (by the definition of  $A_{i+1}$ ). For every  $x_1 \dots x_{k-1} \in A_{n+1}$  there are at most  $2^n - 1$  extensions of it that are in  $A$  since there are only  $2^n - 1$  elements in  $\{0, 1\}^n - 0^n$ .

Hence we have

$$|A| \leq (|A_1| - |A_2|)2^0 + (|A_2| - |A_3|)2^1 + \dots + (|A_n - A_{n+1}|)2^{n-1} + |A_{n+1}|(2^n - 1). \quad (1)$$

By grouping the terms with the same  $|A_i|$  together we can rewrite (1) as

$$\begin{aligned} |A| &\leq 2^0|A_1| + (2^1 - 2^0)|A_2| + \dots + (2^{n-1} - 2^{n-2})|A_n| + (2^n - 1 - 2^{n-1})|A_{n+1}| \\ &= 2^0|A_1| + 2^0|A_2| + 2^1|A_3| + \dots + 2^{n-2}|A_n| + (2^{n-1} - 1)|A_{n+1}|. \end{aligned} \quad (2)$$

Similarly,

$$|B| \leq 2^0|B_1| + 2^0|B_2| + 2^1|B_3| + \dots + 2^{n-2}|B_n| + (2^{n-1} - 1)|B_{n+1}|. \quad (3)$$

Let  $a = \frac{|A|}{H^k}$ ,  $b = \frac{|B|}{H^k}$ ,  $a_i = \frac{|A_i|}{H^{k-1}}$ ,  $b_i = \frac{|B_i|}{H^{k-1}}$ .

Note that we want  $ab \leq p$ . We have

$$a \leq \frac{1}{H}(2^0 a_1 + 2^0 a_2 + 2^1 a_3 + 2^2 a_4 + \dots + 2^{n-2} a_n + (2^{n-1} - 1) a_{n+1})$$

$$b \leq \frac{1}{H}(2^0 b_1 + 2^0 b_2 + 2^1 b_3 + 2^2 b_4 + \dots + 2^{n-2} b_n + (2^{n-1} - 1) b_{n+1}).$$

Hence we want

$$\begin{aligned} &(2^0 a_1 + 2^0 a_2 + 2^1 a_3 + 2^2 a_4 + \dots + 2^{n-2} a_n + (2^{n-1} - 1) a_{n+1}) \times \\ &(2^0 b_1 + 2^0 b_2 + 2^1 b_3 + 2^2 b_4 + \dots + 2^{n-2} b_n + (2^{n-1} - 1) b_{n+1}) \leq pH^2. \end{aligned}$$

Note that  $1 \geq a_1 \geq a_2 \geq \dots \geq a_{n+1}$ . and  $1 \geq b_1 \geq b_2 \geq \dots \geq b_{n+1}$ . Then, to show that Lemma 5.1 is true for  $k$ , we prove the following lemma.

**LEMMA 5.5.** *Let  $1 \geq a_1 \geq a_2 \geq \dots \geq a_{n+1}$ , and  $1 \geq b_1 \geq \dots \geq b_{n+1}$  be such that  $a_i b_{n+2-i} \leq p$  for all  $i \in \{1, \dots, n+1\}$ . Then,*

$$\begin{aligned} &(2^0 a_1 + 2^0 a_2 + 2^1 a_3 + 2^2 a_4 \dots + 2^{n-2} a_n + (2^{n-1} - 1) a_{n+1}) \times \\ &(2^0 b_1 + 2^0 b_2 + 2^1 b_3 + 2^2 b_4 \dots + 2^{n-2} b_n + (2^{n-1} - 1) b_{n+1}) \leq pH^2. \end{aligned} \quad (4)$$

*Proof.* We first claim that we can take  $a_i \geq p$ . Assume  $a_i < p$ . The only constraints involving  $a_i$  are  $a_i b_{n+2-i} \leq p$  and  $a_i \leq 1$ . The only other constraint

involving  $b_{n+2-i}$  is  $b_{n+2-i} \leq 1$ . Hence if you lifted  $a_i$  to  $p$  you would not loosen the constraints on  $b_{n+2-i}$ . Hence there is no reason not to lift  $a_i$  up to  $p$ .

We can assume that  $b_i = p/a_{n+2-i}$  for all  $i \in \{1, \dots, n+1\}$  (because we have  $b_i \leq p/a_{n+2-i}$  and increasing  $b_i$  can only increase the expression on the left-hand side of (4)). Denote

$$f(a_1, \dots, a_{n+1}) = (2^0 a_1 + 2^0 a_2 + 2^1 a_3 + 2^2 a_4 + \dots + 2^{n-2} a_n + (2^{n-1} - 1) a_{n+1}) \times \\ (2^0 \frac{p}{a_{n+1}} + 2^0 \frac{p}{a_n} + 2^1 \frac{p}{a_{n-1}} + 2^2 \frac{p}{a_{n-2}} + \dots + 2^{n-2} \frac{p}{a_2} + (2^{n-1} - 1) \frac{p}{a_1})$$

Then, we have to show that  $f(a_1, \dots, a_{n+1}) \leq pH^2$  for all  $a_1, \dots, a_{n+1}$  satisfying  $1 \geq a_1 \geq a_2 \geq \dots \geq a_{n+1}$ .

We show that  $f(a_1, \dots, a_{n+1})$  is maximized by taking  $a_1 = \dots = a_i, a_{i+1} = \dots = a_{n+1}$  for some  $i$ . Let  $a_i > a_{i+1} = a_{i+2} = \dots = a_j > a_{j+1}$  for some  $1 \leq i < j \leq n+1$ . Then, one can increase  $f$  as follows.

Let  $g(x) = f(a_1, \dots, a_i, x, \dots, x, a_{j+1}, \dots, a_{n+1})$ ,  $x \in [a_{j+1}, a_i]$ . Then,  $g(x) = bx + c + \frac{d}{x}$  for some  $b, c, d \in \mathbb{R}$ . For any interval  $[a_{j+1}, a_i]$ ,  $g(x)$  is maximized by one of its endpoints. Therefore, one can increase  $f(a_1, \dots, a_{n+1})$  by setting  $a_{i+1}, \dots, a_j$  all equal to  $a_i$  or  $a_{j+1}$ .

We show that if  $a_1 = \dots = a_i > a_{i+1} = \dots = a_{n+1}$ , then  $a_1 = 1$  and  $a_{n+1} = p$ . Look at  $g(x) = f(x, \dots, x, a_{i+1}, \dots, a_{n+1})$ . Again  $g(x) = bx + c + \frac{d}{x}$  for some  $b, c, d \in \mathbb{R}$  and  $g$  is maximized either by  $x = a_{i+1}$  or  $x = 1$ . Since  $a_i > a_{i+1}$  we need to take  $x = 1$ . A similar argument, using that  $a_i \geq p$ , shows that  $a_{i+1} = \dots = a_{n+1} = p$ .

If  $a_1 = a_2 = \dots = a_{n+1}$ , then  $f(a_1, \dots, a_{n+1})$  is just

$$(2^0 a_1 + 2^0 a_1 + 2^1 a_1 + 2^2 a_1 + \dots + 2^{n-2} a_1 + (2^{n-1} - 1) a_1) \times \\ (2^0 \frac{p}{a_1} + 2^0 \frac{p}{a_1} + 2^1 \frac{p}{a_1} + 2^2 \frac{p}{a_1} + \dots + 2^{n-2} \frac{p}{a_1} + (2^{n-1} - 1) \frac{p}{a_1}) \\ = p(2^0 + 2^0 + 2^1 + 2^2 + \dots + 2^{n-2} + 2^{n-1} - 1)^2 \\ = p(2^n - 1)^2 \\ \leq pH^2.$$

Otherwise,  $a_1 = \dots = a_i = 1, a_{i+1} = \dots = a_{n+1} = p$  for some  $i \in \{1, \dots, n\}$ . Then,

$$f(a_1, \dots, a_{n+1}) = (2^0 + 2^0 + 2^1 + 2^2 + \dots + 2^{i-2} \\ + 2^{i-1} p + 2^{i+2} p + \dots + 2^{n-2} p + (2^{n-1} - 1) p) \times \\ (2^0 + 2^0 + 2^1 + 2^2 + \dots + 2^{n-i-1} \\ + 2^{n-i} p + \dots + 2^{n-2} p + (2^{n-1} - 1) p) \\ = (2^{i-1} + 2^{i-1}(1 + 2 + 2^2 + \dots + 2^{n-i-1} + 2^{n-i}) p - p) \times \\ (2^{n-i} + 2^{n-i}(1 + 2 + 2^2 + \dots + 2^{i-2} + 2^{i-1}) p - p) \\ = (2^{i-1} + 2^{i-1}(2^{n-i+1} - 1) p - p) \times (2^{n-i} + 2^{n-i}(2^i - 1) p - p) \\ = (2^{i-1} + (2^n - 2^{i-1}) p - p) \times (2^{n-i} + (2^n - 2^{n-i}) p - p) \\ = (2^{i-1} + (2^n - 2^{i-1} - 1) p) \times (2^{n-i} + (2^n - 2^{n-i} - 1) p).$$

If multiplied out  $f(a_1, \dots, a_{n+1})$  would be of the form  $B2^i + C2^{-i} + D$  where  $B, C, D > 0$ . Simple calculus shows that the maximum this function achieves on the interval  $[1, n]$  occurs at one of the endpoints. At  $i = 1$  (or, equivalently  $i = n$ ).

it is equal to

$$((2^{n-1} - 1)p + 2^{n-1})((2^n - 2)p + 1) = \left( \frac{2^{n-1} - 1}{2^n - 4} + 2^{n-1} \right) \left( \frac{2^n - 2}{2^n - 4} + 1 \right) =$$

$$\left( 2^{n-1} + \frac{1}{2} + \frac{1}{2^n - 4} \right) \frac{2(2^n - 3)}{2^n - 4} = \frac{(2^n + 1 + \frac{2}{2^n - 4})(2^n - 3)}{2^n - 4} < \frac{(2^n - 1)^2}{2^n - 3}.$$

**End of proof of Lemma 5.5** ■

**II) What if  $z_k = 1$ ?**

Assume  $k > 1$ . Define  $A_i = \{x_1 \cdots x_{k-1} \mid x_1 \cdots x_k \in A \text{ for at least } 2^{i-1} \text{ } x_k\}$  and  $B_i = \{y_1 \cdots y_{k-1} \mid y_1 \cdots y_k \in B \text{ for at least } 2^{i-1} \text{ } y_k\}$  for all  $i \in \{1, \dots, n\}$ . Again, we consider two cases.

**Case 1:** For some  $i \in \{1, \dots, n\}$ ,  $|A_i||B_{n+1-i}| \geq pH^{2(k-1)}$ .

Then, by inductive assumption, there are  $x_1 \cdots x_{k-1} \in A_i$  and  $y_1 \cdots y_{k-1} \in B_{n+1-i}$  such that  $\text{IP}(x_1, y_1) = z_1, \dots, \text{IP}(x_{k-1}, y_{k-1}) = z_{k-1}$ . Fix such  $x_1, \dots, x_{k-1}, y_1, \dots, y_{k-1}$ . Define  $C = \{x_k \mid x_1 \cdots x_k \in A\}$  and  $D = \{y_k \mid y_1 \cdots y_k \in B\}$ . Note that  $|C| \geq 2^{i-1}$  and  $|D| \geq 2^{n-i}$ . By Lemma 5.2 there exists  $x \in C$  and  $y \in D$  such that  $\text{IP}(x, y) = 1$ . Let  $x_k = x$  and  $y_k = y$ .

**Case 2:** For all  $i \in \{1, \dots, n\}$ ,  $|A_i||B_{n+1-i}| \leq pH^{2(k-1)}$ .

Then, for every  $x_1 \cdots x_{k-1} \in A_i - A_{i+1}$ , there are at most  $2^i - 1$   $x_k$  such that  $x_1 \cdots x_k \in A$ . (Otherwise,  $x_1 \cdots x_{k-1}$  would belong to  $A_{i+1}$ .) Therefore, we have

$$|A| \leq (|A_1| - |A_2|)(2^1 - 1) + (|A_2| - |A_3|)(2^2 - 1) + \cdots + |A_n|(2^n - 1) =$$

$$(2^1 - 1)|A_1| + (2^2 - 2^1)|A_2| + \cdots + (2^n - 2^{n-1})|A_n| =$$

$$2^0|A_1| + 2^1|A_2| + \cdots + 2^{n-1}|A_n|.$$

Define  $a_i$  and  $b_i$  similarly to  $z_k = 0$  case. Then, we have to prove

**LEMMA 5.6.** *Let  $1 \geq a_1 \geq a_2 \geq \cdots \geq a_n$  and  $1 \geq b_1 \geq b_2 \geq \cdots \geq b_n$  be such that  $a_i b_{n+1-i} \leq p$  for all  $i \in \{1, \dots, n\}$ . Then,*

$$(a_1 + 2a_2 + \cdots + 2^{n-1}a_n)(b_1 + 2b_2 + \cdots + 2^{n-1}b_n) \leq pH^2.$$

*Proof.* Similarly to Lemma 5.5 we can assume that all  $a_i$  and  $b_i$  are at least  $p$  and  $b_i = p/a_{n+1-i}$  for all  $i \in \{1, \dots, n\}$ . Then, proving this lemma is equivalent to showing that the function

$$f(a_1, \dots, a_n) = (a_1 + 2a_2 + \cdots + 2^{n-1}a_n) \left( \frac{p}{a_n} + 2\frac{p}{a_{n-1}} + \cdots + 2^{n-1}\frac{p}{a_1} \right).$$

is always at most  $pH^2$ . Again, similarly to the proof of Lemma 5.5, we get that  $f(a_1, \dots, a_n)$  is maximized by  $a_1 = \cdots = a_i = 1$ ,  $a_{i+1} = \cdots = a_n = p$ . Then,

$$f(a_1, \dots, a_n) = (2^i - 1 + (2^n - 2^i)p) \times (2^{n-i} - 1 + (2^n - 2^{n-i})p).$$

If we consider this as a function of  $i$ , the derivative is negative if  $i < n/2$  and positive if  $i > n/2$ . Therefore, it is maximized by  $i = 0$  (or, equivalently,  $i = n$ ). In this case  $f(a_1, \dots, a_n)$  is just  $(2^n - 1)p \times (2^n - 1) = pH^2$ .

**End of proof of Lemma 5.6** ■

**End of proof of Lemma 5.1** ■

## 6. LOWER BOUNDS FOR RANDOMIZED PROTOCOLS

Let  $k$  be a constant. We show that if  $\epsilon < \frac{1}{2^k}$  then  $R_\epsilon^{\text{pub}}(\text{ELIM}(\text{INTER}^k)) = \Omega(\frac{n}{\log(n)\log\log(n)})$  and  $R_\epsilon^{\text{pub}}(\text{ELIM}(\text{IP}^k)) = \Omega(\frac{n}{\log(n)\log\log(n)})$ . Note that if  $\epsilon = \frac{1}{2^k}$  then, for any  $f$ ,  $R_\epsilon^{\text{pub}}(f^k) = 0$  since any random sequence of  $k$  bits has a high probability of not being  $f^k(x, y)$  (both Alice and Bob output the first  $n$  random public bits).

**LEMMA 6.1.** *Let  $k$  and  $\epsilon < \frac{1}{2^k}$  be constants. Let  $Z$  be a set such that  $|Z| \leq 2^k$ . Let  $S$  be a relation on  $\{0, 1\}^n \times \{0, 1\}^n \times Z$  such that  $(\forall x)(\forall y)(\exists z)[S(x, y, z)]$ . If  $R_\epsilon^{\text{pub}}(S) \leq t$  then  $R_{1/\log^2 n}^{\text{pub}}(S) \leq O(t \log \log n)$ .*

*Proof.* Let  $R_\epsilon^{\text{pub}}(S) \leq t$  via protocol  $P$ . Let  $(x, y)$  be an input. We can amplify the probability by running protocol  $P$  on  $(x, y)$   $s$  times and returning the most frequent answer. If incorrect strings (i.e., strings  $z$  such that  $\neg S(x, y, z)$ ) occur less than  $s/2^k$  times then it follows that at least one of the correct strings must occur more than  $s/2^k$  times. In other words we get a correct answer with high probability if the fraction of the occurrences of incorrect answers can be kept strictly less than  $s/2^k$  with high probability. We use Chernoff bounds to get an estimate. If  $S_n$  is the number of occurrences of incorrect strings in  $s$  runs of the protocol then  $\epsilon s$  is the expectation of  $S_n$ . We must keep  $|S_n - \epsilon s|$  strictly less than  $s(1/2^k - \epsilon)$ . Let  $m = s(1/2^k - \epsilon)$ . Recall that Chernoff bounds give

$$\text{Prob}[|S_n - \epsilon s| \geq m] \leq 2e^{-m^2/3\epsilon s}.$$

Which means that for some constant  $c$  (depending on  $k$  and  $\epsilon$ ) this probability is less than  $e^{-cs}$ . Take  $s = \frac{1}{c} \ln \log^2 n = O(\log \log n)$ . ■

We first show a lower bound on the randomized communication complexity of  $\text{ELIM}(\text{INTER}^k)$ . We then make an observation that enables the same proof to yield a lower bound for the randomized communication complexity of  $\text{ELIM}(\text{IP}^k)$ . Recall that  $\text{INTER}$  stands for *not* disjoint. The proof applies a technique from [1, Theorem 3.5][10, Lemma 4.3][41, Theorem 5.1] in a novel way.

**LEMMA 6.2.** *Let  $k, m \in \mathbb{N}$ . Let  $x_1, \dots, x_{2^k-1}, y_1, \dots, y_{2^k-1} \in \{0, 1\}^*$  be such that  $(\forall i)[|x_i| = |y_i|]$ . Let  $X = x_1 \cdots x_{2^k-1}$  and  $Y = y_1 \cdots y_{2^k-1}$ . For  $i = 1, \dots, k$  let  $X_i$  ( $Y_i$ ) be a string obtained from  $X$  ( $Y$ ) as follows: Start with all the  $X_i, Y_i$  being the empty string. Then, for  $j = 1, \dots, 2^k - 1$ , concatenate  $x_j$  to  $X_i$  ( $y_j$  to  $Y_i$ ) if the  $i$ th bit of  $j$  is 1. For example  $X_1 = x_1 x_3 x_5 \cdots x_{2^k-1}$  and  $X_2 = x_2 x_3 x_6 x_7 \cdots x_{2^k-2} x_{2^k-1}$ . Assume  $\text{INTER}^k(X_k X_{k-1} \cdots X_1, Y_k Y_{k-1} \cdots Y_1) \neq b$  and  $b \neq 0^k$ . View  $b$  as a  $k$ -bit binary number (leading bits may be 0). Let  $X'$  ( $Y'$ ) be  $X$  with the  $x_b$  ( $y_b$ ) removed. Then  $\text{INTER}(X, Y) = 1 \Rightarrow \text{INTER}(X', Y') = 1$ .*

*Proof.* If  $\text{INTER}(X, Y) = 1$  and  $\text{INTER}(x_b, y_b) = 0$  then clearly  $\text{INTER}(X', Y') = 1$ . Hence we assume  $\text{INTER}(X, Y) = 1$  and  $\text{INTER}(x_b, y_b) = 1$ .

Let  $b = b_k b_{k-1} \cdots b_1$ . Let  $1 \leq j \leq k$ . If  $b_j = 1$  then  $x_b$  is a substring of  $X_j$  and  $y_b$  is a substring of  $Y_j$  and they are in the same position. Since  $\text{INTER}(x_b, y_b) = 1$  we obtain  $\text{INTER}(X_j, Y_j) = 1 = b_j$ . Since  $\text{INTER}^k(X_k X_{k-1} \cdots X_1, Y_k Y_{k-1} \cdots Y_1) \neq b$  we have  $\bigvee_{1 \leq i \leq k} \text{INTER}(X_i, Y_i) \neq b_i$ . Since  $\text{INTER}(X_i, Y_i) = b_i$  this reduces to  $\bigvee_{1 \leq i \leq k, b_i=0} \text{INTER}(X_i, Y_i) \neq b_i$  hence  $\bigvee_{1 \leq i \leq k, b_i=0} \text{INTER}(X_i, Y_i) = 1$ . Let  $i_0$  be such that  $b_{i_0} = 0$  and  $\text{INTER}(X_{i_0}, Y_{i_0}) = 1$ . Note that  $X_{i_0}$  ( $Y_{i_0}$ ) does not have  $x_b$  ( $y_b$ ) placed in it. Hence  $\text{INTER}(X', Y') = 1$ . ■

LEMMA 6.3.  $R_{1/4}^{\text{pub}}(\text{INTER}) = \Omega(n)$ . Moreover,  $R_{1/4}^{\text{pub}}(\text{INTER}) = \Omega(n)$  even when restricted to

$$D = \{(x, y) \in \{0, 1\}^n \times \{0, 1\}^n : \text{for at most one } i, x_i = y_i\}.$$

*Proof.* It is known ([27], simplified in [44], and also in [34]) that  $R_{1/4}^{\text{pub}}(\text{DISJ}) = \Omega(n)$ . The proofs actually work even when restricted to domain  $D$ . Since  $\text{INTER} = 1 - \text{DISJ}$  the lower bound  $R_{1/4}^{\text{pub}}(\text{DISJ}) = \Omega(n)$  can easily be modified to obtain  $R_{1/4}^{\text{pub}}(\text{INTER}) = \Omega(n)$ , even when restricted to domain  $D$ . ■

THEOREM 6.1. Let  $k$  and  $\epsilon < 1/2^k$  be constants. Then  $R_\epsilon^{\text{pub}}(\text{ELIM}(\text{INTER}^k)) = \Omega\left(\frac{n}{\log(n) \log \log(n)}\right)$ .

*Proof.* Assume  $R_\epsilon^{\text{pub}}(\text{ELIM}(\text{INTER}^k)) = t(n)$  via protocol  $P'$ . By Lemma 6.1 we can obtain a protocol  $P$  such that  $R_{1/\log^2 n}^{\text{pub}}(\text{ELIM}(\text{INTER}^k)) = O(t(n) \log \log n)$  via  $P$ . We can also apply the protocol to  $k$ -tuples of inputs of length  $\leq n$  by having both Alice and Bob pad with 0's. We will still assume it costs  $t(n) \log \log n$ .

We use  $P$  to help show  $R_{1/4}^{\text{pub}}(\text{INTER}) = O(t(n) \log(n) \log \log(n))$ . By Lemma 6.3,  $R_{1/4}^{\text{pub}}(\text{INTER}) = \Omega(n)$ , hence we have  $t(n) = \Omega\left(\frac{n}{\log n \log \log n}\right)$ .

Let  $X$  and  $Y$  be two strings of length  $n$ . Let Alice have  $X$  and Bob have  $Y$ . Alice and Bob divide  $X$  and  $Y$  into  $2^k - 1$  parts so that  $X = x_1 \dots x_{2^k-1}$ ,  $Y = y_1 \dots y_{2^k-1}$ ,  $|x_1| = \dots = |x_{2^k-2}| = |y_1| = \dots = |y_{2^k-1}| = \lfloor n/(2^k - 1) \rfloor$ , and  $|x_{2^k-1}| = |y_{2^k-1}| = n - (2^k - 2) \lfloor n/(2^k - 1) \rfloor \geq \lfloor \frac{n}{2^k-1} \rfloor$ . Let  $X_i$  ( $Y_i$ ) be a string obtained from  $X$  ( $Y$ ) as in Lemma 6.2. Note that  $|X_i| = |Y_i| \leq n$  so we can apply the protocol  $P$  to  $(X_k \cdots X_1, Y_k \cdots Y_1)$ .

Run protocol  $P$  on  $(X_k \cdots X_1, Y_k \cdots Y_1)$ . If the protocol returns  $0^k$  then Alice and Bob stop and reject. Note that if this happens then  $\text{Prob}(\bigvee_{i=1}^k \text{INTER}(X_i, Y_i) = 1) \leq \frac{1}{\log^2 n}$ , so  $\text{Prob}(\text{INTER}(X, Y) = 1) \leq \frac{1}{\log^2 n}$ , hence the probability of error is  $\leq \frac{1}{\log^2 n}$ . If the protocol returns  $b = b_1 \cdots b_k \neq 0^k$  then by Lemma 6.2 with probability greater than  $1 - \frac{1}{\log^2 n}$  we have  $\text{INTER}(X, Y) = 1 \Rightarrow \text{INTER}(X', Y') = 1$  where  $X'$  is  $X$  with the  $x_b$  cut out (and  $Y'$  is similar). Next, Alice and Bob remove  $x_b$  and  $y_b$  from their strings and reiterate the process. In each recursive step, Alice and Bob start with a string of length  $m$  and remove at least  $\lfloor m/(2^k - 1) \rfloor$  bits from that string for the next iteration. The recursion stops when the length of the strings left is less than or equal  $2^k - 1$  bits. Since the invariant preserves, with high probability, that these strings have nonempty intersection if and only if the original

strings had nonempty intersection, the protocol can now determine  $\text{INTER}(X, Y)$  with  $2^k - 1$  extra bits of communication.

Let us first determine the total number of bits exchanged. For this we compute the depth of the recursion. Each step starts with a string of length  $m$  and ends with a string of length at most  $m - \lfloor m/(2^k - 1) \rfloor \leq m(1 - \frac{1}{2^k - 1}) + \frac{2^k - 2}{2^k - 1} \leq \alpha n$  where  $\alpha$  is a constant less than 1 (we are using that  $k$  is a constant). Since Alice and Bob start with a string of length  $n$ , after  $i$  iterations they have a string of length  $\alpha^i n$ . Hence there are  $O(\log n)$  iterations. Each application of  $P$  requires the exchange of at most  $t(n) \log \log n$  bits. The final stage just requires a constant number of bits ( $2^k - 1$ ). It follows that the algorithm in total requires the exchange of  $O(t(n) \log n \log \log n)$  bits.

Let us determine the probability of error. In each step the probability that the string returned by the protocol is correct, i.e., is a string that is indeed the true value of  $f(X_k \cdots X_1, Y_k \cdots Y_1)$  is at least  $(1 - 1/\log^2 n)$ . The probability that all steps are correct is at least  $(1 - 1/\log^2 n)^{c^k \log n}$  for some constant  $c$ . If  $n$  is large enough this is greater than  $\frac{3}{4}$ . ■

**Note:** The proof above is based on the proof that if  $c < 1$  and SAT is  $c \log n$ -membership-comparable [1, 10, 41] then  $P=NP$ . That work has been extended by Sivakumar [51]. It is possible that Sivakumar's techniques can be applied here to obtain stronger results.

**THEOREM 6.2.** *Let  $k$  and  $\epsilon < 1/2^k$  be constants.  $R_\epsilon^{\text{pub}}(\text{ELIM}(\text{IP}^k)) = \Omega(\frac{n}{\log n \log \log n})$ .*

*Proof.* By Lemma 6.3  $R_{1/4}^{\text{pub}}(\text{INTER}) = \Omega(n)$  even when restricted to

$$D = \{(x, y) \in \{0, 1\}^n \times \{0, 1\}^n : \text{for at most one } i \ x_i = y_i\}.$$

On  $D$ ,  $\text{IP} = \text{DISJ}$ . The proof of Theorem 6.1 can now be viewed as a lower bound on  $R_\epsilon^{\text{pub}}(\text{ELIM}(\text{IP}^k))$ . ■

## 7. CONNECTIONS BETWEEN $D(\text{ELIM}(f^2))$ AND $D(\text{ALMOST}(f^M))$

**DEFINITION 7.1.** If  $\sigma, \tau \in \{0, 1\}^*$  are strings of the same length then  $\sigma =^1 \tau$  means that  $\sigma$  and  $\tau$  are either identical or differ on one bit.

**DEFINITION 7.2.** Let  $k, n \in \mathbb{N}$  and  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ .  $\text{ALMOST}(f^k)$  is the relation on  $\{\{0, 1\}^n\}^k \times \{\{0, 1\}^n\}^k \times \{0, 1\}^k$  defined by  $\{(x, y, b) \mid f^k(x, y) =^1 b\}$

Clearly  $D(\text{ALMOST}(f^k)) \leq (k - 1)D(f)$ .

*Conjecture 1.* For any function  $f$ , for any  $k \in \mathbb{N}$ ,  $D(\text{ALMOST}(f^k)) \geq (k - 1)(D(f) - O(1))$ . (Note that for  $k = 2$ , this conjecture is identical to ELC.) ■

Although we believe Conjecture 1 we can obtain consequences from the following weaker conjecture.

*Conjecture 2.* (The *Almost Conjecture (ALC)*) For any function  $f$ , for any  $k \in \mathbb{N}$ ,  $D(\text{ALMOST}(f^k)) \geq \frac{k}{2}D(f)$ . ■

We establish some connections between the complexity of  $\text{ALMOST}(f^k)$  and the complexity of enumeration. We first need a combinatorial lemma.

**DEFINITION 7.3.** If  $X \subseteq \{0, 1\}^m$  and  $1 \leq i_1, \dots, i_k \leq m$  then  $X[i_1, \dots, i_k]$  is the projection of  $X$  onto those coordinates.

**LEMMA 7.1.** Let  $X \subseteq \{0, 1\}^m$ . Let  $b \in X$  be unknown. If  $(\forall i, j)[|X[i, j]| \leq 3]$  then there is an algorithm that requests  $\leq \lceil \frac{m}{2} \rceil - 1$  bits of  $b$  and produces  $b' =^1 b$ .

*Proof.* We show the weaker theorem that there is an algorithm that requests  $\leq \lceil \frac{m}{2} \rceil$  bits of  $b$ . We then show how to modify the algorithm to request  $\leq \lceil \frac{m}{2} \rceil - 1$ .

Let  $U = \{1, \dots, m\}$ ,  $K = G = \emptyset$ . Throughout the algorithm  $U$  will be the set of indices  $i$  such that  $b_i$  is Unknown, nor have we ventured a Guess,  $K$  will be the set of indices  $i$  such that we Know  $b_i$ , and  $G$  will be the set of indices  $i$  such that we have made a Guess for  $b_i$ . At the end of the algorithm we will have  $U = \emptyset$ ,  $K \cup G = \{1, \dots, m\}$ , and at most one of our guesses is wrong.

At all times  $U, K$ , and  $G$  are a partition of  $\{1, \dots, m\}$ . The expression " $K = K \cup \{a, i\}$ " means that wherever  $a, i$  are, they leave those sets and go into  $K$ . Similar conventions apply to other sets. Our final output will be  $b' = b'_1 b'_2 \dots b'_m$ . Initially  $b'_1, \dots, b'_m$  are undefined. They may get set and reset several times; however at the end of the algorithm they will all be defined.

#### ALGORITHM

$U = \{1, \dots, m\}$

$K = \emptyset$

$G = \emptyset$

For  $i=1$  to  $m$

    If  $X[i] = \{c\}$  then  $b'_i = c$ ,  $K = K \cup \{i\}$

End  $i$ -For loop

For  $i = 1$  to  $m$

    For  $j = i + 1$  to  $m$

        If  $X[i, j] \subseteq \{00, 11\}$  then

            ASK( $b_i = ??$ )

            If  $b_i = 1$  then  $b'_i = 1$ ,  $b'_j = 1$ ,  $K = K \cup \{i, j\}$

            If  $b_i = 0$  then  $b'_i = 0$ ,  $b'_j = 0$ ,  $K = K \cup \{i, j\}$

        Else

        If  $X[i, j] \subseteq \{01, 10\}$  then

            ASK( $b_i = ??$ )

            If  $b_i = 1$  then  $b'_i = 1$ ,  $b'_j = 0$ ,  $K = K \cup \{i, j\}$

            If  $b_i = 0$  then  $b'_i = 0$ ,  $b'_j = 1$ ,  $K = K \cup \{i, j\}$

    End  $j$ -For loop

End  $i$ -For loop (Note that if  $|X[i, j]| \leq 2$  then  $i, j \in K$ .)

While  $U \neq \emptyset$

$i = \min(U)$  (The minimum number in  $U$ .)

**Case 1:**  $(\exists j, k \in U \cup G - \{i\})(\exists c_1, c_2 \in \{0, 1\})[0c_1 \notin X[i, j] \wedge 1c_2 \notin X[i, k]]$

        ASK( $b_i = ??$ )

        If  $b_i = 0$  then  $b'_i = 0$ ,  $b'_j = 1 - c_1$ ,  $K = K \cup \{i, j\}$

        If  $b_i = 1$  then  $b'_i = 1$ ,  $b'_k = 1 - c_2$ ,  $K = K \cup \{i, k\}$

(Note that If  $b_i = 0$  then since  $b_i b_j \in X[i, j]$  and  $0c_1 \notin X[i, j]$ , we have  $b_j = 1 - c_1$ .)

Similarly, If  $b_i = 1$  we have  $b_k = 1 - c_2$ .)

**Case 2:**  $(\exists d \in \{0, 1\})(\forall j \in U \cup G - \{i\})[|\{d0, d1\} \cap X[i, j]| \leq 1]$

$$b'_i = 1 - d$$

$$G = G \cup \{i\}$$

(We will show later that either Case 1 or Case 2 holds.)

End While Loop

### END OF ALGORITHM

It is easy to see that the algorithm (a) requests  $\leq \lceil \frac{m}{2} \rceil$  coordinates, (b) sets all the  $b'_i$ , and (c)  $(\forall i \in K)[b_i = b'_i]$ .

**Claim 1:** Either Case 1 or Case 2 occurs.

**Proof:** Assume Case 1 does not occur. We show that Case 2 does. Intuitively Case 1 is saying that there is  $j, k$  such that  $X[i, j]$  and  $X[i, k]$  *exclude* elements of  $\{0, 1\}^2$  that begin with different bits. The negation is that, for all  $j, k$ ,  $X[i, j]$  and  $X[i, k]$  *exclude* elements of  $\{0, 1\}^2$  that begin with the same bit. This bit is the  $d$  in case 2. We proceed more formally. Fix  $j_0 \in U \cup G - \{i\}$ . Since  $|X[i, j_0]| \leq 3$  either  $(\exists c \in \{0, 1\})[0c \notin X[i, j_0]]$  or  $(\exists c \in \{0, 1\})[1c \notin X[i, j_0]]$ . We consider both scenarios.

1)  $(\exists c_1 \in \{0, 1\})[0c_1 \notin X[i, j_0]]$ . (We call it “ $c_1$ ” because it will later play the role of  $c_1$  in Case 1, leading to a contradiction.) We have  $|\{00, 01\} \cap X[i, j_0]| \leq 1$  which looks like Case 2 for  $j_0$  with  $d = 0$ . We show that  $(\forall j \in U \cup G - \{i\})[|\{00, 01\} \cap X[i, j]| \leq 1]$ . Assume, by way of contradiction, that  $(\exists j)[|\{00, 01\} \cap X[i, j]| = 2]$ . Since  $|X[i, j]| \leq 3$  we have  $(\exists c_2 \in \{0, 1\})[1c_2 \notin X[i, j]]$ . Hence

$$(\exists j_0, j \in U \cup G - \{i\})(\exists c_1, c_2 \in \{0, 1\})[0c_1 \notin X[i, j_0] \wedge 1c_2 \notin X[i, j]].$$

This *is* Case 1 with different names for the variables; hence it is really Case 1, a contradiction.

2)  $(\exists c_1 \in \{0, 1\})[1c_1 \notin X[i, j_0]]$ . (We call it “ $c_1$ ” because it will later play the role of  $c_1$  in Case 1, leading to a contradiction.) We have  $|\{10, 11\} \cap X[i, j_0]| \leq 1$  which looks like Case 2 for  $j_0$  with  $d = 1$ . We show that  $(\forall j \in U \cup G - \{i\})[|\{10, 11\} \cap X[i, j]| \leq 1]$ . Assume, by way of contradiction, that  $(\exists j)[|\{10, 11\} \cap X[i, j]| = 2]$ . Since  $|X[i, j]| \leq 3$  we have  $(\exists c_2 \in \{0, 1\})[0c_2 \notin X[i, j]]$ . Hence

$$(\exists j_0, j \in U \cup G - \{i\})(\exists c_1, c_2 \in \{0, 1\})[1c_1 \notin X[i, j_0] \wedge 0c_2 \notin X[i, j]].$$

This *is* Case 1 with different names for the variables; hence it is really Case 1, a contradiction.

### End of Proof of Claim 1

*Claim 2:* There is at most one  $i \in G$  such that  $b_i \neq b'_i$ .

**Proof:** Assume, by way of contradiction, that there exist  $i_1, i_2 \in G$  with  $b_{i_1} \neq b'_{i_1}$  and  $b_{i_2} \neq b'_{i_2}$ . Since  $i_1, i_2 \in G$  we know that (1) they are both the chosen  $i$  in some phase, (2) when they are chosen Case 2 occurs, and (3) they are both always in  $U \cup G$ . Since  $b_{i_1} \neq b'_{i_1}$  when  $i = i_1$  we get Case 2 with  $d = b_{i_1}$ . Since  $i_2 \in U \cup G$  we get  $|\{b_{i_1}0, b_{i_1}1\} \cap X[i_1, i_2]| \leq 1$ . Similarly,  $|\{b_{i_2}0, b_{i_2}1\} \cap X[i_2, i_1]| \leq 1$  which we rewrite as  $|\{0b_{i_2}, 1b_{i_2}\} \cap X[i_1, i_2]| \leq 1$ .

We prove that  $|X[i_1, i_2]| \leq 2$  and hence it must have been dealt with before the while loop even started, which contradicts  $i_1, i_2 \in U$ . Clearly  $b_{i_1}b_{i_2} \in X[i_1, i_2]$ .



Since  $|\{b_{i_1}0, b_{i_1}1\} \cap X[i_1, i_2]| \leq 1$  we get  $b_{i_1}(1 - b_{i_2}) \notin X[i_1, i_2]$ . Since  $|\{0b_{i_2}, 1b_{i_2}\} \cap X[i_1, i_2]| \leq 1$  we get  $(1 - b_{i_1})b_{i_2} \notin X[i_1, i_2]$ . Since  $b_{i_1}(1 - b_{i_2}) \neq (1 - b_{i_1})b_{i_2}$  we have eliminated two elements from  $X[i_1, i_2]$ . Hence  $|X[i_1, i_2]| \leq 2$ .

**End of Proof of Claim 2**

**Claim 3:** The algorithm can be modified to request  $\lceil m/2 \rceil - 1$  bits.

**Proof:** Run the algorithm keeping track of how many queries it makes. If it stops before making  $\lceil m/2 \rceil$  queries then we are done. If it is about to make its  $\lceil m/2 \rceil$ th query then stop it. Each of the first  $\lceil m/2 \rceil - 1$  queries leads to 2 indices being placed in the  $K$  set. Hence  $m - 2$  bits are known for certain. Let the unknown bits be indexed  $i$  and  $j$ . Let  $c_i c_j \notin X[i, j]$ . Set  $b'_i = 1 - c_i$  and  $b'_j = 1 - c_j$ . They cannot both be incorrect since  $b_i b_j \neq c_i c_j$ .

**End of Proof of Claim 3** ■

LEMMA 7.2. *Let  $X \subseteq \{0, 1\}^m$ . Let  $b \in X$  be unknown. Let  $2 \leq k \leq m$ . If  $(\forall i_1, \dots, i_k)[|X[i_1, \dots, i_k]| \leq k + 1]$  then there is an algorithm that requests  $\leq \max\{\lceil \frac{m}{2} \rceil - 1, k - 1\}$  bits of  $b$  and produces  $b' =^1 b$ .*

*Proof.* We prove this by induction on  $k$ . Lemma 7.1 gives the base case of  $k = 2$ . Assume  $k \geq 3$  and that the lemma holds for  $k - 1$ . Assume  $X \subseteq \{0, 1\}^m$  and  $(\forall i_1, \dots, i_k)[|X[i_1, \dots, i_k]| \leq k + 1]$ . If  $(\forall i_1, \dots, i_{k-1})[|X[i_1, \dots, i_{k-1}]| \leq k]$  then we are done by induction. If not then  $(\exists i_1, \dots, i_{k-1})[|X[i_1, \dots, i_{k-1}]| \geq k + 1]$ . Let  $i \in \{1, \dots, m\} - \{i_1, \dots, i_{k-1}\}$ . Since  $|X[i_1, \dots, i_{k-1}, i]| \leq k + 1$  and  $|X[i_1, \dots, i_{k-1}]| \geq k + 1$  for every  $c \in X[i_1, \dots, i_{k-1}]$  exactly one of  $c0$  or  $c1$  is in  $X[i_1, \dots, i_{k-1}, i]$ . Hence if we ask for the values of  $b_{i_1}, \dots, b_{i_{k-1}}$  we can determine the values of all the other  $b_i$ . This takes  $k - 1$  questions. ■

**Note:** In addition to its use here, Lemma 7.2 can also be used to prove the following new theorem: if  $C_k^A$  is  $k + 1$ -enumerable then, for all  $m$ , one can compute  $C_m^A$  with at most one error using  $\max\{\lceil \frac{m}{2} \rceil, k - 1\}$  of the queries given. Further connections between enumerability and computing with errors might be interesting. (See any of [2, 7, 11, 10, 22] for the relevant definitions. Note that the theorem holds for enumerability in the complexity case and for strong enumerability in the computability case.)

**Note:** Lemma 7.2 is optimal in two ways.

1. No algorithm that asks  $\frac{m}{2}$  bit queries can achieve perfect accuracy. In fact, no algorithm that asks  $m - 1$  queries can achieve perfect accuracy. Let  $X$  be the  $m + 1$  vectors  $\{0^m\} \cup \{0^i 10^{m-i-1} \mid 0 \leq i \leq m - 1\}$ . This set satisfies the premise of Lemma 7.2; however, if  $m - 1$  bit queries yield the answer 0 then the string  $b$  is still unknown.

2. No algorithm that asks  $\frac{m}{2} - 2$  bit queries can obtain a string with at most one error. Let  $m$  be even. Let  $X = \{0^{m/2} 1^{m/2}\} \cup \{0^a 10^{m/2-a-1} 1^a 0 1^{m/2-a-1} \mid 0 \leq a \leq m/2 - 1\}$ . (See figure below for  $m = 8$  case.) One can check that, for all  $i, j$ ,  $|X[i, j]| \leq 3$  (there are four cases). Note that for every  $1 \leq i \leq m$  either all the vectors except one have  $b_i = 0$  or all but one have  $b_i = 1$ . If an adversary answers each bit query with the bit that appears most often in that column then every query the algorithm makes eliminates at most one vector. Hence  $m/2 - 2$

queries will leave at least three candidates. Two of the candidates must differ in four places (whichever two are not  $0^{m/2}1^{m/2}$ ). There is no vector that is hamming distance 1 away from both of them; hence an adversary can claim that whatever answer given is wrong in at least two places.

$$\begin{array}{cccccccc} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{array}$$

**THEOREM 7.1.** *Let  $k, m, n \in \mathbb{N}$  and let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Then*

$$D(\text{ALMOST}(f^m)) \leq \binom{m}{k} D(\text{ENUM}(k+1, f^k)) + \max\{\lceil \frac{m}{2} \rceil - 1, k-1\} D(f).$$

*Proof.* We exhibit a protocol for  $\text{ALMOST}(f^m)$  that will invoke a protocol for  $\text{ENUM}(k+1, f^k)$   $\binom{m}{k}$  times, and a protocol for  $f$  at most  $\max\{\lceil \frac{m}{2} \rceil - 1, k-1\}$  times.

- 1) Alice has  $x = x_1x_2 \cdots x_m$ , Bob has  $y = y_1y_2 \cdots y_m$ .
- 2) For all  $i_1 < \cdots < i_k \subseteq \{1, \dots, m\}$  Alice and Bob compute a set of  $k+1$  candidates for  $f^k(x_{i_1}x_{i_2} \cdots x_{i_k}, y_{i_1}y_{i_2} \cdots y_{i_k})$ . This invokes a protocol for  $f^{k+1}$   $\binom{m}{k}$  times.
- 3) Let  $X \subseteq \{0, 1\}^m$  be the set of candidates for  $f^m(x, y)$  that are consistent with the information gathered in step 2. That is,  $b \in X$  iff for every  $i_1, \dots, i_k$  the string  $b_{i_1} \cdots b_{i_k}$  was output when Alice and Bob enumerated  $f^k(x_{i_1} \cdots x_{i_k}, y_{i_1} \cdots y_{i_k})$ . Note that  $X$  is nonempty since  $f(x_1, y_1) \cdots f(x_m, y_m) \in X$ . Note that Alice and Bob both know  $X$  and that  $X$  satisfies Lemma 7.2.
- 4) Alice and Bob perform the algorithm in Lemma 7.2 with  $X$  as in the previous step and  $b = f^k(x, y)$ . Whenever they need to find a particular bit  $f(x_i, y_i)$ , they invoke a protocol for  $f$ . This will happen at most  $\max\{\lceil \frac{m}{2} \rceil - 1, k-1\}$  times. ■

**COROLLARY 7.1.** *Let  $m, n \in \mathbb{N}$  and let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Then*

$$D(\text{ALMOST}(f^m)) \leq \binom{m}{2} D(\text{ELIM}(f^2)) + (\lceil \frac{m}{2} \rceil - 1) D(f).$$

**COROLLARY 7.2.** *Let  $m, n \in \mathbb{N}$  and let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Assume ALC holds for some even  $m$ . Then  $D(\text{ELIM}(f^2)) \geq \Omega(D(f))$ .*

*Proof.* Since ALC holds for  $m$  we have

$$\frac{m}{2} D(f) \leq D(\text{ALMOST}(f^m)) \leq \binom{m}{2} D(\text{ELIM}(f^2)) + (\frac{m}{2} - 1) D(f).$$

Hence  $D(f) \leq \binom{m}{2} D(\text{ELIM}(f^2))$ , so  $D(\text{ELIM}(f^2)) = \Omega(D(f))$ . ■

## 8. CONNECTIONS BETWEEN $N(\text{ENUM}(E, f^k))$ , $N(F)$ , AND $R_{\epsilon}^{\text{pub}}N(F)$

We show that  $N(\text{ENUM}(k, f^k))$  is at least as big as either the nondeterministic randomized complexity of  $f$  or the nondeterministic complexity of  $f$  (modulo a log term).

The proof of the next lemma uses ideas from the proof that p-superterse sets are in P/poly from [2].

LEMMA 8.1. *Let  $e, k, n \in \mathbb{N}$  and let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Either*

$$N(\text{ENUM}(e-1, f^{k-1})) \leq N(\text{ENUM}(e, f^k)) + \log(kn) + O(1)$$

or

$$R_{1/4}^{\text{pub}}N(f) \leq N(\text{ENUM}(e, f^k)).$$

*Proof.* Assume  $N(\text{ENUM}(e, f^k)) \leq t$  via protocol  $P$ . Note that the output of  $P$  is a set  $A$  with  $|A| \leq e$ . We denote the output of  $P$  by  $A$ , and we let  $A_0 = A \cap \{0, 1\}^{k-1}0$  and  $A_1 = \{0, 1\}^{k-1}1$ .

We will try to construct a set  $Z \subseteq \{0, 1\}^n \times \{0, 1\}^n$  such that the following hold:

1.  $|Z| \leq O(kn)$ , and
2. For all  $x_1 \cdots x_{k-1} \in \{0, 1\}^{k-1}$ , for all  $y_1 \cdots y_{k-1} \in \{0, 1\}^{k-1}$ , there exists  $(u, v) \in Z$  such that for all paths of  $P(x_1 \cdots x_{k-1}u, y_1 \cdots y_{k-1}v)$  that output an answer  $A$ , both  $A_0 \neq \emptyset$  and  $A_1 \neq \emptyset$ .

If we succeed then the following nondeterministic protocol shows that

$$N(\text{ENUM}(e-1, f^{k-1})) \leq t + \log(kn) + O(1).$$

The protocol assumes that both Alice and Bob know the set  $Z$  and have agreed ahead of time on some ordering of it. They also know, for every  $(u, v) \in Z$ , the value of  $f(u, v)$ . This is fair since these protocols are non-uniform.

### PROTOCOL

1. Alice has  $x_1 \cdots x_{k-1}$  and Bob has  $y_1 \cdots y_{k-1}$ .
2. The protocol nondeterministically picks a number  $i$  such that  $1 \leq i \leq |Z|$ . This requires  $\log |Z|$  bits.
3. Alice and Bob both find  $(u, v)$ , the  $i$ th element of  $Z$  (according to Alice and Bob's order on  $Z$ ).
4. Alice and Bob run nondeterministic protocol  $P$  with Alice knowing  $x_1 \cdots x_{k-1}u$  and Bob knowing  $y_1 \cdots y_{k-1}v$ . This requires  $\leq t$  bits.
5. If the path outputs "I DON'T KNOW" then that path of the protocol outputs "I DON'T KNOW". If the output is a set  $A$  then we know that  $A_0 \neq \emptyset$  and  $A_1 \neq \emptyset$ . We can assume Alice and Bob both know  $f(u, v) = b$ . Hence they know  $A_b$  contains the correct value of  $f(x, y)$ . The protocol outputs  $A' = \{b_1 b_2 \cdots b_{k-1} | b_1 b_2 \cdots b_{k-1} b \in A_b\}$ . Since  $A_{1-b} \neq \emptyset$  and  $|A| \leq e$ , we know  $|A_b| \leq e-1$ .

**END OF PROTOCOL**

Note that the protocol takes  $t + \log |Z| \leq t + kn + O(1)$  bits, and enumerates  $\leq e - 1$  candidates. Hence  $N(\text{ENUM}(e - 1, f^k)) \leq t + kn + O(1)$ .

Two things may happen in the construction of  $Z$ . If the construction succeeds then we are done, as the protocol above works. If the construction fails then the very reason for the failure will yield a randomized protocol that shows  $R_{1/4}^{\text{pub}}N(f) \leq t$ .

**Definition:** If  $(u, v) \in \{0, 1\}^n \times \{0, 1\}^n$  then  $\mathbf{advisees}(u, v)$  is the set of all  $(x_1 \cdots x_{k-1}, y_1 \cdots y_{k-1})$  such that if the nondeterministic protocol  $P$  is ran on  $(x_1 \cdots x_{k-1}u, y_1 \cdots y_{k-1}v)$  then some leaf outputs an  $A$  such that  $A \cap \{0, 1\}^{k-1}0 \neq \emptyset$  and  $A \cap \{0, 1\}^{k-1}1 \neq \emptyset$ .

**CONSTRUCTION OF  $Z$** 

$$T := (\{0, 1\}^n \times \{0, 1\}^n)^{k-1}$$

$$Z := \emptyset$$

While  $(\exists u, v)[|\mathbf{advisees}(u, v)| \geq \frac{1}{8}|T|]$

choose such a  $(u, v)$

$$T := T - \mathbf{advisees}(u, v)$$

$$Z := Z \cup \{(u, v)\}$$
**END OF CONSTRUCTION**

Note that after the  $i$ th iteration  $|T| \leq (\frac{7}{8})^i 2^{2(k-1)n}$ . Hence there are at most  $O(kn)$  iterations. Since the number of elements in  $Z$  is bounded by the number of iterations,  $|Z| \leq O(kn)$ .

If the construction ends with  $T = \emptyset$  then it succeeds and the protocol above shows

$N(\text{ENUM}(e - 1, f^{k-1})) \leq t + \log(kn) + O(1)$ . If the construction does not succeed then let  $T$  be the set  $T$  at the end. Note that, for every  $u, v$ , there is a set  $W \subseteq T$  such that  $|W| \geq \frac{7}{8}|T|$  and

$$(\forall (x_1 \cdots x_{k-1}, y_1 \cdots y_{k-1}) \in W)$$

if nondeterministic protocol  $P$  is run on  $(x_1 \cdots x_{k-1}u, y_1 \cdots y_{k-1}v)$  then one of the leaves outputs  $A$  such that  $A_0$  or  $A_1$  is empty.

We can use this to devise a protocol for  $f$  which uses public coins over  $T$ .

**PROTOCOL**

1. Alice has  $u$  and Bob has  $v$ .
2. The protocol randomly and publicly picks an element of  $T$ . Let the element be

$$(x_1 \cdots x_{k-1}, y_1 \cdots y_{k-1}).$$

3. Alice and Bob run nondeterministic protocol  $P$  with Alice knowing  $x_1 \cdots x_{k-1}u$  and Bob knowing  $y_1 \cdots y_{k-1}v$ .

4. When they get the answer  $A$  they check and see if either of  $A_0$  or  $A_1$  is empty. If neither is empty then they refuse to give an answer. If  $A_b$  is empty then they output  $1 - b$ .

**END OF PROTOCOL**

The protocol transmit as many bits as  $P$  does, which is  $t$ . The protocol is in error with probability  $\frac{1}{8}$ . The referee used elements in  $T$  to randomize, not elements in some set of strings of bits. By adding a sufficiently number of elements to  $T$  (all of which return I DON'T KNOW) to obtain a set of size a poer of two, one increases the probability of error to at most  $\frac{1}{4}$ . Hence we have  $R_{1/4}^{\text{pub}}N(f) \leq t$ . ■

**THEOREM 8.1.** *Let  $k, n \in \mathbb{N}$ ,  $e \leq k$ ,  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Either*

$$1. R_{1/4}^{\text{pub}}N(f) \leq N(\text{ENUM}(k, f^k)) + k \log(kn) \text{ or}$$

$$2. N(f) \leq N(\text{ENUM}(k, f^k)) + O(k \log(kn)), \text{ coN}(f) \leq N(\text{ENUM}(k, f^k)) + O(k \log(kn)),$$

and  $D(f) \leq O((N(\text{ENUM}(k, f^k)) + k \log(kn))^2)$ .

*Proof.* By Lemma 8.1 we have either

$$R_{1/4}^{\text{pub}}N(f) \leq N(\text{ENUM}(k, f^k))$$

or

$$N(\text{ENUM}(k-1, f^{k-1})) \leq N(\text{ENUM}(k, f^k)) + \log(kn) + O(1).$$

In the former case we are done. In the latter case we apply Lemma 8.1 with  $k-1, k-1$  to get either

$$R_{1/4}^{\text{pub}}N(f) \leq N(\text{ENUM}(k-1, f^{k-1})) \leq N(\text{ENUM}(k, f^k)) + \log(kn) + O(1)$$

or

$$\begin{aligned} N(\text{ENUM}(k-2, f^{k-2})) &\leq N(\text{ENUM}(k-1, f^{k-1})) + \log(kn) + O(1) \\ &\leq N(\text{ENUM}(k, f^k)) + 2(\log(kn) + O(1)). \end{aligned}$$

We repeat the process until we obtain (in the worst case) either

$$R_{1/4}^{\text{pub}}N(f) \leq N(\text{ENUM}(k, f^k)) + k \log(kn)$$

or

$$\begin{aligned} N(\text{ENUM}(1, f^k)) &\leq N(\text{ENUM}(k, f^k)) + (k-1)(\log(kn) + O(1)) \\ &= N(\text{ENUM}(k, f^k)) + O(k \log(kn)). \end{aligned}$$

From the definition of a nondeterministic protocol for a relation we know that  $N(f) \leq N(\text{ENUM}(1, f^k))$  and  $\text{coN}(f) \leq N(\text{ENUM}(1, f^k))$  Hence

$$N(f) \leq N(\text{ENUM}(k, f^k)) + O(k \log(kn))$$

and

$$\text{coN}(f) \leq N(\text{ENUM}(k, f^k)) + O(k \log(kn)).$$

By Theorem 2.11 of [34], (originally proven in [24])  $D(f) \leq O(N(f)\text{coN}(f))$ . Hence  $D(f) \leq O((N(\text{ENUM}(k, f^k)) + k \log(kn))^2)$ . ■

### 9. IF $D(\text{ENUM}(E, f^k)) \leq T$ THEN ...

We present two theorems with the hypothesis that  $D(\text{ENUM}(e, f^k))$  is "small."

LEMMA 9.1 ([6, 13, 42]). *Let  $X \subseteq \{0, 1\}^k$  such that  $|X| \leq k$ . Let  $b \in X$  be unknown. There is an algorithm that requests  $\leq k - 1$  bits of  $b$  that produces  $b$ .*

THEOREM 9.1. *Let  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . For all  $k$ ,*

$$D(f^k) \leq D(\text{ENUM}(k, f^k)) + (k - 1)D(f).$$

*Proof.*

We present a protocol for  $D(f^k)$  that invokes a protocol for  $\text{ENUM}(k, f^k)$  once and a protocol for  $f$   $k - 1$  times.

1) Alice has  $x = x_1x_2 \cdots x_k$ , Bob has  $y = y_1y_2 \cdots y_k$ .

2) Alice and Bob compute a set of  $k$  candidates for  $f^k(x_1x_2 \cdots x_k, y_1y_2 \cdots y_k)$ . This invokes one call to a protocol for  $\text{ENUM}(k, f^k)$ .

3) Let  $X \subseteq \{0, 1\}^k$  be the set of candidates for  $f^k(x, y)$  computed in step 2. Note that  $X$  satisfies the premise of Lemma 9.1.

4) Alice and Bob perform the algorithm in Lemma 9.1 with  $X$ . Whenever they need to find a particular bit  $f(x_i, y_i)$ , they invoke a protocol for  $f$ . This will happen at most  $k - 1$  times. ■

COROLLARY 9.1. *If DSC holds at  $k$  then  $D(\text{ENUM}(k, f^k)) \geq D(f) - O(k)$ .*

If you can just eliminate one possibility, does this imply that you can eliminate more, perhaps for higher values of  $k$ ? The next theorem shows how to do this. The proof is similar to Lemma 5.1 of [5], Lemma 19 in [7] or Theorem 4.4.9 in [22].

DEFINITION 9.1. Let  $k, m \in \mathbb{N}$  such that  $1 \leq k \leq m$ .  $S(m, k) = \sum_{i=0}^{k-1} \binom{m}{i}$ .

LEMMA 9.2 ([4, 7, 18, 47, 53]). *Let  $k, m \in \mathbb{N}$  such that  $1 \leq k \leq m$ , and let  $X \subseteq \{0, 1\}^m$  be such that for any  $k$  coordinates, if you project  $X$  down to those  $k$  coordinates, the resulting set has size  $\leq 2^k - 1$ . Then  $|X| \leq S(m, k)$ .*

THEOREM 9.2. *Let  $k, m, n \in \mathbb{N}$ ,  $k < m$ , and  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Then*

$$D(\text{ENUM}(S(m, k), f^m)) \leq \binom{m}{k} D(\text{ELIM}(f^k)).$$

*Proof.* Suppose that  $D(\text{ELIM}(f^k)) = t$  via protocol  $P$ . Alice is given  $x \in \{\{0, 1\}^n\}^m$  and Bob is given  $y \in \{\{0, 1\}^n\}^m$ . They can compute  $S(m, k)$  candidates for  $f^m(x, y)$  as follows. For each  $k$ -subset  $\{i_1, \dots, i_k\}$  of  $\{1, \dots, m\}$  they run protocol  $P$  on  $(x_{i_1} \cdots x_{i_k}, y_{i_1} \cdots y_{i_k})$ . This takes  $\binom{m}{k}t$  bits. Let  $X$  be the set of elements of  $\{0, 1\}^m$  that are consistent with the information gathered. By Lemma 9.2  $|X| \leq S(m, k)$ . ■

COROLLARY 9.2.  $D(\text{ELIM}(f^2)) \geq \frac{D(\text{ENUM}(m+1, f^m))}{\binom{m}{2}}$ .

## 10. THE COMMUNICATION COMPLEXITY OF SELECTION

We prove lower bounds on  $D(\text{SELECT}(f^2))$  and then note that the proof can easily be modified for  $N(\text{SELECT}(f^2))$ . We then relate the complexity of  $D(\text{SELECT}(f^2))$

to DSC. The proof of the next theorem uses ideas from the proof in [30] that P-selective sets are in P/poly.

THEOREM 10.1. *Let  $n \in \mathbb{N}$  and  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Then*

$$D(\text{SELECT}(f^2)) \geq N(f) - \log(n) - 1$$

and

$$D(\text{SELECT}(f^2)) \geq \text{coN}(f) - \log(n) - 1.$$

*Proof.* Let  $D(\text{SELECT}(f^2)) = t$  via protocol  $P$ . We use  $P$  to build a non-deterministic protocol which will show  $N(f) \leq t + \log(n) + 1$ . We will then know  $\text{coN}(f) \leq t + \log(n) + 1$  since  $D(\text{SELECT}(f^2)) \leq t$  implies  $D(\text{SELECT}(\bar{f}^2)) \leq t$  by a protocol that runs the deterministic protocol for  $\text{SELECT}(f^2)$ , finds answer  $i$ , and outputs  $2 - i$ .

Let  $S = \{(x, y) \mid f(x, y) = 1\}$ . We will denote elements of  $S$  by  $xy$  instead of  $(x, y)$ . Let  $x_1y_1, x_2y_2 \in S$ . Note the following.

1. From  $P(x_1x_2, y_1y_2) = 1$  and  $f(x_2, y_2) = 1$  one can conclude  $f(x_1, y_1) = 1$ .
2. From  $P(x_1x_2, y_1y_2) = 2$  and  $f(x_1, y_1) = 1$  one can conclude  $f(x_2, y_2) = 1$ .

We will try to find an  $x_1y_1 \in S$  (or an  $x_2y_2 \in S$ ) such that there are many  $x_2y_2 \in S$  ( $x_1y_1 \in S$ ) with  $P(x_1x_2, y_1y_2) = 2$ . ( $P(x_1x_2, y_1y_2) = 1$ ). We will repeat this procedure until every  $xy \in S$  has a witness.

Let  $T \subseteq S$ . Let  $H_T : T \times T \rightarrow T \times \{1, 2\}$  be defined by

$$H_T(x_1y_1, x_2y_2) = \begin{cases} (x_1y_1, 1) & \text{if } P(x_1x_2, y_1y_2) = 2; \\ (x_2y_2, 2) & \text{if } P(x_1x_2, y_1y_2) = 1. \end{cases}$$

$H_T$  has domain of size  $|T|^2$  and codomain of size  $2|T|$ . Hence there exists  $(xy, i) \in T \times \{1, 2\}$  such that at least  $|T|/2$  elements of  $T \times T$  map to  $(xy, i)$ . Let  $w_T$  be some such element. If  $i = 1$  then the elements that map to  $w_T$  are of the form  $(xy, x_2y_2)$  and we let  $I_T = \{x_2y_2 \in T \mid H_T(xy, x_2y_2) = (xy, 1)\}$ . If  $i = 2$  then the elements that map to  $w_T$  are of the form  $(x_1y_1, xy)$  and we let  $I_T = \{x_1y_1 \in T \mid H_T(x_1y_1, xy) = (xy, 2)\}$ .

#### CONSTRUCTION OF WITNESSES

$$T_0 = S$$

$$i = 0$$

While  $T_i \neq \emptyset$

$$w_i = w_{T_i}$$

$$T_{i+1} = T_i - I_{T_i}$$

$$i = i + 1$$

#### END OF CONSTRUCTION

By induction  $|T_i| \leq |S|/2^i \leq 2^{2n-i}$ . Hence there are at most  $2n$  iterations. Let  $W$  be the set of the  $w_i$ 's. Note that to specify an element of  $W$  requires at most  $\log(2n) = \log(n) + 1$  bits.

Before the protocol begins both Alice and Bob know the contents of  $W$  and have agreed on an ordering of it.

1. Alice has  $x$ , Bob has  $y$ .
2. The protocol nondeterministically picks an element  $(x'y', i) \in W$ . This takes  $\log(n) + 1$  bits.

3. If  $i = 1$  then Alice and Bob run  $P(x'x, y'y)$ . If it outputs 2 then accept, otherwise reject. If  $i = 2$  then Alice and Bob run  $P(xx', yy')$ . If it outputs 1 then accept, otherwise reject. In either case this takes  $t$  bits.

If  $f(x, y) = 1$  then some  $(x'y', i)$  will work. If  $f(x, y) = 0$  then no  $(x'y', i)$  will work. Hence this is a nondeterministic protocol for  $f$ . It only used  $t + \log(n) + 1$  bits. ■

THEOREM 10.2. *Let  $n \in \mathbb{N}$  and  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ . Then*

$$N(\text{SELECT}(f^2)) \geq N(f) - \log(n) - 1,$$

$$N(\text{SELECT}(f^2)) \geq \text{coN}(f) - \log(n) - 1,$$

and

$$D(f) \leq O(N(f)\text{coN}(f)) \leq O((N(\text{SELECT}(f^2)) + \log n)^2).$$

*Proof.* Let  $N(\text{SELECT}(f^2)) = t$  via protocol  $P$ . We use  $P$  to build a nondeterministic protocol which will show  $N(f) \leq t + \log(n) + 1$ . We will then know  $\text{coN}(f) \leq t + \log(n) + 1$  since  $N(\text{SELECT}(f^2)) \leq t$  implies  $N(\text{SELECT}(\bar{f}^2)) \leq t$  by a protocol that runs the nondeterministic protocol for  $N(\text{SELECT}(f^2))$ , and if the answer would have been  $i$ , and outputs  $2 - i$ . Let

$$H_T(x_1y_1, x_2y_2) = \begin{cases} (x_1y_1, 1) & \text{if some leaf of } P(x_1x_2, y_1y_2) \text{ outputs 2;} \\ (x_2y_2, 2) & \text{otherwise.} \end{cases}$$

From this point on the proof proceeds similar to that of Theorem 10.1.

By Theorem 2.11 of [34], (originally proven in [24])  $D(f) \leq O(N(f)\text{coN}(f))$ . Hence

$$D(f) \leq O(N(f)\text{coN}(f)) \leq O((N(\text{SELECT}(f^2)) + \log n)^2).$$

■

Since  $N(\text{SELECT}(f^2)) \geq N(\text{ELIM}(f^2))$  and, by Theorem 3.2  $N(\text{ELIM}(\text{DISJ}^2)) \geq n - O(\log n)$ , we have

$$N(\text{SELECT}(\text{DISJ}^2)) \geq n - O(\log n).$$

By Theorem 10.2 and  $N(\text{DISJ}) \geq n + 1$  (the fooling set arguments of [34] that show  $D(\text{DISJ}) \geq n + 1$  easily show that  $N(\text{DISJ}) \geq n + 1$ ) we have

$$N(\text{SELECT}(\text{DISJ}^2)) \geq n - \log(n).$$

Using Kolmogorov complexity [36] we can improve this to

$$N(\text{SELECT}(\text{DISJ}^2)) \geq n - O(1).$$



We give a brief informal introduction to Kolmogorov complexity; see [36] for more precise information.  $C : \{0, 1\}^* \rightarrow \mathbb{N}$  maps each binary string  $x$  to the size of the shortest program that, on input 0, prints  $x$ . Since  $x$  can always be printed out by a program that says “PRINT  $x$ ”, which is of length  $|x| + O(1)$ , we always have  $C(x) \leq |x| + O(1)$ . The value of  $C$  may be much shorter. For example  $0^n$  can be printed out by the program “PRINT 0  $n$  times” which has size  $\log n + O(1)$ , hence  $C(0^n) \leq \log n + O(1)$ . A counting argument shows that, for all  $n$ , there are many  $x \in \{0, 1\}^n$  such that  $C(x) \geq |x|$ . The idea is that there are many more strings than programs.

The definition of  $C$  can be extended. Let  $C(x|y_1, \dots, y_k)$  be the size of the shortest program that, on input  $y_1, \dots, y_k$ , prints out  $x$ . A counting argument shows that, given  $n, y_1, \dots, y_k$ , there are many strings  $x$  of length  $n$  such that  $C(x|y_1, \dots, y_k) \geq |x|$ . The idea is that there are many more strings than programs.

**THEOREM 10.3.**  $N(\text{SELECT}(\text{DISJ}^2)) \geq n - O(1)$ .

*Proof.* Assume that  $N(\text{SELECT}(\text{DISJ}^2)) = t$  via protocol  $P$ . Let  $x_1$  and  $x_2$  be strings of length  $n$  such that  $C(x_1|P, x_2) \geq n$  and  $C(x_2|P, x_1) \geq n$ . Let Alice have  $x_1x_2$  and Bob have  $\overline{x_1x_2}$ . (Recall that  $\overline{z}$  means take  $z$  and replace the 0's with 1's and the 1's with 0's.) Let  $b = b_1b_2 \dots b_t$  be a sequence of bits that form a possible path to a real leaf  $L$  that Alice and Bob could go down. (Note that  $b$  includes both the nondeterministic choice bits and the communication bits by the definition of nondeterministic protocols.) Assume that the leaf outputs 2 (the 1 case is similar).

We show that  $C(x_1|x_2, P, b) \leq n + O(1)$ . This shows  $t \geq n - O(1)$  since  $C(x_1|P, x_2) \geq n$ . Recovery algorithm: Enumerate all  $x$  such that  $P(xx_2, \overline{xx_2})$  could end up at leaf  $L$ . There will only be one such  $x$  (proven below) and that one  $x$  is  $x_1$ .

Assume that  $x$  and  $x'$ , get enumerated in the above recovery algorithm. Since  $P(xx_2, \overline{xx_2})$  and  $P(x'x_2, \overline{x'x_2})$  both end up at  $L$ , by a basic theorem in communication complexity [34, Proposition 1.14], the inputs  $(xx_2, \overline{xx_2})$  and  $(x'x_2, \overline{x'x_2})$  will end up at  $L$ . Hence  $\text{DISJ}(x, \overline{x'})\text{DISJ}(x_2, \overline{x_2}) \neq 01$ . Since  $\text{DISJ}(x_2, \overline{x_2}) = 1$  we have  $\text{DISJ}(x, \overline{x'}) = 1$ . We also get  $\text{DISJ}(x', \overline{x})\text{DISJ}(x_2, \overline{x_2}) \neq 01$ . Since  $\text{DISJ}(x_2, \overline{x_2}) = 1$  we have  $\text{DISJ}(x', \overline{x}) = 1$ . Since  $x$  and  $\overline{x'}$  are disjoint sets and  $x'$  and  $\overline{x}$  are disjoint sets,  $x = x'$ . ■

**THEOREM 10.4.**  $D(f^3) \leq 2D(f) + 3D(\text{SELECT}(f^2))$ .

*Proof.* For this theorem we use the definition  $(x_1x_2, y_1y_2, b_1b_2) \in \text{SELECT}(f^2)$  if  $f(x_1, y_1) = b_1$  or  $f(x_2, y_2) = b_2$  and  $b_1 \neq b_2$ . This is easily seen to be equivalent to the usual definition. We present a protocol for  $f^3$  which transmits at most  $2D(f) + 3D(\text{SELECT}(f^2))$  bits. Assume Alice has  $x_1x_2x_3$  and Bob has  $y_1y_2y_3$ . For  $i, j \in \{1, 2, 3\}$  and  $i < j$ , Alice with inputs  $x_i, x_j$  and Bob with inputs  $y_i, y_j$  run the protocol for  $\text{SELECT}(f^2)$  and produce output  $b_{i,j}^1, b_{i,j}^2$ . For each  $i$ , observe that Alice and Bob predict  $f(x_i, y_i)$  exactly twice while running  $\text{SELECT}(f^2)$  thrice. Since the output of the  $\text{SELECT}(f^2)$  protocol is limited to 01 or 10, it must be the case that for some  $i$ , the two predictions of Alice and Bob on  $f(x_i, y_i)$  do not match. Without loss of generality, let us assume that the mismatch happens for  $i = 1$ . Now Alice and Bob compute  $f(x_1, y_1)$  by exchanging at most  $D(f)$  bits. Without loss of generality, let us assume that  $b_{1,2}^1 \neq f(x_1, y_1)$ . Knowing this, Alice and Bob will

correctly conclude that  $f(x_2, y_2) = b_{2,1}^2$ . Finally, Alice and Bob compute  $f(x_3, y_3)$  by exchanging at most  $D(f)$  bits. ■

COROLLARY 10.1. *If DSC holds then  $D(\text{SELECT}(f^2)) \geq \frac{1}{3}D(f) - O(1)$ .*

## 11. OPEN PROBLEMS

The most important open problem is to resolve ENC. As a first step, it is important to resolve ELC. We restate it along with some weaker versions:

1. If  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  then  $D(\text{ELIM}(f^k)) \geq \Omega\left(\frac{D(f)}{\log D(f)}\right)$ ,
2. If  $f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$  then  $D(\text{ELIM}(f^k)) = \Omega(\sqrt{D(f)})$ ,
3. For all monotone functions  $f$ ,  $D(\text{ELIM}(f^k)) = D(f) - O(1)$  (or weaker lower bounds).
4. For all graph properties  $f$ ,  $D(\text{ELIM}(f^k)) = D(f) - O(1)$  (or weaker lower bounds).
5. For all properties  $f$  invariant under some group  $G$ ,  $D(\text{ELIM}(f^k)) = D(f) - O(1)$  (or weaker lower bounds).

## 12. ACKNOWLEDGMENTS

We want to thank László Babai, Richard Chang, Steve Fenner, Anna Gal, James Glenn, Steve Homer, Samir Khuller, Vladik Kreinovich, Clyde Kruskal, Eyal Kushilevitz, Luc Longpre, Jacob Lurie, Noam Nisan, Michael Saks, and Frank Stephan for helpful discussions. In particular Frank Stephan helped on an earlier version of Theorem 3.2, László Babai helped on Theorem 4.1 and 4.2, Anna Gal helped on Note 7, Samir Khuller and Clyde Kruskal helped with Example 4.2, Vladik Kreinovich and Luc Longpre helped on the counterexample to the primitive form of DSC in the introduction. We want to thank an anonymous COMPLEXITY-2000 committee member or subreferee who pointed out some subtleties about DSC. We would like to thank the journal-version referees for many helpful comments and clarifications. Lastly, the third author wants to thank Eyal Kushilevitz and Noam Nisan whose book [34] introduced him to the field and inspired some of this work.

## REFERENCES

1. M. Agrawal and V. Arvind. Quasi-linear truth-table reductions to p-selective sets. *Theoretical Computer Science*, 158(1–2):361–370, 20 May 1996.
2. A. Amir, R. Beigel, and W. Gasarch. Some connections between bounded query classes and non-uniform complexity. Technical Report TR 00-0024, 2000. See [www.ecc.uni-trier.de/eccc/](http://www.ecc.uni-trier.de/eccc/).
3. L. Babai, P. Frankl, and J. Simon. Complexity classes in communication complexity theory (preliminary version). In *Proceedings of the 27th Annual IEEE Symposium on Foundations of Computer Science*, Toronto, Ontario, Canada, pages 337–347, 1986.
4. R. Beigel. *Query-Limited Reducibilities*. PhD thesis, Stanford University, 1987. Also available as Report No. STAN-CS-88-1221.
5. R. Beigel. A structural theorem that depends quantitatively on the complexity of SAT. In *Proceedings of the 2nd IEEE Conference on Structure in Complexity Theory*, Cornell NY, pages 28–32. IEEE Computer Society Press, June 1987.
6. R. Beigel. Bounded queries to SAT and the Boolean hierarchy. *Theoretical Computer Science*, 84:199–223, 1991.

7. R. Beigel, W. Gasarch, J. Gill, and J. Owings. Terse, Superterse, and Verbose sets. *Information and Computation*, 103(1):68–85, Mar. 1993.
8. R. Beigel, W. Gasarch, M. Kummer, G. Martin, T. McNicholl, and F. Stephan. The complexity of  $\text{ODD}_n^A$ . *Journal of Symbolic Logic*, pages 1–18, 2000. Parts of this appeared in MFCS96 with title *On the query complexity of sets*.
9. R. Beigel and T. Hirst. One help bit doesn't help. In *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing*, Dallas TX, 1998.
10. R. Beigel, M. Kummer, and F. Stephan. Approximable sets. *Information and Computation*, 120(2):304–314, 1995.
11. R. Beigel, M. Kummer, and F. Stephan. Quantifying the amount of verboseness. *Information and Computation*, 118(1):73–90, Apr. 1995. Prior version in: *Lecture Notes for Computer Science ("Logic at Tver")*, Vol. 620, pp. 21–32, 1992. This may be available at [springerlink.com](http://springerlink.com) if you or your institution is registered.
12. B. Bollobás. *Extremal Graph Theory*. Academic Press, New York, 1978.
13. J. Bondy and U. Murty. *Graph Theory with applications*. American Elsevier, New York, 1977.
14. N. H. Bshouty. On the extended direct sum conjecture. In *STOC89*, pages 177–185, 1989.
15. N. H. Bshouty. On the direct sum conjecture in the straight line model. *Journal of Complexity*, 14, 1998.
16. J. Cai and L. A. Hemachandra. Enumerative counting is hard. *Information and Computation*, 82(1):34–44, July 1989.
17. J. Cai and L. A. Hemachandra. A note on enumerative counting. *Information Processing Letters*, 38(4):215–219, 1991.
18. S. Clarke, J. Owings, and J. Spriggs. Trees with full subtrees. In *Proc. of the 6th Southeastern Conference on Combinatorics, Graph Theory, and Computing*, pages 169–172, 1975.
19. R. Downey and M. Fellows. *Parameterized Complexity*. Springer-Verlag, 1999.
20. T. Feder, E. Kushilevitz, M. Naor, and N. Nisan. Amortized communication complexity. *SIAM J. Comput.*, 24(4):736–750, Aug. 1995.
21. E. Feig and S. Winograd. On the direct sum conjecture. *Linear Algebra and its Applications*, 63:193–219, 1984.
22. W. Gasarch and G. Martin. *Bounded Queries in Recursion Theory*. Progress in Computer Science and Applied Logic. Birkhäuser, Boston, 1999.
23. A. Hajnal, W. Maass, and G. Turán. The communication complexity of graph properties. In *Proceedings of the Twentieth Annual ACM Symposium on the Theory of Computing*, Chicago IL, pages 186–191, 1988.
24. B. Halstenberg and R. Reischuk. Relations between complexity classes. *Journal of Computer and System Sciences*, 41:402–429, 1990.
25. L. Hemaspaandra and L. Torenvliet. *Theory of Semi-Feasible Algorithms*. EATCS Monographs in Theoretical Computer Science. Springer-Verlag, 2003.
26. C. Jockusch. Semirecursive sets and positive reducibility. *Transactions of the American Mathematical Society*, 131:420–436, May 1968.
27. B. Kalyanasundaram and G. Schnitger. The probabilistic communication complexity of set intersection. *SIAM Journal on Discrete Mathematics*, 5:545–557, 1992.
28. M. Karchmer. *Communication Complexity: A new approach to circuit depth*. MIT Press, Cambridge, MA, 1989.
29. M. Karchmer, R. Raz, and A. Wigderson. Super-logarithmic depth lower bounds via the direct sum in communication complexity. *Computational Complexity*, 5, 1995.
30. K.-I. Ko. On self-reducibility and weak P-selectivity. *Journal of Computer and System Sciences*, 26:209–221, 1983.
31. M. Krentel. The complexity of optimization problems. *Journal of Computer and System Sciences*, 36(3):490–509, 1988. Prior version in *ACM Sym on Theory of Computation*, 1986 (STOC).
32. M. Kummer. A proof of Beigel's cardinality conjecture. *Journal of Symbolic Logic*, 57(2):677–681, June 1992.

33. K. Kuratowski. Sur le probleme des courbes gauches en topologie. *Fundamenta Mathematicae*, 15:271–283, 1930.
34. E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge University Press, 1997.
35. N. Lakshmipathy and K. Winklmann. “Global” graph problems tend to be intractable. *Journal of Computer and System Sciences*, 32(3):407–428, June 1986.
36. M. Li and P. Vitanyi. *An Introduction to Kolmogorov Complexity and Its Applications*. Addison-Wesley, 1991.
37. L. Lovász. Communications complexity: A survey. In B. Korte, editor, *Paths, Flows, and VLSI layout*, Berlin, 1990. Springer-Verlag. Early version is TR CS-TR-204-89, Princeton University, 1989.
38. L. Lovász and M. Saks. Lattices, mobius functions, and communication complexity. *Journal of Computer and System Sciences*, 47:322–349, 1993.
39. W. Mader. Homomorphieeigenschaften und mittlere kantendichte von graphen. *Math. Ann.*, 174:265–268, 1967.
40. N. Nisan, S. Rudich, and M. Saks. Products and help bits in decision trees. *SIAM J. Comput.*, 28, 1998.
41. M. Ogihara. Polynomial-time membership comparable sets. *SIAM J. Comput.*, 24:1068–1081, 1995.
42. J. C. Owings, Jr. A cardinality version of Beigel’s Nonspeedup Theorem. *Journal of Symbolic Logic*, 54(3):761–767, Sept. 1989.
43. W. J. Paul. Realizing Boolean functions on disjoint sets of variables. *Theoretical Computer Science*, 2(3):383–396, Sept. 1976.
44. A. Razborov. On the distributional complexity of disjointness. *Theoretical Computer Science*, 106:385–390, 1992. Prior version in ICALP 1990.
45. N. Robertson and P. D. Seymour. Graph minors XV: Wagner’s conjecture. to appear in *Journal of Combinatorial Theory (Series B)*.
46. N. Robertson and P. D. Seymour. Graph minors XIII: the disjoint paths problem. *Journal of Combinatorial Theory (Series B)*, 63, 1995.
47. N. Sauer. On the density of families of sets. *JCTA*, 13:145–147, 1972.
48. A. Selman. P-selective sets, tally languages, and the behavior of polynomial time reducibilities on NP. *Mathematical Systems Theory*, 13:55–65, 1979.
49. A. Selman. Analogues of semirecursive sets and effective reducibilities to the study of NP complexity. *Information and Computation*, 52(1):36–51, Jan. 1982.
50. A. Selman. Reductions on NP and P-selective sets. *Theoretical Computer Science*, 19:287–304, 1982.
51. D. Sivakumar. On membership comparable sets. *Journal of Computer and System Sciences*, pages 270–280, 1999. Prior version in *IEEE Conf on Complexity Theory, 1998 (STRUCTURES)*.
52. V. Strassen. Vermeidung von divisionen. *J. Reine Angew. Math.*, 264:184–202, 1973.
53. V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probab. and its Applications*, 16(2):264–280, 1971.
54. A. Yao. Some complexity questions related to distributive computing. In *Proceedings of the Eleventh Annual ACM Symposium on the Theory of Computing*, Atlanta GA, 1979.