Logik I

Åsa Hirvonen
Helsingfors universitet

Våren 2013

Inledning

Logik är läran om härledning. Med hjälp av logiken kan vi säga när ett resonemang är korrekt och när det inte är det. För att kunna studera resonemang på ett matematiskt exakt vis, måste vi definiera vad vi menar med resonemang och påståenden. De matematiska begrepp vi definierar kan naturligtvis inte fånga alla delar och nyanser av mänskligt tänkande, men för matematiska bevis är de väl lämpade.

På den här kursen studerar vi satslogik och predikatlogik. Satslogiken är egentligen ett fragment av predikatlogiken. Den befattar sig med enkla påståendesatser som

> Det regnar och vi kör bil.
> Ifall jag glömmer paraplyet blir jag våt.

Predikatlogiken eller första ordningens logik har en större uttrycksförmåga. Med den kan vi undersöka härledningar av formen

$$\frac{\begin{array}{l}\text{Alla människor är dödliga.}\\ \text{Sokrates är en människa.}\end{array}}{\text{Sokrates är dödlig.}}$$

eller formellt uttryckt

$$\frac{\begin{array}{l}\forall x(P(x) \to Q(x))\\ P(a)\end{array}}{Q(a)}$$

Predikatlogiken är den logik vi behöver för att studera matematiska bevis. Av pedagogiska skäl börjar vi ändå med satslogiken som ger oss en bild av de metoder vi skall använda men undviker många av predikatlogikens svårigheter.

Två centrala begrepp inom logiken är *syntax* och *semantik*. Med syntax avser vi reglerna för det formella språk vi använder. Vilka tecken är tillåtna i formlerna? Vilken form får formlerna ha? Semantiken befattar sig med formlernas tolkning. Semantiken inom predikatlogiken avser de modeller vi betraktar. Inom satslogiken innebär semantiken sanningsvärdet i påståendena.

Det här materialet består av två delar. Den första behandlar satslogik, den andra predikatlogik. Vi följer i stort uppläggningen och innehållet i Jouko Väänänens kursmaterial "Logik One".

1 Satslogik

1.1 Formler

Satslogiken studerar ord som *inte*, *och*, *eller* och *om . . . så*. Den studerar hur strukturen av påståenden påverkar deras sanningshalt och beroendeförhållanden oberoende av sakinnehållet i påståendena. Satslogiken behandlar endast påståendesatser dvs. satser som uttrycker ett sakförhållande. Frågesatser, imperativa satser och andra språkliga satser som inte är konstaterande faller utanför satslogikens ramar.

De minsta beståndsdelarna i satslogiken är *enkla* eller *atomära satser*. De består av enskilda påståenden som inte kan spjälkas upp i mindre helheter med hjälp av ord som "inte", "och" och "eller". Exempel på atomära satser är:

Det regnar.
Bilen är blå.

Sammansatta satser byggs upp med hjälp av *konnektiv*. På den här kursen använder vi konnektiven *inte*, *och*, *eller*, *om. . . så* och *om och endast om*. Vi kommer senare dels att se att det finns fler tänkbara konnektiv, dels att förstå varför vi inte behöver granska dem skilt. Exempel på sammansatta satser är:

Jag tar bilen endast om det regnar.
Vi städar och ni diskar.

Eftersom vi är intresserade av satsernas struktur och sanningshalt, men inte själva sakinnehållet, abstraherar vi bort den här "onödiga" informationen och ersätter de atomära satserna med satssymboler

$$p_0, p_1, \dots$$

Vi introducerar också symboler för konnektiven:

Symbol	Tolkning	Namn
$\neg$	inte	negation
$\wedge$	och	konjunktion
$\vee$	eller	disjunktion
$\rightarrow$	om . . . så . . .	implikation
$\leftrightarrow$	om och endast om	ekvivalens

På så sätt kan vi skriva om givna satser på ett sätt som understryker deras struktur. Vi kan t.ex. definiera

p_0	Det regnar.
p_1	Jag glömde paraplyet.
p_2	Jag blir våt.

Nu kan satsen *Om det regnar och jag glömde paraplyet så blir jag våt.* skrivas $(p_0 \wedge p_1) \rightarrow p_2$. Den något pessimistiska satsen *Om jag glömde paraplyet så regnar det.* blir $p_1 \rightarrow p_0$.

För att kunna bevisa något generellt om *alla* satslogiska formler behöver vi en formell definition för vad en formel är. En sådan här definition gör satslogiken till ett *formellt språk*, dvs. ett systematiserat språk med explicita regler för hur språkets symboler får kombineras.

Definition 1. En satslogisk formel är en ändlig följd (teckensträng) bestående av satssymboler $p_0, p_1, p_2, \ldots$, konnektiv $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ och parenteser '(', ')', konstruerad enligt följande:

1. Varje satssymbol $p_0, p_1, p_2, \ldots$ är en satslogisk formel.
2. Om A är en satslogisk formel så är $\neg A$ en satslogisk formel.
3. Om A och B är formler, så är $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$ och $(A \leftrightarrow B)$ satslogiska formler.
4. Endast teckensträngar bildade enligt 1–3 ovan är satslogiska formler.

Det här är ett exempel på en *induktiv* definition. Istället för att explicit säga hur en formel ser ut (vilket skulle vara omöjligt) beskriver definitionen hur formler stegvis byggs upp från enklare formler genom upprepade tillämpningar av en regel. Man kan bevisa att definitionen ovan entydigt definierar den minsta mängd formler som innehåller alla satssymboler och är sluten under de operationer reglerna 2–3 ovan beskriver.

Anmärkning 2. För enkelhets skull utelämnar vi ofta parenteser där tolkningen är otvetydig. Så skriver vi t.ex. $A \wedge B$ istället för $(A \wedge B)$. Däremot är tolkningen av $A \rightarrow B \rightarrow C$ inte unik så här måste vi specifiera om vi menar $(A \rightarrow B) \rightarrow C$ eller $A \rightarrow (B \rightarrow C)$. När vi studerar sanningsvärden skall vi se att satser som $A \wedge B \wedge C$ är entydiga trots att de kan tolkas som både $(A \wedge B) \wedge C$ och $A \wedge (B \wedge C)$. Orsaken är att satserna i fråga är ekvivalenta, dvs. har samma betydelse.

Då vi undersöker en satslogisk formel är strukturen det viktiga. Hur är den uppbyggd? I vilken ordning har delarna pusslats ihop? Behändiga begrepp i de här betraktelserna är *huvudkonnektiv* och *delformler*.

Definition 3. En satslogisk formels *huvudkonnektiv* är det konnektiv som har tillämpats sist enligt 2–3 i Definition 1 i den stegvisa konstruktionen av formeln.

T.ex. i formeln $(\neg A \wedge (B \rightarrow \neg C))$ är huvudkonnektivet en konjunktion.

Definition 4. Om A är en satssymbol så är A den enda delformeln till A.
Delformlerna till $\neg A$ är $\neg A$ samt delformlerna till A.
Delformlerna till $A \wedge B$ är $A \wedge B$ samt delformlerna till A ocn B.
Delformlerna till $A \vee B$ är $A \vee B$ samt delformlerna till A ocn B.
Delformlerna till $A \rightarrow B$ är $A \rightarrow B$ samt delformlerna till A ocn B.
Delformlerna till $A \leftrightarrow B$ är $A \leftrightarrow B$ samt delformlerna till A ocn B.

T.ex. har formeln $((p_0 \rightarrow p_1) \leftrightarrow \neg p_2)$ delformlerna $((p_0 \rightarrow p_1) \leftrightarrow \neg p_2)$, $(p_0 \rightarrow p_1)$, $\neg p_2$, p_0, p_1 och p_2.

De *direkta delformlerna* är de delformler som huvudkonnektivet sammanbinder. Det här innebär att bara sammansatta formler kan ha direkta delformler.

Definition 5. Den direkta delformeln till $\neg A$ är A.
De direkta delformlerna till $A \wedge B$, $A \vee B$, $A \rightarrow B$ och $A \leftrightarrow B$ är A och B.

Lemma 6 (Induktiontionsprincipen för satslgiska formler)**.** *Anta att P är mängden av satslogiska formler och Q är en delmängd av P som uppfyller följande vilkor:*

1. varje satssymbol p_n tillhör Q,

2. om A tillhör Q så tillhör $\neg A$ också Q,

3. om A och B tillhör Q så tillhör $(A \wedge B)$, $(A \vee B)$, $(A \to B)$, och $(A \leftrightarrow B)$ också Q.

Då är $Q = P$.

Bevis. Vi reducerar principen till vanlig induktion (över de naturliga talen) genom att betrakta antalet konnektiv i formlerna. Så vi introducerar en hjälpfunktion

$$kon : P \to \mathbb{N}, \quad kon(A) = \text{antalet konnektiv i } A.$$

Eftersom en formel är en teckensträng är *kon* väldefinierad (dvs. antalet konnektiv i en formel är entydigt). Vi är nu intresserade av egenskapen

$E(n)$: Om A är en satslogisk formel och $kon(A) = n$ så tillhör A mängden Q.

Vi visar att $E(n)$ är sann för alla $n \in \mathbb{N}$.
Bassteg: $n = 0$: Om $A \in P$ och $kon(A) = 0$ så är A en satssymbol och tillhör Q på basen av (1) ovan.
Induktionssteg: Vi antar att $n \geq 0$ och $E(k)$ är sant för alla $k \leq n$ (*induktionshypotes*) och visar att $E(n+1)$ är sant. Så låt A vara en satslogisk formel så att $kon(A) = n + 1$. Från Definition 1 kan vi se alternativen för vad A kan vara. Eftersom $k + 1 > 0$, kan A inte vara en satssymbol, så följande alternativ återstår:

- $A = \neg B$. Eftersom $\neg$ är ett konnektiv är $kon(A) = 1 + kon(B)$. Således är $kon(B) = n$ och enligt induktionshypotesen har vi $B \in Q$. Enligt (2) ovan måste då också $A \in Q$.

- $A = (B \wedge C)$, $A = (B \vee C)$, $A = (B \to C)$ eller $A = (B \leftrightarrow C)$. Då har vi $kon(A) = kon(B) + 1 + kon(C)$ så både $kon(B) \leq n$ och $kon(C) \leq n$. Således ger induktionshypotesen $B \in Q$ och $C \in Q$, vilket enligt (3) ovan ger $A \in Q$.

Slutsats: $E(n)$ håller för alla $n \in \mathbb{N}$ och således har vi $P \subseteq Q$. □

Reduktionen till klassisk induktion kunde ha gjorts på flera olika sätt. Vi kunde t.ex. ha använt hjälpfunktionen $\#A$ = antalet symboler i A. Då hade bassteget börjat vid $n = 1$ men i övrigt hade beviset sett i stort sett likadant ut.

Bevis som använder induktionsprincipen ovan kallas ofta *strukturell induktion*. Nedan ett klassiskt exempel:

Exempel 7. Visa att varje satslogisk formel innehåller lika många höger- som vänsterparenteser.

Bevis. Definiera $HP(A)$=antalet högerparenteser i A och $VP(A)$=antalet vänsterparenteser i A. Vi visar med strukturell induktion att $VP(A) = HP(A)$ för alla satslogiska formler A.
Bassteg: A är en satssymbol. Då är $VP(A) = 0 = HP(A)$.
Induktionssteg: Steget består av följande fall:

- $A = \neg B$ och (induktionshypotes:) $VP(B) = HP(B)$. Då är $VP(A) = VP(B) = HP(B) = HP(A)$.

- $A = (B \wedge C)$, $A = (B \vee C)$, $A = (B \to C)$ eller $A = (B \leftrightarrow C)$ och (induktionshypotes:) $VP(B) = HP(B)$ och $VP(C) = HP(C)$. Då är $VP(A) = 1 + VP(B) + VP(C) = 1 + HP(B) + HP(C) = HP(A)$.

□