Applied Logic	**Lecture 18: Completeness of FOL continued**
CS 4860 Fall 2012	**October 30, 2012**

Plan

(1) Unrealizability (redeux) of $(\forall x.Px \supset C) \supset \exists x.(Px \supset C)$

(2) Review systematic Tableau (Smullyan p.59)

(3) Gödel completeness theorem (Smullyan Thm 3. p.60) – classical

(4) Completeness for iFOL by Judith Underwood – almost constructive *

(5) Completeness relative to uniform validity – fully constructive (we only discuss this new result, no careful proof)

* Item 4. is an advanced topic that we mention as a possible advanced project topic (see www.nuprl.org under Publications look for Judith Underwood, *Aspects of the Computational Content of Proofs*, Chpt 3, Oct. 1994).

The argument sketched in Lecture 17 to show that $(\forall x.Px \supset C) \supset \exists x.(Px \supset C)$ is unrealizable works. It can be made sharp by taking $C = \forall x.Px$. Then $(\forall x.Px \supset \forall x.Px)$ is realizable by $\lambda x.x$. To realize $\exists x.(Px \supset \forall x.Px)$ we consider a domain D and predicate $P(x)$ where $\forall x.P(x)$ is not realizable because for some $d_0 \,\epsilon\, D$, $\sim P(d_0)$. In this case the potential realizer f must pick $\langle d_0, np\rangle$. But by choosing a domain $D = \{d_0, d_1\}$ where p realizes $P(d_1)$, we can pick another P' on the domain so that $P'(d_0)$ and $\sim P'(d_1)$. The realizer f cannot work for both models *uniformly*– it must know D and P. Thus the formula is not uniformly valid.

It might seem bizarre that we can have $P(d) \supset \forall x.Px$, but classically we use $\sim\forall x.P(x) \vee \forall x.P(x)$, then $\sim\forall x.P(x) \Leftrightarrow \exists x.\sim P(x)$, so $\exists x.\sim P(x) \vee \forall x.P(x)$, thus $\sim P(d) \vee \forall x.P(x)$, then $P(d) \Rightarrow \forall x.P(x)$. Hence $\exists x.\big(P(x) \Rightarrow \forall x.P(x)\big)$. So we can realize this with applications of $P \vee \sim P$ for various instances of P. Can we define *dimension* in this example?

Review of systematic tableau

In Lecture 17 we wrote a high level sketch of the procedure Smullyan defines on p.59. We defined it as a procedure that extends a *partial proof tree (ppt)* whose root node has FX as a value. Given a *ppt* of depth d, we define *extension* of it.

Let $extend(ppt)$ produce a new finite partial proof tree ppt' by following the Systematic Tableau procedure. Smullyan's completeness proof works by repeatedly applying $extend$, e.g. start with ppt_0 consisting of the root node FX.

Let $ppt_0 = ppt(FX)$, for a formula X.

$$\text{Define} \quad \begin{array}{lcl} extend^{(0)}(ppt_0) & = & ppt_0 \\ extend^{(n+1)}(ppt_0) & = & extend\big(extend^{(n)}(ppt_0)\big) \end{array}$$

$$\text{Thus} \quad \begin{array}{lcl} extend^{(1)} & = & extend(ppt_0) \\ extend^{(2)} & = & extend\big(extend(ppt_0)\big) \\ \text{etc.} & & \end{array}$$

For Smullyan's classical tableau, we can state a version of Judith Underwood's theorem for *Intuitionistic Tableau* of Fitting. Although we have not studied Fitting's system, we can get an idea of how to transfer her method to Smullyan's system. Let α denote a path in the unbound proof tree $extend^{(n)}(ppt_0)$. Let $\alpha(m)$ be the path up to length m.

Theorem: $\forall x : Form. \forall n : \mathbb{N}.$ If $Valid(x)$ then

(i) if $extend^{(n)}\big(ppt(FX)\big)$ is closed, then it is a tableau proof of X, and

(ii) otherwise $extend^{(n)}\big(ppt(FX)\big)$ can be extended further and $\sim\forall m > n.\ open\big(extend^{(m)}\big(ppt(FX)\big)\big)$, and

(iii) for any path α in the generated tree (unbounded) $\sim\forall m : \mathbb{N}.\ open\big(\alpha(m)\big)$.

Note, Underwood's proof, like Smullyan's, is not constructive. But hers is much closer to a plausible constructive argument. If she could deduce from the statement

> For all paths α in the unbounded proof tree given by $extend^{(n)}\big(ppt(FX)\big)$, we can find an m such that $closed\big(\alpha(m)\big)$,

then by the Fan Theorem, we know that the partial proof tree is finite. Since X is valid, this proof tree is a tableau proof. Thus we would know constructively that:

$$\forall X : Form. \big(Valid(X) \Rightarrow \exists n : \mathbb{N}.\ Proof\big(extend^{(n)}\big(ppt(FX)\big)\big)\big).$$

Classically it is easy to show that given any path α

$$\sim\forall m : \mathbb{N}.\ \big(m > n \Rightarrow open\big(\alpha(m)\big)\big) \text{ implies } \exists m : \mathbb{N}.\ closed\big(\alpha(m)\big).$$

We know that $open\big(\alpha(m)\big)$ is decidable, and $closed\big(\alpha(m)\big)$ means $\sim open\big(\alpha(m)\big)$. The general principle used in this reasoning is called *Markov's Principle* and it can be expressed in iFOL as follows:

> *Markov's Principle*: $\forall x.\big(P(x) \vee \sim P(x)\big) \Rightarrow \big(\sim\forall x.\sim P(x) \Rightarrow \exists x.P(x)\big)$

It seems like the realizer for this when there is a computable function $f : \mathbb{N} \to D$ that enumerates the elements of D (as we have in the Tableau procedure) is a "program" such as *while* $\sim P(x)$ *do* $x := f(x)$ *od* starting at d_0.

We can show that Markov's Principle is not uniformly realizable, hence is not provable in iFOL. The idea is that given the evidence for $\big(\forall x. \sim P(x)\big) \Rightarrow False$, say a function g, the realizer F must on input g compute an element of D, e.g. $F(g)_1 \epsilon D$, and find a proof $F(g)_2$ of $P\big(F(g)_1\big)$. Suppose $F(g)_1 = d$ and $F(g)_2 = p$, p proves $P(d)$. Then we can change the model D so that P is not true on d. Thus MP is not uniformly true.