CSE599s: Extremal Combinatorics October 12, 2011

Lecture 5 More Pigeons, and a Proof Complexity Lower Bound

Lecturer: Anup Rao

1 Dirichlet's Theorem

Theorem 1. *Let x be a positive real number. Then for every positive integer n, there is a rational number p/q such that $1 \leq q \leq n$ and $|x - p/q| > 1/nq \leq 1/q^2$.*

Proof Without loss of generality, assume that $0 < x < 1$. Consider the n holes

$$\left[0, \frac{1}{n}\right), \left[\frac{1}{n}, \frac{2}{n}\right), \ldots, \left[\frac{n-1}{n}, 1\right),$$

and the $n+1$ pigeons which are the fractional parts of the numbers $x, 2x, \ldots, (n+1)x$. By the pigeonhole principle, we must have $a > b$ such that ax, bx are in the same holes. So $ax = y_a + \epsilon_a$, $bx = y_b + \epsilon_b$, where y_a, y_b are positive integers, and $\epsilon_a, \epsilon_b \in [0, 1)$. Then $(a-b)x = (y_a - y_b) + (\epsilon_a - \epsilon_b)$, so $|(a-b)x - (y_a - y_b)| < 1/n$. Dividing through by $(a-b)$ proves the theorem. ■

2 Proving the Pigeonhole Principle can be Hard to prove, a detour.

A proof system is specified by a collection of rules that say how to combine true statements to get new true statements. Formally, the proof system is specified by defining a set of binary strings that are considered "truths", and a subset of these that are considered "axioms" and defining a constant number of functions that can map any two true strings to another string that is true. Once we have such a collection of rules, we can talk about the *proof complexity* of a proving a given true statement using those rules repeatedly applying them to previously derived truths or to axioms.

Investigating the proof complexity in natural proof systems is a whole research area, which has some consequences to the computational complexity of problems. For example, if we can prove that $P = NP$ by proving that some algorithm A solves SAT in polynomial time, all in some fixed proof system $\mathcal{P}$, then one obtains that every satisfiable boolean formula has a polynomial sized proof that proves that the formula is satisfiable in that proof system using a polynomial sized proof. Thus, if we can argue that there are satisfiable formulas such that proving they are satisfiable requires superpolynomial sized proofs, then we will have proved that you cannot prove $P = NP$ in the same proof system.

Here we shall deal with a very simple proof system useful for proving things about boolean formulas. A *literal* is a variable or its complement. A clause is the or of some literals. Given a collection of clauses, the proof system can be used to show that the clauses cannot all be simultaneously satisfied. You can take any two clauses and derive from it another clause that is implied by them. For example, we can derive $(x \vee y)$ from the clauses $(y \vee z)$ and $(x \vee \neg z)$. The proof is complete when a contradiction is derived. Figure 1 shows a proof in this proof system. This proof system is a slight generalization of the proof system called *Resolution*, and a proof in this system is called a *resolution refutation*.

A large clause of SAT solvers can be shown to operate by generating resolution proofs of the underlying sat formulas, so they have been studied extensively. The *size* of the proof is the number of clauses that are derived by it. It can be shown that if there are n unsatisfiable clauses, then there is a resolution refutation proof of size 2^n.

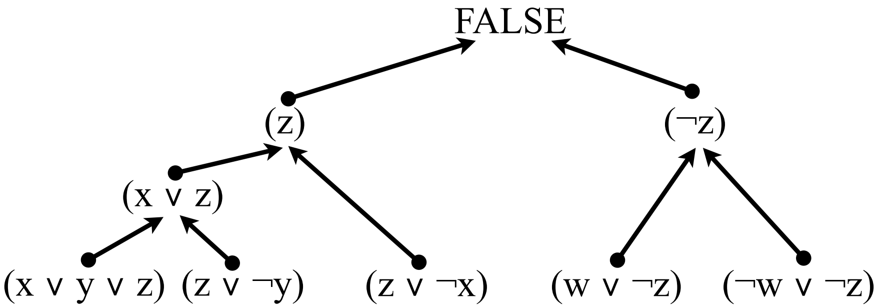

Figure 1: A resolution proof showing that $(x \vee y \vee z) \wedge (z \vee \neg y) \wedge (z \vee \neg x) \wedge (w \vee \neg z) \wedge (\neg w \vee \neg z)$ is false.

2.1 Refutations for the Pigeonhole Principle

For every $i \in [n], j \in [n-1]$, we shall have a boolean variable $x_{i \to j}$. The variable is set to 1 exactly when the i'th pigeon is put in the j'th hole. Define a set of clauses called PGN^n as follows. We shall have two types of clauses:

Every pigeon is in some hole: for all $i \in [n]$, we have the pigeon clause

$$P_i = (x_{i \to 1} \vee x_{i \to 2} \vee \cdots \vee x_{i \to n-1})$$

Every pigeon is in at most one hole: for all $i \in [n]$, and all $j, j' \in [n-1]$ we have the multipigeon clause

$$MP_{i,j,j'} = (\neg x_{i \to j} \vee \neg x_{i \to j'})$$

Every hole has at most one pigeon: for all $j \in [n-1]$ and $i \neq i' \in [n]$, we have the hole clause

$$H_{i,i',j} = (\neg x_{i \to j} \vee \neg x_{i' \to j})$$

The fact that all of the above clauses cannot simultaneously be satisfied is equivalent to the pigeonhole principle.

Haken then proved the following theorem:

Theorem 2 (Haken)**.** *Every resolution refutation for the pigeonhole principle must have size* $2^{\Omega(n)}$.

Here we present a simpler proof due to Beame and Pitassi, building on work of Buss.

The first idea is to restrict the refutation to the following special case:

Assumption: Every hole has exactly one distinct pigeon.

In other words, we let the proof assume this fact for free. Even under this assumption, all the clauses cannot be satisfied, and any proof that finds a contradiction in general must also work under the assumption. We shall show that it takes an exponential number of derivation steps to get a contradiction. Under the assumption, all the multipigeon and hole clauses are immediately satisfied. Also, asserting that the j'th hole does not contain the i'th pigeon is equivalent to asserting that the j'th hole contains one of the other pigeons. So $\neg x_{i \to j}$ is equivalent to $\bigvee_{i' \neq i} x_{i' \to j}$. So we modify the proof by replacing every negated variable by the corresponding or of variables. Henceforth we shall assume that there are no negated variables.

We shall show that any such proof must have a clause with many variables.

Claim 3. *There must be some clause involving $2n^2/9$ variables.*

Before proving the claim, let us see how it can be used to prove Theorem 2. If we assign pigeon i to hole j, then the effect this has on the clauses is essentially to make them look like PGN^{n-1}! Formally, for all $j' \neq j, i' \neq i$, we set $x_{i,j} = 1, x_{i,j'} = 0, x_{i',j} = 0$. Doing this will immediately reduce every clause that contains $x_{i,j}$ to TRUE, and will simplify other clauses by removing the other variables we have set.

Now suppose the proof contained less than ℓ clauses that have at least $n^2/18$ variables. Since there are less than n^2 variables, there must be some variable $x_{i,j}$ that occurs in $\ell/18$ of these clauses, by double counting. Assign the i'th pigeon to the j'th hole. We obtain a proof showing that PGN^{n-1} cannot be satisfied, where the number of clauses with at least $n^2/18$ variables has been reduced to $(17/18)\ell$. After repeating this argument $n/2$ times, we are left with a proof of $\mathsf{PGN}^{n/2}$ with at most $(17/18)^{n/2}\ell$ clauses of length at least $n^2/18 = 2(n/2)^2/9$. Thus $\ell \geq (18/17)^{n/2}$, or else Claim 3 will be contradicted.

It only remains to prove the claim.

Proof For every clause C, call the size of the minimal S such that $\bigwedge_{i \in S} P_i \Rightarrow C$ the *weight* of C. There are n pigeon clauses, and all of them are required to derive a contradiction, so the contradiction has weight n. One of the clauses used to derive the contradiction must have weight at least $n/2$. Continuing in this way, we can find a sequence of clauses $C_1, \ldots, C_t$ such that the weight of C_i is at least half the weight of C_{i-1}, C_1 is the contradiction, and C_t is a pigeon clause. The weight of a pigeon clause is 1. So there must be some clause C in this sequence whose weight is at most $2n/3$, but at least $n/3$. Let S be the corresponding minimal set of pigeons for C. We shall prove that C has at least $2n^2/9$ variables.

Consider any $i \in S$. If $\neg P_i \Rightarrow C$, then P_i is not needed to derive C, since if P_i is false, C is implied anyway. This would contradict the minimality of S. Thus, there must be some assignment of pigeons to holes such that the i'th pigeon is not in a hole and C is not satisfied (i.e. such that $\neg P_i \wedge \neg C$). For any $j \notin S$, we can give the i'th pigeon the j'th pigeon's hole. This will satisfy $P_{i'}$ for all $i' \in S$ (by our assumption all the other pigeons were already in holes), and so *must* satisfy C. Since C does not have any negations, the only way this could happen is if C has the variable $x_{i,k}$, where k was the hole of the j'th pigeon in the original configuration. Thus we have shown that C contains at least $n - |S|$ variables that correspond to the i'th pigeon, one for each $j \notin S$. Repeating this argument for every pigeon in S shows that C has at least $n(n - |S|)$ variables, which is minimized at $2n^2/9$, since $n/3 \leq |S| \leq 2n/3$. ■