Folgern im PK1

- Fl ist syntaktisch beschreibbar mittels Abl
  Fl = Abl „modulo **ag**"
- Fl und Abl sind monoton:
  $\mathbf{X} \subseteq \mathbf{Y} \Rightarrow \text{Fl}(\mathbf{X}) \subseteq \text{Fl}(\mathbf{X})$
  $\mathbf{X} \subseteq \mathbf{Y} \Rightarrow \text{Abl}(\mathbf{X}) \subseteq \text{Abl}(\mathbf{X})$
- $\mathbf{ag} = \text{Fl}(\emptyset) = \text{Abl}(\mathbf{axp})$
- $(H \to G) \in \text{Fl}(\mathbf{X}) \Leftrightarrow G \in \text{Fl}(\mathbf{X} \cup \{H\})$
- $H \in \text{Fl}(\mathbf{X}) \Leftrightarrow (\wedge\mathbf{X} \to H) \in \mathbf{ag}$

Formale Theorien im PK1

Als *Theorie* wird eine bezüglich Folgern (Ableiten) abgeschlossene Menge **Th** von Ausdrücken bezeichnet:

$\mathbf{Th} = \text{Fl}(\mathbf{Th}) = \text{Abl}(\mathbf{Th})$ „modulo **ag**"

- Theorie mit *semantisch bestimmter Satzmenge*:
  Gegeben ist eine Struktur S = [U, I] mit
  $\mathbf{Th} = \{ F \mid F \text{ allgemeingültig in } S\}$
  $= \{ F \mid \text{Wert}_S(F, \beta) = W \text{ für alle } \beta \text{ über } U \}$
- Theorie mit *syntaktisch bestimmter Satzmenge*:
  Gegeben ist ein Ausdruckmenge **Ax** („Axiome") mit
  $\mathbf{Th} = \text{Abl}(\mathbf{Ax} \cup \mathbf{axp})\quad (= \text{Fl}(\mathbf{Ax}))$

Formalisierung mittels PK1

Ziel: Axiome zur Beschreibung von Sachverhalten finden
Analoges Ziel: **Computerverarbeitung ermöglichen**

1. semantisch definierte Theorie **Th** bestimmen
   (Universum, Relationen/Funktionen, Interpretation)
2. axiomatische Beschreibung von **Th** :
   Axiome **Ax** mit $\mathbf{Th} = \mathrm{Abl}(\mathbf{Ax} \cup a\mathbf{xp})$

   (z.B. PROLOG-Programm)

Anwendung PK1

- Formalisierung: Axiome **X**
- Problem durch Ausdruck H beschreiben
- Entscheiden, ob

$$H \in \mathrm{Fl}(\mathbf{X})$$

d.h. ob $H \in \mathrm{Abl}(\mathbf{X} \cup \mathbf{axp})$

Programme dafür: Theorembeweiser

Beweise

- Positiver Kalkül: Allgemeingültigkeit entscheiden

  $\boxed{H \in \mathbf{Th}\ ?}$

- Negativer Kalkül: Unerfüllbarkeit untersuchen

  $\boxed{\{\neg H\} \cup \mathbf{Th} \text{ widersprüchlich ?}}$

- Deduktiver Kalkül:
  Erweitern der Axiome um H zu finden (foward chaining)

- Testkalkül:
  Reduktion von H auf Axiome (backward chaining)

Beispiel: Resolution (PROLOG) ist *Negativer Testkalkül*

Beweise

$H \in Fl(\mathbf{X})$

gilt gdw. $H \in Abl(\mathbf{X} \cup \mathbf{axp})$ ⇨ Positiver Kalkül

$H \in Fl(\mathbf{X})$

gilt gdw. $(\wedge\mathbf{X} \rightarrow H) \in \mathbf{ag}$

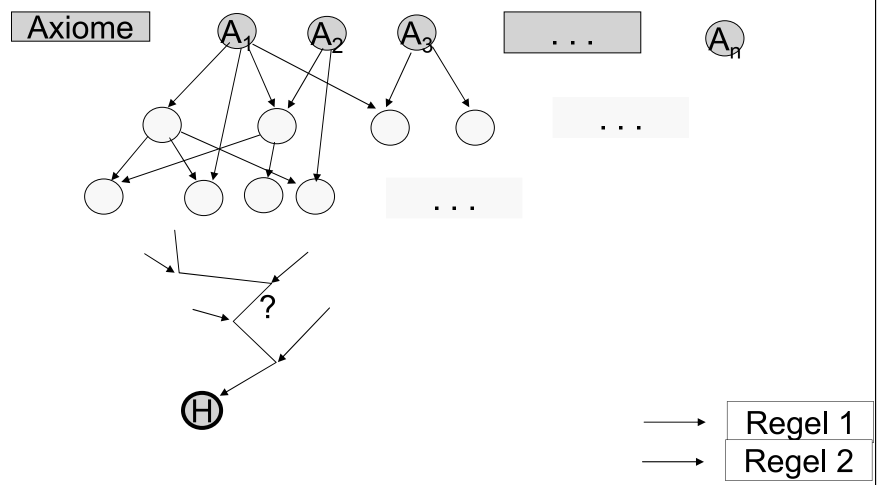
gilt gdw. $\neg(\wedge\mathbf{X} \rightarrow H) \notin \mathbf{ef}$

gilt gdw. Skolemform von $(\wedge\mathbf{X} \wedge \neg H)$ nicht erfüllbar

gilt gdw. Klauselform von $(\wedge\mathbf{X} \wedge \neg H)$ nicht erfüllbar

⇨ Negativer Kalkül

Positiver deduktiver Kalkül

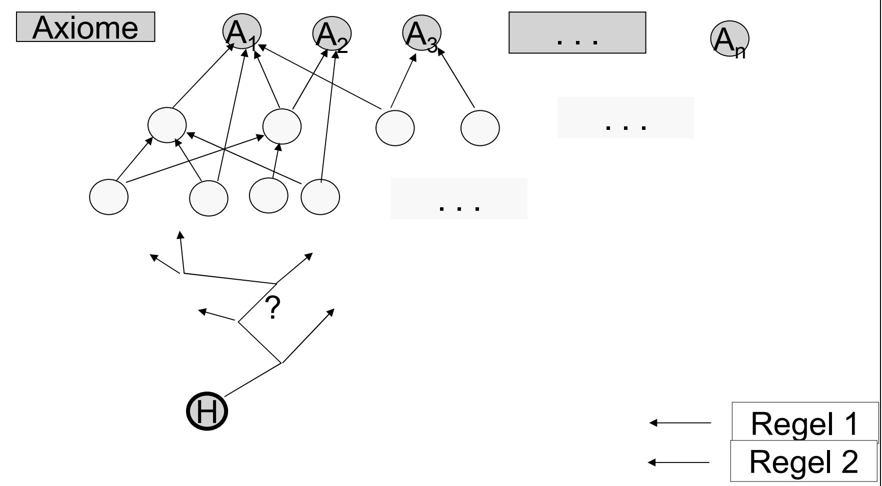


Suchraum bei deduktivem Kalkül

- Klassischer AK : 15 Axiome für **ag**, 2 Regeln
  (ist entscheidbar, allerdings NP)

- Klassischer PK1:
  - abzählbar viele Axiome für **ag** + weitere für **Th** ,
  - 7 Regeln
- Vollständigkeit als Nachteil:
  - Alle allgemeingültigen Ausdrücke im Suchraum **Th** :
    $\mathbf{ag} = FI(\emptyset) \subseteq FI(\mathbf{Th}) = \mathbf{Th}$
  - Mit H viele weitere Ausdrücke im Suchraum **Th** :
    z.B. $H \vee G$ für beliebiges G

Positiver Test-Kalkül



Normalformen

- ein Ausdruck H heißt *bereinigt*, falls gilt:
  - keine Variable x kommt in H sowohl frei als auch gebunden vor,
  - hinter den Quantoren in H vorkommende Variable sind verschieden.
- ein Ausdruck H heißt *pränex*, falls gilt:
  - H hat die Form $Q_1 x_1 \ldots Q_n x_n H'$,
    wobei $Q_1, \ldots, Q_n$ Quantoren sind und H´ keine Quantoren enthält.
- ein Ausdruck H heißt *pränexe Normalform*, falls gilt:
  - H hat pränexe Form $Q_1 x_1 \ldots Q_n x_n H'$,
  - H´ ist eine konjunktive Normalform (KNF):

$$H' = \bigwedge \{ \{ \bigvee L_{ij} \mid i=1\ldots n \} \mid j=1\ldots m_n \}$$

(die L_{ij} sind Literale)

Literal: atomare Formel oder negierte atomare Formel

Normalformen

Für alle S, β :
$\text{Wert}_S(H_1, \beta) = \text{Wert}_S(H_2, \beta)$

Sätze

1. Zu jedem Ausdruck H_1 existiert ein semantisch äquivalenter Ausdruck H_2 in bereinigter Form.
2. Zu jedem Ausdruck H_2 in bereinigter Form existiert ein semantisch äquivalenter Ausdruck H_3 in bereinigter pränexer Form
3. Zu jedem Ausdruck H_3 in bereinigter pränexer Form existiert semantisch äquivalenter Ausdruck H_4 in bereinigter pränexer Normalform.

Zu jedem Ausdruck H existiert ein semantisch äquivalenter Ausdruck G in bereinigter pränexer Normalform.

Skolem-Normalform

Eine Skolem-Normalform ist eine pränexe Normalform, deren Variable sämtlich generalisiert sind.

Umformung einer bereinigten pränexen Normalform G in eine Skolem-Normalform F:

1. $\exists$-Quantoren einführen für alle freien Variablen in G.
   (erfüllbarkeits-äquivalente Umformung !!)
2. Elimimation aller $\exists$-Quantoren durch Skolem-Funktionen.

Die entstehende Formel F ist erfüllbarkeits-äquivalent zu G.

d.h. F erfüllbar gdw. G erfüllbar