CS 4860 Notes on Lecture 16

Tuesday Oct. 23, 2012

These notes outline the topics but do not provide all the details from the lecture. Readers should provide details from the actual lecture using their notes for topics not covered in Smullyan's book.

(1.) In classical FOL we can prove an interesting theorem that all quantifiers can be moved to the front of a formula. This is called prenex normal form. The equivalences that Smullyan list on page 56 at the bottom are used to convert formulas to prenex form. Here is a number list of the equivalences broken into separate implications.

1. $\exists x.(C \Rightarrow P(x)) \Rightarrow (C \Rightarrow \exists x.P(x))$
2. $\exists x.(C \Rightarrow P(x)) \Leftarrow (C \Rightarrow \exists x.P(x))$

3. $\forall x.(C \Rightarrow P(x)) \Rightarrow (C \Rightarrow \forall x.P(x))$
4. $\forall x.(C \Rightarrow P(x)) \Leftarrow (C \Rightarrow \forall x.P(x))$

5. $\forall x.(P(x) \Rightarrow C) \Rightarrow (\exists x.P(x) \Rightarrow C)$
6. $\forall x.(P(x) \Rightarrow C) \Leftarrow (\exists x.P(x) \Rightarrow C)$

7. $\exists x.(P(x) \Rightarrow C) \Rightarrow \forall x.P(x) \Rightarrow C$

8. $\exists x.(P(x) \Rightarrow C) \Leftarrow \forall x.P(x) \Rightarrow C$

Optional exercise: Which implications are constructively true? Give Refinement proofs for them. Explain (prove?) that the others are not realizable.

Examples. $\exists y.\exists x.(P(x) \Rightarrow Q(x)) \Leftrightarrow (\forall x.P(x) \Rightarrow \exists x.Q(x))$

$(\exists y.\forall x.(A(x,y) \Rightarrow \exists v\forall u.B(u,v)) \Leftrightarrow \forall y\exists x\exists v\forall u.(A(x,y) \Rightarrow B(u,v))$.

(2.) The Prelim required us to think more about how to show that a formula is <u>not constructively true</u>, e.g. <u>not realizable</u>. Clearly if the formula F such as $(P \supset Q) \supset \sim P \supset \sim Q$ is not classically true (not a tautology) then it is not constructively true. That's because classical evidence can use a non computable term as a realizer of $P \vee \sim P$, we call it magic(P). It is the only non-uniform realizer.

Recall that we proved that $P \vee \sim P$ is unrealizable. We expressed this by writing the uniform universal quantifier $\forall[P].(P \vee \sim P)$. <u>Fill in the definition from lecture</u>:

We prove $\sim \forall[P].(P \vee \sim P)$ by noting that the evidence term must be given that works for all P in advance of knowing any particular one. Thus it must be either inl( ) or inr( ). If it is inl( ), then pick P to be $0=1$, which is false and thus there is no evidence $p \in [P]$. If the evidence is inr( ) then pick $P = (0=0)$ which has evidence $*$. So inr( ) can't be correct.

In lecture we proved $\sim \forall[P].(\sim\sim P \supset P)$. <u>Fill in the proof.</u>

CS 4860 Notes on Lecture 16

Tuesday Oct. 23, 2012

(3) In today's lecture we will motivate the completeness theorem for tableau which Smullyan proves on p. 60. Intuitively the theorem says that tableau rules are sufficient to prove any classically valid first order formula. Smullyan's proof is not constructive, and there is no known constructive proof. But his proof does have interesting computational meaning.

* Fill in the discussion of that meaning from lecture.

* We reviewed Smullyan's definition of validity on page 49. We used D for a domain, Smullyan uses U on page 49. On page 51 he uses a universe V, and he defines valid sentences (formulas). Add to the notes Smullyan's definition for when an atomic formula is first-order satisfiable (p. 51).

Here is Smullyan's statement of completeness from p. 60 given symbolically

$$\forall x{:}\,\text{Form}.\ \text{Valid}(X) \Rightarrow \exists p{:}\,\text{Proof}_{\text{Tableau}}.\ \text{Proves}(p, X).$$

We could write $\text{Proves}(p,X)$ as $p \vdash X$.

But this proof is not constructive, we cannot find a proof p from Smullyan's argument. However we get a proof procedure which we will discuss.

* Write a summary of Smullyan's proof search procedure from p. 61, called Systematic Tableaux.