

Filosofisk Logik

(FTEA21:4)

föreläsningssanteckningar/kompendium

V

v. 2.0, den 2/5 2014

Fullständighet för FOL

19.1

Att bevisa fullständighet för det härledningssystem vi har använt för FOL, med avseende på den semantik som gått igenom, är ett betydligt svårare problem än att bevisa det för satslogiken. De grundläggande idéerna är dock samma som för satslogiken. För satslogiken visades fullständighet först av Emil Post år 1923, och för predikatlogiken av Kurt Gödel år 1929, även om båda använde helt andra metoder än den vi skall använda av oss.

Det centrala steget är, som vanligt i ett fullständighetsteorem, att visa att om $\mathcal{T}$ inte är inkonsistent, så finns det en modell $\mathcal{M}$ vari alla satser i $\mathcal{T}$ är sanna. Precis som för satslogiken *konstruerar* vi en sådan modell: i detta fall en vars domän, relationer och funktioner utgörs av mängdteoretiska konstruktioner av individkonstanter i språket självt. För FOL görs detta i 4 steg:

1. Tillägg av *vittneskonstanter*. Vi har inga garantier för att de individkonstanter som finns i $\mathcal{L}$ är tillräckliga för att beskriva alla domäner, och behöver därför lägga till några. Vi kallar det resulterande språket $\mathcal{L}_H$.
2. Bilda *Henkinteorin* $\mathcal{H}$ i språket $\mathcal{L}_H$. Denna är en uppsättning axiom som beskriver hur kvantifikatorerna och identitetspredikatet fungerar i termer av konnektiven, d.v.s. en sorts "reduktion" av predikatlogik till satslogik.
3. Visa *eliminationsteoremet*. Detta säger att det som kan visas i Henkinteorin, och som inte innehåller vittneskonstanterna, också kan visas i första ordningens logik *utan* att anta Henkinteorin. Detta behövs för att garantera att den modell vi konstruerar för $\mathcal{T} \cup \mathcal{H}$ gör sanna och falska samma satser som teorin $\mathcal{T}$.
4. Bilda *Henkinkonstruktionen*. Denna är en modell vars domän består av ekvivalensklasser av individkonstanter i $\mathcal{L}_H$, som gör sann alla satser i $\mathcal{T} \cup \mathcal{H}$, och därmed också alla satser i $\mathcal{T}$.

Vittneskonstanter och axiom; eliminationsteoremet

19.2-19.4

För att kunna ge en teori som beskriver kvantifikatorerna måste vi kunna ersätta $\exists x S(x)$ med någon sats som inte innehåller $\exists x$. Detta kan göras genom att byta ut x mot någon konstant c varom vi inte har antagit något mer än $S(c)$. Ett sätt att kunna garantera att sådana konstanter finns är att lägga till dem. För varje sats $\exists x S(x)$ lägger vi därför till konstanten $c_{S(x)}$, och kallar resultatet av detta $\mathcal{L}_1$. Eftersom vi också kan använda dessa konstanter för att bilda nya satser, så innehåller detta språk dock fortfarande satser på formen $\exists x S'(x)$ som inte har motsvarande

vittneskonstanter. Därför upprepar vi processen för att få språken $\mathcal{L}_2, \mathcal{L}_3, \mathcal{L}_4$, etc. Språket $\mathcal{L}_H$ definieras som

$$\mathcal{L}_H =_{df.} \bigcup_{k \in \mathbb{N}} \mathcal{L}_k$$

För att de nya konstanterna skall bete sig som vi vill, och för att vidare kunna beskriva kvantifikatorerna och identitetspredikatet, behöver vi Henkinteorin $\mathcal{H}$ som definieras som följande oändliga mängd satser:

1. $\exists x S(x) \rightarrow S(c_{S(x)})$ för alla satser $\exists x S(x)$.
2. Alla satser på formen $S(c) \rightarrow \exists x S(x)$.
3. Alla satser på formen $\neg \forall x S(x) \leftrightarrow \exists x \neg S(x)$.
4. Alla satser på formen $c = c$.
5. Alla satser på formen $(S(c) \wedge c = d) \rightarrow S(d)$.

Eliminationsteoremet säger att alla satser som inte innehåller någon av vittneskonstanterna och som kan härledas från $\mathcal{T} \cup \mathcal{H}$ också kan härledas från $\mathcal{T}$. Detta kan inses genom att studera Henkinaxiomen: de är direkta tillämpningar av härledningsregler i FOL. För ett mer utarbetat bevis, se B&E, s.550 - 554.

UPPGIFTER: 19.3, 19.7, 19.9, 19.14

Henkinkonstruktionen

19.5

Som vi gått igenom består en *modell* för ett språk $\mathcal{L}$ i FOL av en mängd D som kallas *domänen*, samt extensioner till alla predikat och funktionssymboler. Grundidén i Henkinkonstruktionen är att använda $\mathcal{L}_H$'s individkonstanter för D . Eftersom $c = d$ kräver att c och d är identiska för att vara sann, kan vi emellertid inte använda dem som de är: vi kommer att ha många olika konstanter c, d som visserligen satisfierar samma predikat, men som ändå är distinkta. Lösningen här är att bilda relationen

$$c \equiv d \text{ omm } \mathcal{T} \cup \mathcal{H} \vdash c = d$$

Detta kan visas vara en ekvivalensrelation, så den tillåter att vi bildar ekvivalensklasser av konstanterna i $\mathcal{L}_H$. Vi låter elementen i D vara dessa ekvivalensklasser.

När vi satt ihop domänen är inte extensionerna något större problem. Vi låter extensionen av predikatet $P(x_1, \dots, x_n)$ vara mängden av n -tupler $\langle [c_1]_{\equiv}, \dots, [c_n]_{\equiv} \rangle$ av element i D sådana att $\mathcal{T} \cup \mathcal{H} \vdash P(c_1, \dots, c_n)$. Detta är väldefinierat eftersom alla konstanter i samma ekvivalensklass $[c]_{\equiv}$ satisfierar samma predikat. Detta kan i sin tur härledas med hjälp av Henkinaxiomen och $=\text{elim}$.

Henkinkonstruktionen ger en modell $\mathcal{M}_{\mathcal{T}}$ för varje konsistent mängd satser $\mathcal{T}$. Följer vi samma steg som för beviset av satslogikens fullständighet får vi

$$\text{Om } \mathcal{T} \not\vdash \perp \text{ så finns en modell } \mathcal{M} \text{ sådan att } \mathcal{M} \models \mathcal{T} \Leftrightarrow$$

$$\text{Om } \mathcal{T} \not\vdash \perp \text{ så } \mathcal{T} \not\models \perp \Leftrightarrow$$

$$\text{Om } \mathcal{T} \cup \{P\} \not\vdash \perp \text{ så } \mathcal{T} \cup \{P\} \not\models \perp \Leftrightarrow$$

$$\text{Om } \mathcal{T} \vdash P \text{ så } \mathcal{T} \cup \{P\} \models \perp$$

UPPGIFTER: 19.20, 19.22