COMP 409, Lecture 18

October 17, 2005

1 Proof Complexity

What we expect from any proof is that we should be able to check the steps of the proof 'easily'. We now formalize what it means to be able to check a proof 'easily'. We define $Deductions(\Gamma)$ as follows.

Definition 1 *A* Γ-deduction *is a sequence* $\varphi_1, ..., \varphi_n$ *such that for every* φ_i*, there are* $i_1 < i_2 < ... < i_{k-1} < i$ *such that* $(\varphi_{i_1}\varphi_{i_2}...\varphi_{i_{k-1}}\varphi_i) \in \Gamma$.

$Deductions(\Gamma) = \{(\varphi_1, ..., \varphi_n) \mid (\varphi_1, ..., \varphi_n) \text{ is a } \Gamma - deduction\}$

Now we define *polynomial feasibility* of Γ as follows.

Definition 2 Γ *is polynomially feasible if* $Deductions(\Gamma) \in PTIME$.

How long is the proof going to be? We want reasonably short proofs.

Definition 3 Γ *is a polynomially bounded system iff there is polynomial* ρ *s.t. whenever* $\vdash_\Gamma \varphi$*, there is a deduction* $\varphi_1, ..., \varphi_n \vdash \varphi$ *in* Γ*, such that*

- $n \leq \rho(|\varphi|)$
- $|\varphi_i| \leq \rho(|\varphi|)$

The definition says that a proof is polynomially bounded iff it has polynomial lines and each line has polynomial length.

We list the following desiderata for Γ:

1. Soundness and completeness
2. Polynomial feasibility
3. Polynomial boundedness

Theorem 1 *If* Γ *is a sound and complete deductive system which is polynomially feasible and bounded,* $VALID \in NP$.

The intuition behind this theorem is as follows. Suppose we want to know if $\models \varphi$ is a tautology. $\models \varphi$ iff $\vdash_\Gamma \varphi$ means validity is in NP and therefore NP = co-NP. This is because of the relatively short proof. NP = co-NP is widely believed not to be the case.

Corollary 1 *We don't think we can find Γ, that is sound, complete, polynomially feasible and bounded.*

Theorem 2 (Cook+Reckhow, 1974) *For a more general notion of deductive system, NP=co-NP iff there is a sound, complete, polynomially feasible and bounded deduction system.*

Resolution is sound and complete, but it is not polynomial bounded. Recall that, in Resolution Theorem Proving, we iteratively construct increasingly larger sets of formulas until a contradiction is reached.

$$\begin{array}{rcl} C_0 & = & \varphi \\ C_1 & = & R_{p_1}(C_0) \\ C_2 & = & R_{p_2}(C_1) \\ & \vdots & \\ C_n & = & R_{p_n}(C_{n-1}) \end{array}$$

While each set C_i is only quadratically bigger than C_{i-1}, the overall growth can be exponential. Put otherwise, the length of a resolution refutation can be exponential. We now focus on the length of the refutations. We will give an example formula whose refutation is exponential.

Recall the *Pigeon-Hole Principle*: If we have $f : \{1, \ldots, n\} \mapsto \{1, \ldots, n-1\}$, then for some $1 \leq i \leq j \leq n$ we have $f(i) = f(j)$. It can be written as: $R \subseteq \{1, ..., n\} \times \{1, ..., n-1\}$, for every pigeon i there is some hole k such that $R(i, k)$. There exists $1 \leq i < j \leq n$ and $1 \leq k \leq n-1$, such that both $R(i, k)$ and $R(j, k)$ hold.

We formulate this (for a fixed $n \in \mathbb{N}$) in propositional logic. To do so, we will use the following atomic propositions, for each i, j such that $1 \leq i \leq j \leq n$:

$$P_{ij} \stackrel{\text{def}}{=} R(i, j)$$

PHP_n

$$\left(\bigwedge_{1 \leq i \leq n} \bigvee_{1 \leq k \leq n-1} P_{ik} \right) \rightarrow \left(\bigvee_{1 \leq i < j \leq n} \bigvee_{1 \leq k < n} (P_{ik} \wedge P_{jk}) \right)$$

So, this formula is valid iff the Pigeon-Hole Principle (for a fixed n) holds.

Theorem 3 (Haken, 1975) *Every resolution refutation of $\neg PHP_n$ has length $2^{\Omega(n)}$.*

So, Resolution Theorem Proving is not *polynomially-bounded*: there is no polynomial function of the length of a theorem that places an upper bound on the length of the smallest proof of that theorem.

Conceivably, resolution have long refutations because it is such a weak system. So an active research area is establishing lower bounds for more powerful proof systems.

We have $\Phi \models \varphi$ iff $\Phi \cup \{\neg\varphi\}$ is unsatisfiable. What heppens if Φ is infinite? An example for such a Φ would be a set of formulas describing coloring of an infinite graph. Notice that the proof does not depend in finiteness. In the theorem for resolution we started with a finite number of clauses. We proved earlier that Φ is unsat. iff Φ is refutable (for finite Φ). We want to remove this assumption establishing *strong completeness.*

We thus arrive at a very important result, which we will revisit again later in the course.

Theorem 4 (Compactness Theorem) *Let Φ be a (possibly infinite) set of formulas. Then Φ is satisfiable if and only if every* finite subset *of Φ is satisfiable.*

We will not prove the theorem in class, but the interested students should contact the TA for a sketch of the proof.

With this result we close the first half of the course which was dealing with propositional logic. Next we turn our attention to First Order Logic.