Logic and Computation

CS245

Dr. Borzoo Bonakdarpour

University of Waterloo

(Fall 2012)

Computability and Decidability

Agenda

- Programs as Formulas
- Cantor's Diagonalization
- Decidability

Programs as Formulas

Addition

$\forall x.\mathsf{PLUS}(0, x, x)$

$\forall x, y, z.\mathsf{PLUS}(x, y, z) \Rightarrow \mathsf{PLUS}(s(x), y, s(z))$

where s is the successor function in the set of natural numbers.

Programs as Formulas

Strings

$\forall x.\mathsf{APPEND}([], x, x)$

$\forall x, y, z, h.\mathsf{APPEND}(x, y, z) \Rightarrow \mathsf{APPEND}(h|x, y, h|z)$

where ‘$|$’ denotes concatenation of two strings.

Programs as Formulas

For a program P with the APPEND axioms show that $P \models$ APPEND$([a, b], [c], [a, b, c])$.

Hint: Proof by resolution.

Programs as Formulas

Reverse

$$\forall x.\mathsf{REVERSE}([], x, x)$$

$$\forall x, y, z, w.\mathsf{REVERSE}(y, x|z, w) \Rightarrow \mathsf{REVERSE}(x|y, z, w)$$

Programs as Formulas

Storage

$$\forall x, y, z.\mathsf{LOOKUP}(x, y, \mathsf{cell}(x, y, z))$$

$$\forall x, y, x', y', z.\mathsf{LOOKUP}(x, y, z) \Rightarrow \mathsf{LOOKUP}(x, y, \mathsf{cell}(x', y', z))$$

Decidability

A set is *decidable* iff there exists a formula $\varphi(x)$ such that:

- $\vdash \varphi(t)$ for $t \in S$
- $\nvdash \varphi(t)$ for $t \notin S$

Question: Is the set of terms of FOL decidable? I.e., can we describe FOL by itself?

To answer this question, we should first learn whether a given set is countable. Because if we want to describe an uncountable set by a countable set, we will fail!

Countable Sets

Recall that a set S is *countable* if there is a one-to-one correspondance between S and the $\mathbb{N}$ of natural numbers.

How do we prove that a set is *uncountable*?

Cantor's Diagonal Argument

Considers an infinite sequence $S = (s_1, s_2, \ldots)$, where each element s_i is an infinite sequence of 1s or 0s. Each sequence s_i is countable (why?):

$$
\begin{aligned}
s_1 &= (0,0,0,0,0,0,0,\ldots)\\
s_2 &= (1,1,1,1,1,1,1,\ldots)\\
s_3 &= (0,1,0,1,0,1,0,\ldots)\\
s_4 &= (1,0,1,0,1,0,1,\ldots)\\
s_5 &= (1,1,0,1,0,1,1,\ldots)\\
s_6 &= (0,0,1,1,0,1,1,\ldots)\\
s_7 &= (1,0,0,0,1,0,0,\ldots)\\
&\ldots
\end{aligned}
$$

Diagonalization

It is possible to build a sequence $\overline{s}$ in such a way that if $s_{n,n} = 1$, then $\overline{s}_n = 0$, otherwise $\overline{s}_n = 1$.

$$
\begin{aligned}
s_1 &= (\underline{\mathbf{0}}, 0, 0, 0, 0, 0, 0, \dots)\\
s_2 &= (1, \underline{\mathbf{1}}, 1, 1, 1, 1, 1, \dots)\\
s_3 &= (0, 1, \underline{\mathbf{0}}, 1, 0, 1, 0, \dots)\\
s_4 &= (1, 0, 1, \underline{\mathbf{0}}, 1, 0, 1, \dots)\\
s_5 &= (1, 1, 0, 1, \underline{\mathbf{0}}, 1, 1, \dots)\\
s_6 &= (0, 0, 1, 1, 0, \underline{\mathbf{1}}, 1, \dots)\\
s_7 &= (1, 0, 0, 0, 1, 0, \underline{\mathbf{0}}, \dots)\\
&\dots\\
\overline{s} &= (1, 0, 1, 1, 1, 0, 1, \dots)
\end{aligned}
$$

Diagonalization

By definition, $\overline{s}$ is not contained in the countable sequence S.

Let T be a set consisting of all infinite sequences of 0s and 1s. By definition, T must contain S and $\overline{s}$.

Since $\overline{s}$ is not in S, T cannot coincide with S.

Therefore, T is uncountable because it cannot be placed in one-to-one correspondence with $\mathbb{N}$.

$\mathbb{R}$ is Uncountable

We build a one-to-one correspondance between T and a subset of $\mathbb{R}$.

Let function $f(t) = 0.t$, where t is a string in T. For example, $f(0111\ldots) = 0.0111\ldots$.

Observe that $f(1000\ldots) = 0.1000\cdots = 1/2$, and $f(0111\ldots) = 0.0111\cdots = 1/4 + 1/8 + 1/16 + \cdots = 1/2$.

Hence, f is not a bijection.

$\mathbb{R}$ is Uncountable

To produce a bijection from T to the interval $(0, 1)$:

- From $(0, 1)$, remove the numbers having two binary expansions and form $a = (1/2, 1/4, 3/4, 1/8, 3/8, 5/8, 7/8, \cdots)$.
- From T, remove the strings appearing after the binary point in the binary expansions of 0, 1, and the numbers in sequence a and form $b = (000\cdots, 111\cdots, 1000\cdots, 0111\cdots, 01000\cdots, 00111\cdots, \cdots)$.

$g(t)$ from T to $(0, 1)$ is defined by: If t is the nth string in sequence b, let $g(t)$ be the nth number in sequence a; otherwise, let $g(t) = 0.t$.

$\mathbb{R}$ is Uncountable

To build a bijection from T to $\mathbb{R}$, we use $\tan(x)$, a bijection from $(-\pi/2, \pi/2)$ to $\mathbb{R}$.

The linear function $h(x) = \pi.x - \pi/2$ provides a bijection from $(0, 1)$ to $(-\pi/2, \pi/2)$.

The composite function $\tan(h(x))$ provides a bijection from $(0, 1)$ to $\mathbb{R}$.

Function $\tan(h(g(t)))$ is a bijection from T to $\mathbb{R}$.

Diagonalization

Using diagonalization, one can also show that (for example):

- $|\mathbb{Q}| = |\mathbb{N}| = |\mathbb{Z}|$
- $|\mathbb{N}| < |2^{\mathbb{N}}|$
- The set of all functions from $\mathbb{N}$ to $\mathbb{N}$ is uncountable.

Gödel Numbering

A *Gödel numbering* is a function that assigns to each symbol and well-formed formula of some formal language a unique natural number, called its *Gödel number*.

There are several ways to do this:

- Prime factorization
- ASCII code

Gödel Numbering

Moral of the story: each FOL formula φ is represented by a unique natural number $\lceil\varphi\rceil$.

So what?! Recall that $|\mathbb{N}| < |2^{\mathbb{N}}|$? This means we have too many sets and too few formulas!

This is the core idea of undecidability of FOL.

Decision Problems

Any question about a function can be converted to a “yes/no" problem. This is called a *decision problem*.

For example:

- Find a path a path from s to t in a graph G
  - Does there exist a path from s to t in G?
- Compute function $f(n)$
  - Decide whether $f(n) = m$
  - Or ask whether $\vdash R_f(\lceil n \rceil, \lceil m \rceil)$

Decision Problems and Decidability

Consider the following validity question: $P \vdash R(t_1, \ldots, t_n)$. Three possible answers are:

- There is a proof (e.g., using FOL resolution).
- There is no proof (e.g., using a counter example). This means we have a proof of $P \not\vdash R$.
- We cannot tell (i.e., the proof system *loops*). For example $\forall x.P(x) \leftarrow P(s(x))$

This question is equal to that of decidability/undecidability.

Decidability of FOL

Theorem. The set VALID $= \{\ulcorner\varphi\urcorner \mid\vdash \varphi\}$ is undecidable.

Proof sketch. This means that:

- $\vdash \varphi$ then $\vdash$ VALID$(\ulcorner\varphi\urcorner)$
- $\not\vdash \varphi$ then $\vdash \neg$VALID$(\ulcorner\varphi\urcorner)$

Assume that VALID(x) is a formula.

Diagonalization for FOL

$(\varphi_1)_x^{\ulcorner\varphi_2\urcorner}$	$\ulcorner\varphi_1\urcorner$	$\ulcorner\varphi_2\urcorner$	$\ulcorner\varphi_3\urcorner$	$\dots$
φ_1	$\vdash$	$\nvdash$	$\vdash$	$\dots$
φ_2	$\nvdash$	$\nvdash$	$\vdash$	$\dots$
φ_3	$\nvdash$	$\vdash$	$\vdash$	$\dots$
$.$		$\dots$		
$.$		$\dots$		
$.$		$\dots$		
φ_d	$\nvdash$	$\vdash$	$\nvdash$	$\dots$

Diagonalization for FOL

Note that DIAG$(t, \ulcorner \exists x . \varphi_t \wedge (x = t) \urcorner)$ is decidable.

$\varphi_d = \exists y . (\text{DIAG}(x, y) \wedge \neg \text{VALID}(y))$

Diagonal argument:

- $\vdash \varphi(\ulcorner \varphi_d \urcorner)$
- $\vdash \varphi_d = \exists y . (\text{DIAG}(\ulcorner \varphi_d \urcorner, y) \wedge \neg \text{VALID}(y))$
- $\vdash \neg \text{VALID}(\ulcorner \varphi_d(\ulcorner \varphi_d \urcorner) \urcorner)$
- $\not\vdash \varphi(\ulcorner \varphi_d \urcorner)$

Reducibility

A *reduction* is a transformation of one problem into another problem.

We normally reduce problem 1 to problem 2, because we know how to solve problem 2 and this gives us the answer to problem 1.

Reducibility

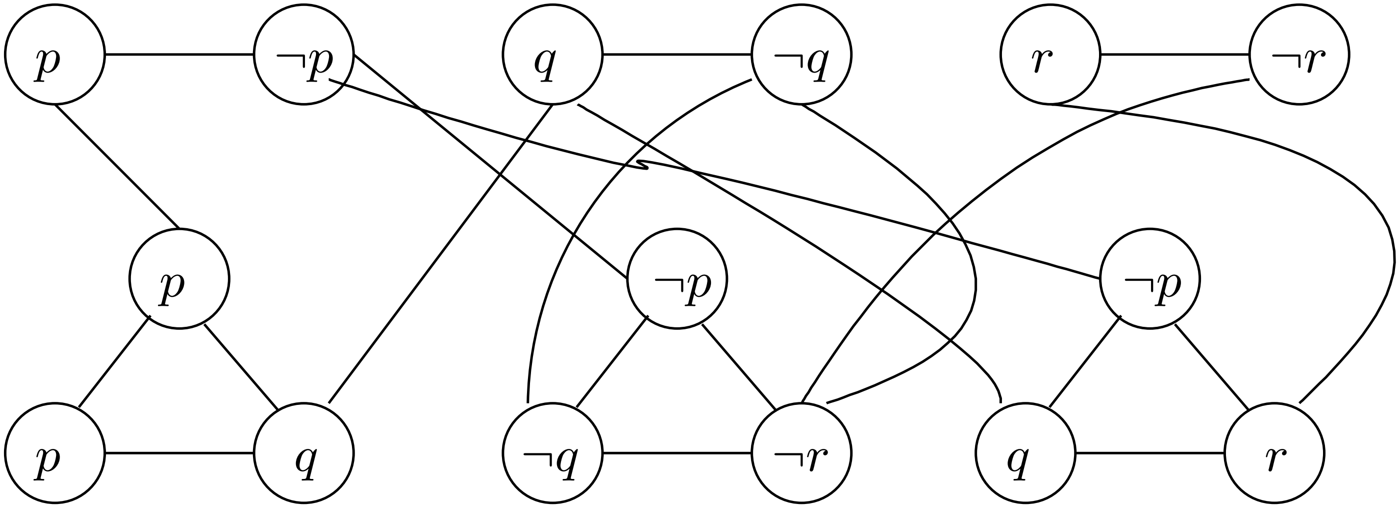
Recall graph connectivity, cyclicity, and vertex cover problems in application of the compactness theorem?

An important consequence of reducibility is to show that two problems belong to the same “class” (e.g., undecidable).

Another Example

Reduction from propositional satifiability to vertex cover:

$$(p \vee q) \wedge (\neg p \vee \neg q \vee \neg r) \wedge (\neg p \vee q \vee r)$$



The Halting Problem

Let HALTS$(\lceil P \rceil)$ be a program that returns true if P halts (recall that a programm P can be encoded as a first-order formula).

Theorem. The *halting problem*, is undecidable.

Undecidability of HALT

Let $\mathsf{RESOLUTION}(\ulcorner\varphi\urcorner)$ be the problem that implements resolution for FOL. I.e., it returns true if $\vdash \varphi$.

Then the program “if $\mathsf{HALTS}(\mathsf{RESOLUTION}(\ulcorner x\urcorner))$, then $\mathsf{EVAL}(\mathsf{RESOLUTION}(x))$, else false” is a decision procedure for VALID.

In other words, if we could solve the halting problem, we could have solved the FOL validity problem.

Gödel's Incompleteness Theroems

Theorem 1. For any consistent, effectively generated formal theory that proves certain basic arithmetic truths, there is an arithmetical statement that is true, but not provable in the theory.

Theorem 2. For any formal effectively generated theory T including basic arithmetical truths and also certain truths about formal provability, if T includes a statement of its own consistency then T is inconsistent.