Deduktive Datenbanksysteme

Problem: Transitiver Abschluss ist in PL1 nicht formulierbar (mit zustandsabhängiger Formulierung möglich)
Diskussion:
Typ 5: $q_1(...), ..., q_n(...) : -.$
Typ 6: $q_1(...), ..., q_n(...) : -p_1(...), ..., p_m(...).$
⇒ Übungsaufgabe

Nur Typ 1 und Typ 4: $q(...) : -p_1(...), ..., p_n(...), n \geq 0$ (ist die Hornklauselform und wird bei definiten Datenbanken genutzt)

Definite Datenbanken

$$\begin{aligned} &q(...) : -.(\text{Fakt}) \\ &q(...) : -p_1(...), ..., p_n(...).(\text{deduktive Regel, } p_{1-n} \text{ Teilziele}) \end{aligned} \tag{1}$$

- Mit IBen (Integritätsbedingungen) (+ Typ2, Typ3) ($: -p_1(...), ..., p_n(...)$)
- Typ5 + Typ6 ⇒ **Disjunktive Datenbank**
- Definite Datenbank + negative Atome im Rumpf von Hornklauseln erlaubt ⇒ Volles Datalog

Formulierung von Anfragen

Klauseln vom Typ: $: -p_1(...), ..., p_n(...)$, geschrieben $? - p_1(...), ..., p_n(...)$
Beispiele:

- $? - ag(X, m).$
  - Bedeutung: Welche Kurse bietet 'm' an?
  - DRC: (x) / ANGEBOT(X, m)
- $? - ag(a3, m).$
  - Bedeutung: Bietet 'm' den Kurs 'a3' an?
  - DRC: () / ANGEBOT(a3, m)
- $? - ag(X, m), bl(X, s, j).$

- Bedeutung: Gib alle von 'm' angebotene Kurse, die 's' als Wiederholer belegt hat
- DRC: (x) / ANGEBOT(x, 'm') ∧ BELEGUNG(x, 's', 'y')

- Wie ist (x) / ANGEBOT(x, 'm') ∧(∃y) BELEGUNG(x, 's', y) formulierbar?
  - Bedeutung: Gib die Dozenten der von s als Wiederholer belegte Kurse
  - Formulierung: $? - Ksm(X)$.
    $Ksm(X) : -ag(X, m), bl(X, s, y)$
  - Bequemer: $? - ag(X, Y*), bl(X, s, y)$., * Kennzeichnet die Ausgabevariable

In Anfragesprachen werden Vergleichsausdrücke benötigt. Dazu sind in Datalog spezielle vordefinierte Prädikate vorhanden. Für jeden Vergleichsoperator wird die Existenz eines solchen Prädikates angenommen.
Zunächst: Beschränkte Variablen in Regeln. Sei eine Regel r gegeben:

- Jede Variable, die als Argument in einem gewöhnlichen Prädikat im Rumpf von r vorkommt ist beschränkt.
- Jede Variable, die in einem Teilziel $X = c$ oder $c = X$ von r vorkommt, ist beschränkt.
- Eine Variable X ist beschränkt, wenn sie in einem Teilziel $X = Y$ oder $Y = X$ von r vorkommt mit Y ist schon als beschränkt bekannt.

Definition: sicher

Eine Regel heißt sicher, wenn alle in ihr vorkommenden Variablen beschränkt sind.

Beispiele:

- $Kls(X, Y) : -bl(Z, s, j), ag(Z, Y), X = Z$. **sicher**
- $vsj(X, Y) : -bl(Y, s, j)$. **nicht sicher** (X ist nicht beschränkt)
- $vs(X, Y) : -vs(X, Z), kp(Z, Y)$. **sicher, wenn vs terminiert**
- $kla(Z, Y) : -bl(Z, V, j), ag(Z, Y), V \neq s$. **sicher**

Bemerkung: Falls keine Build-in Prädikate erlaubt sind (/vorkommen):
Eine Regel ist sicher genau dann wenn jede Variable im Kopf der Regel auch im Rumpf der Regel vorkommt.

Definition: Datalog Programm

Ein Datalog-Programm P (ohne IBen(Integritätsbedingungen)) ist eine endliche Menge von Horn-Klauseln mit Jedes $d \in P$ ist entweder

- ein Fakt $q(...)$. ohne Variable
- eine sichere Regel $q(...) : -p_1(...), ..., p_n(...)$. mit $q \in iPraedikat$

Ein $d \in P$ heißt auch **Datalog-Klausel** Alle Fakten zu extensionalen Prädikaten sind als in DB-Relationen gespeichert zu denken.

Beispiel Datenbankzustand:

$$\begin{aligned}
&ag(a1, m).\\
&kp(c2, a0).\\
&...\\
&rb(a1, r1, t1) \textit{(Kurs a1 im Raum r1 zu t1)}\\
&rb(a3, r2, t4)
\end{aligned} \tag{2}$$

Angebot:

Kursnummer	Dozent
a1	m
...	...

Kursplan:

Kursnummer	Voraussetzung
c2	a0
...	...

Raumbelegung:

Kursnummer	Raum	Zeit
a1	r1	t1
...	...	
	...	

Belegung:

Kursnummer	Teilnehmer	Wiederholer
a1	s	j
...	...	
	...	

$$\begin{aligned} vs(X,Y) &:- Kp(X,Y). \\ vs(X,Y) &:- vs(X,Z), Kp(Z,Y). \\ stdpl(W,X,Y,Z) &:- bl(X,W,V), rb(X,Y,Z). \\ ueberschneidungen(X,Y) &:- Kp(Z,X), Kp(Z,Y), rb(X,V,T), rb(Y,W,T), X \neq Y. \end{aligned} \tag{3}$$

Deklarative Semantik

Extensionale Prädikate eines Programms (ext. Rel, ext. DB): EDB
Intentionale Prädikate eines Programms (int. Rel, int. DB): IDB

Bedeutung Bedeutung eines Datalog-Programms P: Menge derjenigen Grundatome zu den intentionalen Prädikaten von P, die logisch aus P gefolgert werden können. (Jedes Modell von P ist auch ein Modell von $f \in F$). Mit Zielklausel g(...), $g \in Praed$: Aus P logisch folgerbare Grundatome zu g, die von g(..) subsummiert (überdeckt) werden. $\dfrac{\text{g(a,X)}}{\substack{\text{g(a,b)} \\ \text{g(a,c)}}}$

In P werden Werte aus Wertebereichen verwendet, ebenso in Darstellung der extensionalen Prädikate als DB-Relation. Daher können wir $Kost_A$ und Dom idefntifizieren. Mithilfer der Herbrand Interpretation kann die Semantik festgelegt werden (ist möglich).

Herbrand-Interpretation /-Modelle Gewöhnliche Interpretation:

$$\begin{aligned} Konst_A &= \{a,b\}, Dom = \{\circ, \square\} \\ k(a) &= \circ \\ k(b) &= \square \\ ext(p(.,.)) &= \{(\circ,\square),(\square,\square)\} \end{aligned} \tag{4}$$

eine mögliche Herbrand-Interpretation (passt dazu)

$$\begin{aligned} Konst_A &= Dom = \{a, b\} \\ k(a) &= a \\ k(b) &= b \\ ext(p(.,.)) &= \{(a, b), (b, b) \end{aligned} \tag{5}$$

Entsprechende Herbrand-Interpretation. Betrachte alle Paare zu p(.,.), teste gemäß gegebener (gewöhnlicher) Interpretationin ext(p(.,.)).

$$\begin{aligned} Konst_A &= \{a, b\}, Dom = \{\circ, \square\} \\ k(a) &= \square \\ k(b) &= \square \\ ext(p(.,.)) &= \{(\circ, \square), (\square, \square) \end{aligned} \tag{6}$$

(a,a) wird zu $(\square, \square) \in ext(p(.,.))$

Herbrand-Interpretation

$$\begin{aligned} Konst_A &= Dom = \{a, b\} \\ k(a) &= a \\ k(b) &= b \\ ext(p(.,.)) &= \{(a, a), (a, b), (b, a), (b, b)\} \end{aligned} \tag{7}$$

Bei beiden Interpretationen sind die gleichen Formeln gültig bei Beschränkung auf quantorenfreie Formeln ohne Variablen (und ohne Funktionen).

Beispiel Erste Interpretation: $p(a, b) \wedge p(b, a) \Rightarrow (\square, \square) \in ext(p(.,.)) \wedge ...$ bzw. $(a, b) \in ext(p(.,.)) \wedge (b, a) \in ext(p(.,.))$

Menge von Konstanten und Prädikatensymbole ist endlich, daher ist die Anzahl der möglichen Herbrand-Interpreationen endlich.

Satz von Gödel / Skolem (Herbrand, 1930)

Eine Klauselmenge P hat ein Modell genau dann wenn P hat ein Herbrand-Modell. Daraus folgt, dass ein Verfahren analog zu Wahrheitstabellen in der Aussagenlogik möglich ist.

Beispiel $F = \{p(a) \Rightarrow q(b), p(a) \wedge q(b)\}$, $q(b)$?

p	q	$p(a) \Rightarrow q(b)$erfüllt?	$p(a) \wedge q(b)$erfüllt?	$p(a) \Rightarrow q(b)$und$p(a) \wedge q(b)$erfüllt?
$\{\}$	$\{\}$	✓	-	-
$\{\}$	$\{a\}$	✓	-	-
$\{\}$	$\{b\}$	✓	✓	✓
$\{\}$	$\{a,b\}$	✓	✓	✓
$\{a\}$	$\{\}$	-	✓	-
...	...	...	...	....
$\{b\}$	$\{b\}$	✓	✓	✓

Jedes Modell von F ist auch ein Modell von q(b), d.h. q(b) kann aus F logisch gefolgert werden. Gilt bei Klauselmengen, aber **Vorsicht bei allgemeinen Formeln**.

Beispiel: $\{p(a), (\exists X)(\neg p(X))\}$ Formelmenge, keine Klauselmenge

Modell (vgl. Übung):

$$\begin{aligned} Dom &= \{0,1\} \\ k(a) &= 0 \\ ext(p(.)) &= \{(0)\} \end{aligned} \tag{8}$$

Aber: Es gibt kein durch `ext` bestimmtes Herbrand-Modell:

1. $ext(p(.)) = \{(a)\}, Konst = Dom = \{a\}$
2. $ext(p(.) = \{\})$

Herbrand-Modell muss genügend viele Elemente enthalten, damit der Satz von Gödel / Skolem gelten kann. **Skolemisierung** bedeutet, dass man alle Existenzquantoren durch Funktionen ersetzt:

$$(\forall x_1, ..., x_n)(\exists y)(F) \rightsquigarrow (\forall x_1, ..., x_n)(F[f(x_1, ...x_n)/y]) \tag{9}$$

0.0.1 Bemerkung: Skolemisierung

Jede Formel der PL1 Logik kann man in einer erfüllbarkeitsäquivalente Formel in Skolemform umformen:

1. Pränexnormalform

2. Umformungen à la $(\forall x_1, ..., x_n)(\exists y)(F) \rightsquigarrow (\forall x_1, ..., x_n)(F[f(x_1, ...x_n)/y])$ mit jeweils einem neuen Funktionssymbol

Dies ist eine Art "Materialisierung" der durch den Existenzquantor gebundenen Variablen.

Beispiel (von oben) $\{p(a), \neg p(y)^1\}$ erfüllbar $\Longleftrightarrow$ $\{p(a), (\exists X)(\neg p(X))\}$ erfüllbar

Vorsicht: Semantische Äquivalenz von Formeln und ihren Skolem-Normalformen im Allgemeinen bicht gegeben.

Skolem-NF: $(\exists X)(p(X)) : p(a)$

[1] neue Variable