Lecture 34

November 30, 2005

1 Undecidability

In the previous lecture we looked at the validity of formulas. Gödel Completeness Theorem states that validity for first-order logic is recursively enumerable. This was good news, because the logicians back in the days thought that with a little more work, they could prove that validity is decidable. However, this is impossible. In 1936, Church and Turing independently showed that validity for first-order logic is undecidable.

So, how we can prove this? Remember that if $L_1 \leq_r L_2$ and L_1 is undecidable, then L_2 is undecidable. As in NP-completeness, we need the "first" undecidable problem. Such a problem is the Halting problem, and it is proved by the diagonalizing method.

2 Simple Programming Language

Consider the Simple Programming Language–SPL, having the following instructions

1. Assignments: $x = x + 1$, and $x = x - 1$
2. Branching: goto line l
3. Conditional: If $x = 0$, then goto line
4. Termination: *stop*

We also call this a counter machine (CM), because it essentially counts up or down using one or more variables. The decision problem for a CM is: Starting at statement 0, with all variables initialized to 0, will the machine reach a statement k with all variables (counters) equal to 0?

We denote the decision problem for a CM with 2 variables as 2CM. The Halting problem for 2CM is undecidable.

Consider $S_i(t_1, t_2)$ to mean that the machine is in line i with t_1, t_2 in its counters. Let our starting state be $S_0(0, 0)$.

We can model the properties of the machine as a set of the following axioms:

1. $(\forall x)(\forall y)(S_i(x, y)) \to S_{i+1}(x + 1, y))$ for $x = x + 1$.
2. $(\forall x)(\forall y)(S_i(x + 1, y)) \to S_{i+1}(x, y))$ for $x = x - 1$.
3. $(\forall x)(\forall y)(S_i(x, y)) \to S_l(x, y)$ for goto l.
4. $(\forall x)(\forall y)(S_i(0, y)) \to S_l(0, y)$ for if $x = 0$, goto l.

Axioms for variable y are similar to those above.

Lemma 1 *The machine M reaches statement k with all zeroed variables iff*

$$\models (Axioms \to S_k(0, 0)).$$

Here, *Axioms* corresponds to the set of all axioms representing the counter machine M.

Now, suppose that M does not reach k with zeroes variables. We need to construct a structure A such that $A \models Axioms$ but $A \not\models S_k(0, 0)$, i.e., $A \models Axioms \wedge \neg S_k(0, 0)$

It is easy to see that we can construct a Herbrand structure A such that $A \models Axioms \wedge \neg S_k(0, 0)$, since the specification of the machine and execution has all universal sentences.

Coming back to validity of first-order logic.

Theorem 1 (Church-Turing Theorem) *Validity of first-order logic is undecidable.*

φ is satisfiable iff $\neg\varphi$ is not valid.

Corollary 1 *Satisfaction is not recursively enumerable.*

This follows from the fact that if a problem is both CO-RE and RE, it is recursive. This corollary is known as an *Incompleteness* Theorem, since it implies that there is no sound and complete proof system for satisfiability.

Now, what about validity over all finite structures. Unfortunately, there we have another negative result.

Definition 1 *φ is valid in the finite if $A \models \varphi$ holds whenever A is finite.*

Theorem 2 (Trakhtenbrot's Theorem) *Validity over finite structures for first-order logic is not recursive.*

Definition 2 *φ is satisfiable in the finite if there is a finite structure A, such that $A \models \varphi$*

What about satisfiability in a finite structure? If the vocabulary is finite, then there are finitely many structures (up to an isomorphism) of a given finite cardinality.

Corollary 2 *Satisfiability in some finite structure is recursively enumerable.*

Hence, validity over finite structures cannot be recursively enumerable.

Let Arith $= (\mathbf{N}, 0, +, *)$. The truth problem for arithmetics is then $\{\varphi : \text{Arith} \models \varphi\}$.

Theorem 3 (Gödel's Incompleteness Theorem) *The truth problem for arithmetics is not recursive. Hence, it is also not recursively enumerable.*

A consequence of this theorem is that we cannot enumerate all theorems in arithmetics.