

UPPSALA UNIVERSITET

FÖRELÄSNINGSANTECKNINGAR VT22

# Logik och Bevisteknik

*Rami Abou Zahra*

## CONTENTS

1. Föreläsning - Introduktion till Satslogik	2
1.1. Historia	2
1.2. Vad behövs för ett matematiskt bevis?	2
1.3. Exempel	2
1.4. Satslogik	3
1.5. Predikatlogik (1:a ordningens logik)	3
2. Satslogik (Propositional calculus)	4
2.1. Språk (LP)	4
2.2. Exempel	4
3. Föreläsning - Naturlig deduktion (syntax)	5
3.1. Parantesers roll	5
3.2. Syntax	5
3.3. Naturlig deduktion	6
4. Förtydligande	7
4.1. Sanningstabeller	8

## 1. FÖRELÄSNING - INTRODUKTION TILL SATSLOGIK

1.1. **Historia.** Vad är ett matematiskt bevis?

Vad får användas i bevis?

Är matematiken motsägelsefri (Konsistent = motsägelsefritt)?

1.2. **Vad behövs för ett matematiskt bevis?**

- Ett påstående (även kallad utsaga) (ex.vis  $\sqrt{2}$  är irrationellt), dessa har sanningsvärde *sant* eller *falskt*
- (Giltigt) Argument (resonemang)

1.3. **Exempel.**

- Påstående: *Varje kvadrat är en rektangel*
- Påstående: *Det finns en fyrhörning som inte är en rektangel*

---

Alltså finns en fyrhörning som inte är kvadrat

Detta är syftet med kursen, låt oss nu göra det mer abstrakt. (Låt  $x$  vara form,  $K(x)$  kvadrat,  $R(x)$  rektangel,  $F(x)$  fyrhörning) Då blir påståenden:

- $\forall x(K(x) \Rightarrow R(x))$
- $\exists x(F(x) \wedge \neg R(x))$

---

$\exists x(F(x) \wedge \neg(K(x)))$

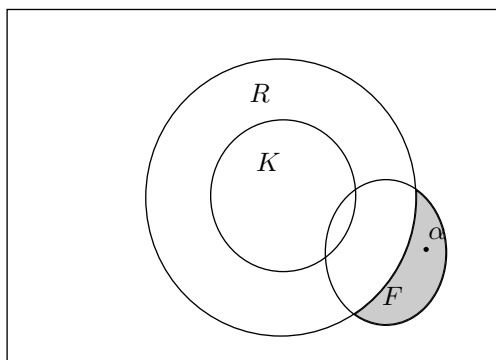


FIGURE 1. Grafisk tolkning

**Sats 1.1: Logiskt giltighet (Predikat-logik = 1:a ordningens logik)**

En följd av logiska steg kallas för *logiskt giltig* om det gäller  $\forall$  tolkningar av  $K, R, F$  (även vovvisar)

#### 1.4. Satslogik. Vi behöver:

- Ett skriftligt språk (mängd av teckensträngar som betyder utsagor/satser)
- Regler/Formella bevis (Hur får man dra slutsatser av nya teckensträngar), *syntax*
- Tolka teckensträngarna i en "verklighet" (sant eller falskt)

#### 1.5. Predikatlogik (1:a ordningens logik). Skiljer sig från satslogik i och med att vi inte hanterar satser utan predikaten.

- Språk behövs, liknande med teckenmängd men vi kan även hantera elementen
- Formella bevis består av teckensträngsmanipulation, relation mellan teckensträngar  
(Exvis  $A_1, A_2, A_3 \vdash B$  (A bevisar B))
- Vad betyder det att något är sant eller falskt? Om det går att visa B utan något så är det sant.

##### Sats 1.2: Definiera sanning i struktur

$\vdash B$  = Sant om det inte krävs något för att visa B:s sanningsvärde.

$A_1, A_2, A_3 \models B$  det vill säga om alla A krav är uppfyllda så gäller B

- Samband mellan  $\vdash$  (formell bevisbarhet) och  $\models$  (hur man tolkar något som sant eller falskt)

##### Sats 1.3: Sundhetssatsen

Detta säger att bevissystemet är sunt, allt man visar är sant

$A_1, A_2, A_3 \vdash B \Rightarrow A_1, A_2, A_3 \models B$

##### Sats 1.4: Adekvathet

Om det är så att B är sann så fort alla A är sanna så finns det ett bevis, vi kommer kunna påstå att det finns ett formellt bevis för samma sak.

$A_1, A_2, A_3 \models B \Rightarrow A_1, A_2, A_3 \vdash B$

##### Sats 1.5: Fullständighet

Slår vi ihop dessa 2 (Sundhetssatsen och Adekvathet) får vi fullständighet

$A_1, A_2, A_3 \vdash B \Leftrightarrow A_1, A_2, A_3 \models B$

## 2. SATSLOGIK (PROPOSITIONAL CALCULUS)

## 2.1. Språk (LP).

- Satssymboler:  $\sigma = \{p_0, p_1, \dots, p_n\}$  (en *satslogisk signatur*, även kallas *språkets signatur*)

$LP(\sigma)$  kallas för en *dialekt* av LP

- Alfabet i  $LP(\sigma)$ :
  - Alla symboler i  $\sigma$
  - Konnektiver:  $\Leftrightarrow, \Leftarrow, \Rightarrow, \vee, \wedge, \neg, \perp$
  - Paranteser:  $(, )$

Viktigt att notera,  $\perp$  är konnektiv men även sats, så den skapar inga nya formler.

**Sats 2.1: Mängden av formler i  $LP(\sigma)$** 

Mängden definieras rekursivt (likt naturliga talen).

- Basfall:
  - Alla tecken i  $\sigma$  är en formel
  - $\perp$  är en formel
  - Dessa kallas för *Atomära* formler
- Induktion:
  - Om  $\varphi$  är en formel, så är  $(\neg\varphi)$  en formel
- Låt  $\Box$  vara konnektiv:
  - Induktion  $\Box$ :
    - \* Om  $\varphi_1$  och  $\varphi_2$  är formler, så är  $(\varphi_1\Box\varphi_2)$  en formel

2.2. **Exempel.** Några exempel på formler i  $LP(\sigma)$  där  $\sigma = \{p, q, r\}$ :

- $\perp, r, (p \Rightarrow (\neg r)), (p \vee (\neg p))$

Några som *inte* är formler:

- $(p \wedge, p \vee q \wedge r$  (inga paranteser!)

## 3. FÖRELÄSNING - NATURLIG DEDUKTION (SYNTAX)

## 3.1. Parantesers roll.

Exempel:  $p \rightarrow q \wedge r$ . Hur skall vi placera ut paranteser, och behåller det formelns sanningsvärde? Vad händer om vi skriver  $(q \wedge r)$  istället så att  $(p \rightarrow (q \wedge r))$ , eller motsatsen,  $(p \rightarrow q)$  så att  $((p \rightarrow q) \wedge r)$ .

Det finns en konvention som hjälper oss att hålla koll på var och när och hur många paranteser som behövs.

- Skriv inte ut yttersta paranter
- $\neg$  binder starkare än  $\wedge, \vee, \rightarrow, \leftrightarrow$ 
  - $\neg p \wedge q$  betyder  $(\neg p) \wedge q$
- $\wedge, \vee$  binder starkare än pilarna  $\rightarrow, \leftrightarrow$ 
  - $p \wedge q \rightarrow r \vee s \Leftrightarrow (p \wedge q) \rightarrow (r \vee s)$
- $\wedge, \vee$  binder lika hårt.
- $\rightarrow, \leftrightarrow$  binder lika hårt.
  - Ex:  $p \vee q \wedge r$  ej klart vad som menas, här måste paranteser placeras ut

Man kan formulera detta genom *parsingträd* där subnoderna *inte* får kommutera, detta kallas för att trädet är *ordnat*.

Ex:  $(p \wedge (\neg q))$

Ex:  $(p_1 \wedge p_2) \rightarrow (\neg p_2 \rightarrow (p_1 \rightarrow \perp))$ . Detta är en *pilformel* och då kallas det för ett *huvudkonnektiv*.

**Sats 3.1**

Det sist tillagda konnektivet i formeln (motsvarar högsta noden i trädet) kallas för *huvudkonnektiv*.

Varje formel har ett entydigt träd. Men också tvärtom, givet ett träd så kan vi "bygga upp" en entydig formel.

**Sats 3.2: Parsingträd**

Låt  $T$  stå för ett träd.  $T : LP(\sigma) \rightarrow \{\text{parsingträd}\}$ .

Vi definierar det induktivt, där basen är en  $p$  atom:  $T(p) = \bullet p$ . Sedan påbörjar induktionen:

- Om  $\varphi$  formel med träd  $T(\varphi)$ , så  $T((\neg\varphi)) = \text{fig}$ .
- Om  $\varphi$  och  $\phi$  formler med träd  $T(\varphi)$  resp  $T(\phi)$ , så  $T((\varphi \square \phi)) = \text{fig}$ .

**Sats 3.3: Delformel**

En *delformel* till en formel  $\varphi$  är en teckensträng från  $\varphi$  som själv är en formel, då är det en delformel. Den triviala delformeln är  $\varphi$  själv.

Exempel:  $\varphi = (p_1 \wedge \neg p_2) \rightarrow ((p_2 \rightarrow p_3) \wedge p_4)$ . Då är exempelvis  $(p_2 \rightarrow p_3)$  en delformel eller  $\neg p_2$ . Däremot så är  $p_3) \wedge p_4$  *inte* en delformel. I parsingträdet så motsvarar varje nod en delformel.

## 3.2. Syntax.

Hur man kan dra slutsatser på ett syntaktiskt sett:

Om  $\Gamma$  är en mängd av formler och  $\varphi$  är en formel vill vi studera relationen mellan dessa. Följer  $\varphi$  av formlerna i  $\Gamma$ ?

Vi kommer studera denna frågan på 2 sätt, syntaktiskt och semantiskt.

- Syntax:
  - Formella bevisregler, tex.  $\frac{AB}{(a \wedge B)}$
  - $\Gamma \vdash \varphi$ ,  $\Gamma$  bevisar  $\varphi$

- Teckensträngsmanipulation
- Naturlig deduktion
- Semantik:
  - Är  $\varphi$  sann om alla formler i  $\Gamma$  är sanna?
  - $\Gamma \models \varphi$

### 3.3. Naturlig deduktion.

För varje konnektiv som vi har kommer vi introducera 2 regler, en för att introducera konnektivet och en för att ta bort.

- $\wedge$  intro ( $\wedge I$ ). Om vi har teckensträng  $A$  och  $B$  så kan vi  $\frac{AB}{(A \wedge B)}$
- $\wedge$ -elimination.  $\frac{(A \wedge B)}{A}$  och  $\frac{(A \wedge B)}{B}$
- $\rightarrow$ -intro.  $A \cdots B$ , jag börjar med  $A$  och jobbar mig mot  $B$  S.T  $\frac{A:B}{(A \rightarrow B)}$ . Efter  $B$  får man dra vilken slutsats  $A \rightarrow B$  där  $A$  vilken formel som helst. Om  $A$  är en premiss ovanför  $B$  så får  $A$  strykas.
- $\rightarrow$ -elimination. Om  $A$  gäller och  $A \rightarrow B$  så vet vi att  $B$  gäller (Modus ponens)

Exempel: Vi vill göra ett bevissträd som har följande slutsats,  $(A \rightarrow (B \rightarrow A \wedge B))$ . I detta träd kommer alla antaganden vara längst upp och slutsatser längst ner. Vi tittar på slutsatsen och märker att huvudkonnektivet är en pil, så vi kommer behöva använda pilintro. Då skall jag antag  $A$  och försöka härleda  $B \rightarrow (A \wedge B)$ , men då måste jag visa att  $B \rightarrow (A \wedge B)$  så vi måste anta  $B$  för att visa  $A \wedge B$  och nu kan jag visa det jag ville visa. Vi använder  $\wedge$ -intro. Per vårt antagande gäller  $A, B$  och därmed  $A \wedge B$ . Nu kan vi använda pil-elimination för att få bort pilen  $B \rightarrow (A \wedge B)$

#### Sats 3.4: Premiss

En *premiss* är en formel som ej är struken och förekommer högst upp i bevisträdet.

#### Sats 3.5: Slutsats

Formeln som står längst ner i bevisträdet.

## 4. FÖRTYDLIGANDE

**Sats 4.1: Disjunktion**

$A$  eller  $B$ . Uttrycker att minst en av  $A$  och  $B$  är fallet. Betecknas  $A \vee B$  där  $A, B$  kallas *disjunktionsled* eller *disjunkter*.

Det finns 2 typer av disjunktion, *uteslutande* och *icke-uteslutande*. Uteslutande disjunktion är helt enkelt  $A$  eller  $B$  (men inte båda) och icke-uteslutande är motsatt.

Disjunktionen är så kallad *inklusive*, det vill säga det är helt okej att både  $A$  och  $B$  är sann. Det finns en så kallad *exklusiv* disjunktion vilket kommer lite senare (kanske  $A \wedge B$ ?).

**Sats 4.2: Implikation**

Om  $A$  så  $B$  uttrycker att  $B$  är fallet givet att  $A$  är det. Detta betecknas  $A \rightarrow B$ . Här är  $A$  *antecedenten* eller även *förledet* och  $B$  är *konsekventen* eller *efterledet*. En implikation kallas också ivland en *materiell implikation*, *konditionalsats*, *villkorssats*.

**Sats 4.3: Ekvivalens**

$A$  omm  $B$  uttrycker konjunktionen av två implikationer, det vill säga om  $A$  så  $B$  oh om  $B$  så  $A$ . Skrivs  $A \leftrightarrow B$  och kallas även för *materiella ekvivalenser* eller *bikonditionalsatser*.

**Sats 4.4: Molekyler och Atomär**

En sats är *molekylär* om den är uppbyggd av en eller två andra satser med hjälp av ett konnektiv. I motsatt fall är satsen *atomär*. En molekylär sats innehåller minst ett konnektiv.

**Sats 4.5: n-ställig satsoperator**

En *n-ställig satsoperator* är en operator som "tar in"  $n$  variabler. Exempelvis är negationen  $\neg$  en 1-ställig operator, medan  $\wedge$  är en 2-ställig operator.

**Sats 4.6: Huvudoperator**

En *huvudoperator* är den operator som har tillämpats sist i uppbyggnaden av satsen. Exempelvis, i  $(A \rightarrow (B \vee \neg A))$  är  $\rightarrow$  huvudoperatoren. Däremot är  $\vee$  huvudoperatoren i delformen  $(B \vee \neg A)$ .



#### 4.1. Sanningstabeller.

För negation  $\neg$ :

$\neg A$  är sann  $\Leftrightarrow A$  är falsk  $\Leftrightarrow A$  inte är sann.

A	$\neg A$
S	F
F	S

För Konjunktion:

$A \wedge B$  är sann  $\Leftrightarrow A$  är sann och  $B$  är sann.

A	B	$A \wedge B$
S	S	S
S	F	F
F	S	F
F	F	F

För disjunktion:

$A \vee B$  är sann  $\Leftrightarrow A$  är sann eller  $B$  är sann  $\Leftrightarrow$  minst en av  $A$  och  $B$  är sann.

A	B	$A \vee B$
S	S	S
S	F	S
F	S	S
F	F	F

För implikation:

$A \rightarrow B$  är sann  $\Leftrightarrow$  om  $A$  är sann så  $B$  är sann  $\Leftrightarrow A$  är falsk eller  $B$  är sann. (Tips, vad är sista sanningsvärdet, dvs sanningsvärdet på  $B$ ?)

A	B	$A \rightarrow B$
S	S	S
S	F	F
F	S	S
F	F	S

För ekvivalens:

$A \leftrightarrow B$  är sann  $\Leftrightarrow$  om  $A$  är sann så är  $B$  sann och om  $B$  är sann så är  $A$  sann.  $\Leftrightarrow A \rightarrow B$  är sann och  $B \rightarrow A$  är sann  $\Leftrightarrow A$  och  $B$  har samma sanningsvärde.

A	B	$A \leftrightarrow B$
S	S	S
S	F	F
F	S	F
F	F	S

#### Sats 4.7: Satsparametrar

Antalet satsparametrar är antalet "variabler" i vår utsaga. Exvis, i  $A \rightarrow (B \wedge C \leftrightarrow \neg A)$  har 3st satsparametrar.