729G78 Artificiell intelligens
Kunskapsrepresentation - Satslogik

Arne Jönsson

HCS/IDA

LINKÖPINGS UNIVERSITET

Kunskapsrepresentation

- Introduktion
  - Wumpus-världen
- Logik
  - Satslogik
  - Predikatlogik, FOPL
- FOPL, Inferens
  - Resolution, Unifiering
- Representation av kunskap
  - Ontologi
  - Strukturerad representation/semantiska nät

Kunskapsbaserade agenter

- Generell kunskap om världen
  - Deklarativ beskrivning
- Härleda ny kunskap
  - Inferens
- Flexibla
  - Explicita mål
- Partiellt observerbara världar (jfr sökning)

Kunskapsbaserad agent

- Har en kunskapsbas som innehåller fakta och mekanismer för att dra slutsatser baserat på dessa fakta
- Agenten
  - tolkar percept vid en tid t
  - uppdaterar KB med perceptet
  - väljer handling baserat på KB
  - uppdaterar KB med handling vid t
  - utför handling och uppdaterar tiden

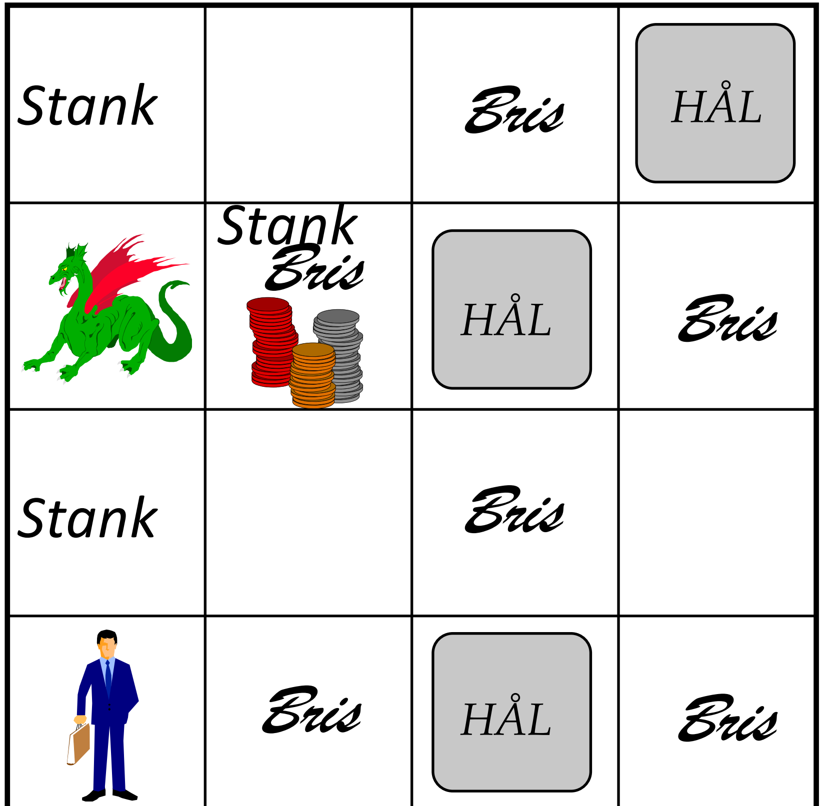
Kunskapsbaserad agent

```
KB = [ ]
t = 0
def KBagent(percept):
  tell(KB, makePerceptSentence(percept, t))
  action = ask(KB, makeActionQuery(t))
  tell(KB, makeActionSentence(action, t))
  t = t + 1
  return action
```

Modelleringsparadigm

- Tillståndsbaserade modeller
  - Tillstånd, handlingar, kostnader
  - Sökning
- Variabelbaserade modeller
  - Variabler, tilldelningar
  - CSP
- Logikbaserade modeller
  - Satslogik, predikatlogik
  - Logiska formler och inferensregler

Wumpus-världen



Wumpus-världen

- **Percept** (Stank, Bris, Glitter, Bump, Skrik)
  - Wumpusen stinker en ruta vertikalt och horsiontellt
  - Hål ger bris en ruta vertikalt och horisontellt
  - Guld glittrar
  - Bump i vägg
  - Döende Wumpus skriker

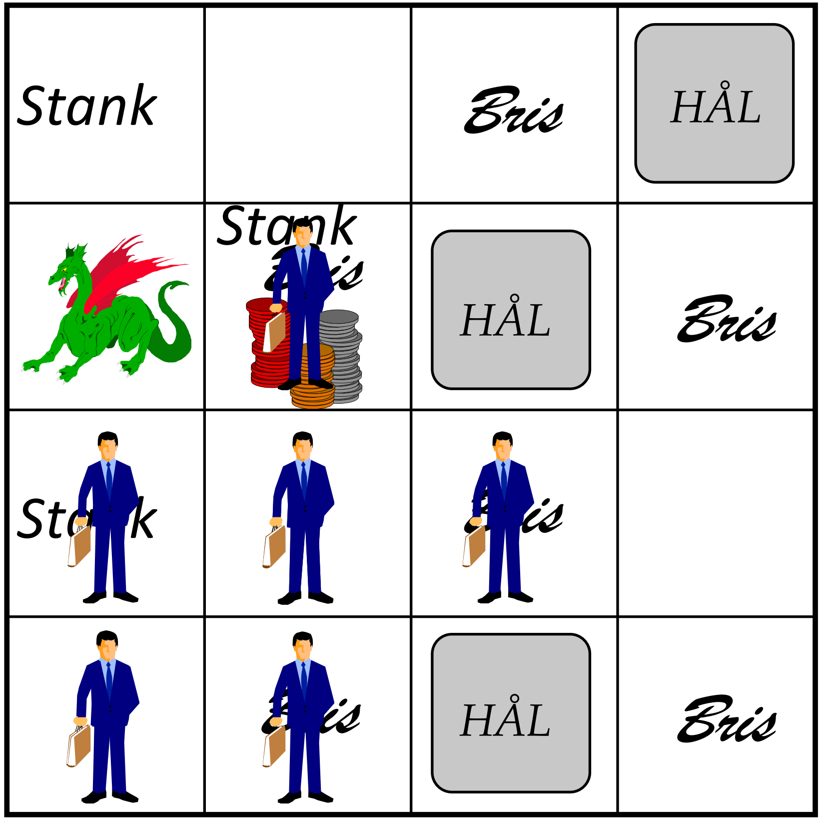
Wumpus-världen

- **Handlingar** (framåt, (sväng höger), (sväng vänster), ta, skjut, klättra)
  - Om agenten är vid skatten innebär “ta” att agenten har guldet
  - Om agenten står riktad mot wumpusen och skjuter dör wumpusen
  - Om agenten är vid startplatsen och klättrar kommer den ut
  - Om agenten hamnar i hål eller hos wumpusen dör den

Wumpus-världen resonemang

- Om det blåser på ruta x och ruta a, b, c och d är angränsande rutor så kan det finnas hål på a, b, c och d
- Om det inte blåser på ruta x och a, b, c och d är angränsande rutor så kan det inte finnas hål på a, b, c eller d
- Om det finns ett hål på ruta x eller y och det inte finns ett hål på x så finns det ett hål på y
- …

Wumpus-världen



Logik

- Syntax
  - Uttryck som beskriver världen
- Semantik
  - Uttryckens betydelse
  - En mängd modeller
- Inferensregler
  - Ger nya beskrivningar av världen

Syntax vs semantik

- Syntax
  - Vilka formler (uttryck) är tillåtna
  - Ex matematik:
    - x + y = 4
    - x 4 y = +
- Semantik
  - Vad betyder formlerna. När är de sanna?
  - Ex matematik:
    - x + y = 4 är sann för x = 1 och y = 3 eller x = 2 och y = 2 etc
    - Flera olika **modeller**

Satslogik

Syntax för satslogik

- Atomära satser. Kan vara sanna eller falska
  - Ofta representerade med bokstäver, t.ex. ”AI är kul” kan representeras med K
- Logiska konnektiv
  - $\neg$ Negation (inte)
  - $\wedge$ Konjunktion (och)
  - $\vee$ Disjunktion (eller)
  - $\Rightarrow$ Implikation (medför)
  - $\Leftrightarrow$ Ekvivalens
- Parenteser ( )

Syntaxexempel

Formler:

A

$A \land B$

$\neg\neg A$

$A \lor \neg B \Longrightarrow \neg A \land B$

$\neg (A \land B) \lor (A \land \neg B) \land A$

Inte formler:

$\land A$

$A \neg B$

Konjunktion

Syntax, exempel

K: AI är kul

L: Folk är lyckliga

K ∧ L - AI är kul **och** folk är lyckliga

Semantik

K ∧ L är sant om både K och L är sanna

Disjunktion

Syntax, exempel

K: AI är kul

L: Folk är lyckliga

K ∨ L - AI kul **eller** folk är lyckliga.

Semantik

K ∨ L är sant om någon av K eller L är sann, **eller båda**

Negation

Syntax, exempel

K: AI är kul

¬ K - Det är **inte** så att AI är kul

Semantik

¬K är falsk när K är sann. Annars sann.

Implikation

Syntax, exempel

K: AI är kul

L: Folk är lyckliga

$K \Longrightarrow L$ – **Om** AI är kul **så** folk är lyckliga**.**

OBS! Inget om vad som händer om AI inte är kul.

Semantik

$K \Longrightarrow L$ är enbart falsk om K är sann och L är falsk. Annars sann.

Ekvivalens

Syntax, exempel

K: AI är kul

L: Folk är lyckliga

$K \Leftrightarrow L$ – Folk är lyckliga **om och endast om (omm)** AI är kul**.** Är AI kul är folk lyckliga. Är folk lyckliga är AI kul.

Semantik

$K \Leftrightarrow L$ är sann om L och K har samma sanningsvärde. Annars falsk.

Sanningstabell

Kan lista alla sanningsvärden

A	B	¬A	A ∧ B	A ∨ B	A ⟹ B	A ⟺ B
F	F	S	F	F	S	S
F	S	S	F	S	S	F
S	F	F	F	S	F	F
S	S	F	S	S	S	S

Implikation

- $A \Longrightarrow B$ om båda sanna måste satsen vara sann och om A är sann och B falsk måste satsen vara falsk

A	B	$A \Longrightarrow B$ Alt 1	$A \Longrightarrow B$ Alt 2	$A \Longrightarrow B$ Alt 3	$A \Longrightarrow B$ Alt 4
F	F	F	F	S	S
F	S	F	S	F	S
S	F	F	F	F	F
S	S	S	S	S	S
		$\wedge$	B	$\Leftrightarrow$	OK

- Dessutom måste $A \Leftrightarrow B \equiv (A \Longrightarrow B) \wedge (B \Longrightarrow A)$
  - $\equiv$ betyder logisk ekvivalens, sanna i samma modeller (metaspråk)

XOR

- Exklusivt eller (antingen eller), dvs bara sant då en av satserna sann
  $A \oplus B \equiv (A \lor B) \land \neg(A \land B)$

A	B	A ⊕ B
F	F	F
F	S	S
S	F	S
S	S	F

Sanningstabell, exempel

$P \land \neg Q \implies R$

P	Q	R	$P \land \neg Q$	$P \land \neg Q \implies R$
F	F	F	F	S
F	F	S	F	S
F	S	F	F	S
F	S	S	F	S
S	F	F	S	F
S	F	S	S	S
S	S	F	F	S
S	S	S	F	S

Syntax för satslogik

Syntax byggs rekursivt

- Låt satser betecknas med stora bokstäver och formler (som består av satser) med små grekiska
- Om α och β är formler så är följande också formler:
  - $\neg\alpha$
  - $\alpha \wedge \beta$
  - $\alpha \vee \beta$
  - $\alpha \Longrightarrow \beta$
  - $\alpha \Leftrightarrow \beta$
- Ex. Om α = A ∧ B, β = C ∨ D då är $\alpha \wedge \beta$ = A ∧ B ∧ C ∨ D

Prioritesordning och parenteser

$\alpha \land \beta \lor \delta$ kan tolkas som

$(\alpha \land \beta) \lor \delta$ eller $\alpha \land (\beta \lor \delta)$

Formellt krävs parenteser men ofta skippar man dem, ex

$\alpha \land \beta \lor \delta$ tolkas som $(\alpha \land \beta) \lor \delta$

$\alpha \lor \beta \Longrightarrow \delta$ tolkas som $(\alpha \lor \beta) \Longrightarrow \delta$

$\neg\alpha \land \beta$ tolkas som $(\neg\alpha) \land \beta$

Prioritetsordning

$\neg \; \land \; \lor \; \Longrightarrow \; \Leftrightarrow$

Några viktiga logiska ekvivalenser

1.	$\alpha \land \beta \equiv \beta \land \alpha$	kommutativ lag
2.	$\alpha \lor \beta \equiv \beta \lor \alpha$	kommutativ lag
3.	$\alpha \land (\beta \land \delta) \equiv (\alpha \land \beta) \land \delta$	associativ lag
4.	$\alpha \lor (\beta \lor \delta) \equiv (\alpha \lor \beta) \lor \delta$	associativ lag
5.	$\neg\neg\alpha \equiv \alpha$	dubbelnegationseliminering
6.	$\alpha \Longrightarrow \beta \equiv \neg\beta \Longrightarrow \neg\alpha$	kontraposition
7.	$\alpha \Longrightarrow \beta \equiv \neg\alpha \lor \beta$	eliminering av implikation
8.	$\alpha \land (\beta \lor \delta) \equiv (\alpha \land \beta) \lor (\alpha \land \delta)$	distributiv lag
9.	$\alpha \lor (\beta \land \delta) \equiv (\alpha \lor \beta) \land (\alpha \lor \delta)$	distributiv lag
10.	$\neg(\alpha \lor \beta) \equiv \neg\alpha \land \neg\beta$	De Morgan
11.	$\neg(\alpha \land \beta) \equiv \neg\alpha \lor \neg\beta$	De Morgan

”Sanningstabellsbevis”

		Eliminering av implikation		De Morgan		De Morgan	
A	B	$A \implies B$	$\neg A \lor B$	$\neg(A \land B)$	$\neg A \lor \neg B$	$\neg(A \lor B)$	$\neg A \land \neg B$
F	F	S	S	S	S	S	S
F	S	S	S	S	S	F	F
S	F	F	F	S	S	F	F
S	S	S	S	F	F	F	F

Översättningar, 1

Per och Pål är starka

egentligen

Per är stark och Pål är stark

E*: Per är stark*

Å*: Pål är stark*

E ∧ Å

Översättningar, 2

Steken skall kryddas med svartpeppar och vitpeppar eller kryddpeppar

S*: Steken skall kryddas med svartpeppar*

V*: Steken skall kryddas med vitpeppar*

K*: Steken skall kryddas med kryddpeppar*

$(S \land V) \lor K$

$S \land (V \lor K)$

Två olika tolkningar

Översättningar, 3

Du skall ha registrerat dig annars får du inte tenta

annars signalerar att det är nödvändigt att registrera sig för att få tenta

Om du inte registrerat dig får du inte tenta

T*: Du får tenta*

R*: Du har registrerat dig*

$\neg R \Longrightarrow \neg T$

$T \Longrightarrow R$

Konjunktiv normalform

- Standardiserat sätt att skriva formler
- En literal är en sats eller negerad sats
- En formel består bara av en itererad konjunktion av disjunktiva literaler
- $\alpha_1 \land \alpha_2 \land \alpha_3 \dots \land \alpha_n$ där α_i är $\beta_i \lor \delta_i$ och β eller δ är literaler, eller tomma
- T.ex. $(A \lor B) \land (C \lor \neg D) \land E$
- Alla uttryck går att skrivas om till konjunktiv normalform (eller disjunktiv normalform)

Konvertering till konjunktiv normalform

För satslogik i princip tre enkla samband

1. Eliminera implikation
   $\alpha \Longrightarrow \beta \equiv \neg\, \alpha \vee \beta$
2. Reducera negationernas räckvidd
   de Morgan
   $\neg(\alpha \vee \beta) \equiv \neg\alpha \wedge \neg\beta$ och $\neg(\alpha \wedge \beta) \equiv \neg\alpha \vee \neg\beta$
3. Konvertera till konjunktion av disjunktioner
   Distributiva lagen
   $\alpha \wedge (\beta \vee \delta) \equiv (\alpha \wedge \beta) \vee (\alpha \wedge \delta)$
   $\alpha \vee (\beta \wedge \delta) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \delta)$

Exempel

$A \Longrightarrow (B \land \neg(C \land D))$

1. $\neg A \lor (B \land \neg(C \land D))$
2. $\neg A \lor (B \land (\neg C \lor \neg D))$
3. $(\neg A \lor B) \land (\neg A \lor \neg C \lor \neg D)$