CS277: Concrete Complexity	Fall 1996
Lecture 17: October 29, 1996	
Lecturer: Uri Zwick	*Scribe: Deborah Goldman*

17.1 A Lower Bound on $C_{M_2}(CLIQUE_k(n))$

In this lecture, using the method of approximations, we prove a near-exponential lower bound on the size of a monotone circuit computing the function $CLIQUE_k(n)$.

17.1.1 Review of Set-up

We first briefly review from the previous lecture the definitions of the approximators, operations, and test graphs which we need to use the method of approximations. Recall that an approximator is an expression of the form $\vee_{i=1}^{m'} \lceil X_i \rceil$, where $\lceil X_i \rceil$ is a clique indicator function on the vertices in X_i. Further, our approximators are also restricted by two parameters p and l (which we will choose) so that $|X_i| \le l$ for all i and $m' \le m$, where $m = (p-1)^l \cdot l!$.

Next, we present the definitions of the approximate OR and AND, which rely heavily on the use of sunflowers. The approximate OR is defined in the following way:

$$\bigvee_{i=1}^{m_1} \lceil X_i \rceil \sqcup \bigvee_{j=1}^{m_2} \lceil Y_j \rceil = \begin{cases} \bigvee_{i=1}^{m_1} \lceil X_i \rceil \vee \bigvee_{j=1}^{m_2} \lceil Y_j \rceil & \text{if } m_1 + m_2 \le m \\ \bigvee_{i=1}^{m'} \lceil W_i \rceil, m' \le m & \text{otherwise,} \end{cases}$$

where the W_i are derived by plucking sunflowers from the set consisting of the X_i and Y_j until at most m sets remain. The approximate AND is defined in a similar manner:

$$\bigvee_{i=1}^{m_1} \lceil X_i \rceil \sqcap \bigvee_{j=1}^{m_2} \lceil Y_j \rceil = \begin{cases} \bigvee_{(i,j):|X_i \cup Y_j| \le l} \lceil X_i \cup Y_j \rceil & \text{if } |\{(i,j) : |X_i \cup Y_j| \le l\}| \le m \\ \bigvee_{i=1}^{m'} \lceil W_i \rceil, m' \le m & \text{otherwise,} \end{cases}$$

where the W_i are derived by plucking sunflowers from the set consisting of the $X_i \cup Y_j$ such that $|X_i \cup Y_j| \le l$ until at most m sets remain.

Finally, we need only review the choice of test graphs. Positive text graphs will be cliques of size k. Negative test graphs will be $(k-1)$-partite graphs which correspond to all possible ways of coloring n vertices with $k-1$ colors.

17.1.2 Proof

The proof of a lower bound on $C_{M_2}(CLIQUE_k(n))$ follows the same format as our proof of a lower bound on $C_{M_2}(CLIQUE_3(n))$. First, we show in the following claim that our approximator circuits perform badly on our test graphs.

Claim 17.1 *Every approximator circuit either outputs 0 on all positive test graphs or outputs 1 on at least half of the negative test graphs.*

Proof: Suppose the approximator C is not identically 0. Then by our choice of approximators we must have $C \geq \lceil X_1 \rceil$ for some indicator $\lceil X_1 \rceil$, where $|X_1| \leq l$. Now, $\lceil X_1 \rceil$ will evaluate to 1 on a negative test graph if and only if all the vertices in X_1 have different colors. Noting that, for a random negative test graph, the probability that two vertices in X_1 have the same color is certainly less than or equal to $\frac{\binom{l}{2}}{k-1}$, we see that

$$Pr[\lceil X_1 \rceil(\text{and thus C) outputs 1 on a random negative test graph}] \geq 1 - \frac{\binom{l}{2}}{k-1}.$$

It suffices to choose $l = \lfloor \sqrt{k} \rfloor$ to ensure the probability is greater than or equal to $\frac{1}{2}$. ■

Next, in the following two claims we show that our approximate OR's and AND's still serve as fairly good local approximations. In Claim 17.2 we bound the number of positive test graphs for which a gate could output the wrong answer.

Claim 17.2 *For every pair of approximators C_1 and C_2,*

1. $C_1 \sqcup C_2 \geq C_1 \vee C_2$ *and*
2. $C_1 \sqcap C_2 < C_1 \wedge C_2$ *for at most $m^2\binom{n-l-1}{k-l-1}$ positive text graphs.*

Proof: To prove the first item we need only look back at the definition of the approximate OR and observe that by plucking sunflowers we might only make $C_1 \sqcup C_2$ larger than $C_1 \vee C_2$. Let $C_1 = \vee_{i=1}^{m_1} \lceil X_i \rceil$ and $C_2 = \vee_{j=1}^{m_2} \lceil Y_j \rceil$. To prove item 2, we note that for cliques (our positive test graphs),

$$C_1 \wedge C_2 = \bigvee_{i=1}^{m_1} \lceil X_i \rceil \wedge \bigvee_{j=1}^{m_2} \lceil Y_j \rceil = \bigvee_{(i,j)} \lceil X_i \cup Y_j \rceil.$$

Thus, our approximate AND might only be smaller than this value because we have omitted terms $\lceil X_i \cup Y_j \rceil$ with $|X_i \cup Y_j| \geq l+1$ (since, again, plucking sunflowers can only increase the value). However, only at most $\binom{n-l-1}{k-l-1}$ cliques can contain a set $X_i \cup Y_j$ of size at least $l+1$. Finally, noting that there are at most m^2 different sets $X_i \cup Y_j$, we see that the inequality can hold for at most $m^2\binom{n-l-1}{k-l-1}$ positive test graphs. ■

Similarly, in Claim 17.3 we bound the number of negative test graphs for which a gate could output the wrong answer.

Claim 17.3 *For every pair of approximators C_1 and C_2,*

1. *$C_1 \sqcup C_2 > C_1 \vee C_2$ for at most $m2^{-p}(k-1)^n$ negative test graphs and*
2. *$C_1 \sqcap C_2 > C_1 \wedge C_2$ for at most $m^2 2^{-p}(k-1)^n$ negative test graphs.*

Proof: To prove both items, we note that the only way the value of an approximate OR or AND could be strictly larger than the real value is if we plucked a sunflower for which its center, Z, was a clique, but none of the petals, $Z_1, \ldots, Z_p$, were cliques. Recall that a negative test graph contains a clique on a set S if and only if each vertex of S has a different color. Let us call a set with such differently colored vertices well-colored (wc). We wish to evaluate the probability $Pr[Z \text{ is wc and } Z_1, \ldots, Z_p \text{ are not wc}]$ for a randomly chosen negative test graph, and do so using the following chain of inequalities:

$$
\begin{aligned}
Pr[Z \text{ is wc and } Z_1, \ldots, Z_p \text{ are not wc}] &\leq Pr[Z_1, \ldots, Z_p \text{ are not wc} | Z \text{ is wc}] && (17.1)\\
&= \prod_{i=1}^{p} Pr[Z_i \text{ is not wc} | Z \text{ is wc}] && (17.2)\\
&\leq \prod_{i=1}^{p} Pr[Z_i \text{ is not wc}] && (17.3)\\
&\leq \left(\frac{\binom{l}{2}}{k-1}\right)^p && (17.4)\\
&\leq 2^{-p}. && (17.5)
\end{aligned}
$$

The first inequality follows easily from the definition of conditional probability. The second follows from the fact that the events Z_i is well-colored are independent given that Z is well-colored. The third holds since the event Z_i is well-colored might only happen with greater probability if we don't assume Z is well-colored. Finally, the fourth and fifth inequalities follow from our earlier calculation and choice of l.

Now, when computing approximate OR's and AND's, we certainly pluck no more than m and m^2 sunflowers, respectively. Thus, in total we find $C_1 \sqcup C_2 > C_1 \vee C_2$ for at most $m2^{-p}(k-1)^n$ negative test graphs and $C_1 \sqcap C_2 > C_1 \wedge C_2$ for at most $m^2 2^{-p}(k-1)^n$ negative test graphs. ■

We can now find a lower bound on the number of gates in a monotone circuit computing $CLIQUE_k(n)$. From the previous two claims, we know that an approximate gate can adversely affect no more than $m^2\binom{n-l-1}{k-l-1}$ positive test graphs and no more than $m^2 2^{-p}(k-1)^n$ negative test graphs. As a result of Claim 17.1, we see that

$$C_{M_2}(CLIQUE_k(n)) \geq \min\left\{\frac{\binom{n}{k}}{m^2\binom{n-l-1}{k-l-1}}, \frac{2^{p-1}}{m^2}\right\}.$$

Let us choose $p = \lceil \sqrt{k} \log n \rceil$, so then

$$m = (p-1)^l \cdot l! \leq (\sqrt{k} \log n)^{\lfloor \sqrt{k} \rfloor} \sqrt{k}^{\lfloor \sqrt{k} \rfloor} = (k \log n)^{\lfloor \sqrt{k} \rfloor}.$$

We find

$$\frac{\binom{n}{k}}{m^2 \binom{n-l-1}{k-l-1}} \geq \frac{1}{m^2} (\frac{n}{k})^l \geq \frac{1}{(k \log n)^{2\lfloor \sqrt{k} \rfloor}} (\frac{n}{k})^{\lfloor \sqrt{k} \rfloor} = (\frac{n}{k^3 \log^2 n})^{\lfloor \sqrt{k} \rfloor}$$

and

$$\frac{2^{p-1}}{m^2} \geq \frac{1}{2} \cdot \frac{2^{\lfloor \sqrt{k} \rfloor \log n}}{(k \log n)^{2\lfloor \sqrt{k} \rfloor}} = \frac{1}{2} (\frac{n}{k^2 \log^2 n})^{\lfloor \sqrt{k} \rfloor}.$$

Consequently, we derive our desired lower bound:

Theorem 17.4 $C_{M_2}(CLIQUE_k(n)) \geq (\frac{n}{k^3 \log^2 n})^{\lfloor \sqrt{k} \rfloor}$, *i.e.* $C_{M_2}(CLIQUE_k(n)) \geq 2^{\lfloor \sqrt{k} \rfloor}$ *provided* $k \geq (\frac{n}{2 \log^2 n})^{\frac{1}{3}}$.

Corollary 17.5 *For* $k = (\frac{n}{2 \log^2 n})^{\frac{1}{3}}, C_{M_2}(CLIQUE_k(n)) \geq 2^{(\frac{n}{2 \log^2 n})^{\frac{1}{6}}}$.

CS277: Concrete Complexity **Fall 1996**

Lecture 18: October 31, 1996

Lecturer: Uri Zwick *Scribe: Deborah Goldman*

18.2 Communication Complexity

In this lecture, we define the notion of communication complexity and then prove a theorem relating communication complexity to circuit depth.

18.2.1 Definitions and Example

The communication problem corresponding to a relation R is as follows: given a relation $R \subseteq X \times Y \times Z$, two players Alice and Bob are told $x \in X$ and $y \in Y$, respectively. Their goal is to find $z \in Z$ such that $(x, y, z) \in R$, or decide that no such z exists. For example, if R is equal to a function g, then Alice and Bob should compute $g(x, y)$.

Alice and Bob are allowed to communicate with each other by sending each other bits, and this communication should be done according to a deterministic protocol. In a deterministic protocol, the next player to communicate is completely determined by the number and content of the previous bits sent, and the new bit to be sent is a function of the previous communication and the player's string.

The communication complexity of a protocol for R is the maximum number of bits that Alice and Bob send using the protocol, taken over all pairs of inputs $(x, y) \in X \times Y$. We can now define the communication complexity of R:

Definition 18.6 *The communication complexity of the relation R, denoted $C(R)$, is the minimum of the communication complexities of all protocols that solve the communication problem corresponding to R.*

There are two types of relations in which we will be particularly interested. First, for sets $A, B \subseteq \{0,1\}^n$ such that $A \cap B = \emptyset$ we define the relation

$$R_{A,B} = \{(x, y, i) : x \in A, y \in B, x_i \neq y_i\}.$$

Note that $R_{A,B} \subseteq A \times B \times [n]$. Next, for a function f such that $f : \{0,1\}^n \to \{0,1\}$, we define

$$R_f = R_{f^{-1}(0), f^{-1}(1)}.$$

Thus in the communication problem corresponding to f, Alice receives $x \in f^{-1}(0)$ and Bob receives $y \in f^{-1}(1)$, and their goal is to find $i \in [n]$ such that $x_i \neq y_i$.

Example: In this example, we illustrate a protocol for the relation corresponding to the function (XOR_n). We assume Alice (A) is given $x \in \{0,1\}^n$ such that $x_1 \oplus \cdots \oplus x_n = 0$ and Bob (B) is given $y \in \{0,1\}^n$ such that $y_1 \oplus \cdots \oplus y_n = 1$. The protocol is as follows:

- If $n > 1$, then A sends $x_1 \oplus \cdots \oplus x_{\frac{n}{2}}$.
- B sends $x_1 \oplus \cdots \oplus x_{\frac{n}{2}} \oplus y_1 \oplus \cdots \oplus y_{\frac{n}{2}}$.
- If B sent the bit 1, then A and B repeat the first two steps using the first halves of their strings, else they repeat using the second halves.

The protocol ends when A and B each have one bit remaining, which is the bit they agree they differ on. Notice the number of bits sent in this protocol is at most $2\lceil \log n \rceil$, which happens to be the depth complexity of the function XOR_n over U_2. Thus $C(R_{XOR_n}) \leq 2\lceil \log n \rceil$, and, in fact, using the next theorem we will see the above protocol is optimal. ■

18.2.2 Communication Complexity and Depth Complexity

In the following theorem we prove that, for a Boolean function f, the communication complexity of R_f is equal to the depth complexity of f over the basis U_2. This theorem will prove useful in the future for proving bounds on $D_{U_2}(f)$.

Theorem 18.7 *(Karchmer, Wigderson) For every Boolean function $f : \{0,1\}^n \to \{0,1\}$, $D_{U_2}(f) = C(R_f)$.*

Proof: First, we show $C(R_f) \leq D_{U_2}(f)$. We may assume that Alice and Bob have agreed on a formula of depth $D_{U_2}(f)$ computing f. Further, we may assume, using de Morgan's laws, that negations are applied only to the variables (leaves) and, thus, that all gates are $\wedge$'s and $\vee$'s. Suppose the output gate is an $\wedge$-gate, that is, we can write $f = f_1 \wedge f_2$. Then Alice sends a bit i corresponding to a function f_i such that $f_i(x) = 0$. We know that we must have $f_i(y) = 1$. We can then repeat this step at the gate corresponding to the output gate of f_i, where Alice answers if the gate is an $\wedge$ and Bob answers if the gate is an $\vee$ (he sends a bit corresponding to a function which outputs 1). Alice and Bob repeat this process until they reach a leaf, at which point they agree they differ on that bit.

To prove $D_{U_2}(f) \leq C(R_f)$, it suffices to prove the following claim:

Claim 18.8 *If $A, B \subseteq \{0,1\}^n, A \cap B = \emptyset$, and $C(R_{A,B}) \leq d$, then there exists a function $f : \{0,1\}^n \to \{0,1\}$ such that $f(A) = 0, f(B) = 1$, and $D_{U_2}(f) \leq d$.*

We prove the claim by induction on d. Suppose $d = 0$. Then we must have, for some index i, that $x_i \neq y_i$ for all pairs (x, y) in $A \times B$. Thus we may choose either $f = x_i$ or $f = \bar{x}_i$ according to which function satisfies $f(A) = 0$ and $f(B) = 1$. Next, we prove the claim is true for d assuming it is true for $d-1$. Consider a protocol for $R_{A,B}$ that uses at most d bits. Let us assume Alice sends the first bit. Then there is a partition $A = A_0 \cup A_1, A_0 \cap A_1 = \emptyset$, such that for $x \in A_0$, Alice sends the bit 0 and for $x \in A_1$, Alice sends the bit 1. We then know that the relations $R_{A_0,B}$ and $R_{A_1,B}$ both have $C(R_{A_0,B}), C(R_{A_1,B}) \leq d-1$. Applying our induction hypothesis, we find there exist f_0 such that $f_0(A_0) = 0, f_0(B) = 1$, and $D_{U_2}(f_0) \leq d-1$ and f_1 such that $f_1(A_1) = 0, f_1(B) = 1$, and $D_{U_2}(f_1) \leq d-1$. We define $f = f_0 \wedge f_1$, for then $f(A) = 0, f(B) = 1$, and $D_{U_2}(f) \leq d$ as desired. Note that, if Bob had sent the first bit, we would have partitioned B and defined $f = f_1 \vee f_2$. ∎