

# Was bisher geschah

Wissensrepräsentation und -verarbeitung in Logiken:

**Wissensbasis:** Kontextwissen

**Formulierung der Aufgabe:** fallspezifisches Wissen

**Lösung:** Bedingungen

**Lösungsverfahren**

WH – klassische Aussagenlogik:

- ▶ Syntax
- ▶ Semantik
- ▶ semantisches Schließen

# Wiederholung: Syntaktisches Ableiten

gegeben: Formelmenge  $\Phi$

Formel  $\psi$

Frage : Gilt  $\Phi \models \psi$  ?

Ziel: Verfahren zur Beantwortung dieser Frage durch **syntaktische** Operationen

(ohne Benutzung der Semantik, Modellmengen)

**Syntaktische Ableitungsrelation**  $\vdash \subseteq 2^{\text{AL}(P)} \times \text{AL}(P)$

**passend** zur

semantischen Folgerungsrelation  $\models \subseteq 2^{\text{AL}(P)} \times \text{AL}(P)$

$\vdash$  **passt** zu  $\models$ , falls für jede Formelmenge  $\Phi \in \text{AL}(P)$  und jede Formel  $\psi \in \text{AL}(P)$  gilt

$$\Phi \vdash \psi \quad \text{gdw.} \quad \Phi \models \psi$$

# Wiederholung: Syntaktisches Ableiten

gegeben: Formel  $\varphi$  (Formelmenge  $\Phi$ )

Formel  $\psi$

Frage: Gilt  $\Phi \models \psi$

Idee: schrittweises Ableiten (ohne Zugriff auf die Semantik der Formeln) von Folgerungen aus einer Formelmenge durch **syntaktische Umformungen**

**logischer Kalkül** Menge von Regeln zur syntaktischen Umformung von Formeln (Formelmengen)  
(ohne Änderung der Semantik der Formelmengen)

Ein logischer Kalkül  $K$  ist sinnvoll, wenn man zeigen kann:

**Korrektheit** Jede in  $K$  ableitbare Formel ist allgemeingültig.

**Vollständigkeit** Jede allgemeingültige Formel ist in  $K$  ableitbar.

# Wiederholung: Aussagenlogische Resolution

Formeln  $p \vee \psi, \neg p \vee \eta$  haben die **Resolvente**  $\psi \vee \eta$

## Satz (Resolutionslemma)

*Für jede CNF (Klauselmeng)e  $\Phi$  und die Resolvente  $R$  zweier Klauseln aus  $\Phi$  gilt*

$$\text{Mod}(\Phi) = \text{Mod}(\Phi \cup \{R\})$$

Idee: Schrittweise Erweiterung der Formelmeng)e  $\Phi$  um Resolventen

Anwendung der **Resolutionsregel**:

$$\{\psi \vee p, \neg p \vee \eta\} \rightarrow \{\psi \vee p, \neg p \vee \eta, \psi \vee \eta\}$$

alternative Darstellung:

$$\{\neg\psi \rightarrow p, p \rightarrow \eta\} \rightarrow \{\neg\psi \rightarrow p, p \rightarrow \eta, \neg\psi \rightarrow \eta\}$$

Spezialfall: endliche Menge  $\Phi$  von Formeln in CNF

## Wiederholung: Ableitungen durch Resolution

**Resolutionsableitung** aus einer Klauselmenge  $\Phi$  (CNF):  
endliche Folge  $C_1, \dots, C_n$  von Klauseln, wobei für jede Klausel  $C_i$  gilt:

- ▶  $C_i \in \Phi$  oder
- ▶  $C_i$  ist eine Resolvente von Klauseln  $C_j, C_k$  mit  $j < i$  und  $k < i$ .

Resolutionsableitung **der Klausel  $\psi$**  aus Klauselmenge  $\Phi$ :  
Resolutionsableitung  $C_1, \dots, C_n$  in  $\Phi$  mit  $C_n = \psi$

Beispiel: Resolutionsableitung von  $d$  aus

$$\Phi = \{a \vee b \vee c, \neg b \vee d, \neg a \vee d, \neg c \vee d\}$$

Baumdarstellung (Tafel)

## Wiederholung: Resolutionsableitungen von $\mathbf{f}$

Problem:

Es existiert **keine** Resolutionsableitung von  $\neg a \vee \neg b \vee d$  aus

$$\Phi = \{a \vee b \vee c, \neg b \vee d, \neg a \vee d, \neg c \vee d\}$$

aber es gilt  $\Phi \models \neg a \vee \neg b \vee d$ .

Lösungsidee:

Es gilt  $\Phi \models \psi$  gdw.  $\Phi \cup \{\neg\psi\}$  unerfüllbar.

Unerfüllbarkeitsbeweis für  $\Phi \cup \{\psi\}$  durch Resolutionsableitung von  $\mathbf{f}$  aus  $\Phi \cup \{\neg\psi\}$  (Klauselform)

Beispiel (Tafel): Resolutionsableitung von  $\mathbf{f}$  aus

$$\Phi \cup \{\neg\psi\} = \{a \vee b \vee c, \neg b \vee d, \neg a \vee d, \neg c \vee d, a, b, \neg d\}$$

## Wiederholung: Syntaktische Ableitungsrelation $\vdash_R$

Schon gezeigt:

Für jede Formelmenge  $\Phi \subseteq \text{AL}(P)$  und jede Formel  $\psi \in \text{AL}(P)$  gilt:

$$\Phi \models \psi \quad \text{gdw.} \quad \Phi \cup \{\neg\psi\} \text{ unerfüllbar}$$

Syntaktische Ableitungsrelation  $\vdash_R \subseteq 2^{\text{AL}(P)} \times \text{AL}(P)$ :

$\Phi \vdash_R \psi$  gdw.

eine Resolutionsableitung für  $\mathbf{f}$  aus  $\Phi \cup \{\neg\psi\}$  existiert.

Beispiele:

- ▶  $\{a \vee b \vee c, (a \vee b) \rightarrow d, c \rightarrow e, \neg d\} \vdash_R e$
- ▶  $(\neg p \vee q) \wedge (\neg q \vee r) \wedge p \wedge \neg r$  ist unerfüllbar.
- ▶  $\phi = (q \wedge r) \vee (\neg p \wedge \neg q \wedge r) \vee p \vee (\neg p \wedge \neg r)$  ist allgemeingültig.

## Wiederholung: Korrektheit und Vollständigkeit

Die folgenden beiden Sätze zeigen, dass  $\vdash_R$  zu  $\models$  **passt**, d.h.  
 $\Phi \vdash_R \psi$  gdw.  $\Phi \models \psi$

### Satz (Korrektheit der Ableitungsrelation $\vdash_R$ )

Für jede Formelmenge  $\Phi \subseteq \text{AL}(P)$  und jede Formel  $\psi \in \text{AL}(P)$  gilt:  
Aus  $\Phi \vdash_R \psi$  folgt  $\Phi \models \psi$   
(Wenn eine Resolutionsableitung von  $\mathbf{f}$  aus einer zu  $\Phi \cup \{\neg\psi\}$  äquivalenten Klauselmenge existiert, dann gilt  $\Phi \models \psi$ .)

### Satz (Vollständigkeit der Ableitungsrelation $\vdash_R$ )

Für jede Formelmenge  $\Phi \subseteq \text{AL}(P)$  und jede Formel  $\psi \in \text{AL}(P)$  gilt:  
Aus  $\Phi \models \psi$  folgt  $\Phi \vdash_R \psi$   
(Wenn  $\Phi \models \psi$  gilt, dann existiert eine Resolutionsableitung von  $\mathbf{f}$  aus einer zu  $\Phi \cup \{\neg\psi\}$  äquivalenten Klauselmenge.)

# Modellierungsbeispiel in Prädikatenlogik (1. Stufe)

Wissensbasis (Aufgabenbereich):

allgemein:

- ▶ Personen mit einem gleichen Elternteil sind Geschwister.
- ▶ Nichten sind weibliche Kinder von Geschwistern.

speziell:

- ▶ Tina ist die Tochter von Anna und Max.
- ▶ Paul und Berta sind die Eltern von Anna und Otto.

Formeln ...

Frage Wer ist wessen Nichte?

Lösung ...

# Wiederholung Prädikatenlogik: Syntax

Ziel: Modellierung von Aussagen über Eigenschaften und Beziehungen von Objekten eines bestimmten Bereiches

**Signatur**  $\Sigma = (\Sigma_F, \Sigma_R)$  Funktions- und Relationssymbole

**(Individuen-)Variablen**  $\mathbb{X}$

**Terme**  $\text{Term}(\Sigma_F, \mathbb{X})$ , induktive Definition:

**IA:**  $\mathbb{X} \subseteq \text{Term}(\Sigma_F, \mathbb{X})$

**IS:** Aus  $(f, n) \in \Sigma_F$  und  $t_1, \dots, t_n \in \text{Term}(\Sigma_F, \mathbb{X})$   
folgt  $f(t_1, \dots, t_n) \in \text{Term}(\Sigma_F, \mathbb{X})$ .

**Atome**  $\text{Atom}(\Sigma, \mathbb{X})$ :

Aus  $(p, n) \in \Sigma_R$  und  $t_1, \dots, t_n \in \text{Term}(\Sigma_F, \mathbb{X})$  folgt  
 $p(t_1, \dots, t_n) \in \text{Atom}(\Sigma, \mathbb{X})$

**Formeln**  $\text{FOL}(\Sigma, \mathbb{X})$  induktive Definition:

**IA:**  $\text{Atom}(\Sigma, \mathbb{X}) \subseteq \text{FOL}(\Sigma, \mathbb{X})$

**IS:** Falls  $j$  ein  $n$ -stelliger Junktor ist,  $x \in \mathbb{X}$  und  
 $\varphi_1, \dots, \varphi_n \in \text{FOL}(\Sigma, \mathbb{X})$ , dann gilt

$j(\varphi_1, \dots, \varphi_n) \in \text{FOL}(\Sigma, \mathbb{X})$ ,  $\forall x \varphi \in \text{FOL}(\Sigma, \mathbb{X})$   
und  $\exists x \varphi \in \text{FOL}(\Sigma, \mathbb{X})$ ,

# Wiederholung Prädikatenlogik: Semantik

$\Sigma$ -Struktur  $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$  mit

- ▶ nichtleerer Menge  $A$  (Trägermenge)
- ▶ Interpretation  $\llbracket \cdot \rrbracket_{\mathcal{A}}$  der Funktions- und Relationssymbole aus  $\Sigma$ 
  - ▶ für jedes  $(f, n) \in \Sigma_F$  eine Funktion  $\llbracket f \rrbracket_{\mathcal{A}} : A^n \rightarrow A$
  - ▶ für jedes  $(p, n) \in \Sigma_R$  eine Relation  $\llbracket p \rrbracket_{\mathcal{A}} \subseteq A^n$

**Belegung**  $\beta : X \rightarrow A$  der Individuenvariablen

Eine **Interpretation**  $(\mathcal{A}, \beta)$  für Term  $t \in \text{Term}(\Sigma_F, X)$  oder Formel  $\varphi \in \text{FOL}(\Sigma, X)$

- ▶ einer  $\Sigma$ -Struktur  $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$  und
- ▶ einer Variablenbelegung  $\beta : X \rightarrow A$ .

Menge aller Modelle der Formel  $\varphi \in \text{FOL}(\Sigma, X)$

$$\text{Mod}(\varphi) = \left\{ (\mathcal{S}, \beta) \mid \begin{array}{l} (\mathcal{S}, \beta) \text{ ist } \Sigma\text{-Interpretation und} \\ \llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)} = 1 \end{array} \right\}$$

## Wiederholung: Einbettung Aussagen- in Prädikatenlogik

Jede Formel  $\varphi \in \text{AL}(P)$  ist (syntaktisch) auch eine Formel in  $\text{FOL}(\Sigma, \emptyset)$  mit

$$\begin{aligned}\Sigma &= (\Sigma_F, \Sigma_R) \\ \Sigma_F &= \emptyset \quad \text{und} \quad \Sigma_R = \{(p, 0) \mid p \in P\}\end{aligned}$$

Semantik:

Jede Belegung  $W : P \rightarrow \{0, 1\}$  für  $\varphi \in \text{AL}(P)$  definiert eine  $\Sigma$ -Struktur  $\mathcal{S}_W = (\mathcal{S}, \llbracket \cdot \rrbracket_{\mathcal{S}_W})$  durch

$$\forall p \in P : \llbracket \cdot \rrbracket_{\mathcal{S}_W} = W(p)$$

$W$  ist Modell (erfüllende Belegung) für  $\varphi \in \text{AL}(P)$

gdw.  $W(\varphi) = 1$

gdw.  $\mathcal{S}_W$  ist Modell für  $\varphi \in \text{FOL}(\Sigma)$ .

( $\varphi \in \text{FOL}(\Sigma)$  enthält keine Individuenvariablen, Variablenbelegung deshalb irrelevant)

Aussagenlogik ist ein **Fragment** der Prädikatenlogik

# Wiederholung: Übersetzung Prädikaten- in Aussagenlogik

**Grundinstanziierung** einer Formelmenge  $\Phi \in \text{FOL}(\Sigma, \mathbb{X})$

in der  $\Sigma$ -Struktur  $\mathcal{A} = (A, [\cdot]_{\mathcal{A}})$ :

1. Definition eines neuen Konstantensymbols für jedes  $d \in A$
2. Ersetzung der Formeln  $\varphi \in \Phi$  durch ihre Grundinstanzen (induktiv):
  - IA: für Atome  $\varphi = p(t_1, \dots, t_n)$ :  
 $G(\varphi) = G((p(t_1, \dots, t_n)) = \{\beta(p(t_1, \dots, t_n)) \mid \beta : \mathbb{X} \rightarrow A\}$
  - IS: für Formeln mit  $n$ -stelligen Junktor  $j$   
 $\varphi = j(\varphi_1, \dots, \varphi_n)$ :  $G(\varphi) = j(G(\varphi_1), \dots, G(\varphi_n))$
3. Übersetzung  $\exists x \varphi \rightarrow \bigvee_{d \in A} \varphi[x \mapsto d]$  und  $\forall x \varphi \rightarrow \bigwedge_{d \in A} \varphi[x \mapsto d]$
4. Ersetzung jedes Grundatoms durch eine Aussagenvariable,
5.  $G(\Phi) = \bigcup_{\varphi \in \Phi} \{\beta(\varphi) \mid \beta : \mathbb{X} \rightarrow A\}$

Beispiele:  $p(x, f(y)) \wedge \exists x p(y, x)$  in  $A = \{1, 2, 3\}$ , n-Damen-Problem

**Vorteil:** aussagenlogische Methoden anwendbar, Entscheidbarkeit

- Nachteil:**
- ▶ nur für Interpretationen in Strukturen mit **endlicher** Trägermenge und Signaturen ohne  $< 0$ -stellige Funktionen (nur Konstanten) möglich
  - ▶ große unübersichtliche Formelmengen

## Wiederholung: Prolog-Syntax

**Regel** (Horn-Klausel)  $a :- a_1, \dots, a_m.$

Bedeutung in Prädikatenlogik (der ersten Stufe)

$$(\forall X_1 \dots \forall X_n ((a_1 \wedge \dots \wedge a_m) \rightarrow a))$$

wobei  $X_1, \dots, X_n$  alle in  $a, a_1, \dots, a_m$  vorkommenden Variablen sind.

Rumpf  $a_1 \wedge \dots \wedge a_m$ , Kopf  $a$

**Fakt** Atom  $a.$  (positives Literal, Regel ohne Rumpf)

Bedeutung in Prädikatenlogik (der ersten Stufe):

$$\forall X_1 \dots \forall X_n a, \text{ wobei } X_1, \dots, X_n \text{ alle in } a$$

vorkommenden Variablen sind.

**Zielklausel** (Query, Anfrage)  $?- a_1, \dots, a_n.$

Bedeutung in Prädikatenlogik (der ersten Stufe):

$$(\forall X_1 \dots \forall X_n (a_1 \wedge \dots \wedge a_m))$$

wobei  $X_1, \dots, X_n$  alle in  $a_1, \dots, a_m$  vorkommenden Variablen sind.

Variablenamen beginnen mit Großbuchstaben,

Funktions- und Relationssymbole mit Kleinbuchstaben

## Wiederholung: Prolog-Programme

Programm  $P$  (Wissensbasis):

endliche Menge von Fakten und Regeln,  
repräsentiert eine prädikatenlogische Formelmenge  $\Phi$ ,  
(repräsentiert eine prädikatenlogische Formel  
 $\varphi = \bigwedge_{\psi \in \Phi} \psi$ )

Beispiel: Programm  $P$

liest(paul,krimi).

liest(bob,zeitung).

liest(tina,arztroman).

mag(tina,X) :- liest(X,krimi).

repräsentiert die Formelmenge

$$\Phi = \{l(p, k), l(b, z), l(t, a), \forall x(l(x, k) \rightarrow m(t, x))\}$$

# Wiederholung: Prolog-Anfragen

Zielklausel Atom

repräsentiert eine prädikatenlogische Formel  $\psi$

Beispiel:  $?- \text{mag}(\text{tina}, X).$

repräsentiert die Frage: Wen mag Tina?

repräsentiert durch die Formel (Behauptung):  $\psi = \exists X \text{mag}(\text{tina}, X)$

negiertes Einfügen in den Kontext:  $\neg\varphi = \forall X \neg \text{mag}(\text{tina}, X)$

also Einfügen der Klausel:  $\neg \text{mag}(\text{tina}, X)$

## Wiederholung: Prolog-Auswertung

Ausgewertet werden Paare  $(\Phi, \psi)$  aus

- ▶ Programm  $\Phi$
- ▶ Zielklausel  $\psi$

(prädikatenlogische Darstellung von Programm und Anfrage)

Antwort: Substitution  $\theta$  mit  $\Phi \models \theta(\psi)$

(Prolog-Ausgabe: Grundinstanzen  $\theta(\psi)$  der Zielklausel  $\psi$ )

## Wiederholung Prolog: Bestimmung der Antworten

- ▶ durch Lösung der Aufgabe:  
Für welche Substitutionen  $\theta$  gilt  $\Phi \models \theta(\psi)$ ?
- ▶ durch Lösung der äquivalenten Aufgabe:  
Für welche Substitutionen  $\theta$  ist die Formelmenge  $\Phi \cup \neg\theta(\psi)$  unerfüllbar?
- ▶ durch Lösung der äquivalenten Aufgabe:  
Für welche Substitutionen  $\theta$  gilt  $\Phi \cup \neg\theta(\psi) \models \mathbf{f}$  ?
- ▶ durch Bestimmung der Substitutionen  $\theta$ , für die  $\mathbf{f}$  aus  $\Phi \cup \neg\theta(\psi)$  syntaktisch herleitbar ist.
- ▶ durch **prädikatenlogische Resolution** mit festgelegter Auswertungsreihenfolge (SLD-Resolution)

Ausführung durch Prolog-Interpreter,  
z.B. SWI-Prolog <http://www.swi-prolog.org>  
oder online <http://www.learnprolognow.org>

# Wiederholung: Beispiel für Prolog-Auswertung

Programm  $P$ :

$p(a,b) . p(b,c) . p(c,d) .$

$e(X,Y) :- p(X,Y) .$

$e(X,Y) :- p(X,Z) , e(Z,Y) .$

Zielklausel  $e(X, d)$

Prädikatenlogische Bedeutung:

► Programm  $P$ :

$$\Phi = \left\{ \begin{array}{l} p(a,b), p(b,c), p(c,d), \forall X \forall Y (p(X,Y) \rightarrow e(X,Y)), \\ \forall X \forall Y \forall Z (p(X,Z) \wedge e(Z,Y) \rightarrow e(X,Y)) \end{array} \right\}$$

► Zielklausel  $\psi = e(X, d)$ , negiert  $\neg\psi = \neg e(X, d)$

► kombiniert als Formelmenge

$$\{\neg\psi\} \cup \Phi = \left\{ \begin{array}{l} \neg e(X, d), p(a,b), p(b,c), p(c,d), \\ \forall X \forall Y (p(X,Y) \rightarrow e(X,Y)), \\ \forall X \forall Y \forall Z (p(X,Z) \wedge e(Z,Y) \rightarrow e(X,Y)) \end{array} \right\}$$

Antworten:  $X = c$  ;  $X = a$  ;  $X = b$  ; No

# Prädikatenlogische Resolution

Berechnung einer prädikatenlogischen Resolvente der Klauseln  $l_1 \vee \dots \vee l_n$  und  $l'_1 \vee \dots \vee l'_m$ :

- ▶ Variablenumbenennung (Klauseln haben keine gemeinsamen Variablen)
- ▶ Bestimmung eines allgemeinsten Unifikators  $\sigma$  für ein Paar von Literalen  $l_i$  und  $\neg l'_j$
- ▶ Resolvente:  $\sigma(l_1) \vee \dots \vee \sigma(l_{i-1}) \vee \sigma(l_{i+1}) \vee \dots \vee \sigma(l_n) \vee \sigma(l'_1) \vee \dots \vee \sigma(l'_{j-1}) \vee \sigma(l'_{j+1}) \vee \dots \vee \sigma(l'_m)$

# Beispiel

Klauselmeng

$$\Phi = \{P(x, b) \vee P(a, y) \vee Q(x, f(y)), \neg P(z, w), \neg Q(w, z)\}$$

Resolution:

positive Literale  $P(x, b), P(a, y)$

negatives Literal  $\neg P(z, w)$

Substitution  $\sigma = [x \mapsto a, y \mapsto b, z \mapsto a, w \mapsto b]$

Resolvente  $Q(a, f(b)) \vee \neg Q(b, a)$

# Prolog – Operationale Semantik (SLD-Resolution)

für Mengen von Hornklauseln

Auswahl der angewendeten Regeln in einer festen Reihenfolge:

- ▶ erste anwendbare Regel im Programm,
- ▶ in jedem Schritt entsteht eine Hornklausel
- ▶ neu erzeugte Regel sofort anwenden
- ▶ zuerst das erste Rumpf-Literal resolvieren
- ▶ Auswahl nur für dessen Resolutions-Partner (Suchbaum, meist Tiefensuche)

(immer Substitution  $\beta$  mitführen)

# Prolog – Deklarative Semantik

Prolog-Programm  $P$  = endliche Menge von Fakten und Regeln  
(Formelmeng)

deklarative Semantik (Bedeutung) von  $P$ :

Menge aller Modelle der Formelmeng  $P$

enthält für Hornklausel-Programme  $P$  genau ein **minimales** Modell  
(bzgl.  $\subseteq$ )

Prolog-Programm  $P$ , Anfrage  $\psi$

deklarative Semantik (Bedeutung) von  $P$  mit Anfrage  $\psi$ :

Menge aller Variablenbelegungen  $\beta$  (Antwort), für die  $\beta(\psi)$  im  
minimalen Modell von  $P$  gilt

# Ideales und Reales Prolog

wie hier definiert (ideal, genügt hier):

- ▶ Semantik ist deklarativ
- ▶ Reihenfolge der Regeln im Programm und Atome in Regel-Rumpf beeinflusst Effizienz, aber nicht Korrektheit

reales Prolog:

- ▶ cut (!) zum Abschneiden der Suche
  - ▶ green cut: beeinflusst Effizienz
  - ▶ red cut: ändert Semantik

merke: cut  $\approx$  goto, grün / rot schwer zu unterscheiden

- ▶ Regeln mit Nebenwirkungen (u. a. für Ein/Ausgabe)

für beides: keine einfache denotationale Semantik