PREDIKAATTILOGIIKKA

1. Predikaattilogiikan kieli
2. Predikaattilogiikan semantiikka
3. Semanttiset taulut predikaattilogiikalle
4. Hilbertin järjestelmä
5. Normaalimuodot
6. Tietämyksen esittämisestä
7. Herbrandin teoreema
8. Unifikaatio
9. Resoluutiosääntö ja -todistukset

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

1 Predikaattilogiikan kieli

- Motivaatio
- Predikaattilogiikan aakkosto
- Kielen määritelmä
- Kvanttoreihin liittyviä määritelmiä
- Lauseiden muodostaminen

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

1.1 Motivaatio

- Lauselogiikka on useisiin tarkoituksiin liian yksinkertainen: olkoon $A =$ "a on viallinen", $B =$ "b on viallinen", $C =$ "c on viallinen".

  Tällöin "kaikki ovat viallisia" $= A \wedge B \wedge C$ ja
  "jokin on viallinen" $= A \vee B \vee C$.

- Erityisesti objektien välisten suhteiden kuvaaminen on hankalaa (tarvitaan paljon lauseita, jotka ovat muodoltaan samankaltaisia).

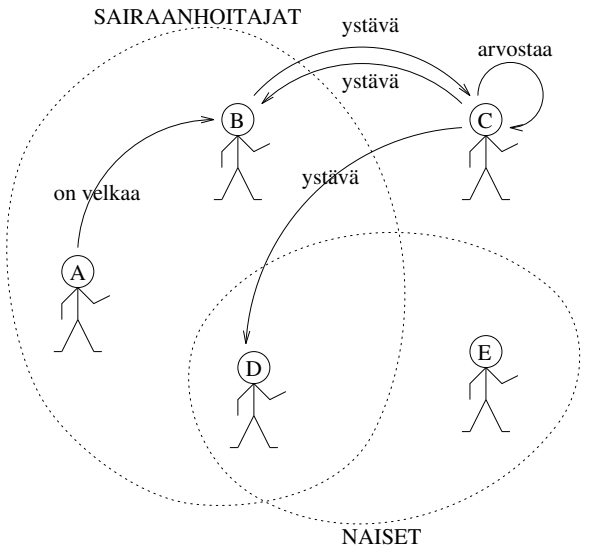
  Esimerkki.

  "Jos x on isompi kuin y ja y on isompi kuin z,
  niin x on isompi kuin z".

  $C_d =$ "c on isompi kuin d", $D_e =$ "d on isompi kuin e", ...

  $(C_d \wedge D_e \rightarrow C_e) \wedge (C_e \wedge E_d \rightarrow C_d) \wedge (D_e \wedge E_c \rightarrow D_c) \wedge \cdots$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio



© 2002 TKK / Tietojenkäsittelyteorian laboratorio

1.2 Predikaattilogiikan aakkosto

Predikaattilogiikan kielessä $\mathcal{L}$ käytetään seuraavia symboleja:

- Muuttujasymbolit $\mathcal{V} = \{x, y, z, \ldots\}$
- Vakiosymbolit $\mathcal{C} = \{a, b, c, \ldots\}$
- Funktiosymbolit $\mathcal{F} = \{f, g, h, \ldots\}$
- Predikaattisymbolit $\mathcal{P} = \{=, P, Q, R, \ldots\}$
- Lauselogiikan konnektiivit $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
- Kvanttorisymbolit $\exists, \forall$
- Sulut () ja pilkku ,

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä.

- Jokaisella funktiosymbolilla $f \in \mathcal{F}$ on *paikkaluku* $n > 0$ (mikä määrää f:n argumenttien lukumäärän).
- Vastaavasti predikaattisymboleilla $P \in \mathcal{P}$ on paikkaluvut $n \geq 0$.
- Määritellään $\mathcal{F}_n = \{f \in \mathcal{F} \mid f\text{:n paikkaluku on } n\}$ ja $\mathcal{P}_n = \{P \in \mathcal{P} \mid P\text{:n paikkaluku on } n\}$.
- Yhtäsuuruuspredikaatti $= \in \mathcal{P}$ on kaksipaikkainen, eli $= \in \mathcal{P}_2$.
- Täten $\mathcal{F} = \cup\{\mathcal{F}_n \mid n > 0\}$ ja $\mathcal{P} = \cup\{\mathcal{P}_n \mid n \geq 0\}$.

Huomioita.

- Vakiosymbolien joukko $\mathcal{C}$ voitaisiin vaihtoehtoisesti määritellä 0-paikkaisten funktiosymbolien joukkona $\mathcal{F}_0$.
- Joukon $\mathcal{P}_0$ symbolit vastaavat lauselogiikan atomisia lauseita.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

1.3 Kielen määritelmä

Predikaattilogiikan kielen $\mathcal{L}$ määritelmä on kolmitasoinen: ensin määritellään termit, sitten atomikaavat ja lopulta varsinaiset kaavat.

Määritelmä.

1. Jokainen muuttujasymboli $v \in \mathcal{V}$ on *termi*.
2. Jokainen vakiosymboli $c \in \mathcal{C}$ on *termi*.
3. Jos $f \in \mathcal{F}_n$ on n-paikkainen funktiosymboli ja $t_1, \ldots, t_n$ ovat termejä, niin myös $f(t_1, \ldots, t_n)$ on *termi*.
4. Muita termejä ei ole.

Esimerkki. x, c, $f(x)$, $f(f(f(f(f(x)))))$, $g(f(x), g(f(x), g(x, c)))$.

Määritelmä. Termi, jossa ei esiinny muuttujia, on *muuttujaton termi* (engl. *ground term*).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä.

1. Jos t_1 ja t_2 ovat termejä, niin $t_1 = t_2$ on *atomikaava*.
2. Jos $P \in \mathcal{P}_0$ on 0-paikkainen predikaattisymboli, niin P on *atomikaava*.
3. Jos $P \in \mathcal{P}_n$ on n-paikkainen predikaattisymboli (missä $n > 0$) ja $t_1, \ldots, t_n$ ovat termejä, niin $P(t_1, \ldots, t_n)$ on *atomikaava*.
4. Muita atomikaavoja ei ole.

Esimerkki. Atomikaavoja ovat mm.

$$P(x_1), \quad Q, \quad x_1 = x_2, \quad g(x_1, x_2) = f(f(c_1)),$$
$$R(c_1, x_1, y_1) \quad \text{ja} \quad S(x_1, c_1, f(x), h(f(x_1), c_1, x_1), x_2, x_2, x_3),$$

mutta esim. $f(R(x), c)$ ei ole atomikaava (eikä edes termi).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä.

1. Jokainen atomikaava ϕ on *kaava*.
2. Jos ϕ ja ψ ovat kaavoja ja x on muuttuja, niin myös
   $(\neg\phi)$, $(\phi \vee \psi)$, $(\phi \wedge \psi)$, $(\phi \rightarrow \psi)$, $(\phi \leftrightarrow \psi)$, $(\forall x\phi)$, $(\exists x\phi)$
   ovat *kaavoja*.
3. Muita kaavoja ei ole.

Esimerkki. Predikaattilogiikan kaavoja: $P(c)$, $(\forall x(P(x) \rightarrow Q(x)))$, $(\forall x(P(x) \vee (\exists y Q(x,y))))$, $(\exists x(\forall y(\forall z P(x,y,z))))$.

Symbolijoukkoihin $\mathcal{V}, \mathcal{C}, \mathcal{F}$ ja $\mathcal{P}$ perustuva predikaattilogiikan *kieli* $\mathcal{L}$ määritellään edellä annetuilla periaatteilla muodostettavissa olevien kaavojen joukkona.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Sopimukset sulkeiden käytöstä

- Konnektiivien presedenssiluokat ovat seuraavat:
  1. $\neg$, $\forall v$ ja $\exists v$ (missä $v \in \mathcal{V}$) ovat vahvimmat konnektiivit.
  2. $\vee$ ja $\wedge$ ovat näitä heikompia, mutta vahvempia kuin $\rightarrow$ ja $\leftrightarrow$.
  3. $\rightarrow$ ja $\leftrightarrow$ ovat heikoimmat konnektiivit.
- Lauselogiikan yhteydessä käyttöönotettuja periaatteita sulkeiden vähentämiseksi käytetään myös kaavoja kirjoitettaessa.

Esimerkki. Täten kaava

$$(\exists x(\forall y((\exists z(P(x,z) \wedge P(z,y))) \rightarrow ((Q(x) \vee Q(y)) \vee R(x,y)))))$$

voidaan kirjoittaa selkeämmin

$$\exists x \forall y(\exists z(P(x,z) \wedge P(z,y)) \rightarrow Q(x) \vee Q(y) \vee R(x,y)).$$

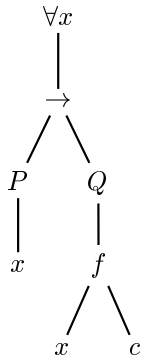
© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Kaavojen jäsennyspuut

Predikaattilogiikan kaavoilla on yksikäsitteinen jäsennyspuu.

Kaavan $\forall x(P(x) \rightarrow Q(f(x,c)))$ jäsennyspuu on annettu oikealla.

Kyseinen kaava on muodoltaan *universaalisti kvantifioitu kaava*.



Huomio. Jäsennyspuun juuressa oleva konnektiivi määrää edelleen, mitä muotoa lause on. Esimerkiksi $\exists x P(x) \rightarrow \forall x P(x)$ on muodoltaan implikaatio, kun taas $\exists x(P(x) \rightarrow \forall y Q(y))$ on muodoltaan eksistentiaalisesti kvantifioitu kaava.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

1.4 Kvanttoreihin liittyviä määritelmiä

Kaavan **alikaavat** määräytyvät seuraavasti:

- Atomisen kaavan ψ ainoa alikaava on ψ itse.
- Kaavan $\exists x\psi$ ($\forall x\psi$) alikaavat ovat $\exists x\psi$ ($\forall x\psi$) ja ψ:n alikaavat.
- Lauselogiikan konnektiivit ($\neg, \rightarrow, \leftrightarrow, \vee, \wedge$) käsitellään vastaavalla tavalla (vrt. alilauseiden määritelmä lauselogiikan tapauksessa).

Esimerkki. Kaavan $\phi = \exists x A(x,x) \wedge \exists x(S(x) \wedge N(x))$ alikaavoja ovat

ϕ, $\exists x A(x,x)$, $\exists x(S(x) \wedge N(x))$, $A(x,x)$, $S(x) \wedge N(x)$, $S(x)$ ja $N(x)$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Vapaat ja sidotut muuttujat kaavassa

Määritelmä. Olkoot $\exists x\psi$ ja $\forall x\phi$ predikaattilogiikan kaavoja. Alikaava ψ on kvanttorin $\exists x$ *vaikutusalue* kaavassa $\exists x\psi$. Vastaavasti alikaava ϕ on kvanttorin $\forall x$ vaikutusalue kaavassa $\forall x\phi$.

Esimerkki.

$$\forall x \overbrace{(P(x) \rightarrow \exists y \underbrace{Q(x,y)}_{\exists y})}^{\forall x}$$

$$\exists x \overbrace{(Q(x) \leftrightarrow \forall x \underbrace{P(x,z)}_{\forall x})}^{\exists x} \quad \vee \exists x \overbrace{R(x)}^{\exists x}$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä. Muuttujan x esiintymä kaavassa ψ on *sidottu*, jos se sijaitsee kvanttorin $\forall x$ (tai $\exists x$) vaikutusalueessa. Kvanttorisymbolia seuraava muuttujaesiintymä on sidottu.

Jos muuttujan x esiintymä ei ole sidottu, niin x:n esiintymä on *vapaa*.

Muuttuja x *esiintyy vapaana* ϕ:ssä, jos sillä on vapaa esiintymä ϕ:ssä.

Esimerkki.

$$\forall \overbrace{x}^{\text{sid.}}(P(\overbrace{x}^{\text{sid.}}, \overbrace{y}^{\text{vap.}}, \overbrace{z}^{\text{vap.}}) \vee \exists \overbrace{y}^{\text{sid.}}(Q(\overbrace{y}^{\text{sid.}}) \rightarrow R(\overbrace{x}^{\text{sid.}}, \overbrace{z}^{\text{vap.}})))$$

Määritelmä.
Kaava ϕ on *lause*, jos siinä ei ole vapaita muuttujaesiintymiä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Termin sijoittaminen kaavaan

Määritelmä. Olkoon $\phi(x)$ kaava, jossa muuttuja x mahdollisesti esiintyy vapaana ja t termi.

1. Kaavalla $\phi(t)$ tarkoitetaan kaavaa $\phi(x)$, jossa jokainen muuttujan x vapaa esiintymä on korvattu termillä t.
2. Termi t on sijoitettavissa kaavaan ϕ, mikäli mikään termin t sisältämä muuttujan x esiintymä ei joudu sijoituksessa kvanttorin ($\forall x$ tai $\exists x$) sitomaksi.

Huomio. Muuttujaton termi t on aina sijoitettavissa.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki.

1. Olkoon $\phi(y) = \exists x(P(x,y) \vee Q(y,x))$.
   - Sijoittamalla muuttujattomat termit c ja $f(f(d))$ kaavaan $\phi(y)$ saadaan $\phi(c) = \exists x(P(x,c) \vee Q(c,x))$ ja
   $$\phi(f(f(d))) = \exists x(P(x, f(f(d))) \vee Q(f(f(d)), x)).$$
2. Olkoon $\psi(x) = \exists x P(x) \wedge Q(x)$.
   - Sijoittamalla c saadaan $\psi(c) = \exists x P(x) \wedge Q(c)$.
3. Olkoon $\xi(y) = \exists x(P(x) \wedge Q(y))$.
   - Termi $f(x)$ ei ole sijoitettavissa kaavaan $\xi(y)$, koska $f(x)$:n sisältämä muuttujaesiintymä x joutuisi kvanttorin $\exists x$ sitomaksi.
   - Sen sijaan termi $f(z)$ on sijoitettavissa ja tulokseksi saadaan $\xi(f(z)) = \exists x(P(x) \wedge Q(f(z)))$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

1.5 Lauseiden muodostaminen

1. Tunnistetaan kuvattavaan järjestelmään liittyvät objektit:
   - Otetaan käyttöön vakiosymboli jokaiselle objektille, johon on tarve viitata erikseen, eli *nimetään* tarvittavat objektit.
   - Mikäli objektien välillä on funktionaalisia riippuvuuksia, otetaan käyttöön vastaavat funktiosymbolit.
2. Tutkitaan millaisia relaatioita objektien välillä on ja otetaan käyttöön näitä vastaavat predikaattisymbolit.
3. Kuvataan relaatioiden väliset riippuvuudet kirjoittamalla niille määritelmät predikaattilogiikan lausein.

Huomio. Funktiosymbolin voi tarvittaessa korvata predikaattisymbolilla, mutta tällöin tällöin määritelmään tulee liittää funktionaalisuusehto.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki.

Olkoon $t =$ "tuoli", $h =$ "hattu", $s =$ "sateenvarjo" vakioita ja $P(x, y) =$ "x on y:n päällä" kaksipaikkainen predikaatti. Tällöin:

$\neg P(s, t) =$ "sateenvarjo ei ole tuolin päällä".

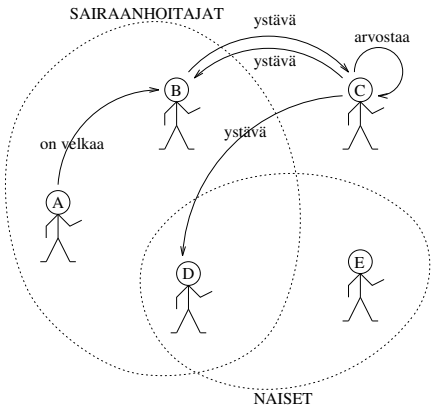
$\exists x P(x, h) =$
"on olemassa x, joka on hatun päällä",
eli "hatun päällä on jotakin".

$\exists x \forall y \neg P(y, x) =$
"on olemassa x siten, että mikään y ei ole x:n päällä",
eli "jonkin päällä ei ole mitään".

$\forall x(P(x, h) \rightarrow P(x, t)) =$
"kaikille x: jos x on hatun päällä, niin x on tuolin päällä",
eli "kaikki hatun päällä olevat ovat tuolin päällä".

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki.



$N(e) \rightarrow A(c, c)$ $\quad \exists x \exists y(Y(x, y) \wedge Y(y, x))$

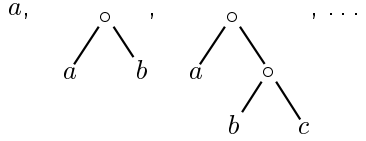
$\exists x \exists y(S(x) \wedge S(y) \wedge V(x, y))$ $\quad \exists x A(x, x) \wedge \exists x(S(x) \wedge N(x))$

$\neg \forall x(S(x) \rightarrow N(x))$ $\quad \forall x(Y(x, c) \rightarrow V(a, x))$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Funktiosymbolit tarjoavat tavan esittää induktiivisia tietorakenteita termien avulla.

1. Luonnolliset luvut: vakiosymboli 0 ja funktiosymboli $s \in \mathcal{F}_1$.
   - Termit $0, s(0), s(s(0)), \ldots$ vastaavat luonnollisia lukuja $0, 1, 2, \ldots$
2. Listat: vakiosymboli e (tyhjä lista) ja funktiosymboli $c \in \mathcal{F}_2$.
   - Termit $e, c(a, e), c(a, c(b, e)), \ldots$ vastaavat listoja $[\,], [a], [a, b], \ldots$
3. Binääripuut: funktiosymbolit $l \in \mathcal{F}_1$ (lehtisolmut) ja $t \in \mathcal{F}_2$ (sisäsolmut).
   - Termit $l(a), t(l(a), l(b)), t(l(a), t(l(b), l(c))), \ldots$ vastaavat puita



© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Esitetään binääripuut kuten edellä funktiosymbolien l ja t avulla. Kirjoitetaan määritelmä seuraavalle predikaatille:

$$P(x, y) = \text{“binäripuu } x \text{ on binääripuun } y \text{ peilikuva”}.$$

Koska binääripuut muodostavat induktiivisen tietorakenteen, on luontevaa, että predikaatille $P(x, y)$ joudutaan kirjoittamaan induktiivinen/rekursiivinen määritelmä seuraavalla tavalla.

- Perustapaus (pelkästä lehtisolmusta koostuvat binääripuut):
$$\forall x(P(l(x), l(x)))$$
- Induktioaskel (monimutkaisemmat binääripuut):
$$\forall x_1 \forall y_1 \forall x_2 \forall y_2 (P(x_1, y_1) \wedge P(x_2, y_2) \rightarrow P(t(x_1, x_2), t(y_2, y_1))).$$

$\Longrightarrow$ Määritelmä kattaa kaikki binääripuut.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

2 Predikaattilogiikan semantiikka

- Struktuurit
- Predikaattilogiikan totuusmääritelmä
- Semanttiset peruskäsitteet

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

2.1 Struktuurit

Predikaattilogiikassa totuusjakelut korvataan struktuureilla.

Määritelmä. *Struktuuri* (rakenne) $\mathcal{A}$ kielelle $\mathcal{L}$ koostuu

- universumista A, joka on jokin ei-tyhjä joukko, sekä
- vakio-, muuttuja-, funktio- ja predikaattisymbolien tulkinnoista:
  1. Vakiosymbolin $c \in \mathcal{C}$ tulkintana on alkio $c^{\mathcal{A}} \in A$.
  2. Muuttujasymbolin $v \in \mathcal{V}$ tulkintana on alkio $v^{\mathcal{A}} \in A$.
  3. Funktiosymbolin $f \in \mathcal{F}_n$ tulkintana on funktio $f^{\mathcal{A}}\colon A^n \rightarrow A$.
  4. Predikaattisymbolin $P \in \mathcal{P}_n$ tulkintana on relaatio $P^{\mathcal{A}} \subseteq A^n$.

Struktuuri voidaan edelleen ymmärtää yhden asiaintilan kuvauksena.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

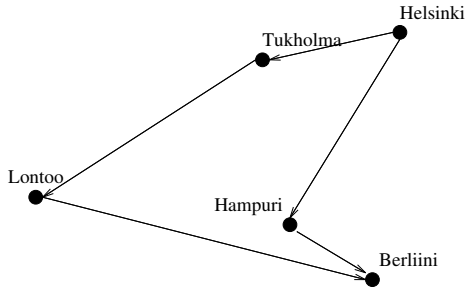
Huomioita.

- Joukon $A^n = \overbrace{A \times \cdots \times A}^{n \text{ kpl}}$ alkiot ovat monikkoja (tai jonoja) $\langle a_1, \ldots, a_n \rangle$, missä alkiot $a_1 \in A$, $\ldots$, $a_n \in A$.
- Erikoistapaukset: $A^1 = A$ ja $A^0 = \{\langle\,\rangle\}$, missä $\langle\,\rangle$ on tyhjä jono.
- Kvanttoreilla $\exists v$ ja $\forall v$ tullaan jatkossa viittaamaan universumin eri alkioihin. Muuttujan v arvon vaihtaminen struktuurissa $\mathcal{A}$ tapahtuu seuraavasti:

  Määritelmä. Struktuurilla $\mathcal{A}[v \mapsto a]$ tarkoitetaan struktuuria $\mathcal{A}'$, joka on muuten sama kuin $\mathcal{A}$, mutta muuttujasymbolin $v \in \mathcal{V}$ tulkintana $v^{\mathcal{A}'}$ onkin annettu alkio $a \in A$ (alkion $v^{\mathcal{A}}$ sijaan).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki.



$$
\begin{aligned}
A &= \{he, tu, ha, be, lo\} & \text{Helsinki}^{\mathcal{A}} &= he \\
\text{Tukholma}^{\mathcal{A}} &= tu & \text{Hampuri}^{\mathcal{A}} &= ha \\
\text{Berliini}^{\mathcal{A}} &= be & \text{Lontoo}^{\mathcal{A}} &= lo \\
\text{pääkaupunki}^{\mathcal{A}} &= \{he, tu, be, lo\} \subseteq A^1 = A \\
\text{lento}^{\mathcal{A}} &= \{\langle he, tu\rangle, \langle tu, lo\rangle, \langle lo, be\rangle, \langle he, ha\rangle, \langle ha, be\rangle\} \subseteq A^2
\end{aligned}
$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

2.2 Predikaattilogiikan totuusmääritelmä

Termien tulkinta struktuurissa

Määritelmä. Olkoon $\mathcal{A}$ struktuuri kielelle $\mathcal{L}$.

- Vakio $c \in \mathcal{C}$ *nimeää* universumin alkion $c^{\mathcal{A}}$.
- Muuttuja $v \in \mathcal{V}$ *nimeää* universumin alkion $v^{\mathcal{A}}$.
- Jos termit $t_1, \ldots, t_n$ nimeävät universumin alkiot $t_1^{\mathcal{A}}, \ldots, t_n^{\mathcal{A}}$ ja $f \in \mathcal{F}_n$, niin termi $f(t_1, \ldots, t_n)$ *nimeää* universumin alkion $f^{\mathcal{A}}(t_1^{\mathcal{A}}, \ldots, t_n^{\mathcal{A}})$.

Näin voimme viitata kielen $\mathcal{L}$ termeillä universumin A alkioihin (kunhan vakio-, muuttuja-, ja funktiosymbolien tulkinnat on annettu).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Tarkastellaan vakiosymbolia $c \in \mathcal{C}$ ja funktiosymboleja $f \in \mathcal{F}_1$ ja $g \in \mathcal{F}_2$. Olkoon struktuurin $\mathcal{A}$ universumi A luonnolliset luvut $0, 1, 2, \ldots$. Valitaan em. symbolien tulkinnat seuraavasti:

$$
\begin{aligned}
c^{\mathcal{A}} &= 0, \\
f^{\mathcal{A}} &= \text{seuraajafunktio, eli } f^{\mathcal{A}}(n) = n + 1, \text{ ja} \\
g^{\mathcal{A}} &= \text{summafunktio, eli } g^{\mathcal{A}}(n, m) = n + m.
\end{aligned}
$$

Tällöin c	nimeää alkion 0,
$f(c)$	nimeää alkion 1,
$f^n(c) = \underbrace{f(f(\ldots f}_{n \text{ kpl}}(c)\ldots))$	nimeää alkion n ja
$g(f(c), f(f(c)))$	nimeää alkion 3.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Kaavojen totuusarvojen laskeminen struktuurissa

Olkoon $\mathcal{A}$ struktuuri kielelle $\mathcal{L}$.

Määritelmä. Seuraavassa määritellään, milloin kaava $\phi \in \mathcal{L}$ on **tosi** struktuurissa $\mathcal{A}$ (merk. $\mathcal{A} \models \phi$) ja milloin **epätosi** (merk. $\mathcal{A} \not\models \phi$).

1. $\mathcal{A} \models t_1 = t_2 \iff t_1^{\mathcal{A}}$ ja $t_2^{\mathcal{A}}$ ovat sama universumin A alkio (t_1 ja t_2 ovat mitä tahansa termejä).
2. $\mathcal{A} \models P(t_1, \ldots, t_n) \iff$ $\langle t_1, \ldots, t_n\rangle^{\mathcal{A}}$ (eli jono $\langle t_1^{\mathcal{A}}, \ldots, t_n^{\mathcal{A}}\rangle$) kuuluu tulkintaan $P^{\mathcal{A}}$ ($n > 0$, $t_1, \ldots, t_n$ ovat mitä tahansa termejä ja $P \in \mathcal{P}_n$).
3. $\mathcal{A} \models P \iff$ tyhjä jono $\langle\rangle$ kuuluu tulkintaan $P^{\mathcal{A}}$ ($P \in \mathcal{P}_0$).
4. $\mathcal{A} \models \neg\alpha \iff \mathcal{A} \not\models \alpha$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

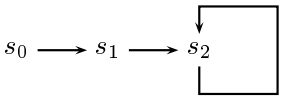
5. $\mathcal{A} \models \alpha \wedge \beta \iff \mathcal{A} \models \alpha$ ja $\mathcal{A} \models \beta$.
6. $\mathcal{A} \models \alpha \vee \beta \iff \mathcal{A} \models \alpha$ tai $\mathcal{A} \models \beta$.
7. $\mathcal{A} \models \alpha \to \beta \iff \mathcal{A} \not\models \alpha$ tai $\mathcal{A} \models \beta$.
8. $\mathcal{A} \models \alpha \leftrightarrow \beta \iff$ joko $\mathcal{A} \models \alpha$ ja $\mathcal{A} \models \beta$, tai $\mathcal{A} \not\models \alpha$ ja $\mathcal{A} \not\models \beta$.
9. $\mathcal{A} \models \exists x \alpha(x) \iff$
   $\mathcal{A}[x \mapsto a] \models \alpha(x)$ *jollekin* universumin alkiolle $a \in A$.
10. $\mathcal{A} \models \forall x \alpha(x) \iff$
    $\mathcal{A}[x \mapsto a] \models \alpha(x)$ *kaikille* universumin alkioille $a \in A$.

Väite. Kaikille kaavoille $\phi \in \mathcal{L}$ pätee joko $\mathcal{A} \models \phi$ tai $\mathcal{A} \not\models \phi$.

Väite. Jos kaava $\phi \in \mathcal{L}$ on lisäksi *lause*, sen totuusarvo ei riipu muuttujien $v \in \mathcal{V}$ tulkinnoista struktuurissa $\mathcal{A}$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Tarkastellaan graafin solmujen joukkoa (universumi) ja esitetään kaaret kaksipaikkaisen predikaatin K avulla. Nyt esim. graafia



vastaa struktuuri $\mathcal{A}$, jonka universumi on $A = \{s_0, s_1, s_2\}$ ja K:n tulkinta $K^{\mathcal{A}} = \{\langle s_0, s_1\rangle, \langle s_1, s_2\rangle, \langle s_2, s_2\rangle\}$.

1. $\langle s_0, s_1\rangle \in K^{\mathcal{A}} \implies \langle x, y\rangle^{\mathcal{A}[x\mapsto s_0, y\mapsto s_1]} \in K^{\mathcal{A}[x\mapsto s_0, y\mapsto s_1]}$
   $\implies \mathcal{A}[x \mapsto s_0, y \mapsto s_1] \models K(x, y)$
   $\implies \mathcal{A}[x \mapsto s_0] \models \exists y K(x, y)$.
2. Vastaavasti $\langle s_1, s_2\rangle \in K^{\mathcal{A}} \implies \mathcal{A}[x \mapsto s_1] \models \exists y K(x, y)$.
3. Vastaavasti $\langle s_2, s_2\rangle \in K^{\mathcal{A}} \implies \mathcal{A}[x \mapsto s_2] \models \exists y K(x, y)$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. (jatkoa)

4. Koska $\mathcal{A}[x \mapsto s_i] \models \exists y K(x, y)$ jokaiselle universumin alkiolle $s_i \in A$, saamme $\mathcal{A} \models \forall x \exists y K(x, y)$.
5. Lisäksi esim.

$$\begin{aligned} \langle s_2, s_2\rangle \in K^{\mathcal{A}} &\implies \langle x, x\rangle^{\mathcal{A}[x\mapsto s_2]} \in K^{\mathcal{A}[x\mapsto s_2]} \\ &\implies \mathcal{A}[x \mapsto s_2] \models K(x, x) \\ &\implies \mathcal{A}[x \mapsto s_2] \not\models \neg K(x, x) \\ &\implies \mathcal{A} \not\models \forall x \neg K(x, x) \\ &\implies \mathcal{A} \models \neg \forall x \neg K(x, x). \end{aligned}$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

2.3 Semanttiset peruskäsitteet

- Semanttisten peruskäsitteiden määritelmät ovat muodoltaan samat.
- Olennaisena erona lauselogiikkaan on, että lauseiden rakenne on monipuolisempi ja että struktuurit korvaavat totuusjakelut.

Määritelmä. Struktuuri $\mathcal{A}$ on lauseen $\alpha \in \mathcal{L}$ *malli*, joss lause α on tosi $\mathcal{A}$:ssa eli $\mathcal{A} \models \alpha$.

Määritelmä. Struktuuri $\mathcal{A}$ on lausejoukon $\Sigma \subseteq \mathcal{L}$ malli, joss kaikille lausejoukon Σ lauseille $\sigma \in \Sigma$ pätee $\mathcal{A} \models \sigma$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä. Lause $\alpha \in \mathcal{L}$ (tai lausejoukko $\Sigma \subseteq \mathcal{L}$) on *toteutuva*, joss ainakin yksi struktuuri $\mathcal{A}$ on sen malli.

Esimerkki. $\exists x \forall y P(x,y)$ on toteutuva.
Olkoon $\mathcal{A}$:n universumi $A = \{1,2\}$ ja $P^{\mathcal{A}} = \{\langle 1,1\rangle, \langle 1,2\rangle\}$.

1. $\langle 1,1\rangle \in P^{\mathcal{A}} \implies \langle x,y\rangle^{\mathcal{A}[x\mapsto 1, y\mapsto 1]} \in P^{\mathcal{A}[x\mapsto 1, y\mapsto 1]}$
   $\implies \mathcal{A}[x\mapsto 1, y\mapsto 1] \models P(x,y).$
2. $\langle 1,2\rangle \in P^{\mathcal{A}} \implies \langle x,y\rangle^{\mathcal{A}[x\mapsto 1, y\mapsto 2]} \in P^{\mathcal{A}[x\mapsto 1, y\mapsto 2]}$
   $\implies \mathcal{A}[x\mapsto 1, y\mapsto 2] \models P(x,y).$
3. Siis $\mathcal{A}[x\mapsto 1] \models \forall y P(x,y)$ ja $\mathcal{A} \models \exists x \forall y P(x,y)$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä. Kielen $\mathcal{L}$ lause α on *pätevä* (merkitään $\models \alpha$), joss $\mathcal{A} \models \alpha$ kaikissa $\mathcal{L}$:n struktuureissa $\mathcal{A}$.

Esimerkki. $\models (\forall x P(x)) \leftrightarrow (\neg \exists x \neg P(x))$

Kaikille kielen $\mathcal{L}$ struktuureille $\mathcal{A}$ pätee:

$\mathcal{A} \models \forall x P(x)$
$\iff$ $\mathcal{A}[x\mapsto a] \models P(x)$ kaikille $a \in A$
$\iff$ $x^{\mathcal{A}[x\mapsto a]} \in P^{\mathcal{A}[x\mapsto a]}$ kaikille $a \in A$
$\iff$ ei ole niin, että $x^{\mathcal{A}[x\mapsto a]} \notin P^{\mathcal{A}[x\mapsto a]}$ jollekin $a \in A$
$\iff$ ei ole niin, että $\mathcal{A}[x\mapsto a] \not\models P(x)$ jollekin $a \in A$
$\iff$ ei ole niin, että $\mathcal{A}[x\mapsto a] \models \neg P(x)$ jollekin $a \in A$
$\iff$ $\mathcal{A} \not\models \exists x \neg P(x)$
$\iff$ $\mathcal{A} \models \neg \exists x \neg P(x)$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. $\models \forall x P(x) \vee \neg \forall x P(x)$
Olkoon $\mathcal{A}$ mielivaltainen $\mathcal{L}$:n struktuuri.
Nyt $\mathcal{A} \models \forall x P(x) \iff \mathcal{A} \not\models \neg \forall x P(x)$, joten $\mathcal{A} \models \forall x P(x) \vee \neg \forall x P(x)$.

Määritelmä. Kielen $\mathcal{L}$ lause α on $\mathcal{L}$:n lauseiden Σ *looginen seuraus* (merkitään $\Sigma \models \alpha$), joss α on tosi jokaisessa Σ:n mallissa.

Esimerkki. $\{\forall x P(x)\} \models \exists x P(x)$

Olkoon $\mathcal{A}$ struktuuri siten, että $\mathcal{A} \models \forall x P(x)$. Tällöin
$\implies$ kaikille $a \in A$ pätee $\mathcal{A}[x\mapsto a] \models P(x)$
$\implies$ jollekin $a \in A$ pätee $\mathcal{A}[x\mapsto a] \models P(x)$
(universumi A on ei-tyhjä struktuurin määritelmän perusteella)
$\implies$ $\mathcal{A} \models \exists x P(x)$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Peruskäsitteiden väliset yhteydet

Seuraavat pätevät myös predikaattilogiikan tapauksessa:

- $\models \alpha \iff \neg\alpha$ on toteutumaton.
- $\Sigma \models \alpha \iff \Sigma \cup \{\neg\alpha\}$ on toteutumaton.
- $\models \alpha \iff \emptyset \models \alpha$.
- $\{\phi_1, \ldots, \phi_n\} \models \phi \iff \models \phi_1 \wedge \cdots \wedge \phi_n \to \phi$.
- Kompaktius: jos $\Sigma \models \phi$, niin on olemassa äärellinen osajoukko $\Sigma' \subseteq \Sigma$ siten, että $\Sigma' \models \phi$.
- $\Sigma \subseteq \mathrm{Cn}(\Sigma)$.
- Monotonisuus: $\Sigma_1 \subseteq \Sigma_2 \implies \mathrm{Cn}(\Sigma_1) \subseteq \mathrm{Cn}(\Sigma_2)$.
- $\mathrm{Cn}(\mathrm{Cn}(\Sigma)) = \mathrm{Cn}(\Sigma)$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki.

$$\Sigma = \left\{ \begin{array}{l} \text{tentti(tiistai)}, \\ \text{tentti(keskiviikko)}, \\ \text{luento(keskiviikko)}, \\ \forall x(\neg\text{tentti}(x) \wedge \neg\text{luento}(x) \rightarrow \text{vapaa}(x)) \end{array} \right\}$$

Onko $\Sigma \models \text{vapaa(perjantai)}$? **Ei**, koska löytyy *vastamalli* $\mathcal{A}$, jonka perusteella $\Sigma \not\models \text{vapaa(perjantai)}$ eli $\mathcal{A} \models \Sigma$ ja $\mathcal{A} \not\models \text{vapaa(perjantai)}$!

Olkoon universumi $A = \{t, k, p\}$ ja symbolien tulkinnat seuraavat:

$$\begin{array}{rclrcl} \text{tiistai}^{\mathcal{A}} & = & t & \text{keskiviikko}^{\mathcal{A}} & = & k \\ \text{perjantai}^{\mathcal{A}} & = & p & \text{tentti}^{\mathcal{A}} & = & \{t, k\} \\ \text{luento}^{\mathcal{A}} & = & \{k, p\} & \text{vapaa}^{\mathcal{A}} & = & \{\} \end{array}$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

3 Semanttiset taulut predikaattilogiikalle

- Taulusäännöt kvanttoreiden käsittelyyn
- Semanttiset taulut predikaattilogiikalle
- Ohjeita taulutodistusten laadintaan
- Systemaattinen taulu
- Vastamallin konstruointi

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

3.1 Taulusäännöt kvanttorien käsittelyyn

- Muotoa $T\,\exists x\varphi(x)$ (tai $E\,\forall x\varphi(x)$) oleva solmu tulee hajoittaa *kertaalleen* käyttäen jotain (hajoitushetkellä) *uutta vakiota* c.

$T\,\exists x\varphi(x)$ $\mid$ $T\varphi(c)$ c uusi vakio	$E\,\forall x\varphi(x)$ $\mid$ $E\varphi(c)$ c uusi vakio

- Olkoon P polku (juurisolmusta lehtisolmuun), jolla solmu $T\,\exists x\varphi(x)$ ($E\,\forall x\varphi(x)$) esiintyy ja jota on tarkoitus jatkaa ao. taulusäännöllä. Vakio c on *uusi*, mikäli se ei esiinny polulla P.

Huomio. Uuden vakion käyttöönotto johtuu siitä, ettemme tiedä, millä universumin alkiolla on ko. ominaisuus φ (tai ei ole ominaisuutta φ).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Muotoa $T\,\forall x\varphi(x)$ (tai $E\,\exists x\varphi(x)$) oleva solmu tulee (tarvittaessa) hajoittaa kaikille *muuttujattomille termeille* t (vakioita tai vakioista ja funktiosymboleista rakentuvia termejä).

$T\,\forall x\varphi(x)$ $\mid$ $T\varphi(t)$ t muuttujaton termi	$E\,\exists x\varphi(x)$ $\mid$ $E\varphi(t)$ t muuttujaton termi

Huomio.

- Muuttujattomia termejä voi olla ääretön määrä, joten muotoa $T\,\forall x\varphi(x)$ (tai $E\,\exists x\varphi(x)$) olevaa solmua ei välttämättä saada hajoitetuksi soveltamalla ao. taulusääntöä äärellisen monta kertaa.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki.

1. $T\ \exists x \forall y \exists z P(x,y,z)$
2. $T\ \forall y \exists z P(a,y,z)$ 1. x/a (a uusi vakio)
3. $T\ \exists z P(a,a,z)$ 2. y/a
4. $T P(a,a,b)$ 3. z/b (b uusi vakio)
5. $T\ \exists z P(a,b,z)$ 2. y/b
6. $T P(a,b,c)$ 5. z/c (c uusi vakio)

jne.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

3.2 Semanttiset taulut predikaattilogiikalle

- Semanttisten taulujen määritelmä säilyy ennallaan.
- Ehtoja, millä polun solmu on *hajoitettu*, joudutaan täydentämään:

  Olkoon τ semanttinen taulu ja P polku juurisolmusta lehtisolmuun τ:ssa. P:n solmu $T\forall x\varphi(x)$ ($E\exists x\varphi(x)$) hajoitettu polulla P, jos
  - polulla P esiintyy $T\varphi(t)$ ($E\varphi(t)$) kaikille muuttujattomille termeille t, jotka voidaan muodostaa polulla P esiintyvistä vakio- ja funktiosymboleista (vakiosymboleja on oltava ainakin yksi).

  Huomio. Mikäli polulla P ei esiinny vakiosymboleita, $T\forall x\varphi(x)$ ($E\exists x\varphi(x)$) tulee hajoittaa käyttäen jotain uutta vakiosymbolia c.
- Muilta osin semanttisen taulun (ja sen polkujen) ristiriitaisuuden ja valmiuden määritelmät säilyvät ennallaan.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Taulumenetelmää voidaan käyttää erilaisten loogisten ongelmien ratkomiseen aivan kuten lauselogiikankin tapauksessa.

Määritelmä. Taulu τ on lauseen ϕ *todistus*, jos taulun τ juurisolmuna on $E\phi$ ja τ on ristiriitainen. Jos lauseella ϕ on todistus, on ϕ *teoreema/todistuva* (merkitään $\vdash \phi$).

Määritelmä. Lause ϕ *on johdettavissa* äärellisestä lausejoukosta $\Sigma = \{\phi_1, \ldots, \phi_n\}$ (merkitään $\Sigma \vdash \phi$), joss juurisolmusta

$$E(\phi_1 \wedge \cdots \wedge \phi_n \rightarrow \phi)$$

muodostettu valmis semanttinen taulu on ristiriitainen.

Väite. $\Sigma \vdash \phi \iff \Sigma \models \phi$ (virheettömyys ja täydellisyys).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Onko $\{P(a), \forall x(P(x) \rightarrow Q(x))\} \vdash Q(a)$?

1. $T\ \forall x(P(x) \rightarrow Q(x))$
2. $T\ P(a)$
3. $E\ Q(a)$
4. $T(P(a) \rightarrow Q(a))$ 1. x/a
5. $E\ P(a)$ 4. × | $T\ Q(a)$ 4. ×

Taulu on ristiiriitainen. Lause $Q(a)$ on siis johdettavissa lausejoukosta $\{P(a), \forall x(P(x) \rightarrow Q(x))\}$ (sekä lausejoukon looginen seuraus).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Onko $\{\exists x(P(x) \wedge Q(x))\} \vdash P(a)$?

1. $T\ \exists x(P(x) \wedge Q(x))$
2. $EP(a)$
3. $T(P(b) \wedge Q(b))$ 1. x/b
4. $TP(b)$ 3.
5. $TQ(b)$ 3.

Taulu saatiin valmiiksi muttei ristiriitaiseksi. Lause $P(a)$ ei ole johdettavissa lausejoukosta $\{\exists x(P(x) \wedge Q(x))\}$ (eikä myöskään lausejoukon looginen seuraus).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

3.3 Ohjeita taulutodistusten laadintaan

- Lauseen rakenne määrää edelleen, mitä taulusääntöä tulee käyttää (jäsennyspuun juuressa oleva konnektiivi).
- Solmujen hajoittamisjärjestyksellä voi usein vaikuttaa taulun kokoon (haarautumista kannattaa välttää).
- Jälkimmäisissä kvanttorisäännöissä esiintyvän t:n tilalle valitaan *hajoittamishetkellä* (eikä myöhemmin) jokin vakio tai funktio- ja vakiosymboleista rakentuva termi.
- Valitsemalla muuttujattomat termit t sopivasti voidaan usein nopeuttaa taulun valmistumista.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

1. Valitaan muuttujattomaksi termiksi t vakio, joka ei esiinny lauseissa.

Esimerkki. Osoitetaan $\{\forall x P(x)\} \vdash \exists x P(x)$.

1. $T\ \forall x P(x)$
2. $E\ \exists x P(x)$
3. $TP(c)$ 1. x/c
4. $EP(c)$ 2. x/c
   $\times$

Huomio. Tämä on perusteltua, koska universumissa A on aina vähintään yksi alkio $a \in A$, joka voidaan nimetä (eli $c^{\mathcal{A}} = a$).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

2. Solmu $T\ \forall x\varphi(x)$ (tai $E\ \exists x\varphi(x)$) joudutaan hajoittamaan useasti.

Esimerkki. $\{\forall x P(x)\} \vdash P(a) \wedge P(b)$

1. $T\ \forall x P(x)$
2. $E(P(a) \wedge P(b))$
3. $TP(a)$ 1. x/a
4. $TP(b)$ 1. x/b

5. $EP(a)$ 2. $\times$ 5. $EP(b)$ 2. $\times$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

3. Muuttujien korvaaminen sopivilla muuttujattomilla termeillä.

Esimerkki.
$\{\forall x\forall y\forall z(P(x,y) \wedge P(y,z) \rightarrow P(x,z)), P(a,b), P(b,c)\} \vdash P(a,c)$

- Semanttiseen tauluun tulee solmu
  $T\,\forall x\forall y\forall z(P(x,y) \wedge P(y,z) \rightarrow P(x,z))$, josta voidaan johtaa kvanttorisäännöillä 27 erilaista todeksi merkittyä implikaatiota.
- Ristiriidan johtamisen kannalta olennaisia ovat implikaatioista ne, joissa esiintyy atomisia lauseita $P(a,b)$, $P(b,c)$ ja $P(a,c)$.
- Esimerkin tapauksessa tämä johtaa ensimmäiseksi x:n, y:n ja z:n korvaamiseen vakoilla a, b, ja c (näin saatava implikaatio riittää).
- Muita implikaatioita ei tarvita, ja niiden johtaminen ja mahdollinen hajoittaminen johtaa semanttisen taulun tarpeettomaan kasvuun.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Taulutodistus muodostuu kokonaisuudessaan seuraavaksi:

1. $T\,\forall x\forall y\forall z(P(x,y) \wedge P(y,z) \rightarrow P(x,z))$
2. $T\,P(a,b)$
3. $T\,P(b,c)$
4. $E\,P(a,c)$
5. $T\,\forall y\forall z(P(a,y) \wedge P(y,z) \rightarrow P(x,z))^{1.\ x/a}$
6. $T\,\forall z(P(a,b) \wedge P(b,z) \rightarrow P(a,z))^{5.\ y/b}$
7. $T(P(a,b) \wedge P(b,c) \rightarrow P(a,c))^{6.\ z/c}$
8. $E(P(a,b) \wedge P(b,c))^{7.}$ — 8. $T\,P(a,c)^{7.}$ ×
9. $E\,P(a,b)^{8.}$ × — 9. $E\,P(b,c)^{8.}$ ×

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

4. Solmujen keskinäinen hajoitusjärjestys voi olla ratkaisevassa roolissa.

Esimerkki. $\{\forall xP(x), \forall x(P(x) \rightarrow Q(x)), Q(a)\} \vdash \forall xQ(x)$

1. $T\,\forall xP(x)$
2. $T\,\forall x(P(x) \rightarrow Q(x))$
3. $T\,Q(a)$
4. $E\,\forall xQ(x)$
5. $E\,Q(c)^{4.\ x/c}$
6. $T\,P(c)^{1.\ x/c}$
7. $T(P(c) \rightarrow Q(c))^{2.\ x/c}$
8. $E\,P(c)^{7.}$ × — 8. $T\,Q(c)^{7.}$ ×

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Kvanttorisekvenssien käsittely

Jatkossa sallimme seuraavien johdettujen taulusääntöjen käytön kvanttorisekvenssien käsittelyssä:

$T\,\forall x_1 \cdots \forall x_n\varphi(x_1,\ldots,x_n)$ $\vert$ $T\varphi(t_1,\ldots,t_n)$	$E\,\exists x_1 \cdots \exists x_n\varphi(x_1,\ldots,x_n)$ $\vert$ $E\varphi(t_1,\ldots,t_n)$
$T\,\exists x_1 \cdots \exists x_n\varphi(x_1,\ldots,x_n)$ $\vert$ $T\varphi(c_1,\ldots,c_n)$	$E\,\forall x_1 \cdots \forall x_n\varphi(x_1,\ldots,x_n)$ $\vert$ $E\varphi(c_1,\ldots,c_n)$

Yllä $c_1,\ldots,c_n$ ovat ao. taulusääntöjen edellyttämiä *uusia vakioita* ja vastaavasti $t_1,\ldots,t_n$ ovat valittuja *muuttujattomia termejä*.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

3.4 Systemaattinen taulu

- Lauselogiikan keskeiset päättelyongelmat ovat *ratkeavia*.

Esimerkki. Voidaan konstruoida *deterministinen* Turing-kone T, jonka suoritus pysähtyy syötteeksi annetulla lauselogiikan lauseella φ

1. hyväksyvään tilaan k (kyllä), jos syöte φ on pätevä, ja
2. hylkäävään tilaan e (ei), jos syöte φ ei ole pätevä.

Huomioita.

- Tällainen algoritmi voi perustua esim. totuustaulukkoihin, semanttisiin tauluihin tai resoluutioon.
- Myös looginen ekvivalenssi, looginen seuraavuus ja toteutuvuus ovat lauselogiikan tapauksessa ratkeavia ongelmia.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Predikaattilogiikka ei ole ratkeava, vaan *puoliratkeava*.

Esimerkki. Lauseen φ pätevyyden tarkastamista varten voidaan konstruoida seuraavanlainen deterministinen Turing-kone T:

1. Jos syöte φ on pätevä, T pysähtyy hyväksyvään tilaan k (kyllä).
2. Jos syöte φ ei ole pätevä, T pysähtyy *joskus* hylkäävään tilaan e (ei) ja *joskus T ei pysähdy lainkaan*.

Huomio. Tällainen algoritmi voi perustua semanttisiin tauluihin:

- Rakentamalla semanttinen taulu tietyllä tavalla *systemaattisesti*, voidaan taata, että taulu saadaan aina ristiriitaiseksi, kun sen juuressa on $E\varphi$ ja φ on pätevä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Systemaattisen taulun periaatteita

- Tuotetaan indeksoimalla riittävä määrä uusia vakioita $c_1, c_2, c_3, \ldots$, kun hajoitetaan muotoa $T\,\exists x\psi(x)$ tai $E\,\forall x\psi(x)$ olevia solmuja.
- Tuotetaan tarpeen mukaan muuttujattomia termejä $t_1, t_2, t_3, \ldots$, jotka rakentuvat $E\varphi$:ssä esiintyvistä vakio- ja funktiosymboleista sekä mahdollisesti käyttöönotetuista uusista vakioista $c_1, c_2, c_3, \ldots$.
- Sekvenssin $t_1, t_2, t_3, \ldots$ on oltava *reilu*: jokainen em. symboleista rakentuva muuttujaton termi t esiintyy siinä jonain terminä t_i.
- *Hajoitusten reiluus*: taataan, että taulun keskeneräisillä poluilla esiintyvät hajoittamattomat solmut tulevat hajoitusvuoroon (seuraavan kerran) äärellisen monen muun hajoituksen jälkeen.
- Muotoa $T\,\forall x\psi(x)$ tai $E\,\exists x\psi(x)$ olevia solmuja hajoitetaan järjestyksessä käyttäen muuttujattomia termejä $t_1, t_2, t_3, \ldots$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Huomio. Muotoa $T\,\forall x\phi(x)$ tai $E\,\exists x\phi(x)$ olevia solmuja sisältäviä polkuja ei välttämättä saada hajoitetuksi äärellisellä askelmäärällä, jolloin valmis taulu muodostuu äärettömäksi (puoliratkeavuuden ilmentymä).

Esimerkki. Kirjoitetaan luonnollisten lukujen $>$-relaatiolle määritelmä. Olkoon $G(x, y) =$ "$x > y$"ja $s(x)$ luvun x seuraaja (eli $x + 1$).

- Määritelmä: $\forall x\, G(s(x), x)$ ja $\forall x \forall y (G(x, y) \rightarrow G(s(x), y))$.
- Kysely: onko $G(s(s(s(0))), s(0))$ määritelmän looginen seuraus?

Hajoitetaan systemaattisesti semanttisen taulun solmuja

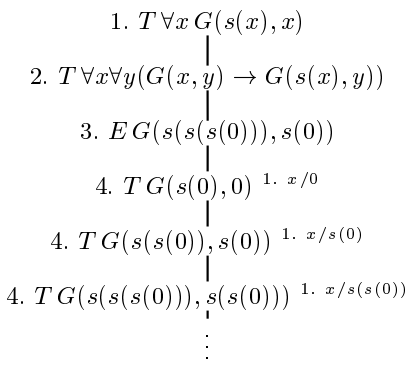
$$T\,\forall x\, G(s(x), x) \text{ ja } T\,\forall x \forall y (G(x, y) \rightarrow G(s(x), y))$$

käyttäen muuttujattomia termejä: $0, s(0), s(s(0)), s(s(s(0))), \ldots$

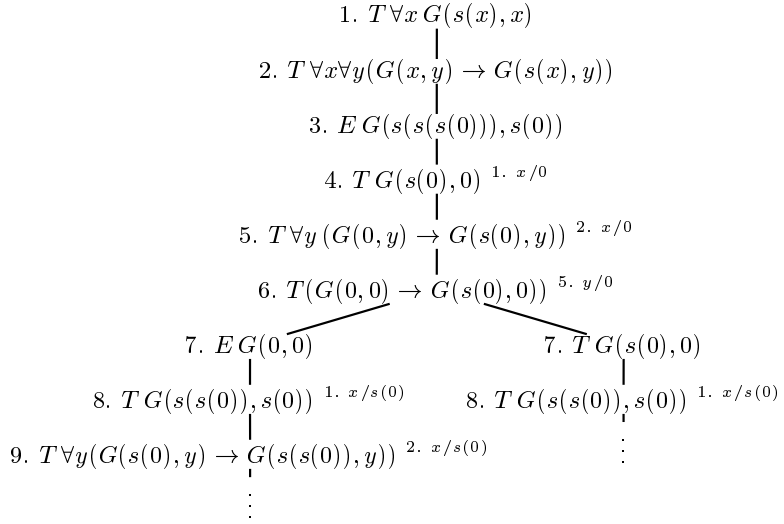
© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Mikäli solmujen hajoitusjärjestys rikkoo edellä esitettyä reiluusperiaatetta, todistuksen löytyminen ei ole taattu.

Esimerkki. Järjestys, missä hajoitetaan ainoastaan solmua $T\,\forall x\,G(s(x),x)$ em. muuttujattomien termien suhteen, ei ole reilu:



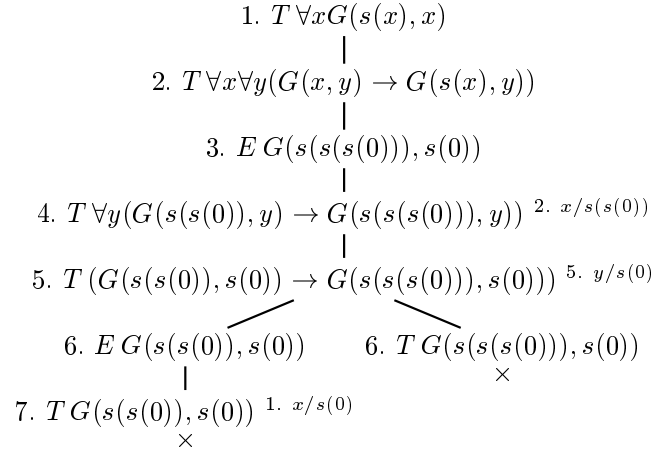
© 2002 TKK / Tietojenkäsittelyteorian laboratorio



Systemaattinen taulu voi tehdä turhaa työtä $\Longrightarrow$ heuristiikkaa tarvitaan!

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Valitsemalla muuttujattomat termit aikaisemmin esitetyillä periaatteilla semanttinen taulu jää huomattavasti pienemmäksi:



© 2002 TKK / Tietojenkäsittelyteorian laboratorio

3.5 Vastamallien konstruointi

- Vastamallin (struktuuri) konstruoinnissa voidaan hyödyntää semanttisen taulun ristiriidattomasta polusta saatavia atomisia lauseita koskevia totuusarvovaatimuksia $T\,P(t_1,\ldots,t_n)$, $E\,Q(s_1,\ldots,s_m)$, ... (t_i:t ja s_j:t ovat muuttujattomia termejä).
- Valitaan riittävän iso universumi A, jotta pystytään antamaan tulkinnat totuusarvovaatimuksissa esiintyville vakio- ja funktiosymboleille.
- Tämän jälkeen valitaan predikaattien tulkinnat totuusarvovaatimusten mukaisesti:
  1. Jos $T\,P(t_1,\ldots,t_n)$ on polulla, $\langle t_1^{\mathcal{A}},\ldots,t_n^{\mathcal{A}}\rangle \in P^{\mathcal{A}}$.
  2. Jos $E\,Q(s_1,\ldots,s_m)$ on polulla, $\langle s_1^{\mathcal{A}},\ldots,s_m^{\mathcal{A}}\rangle \notin Q^{\mathcal{A}}$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Menettely on käyttökelpoinen erityisesti, jos *valmiin semanttisen taulun* ristiriidaton polku muodostuu *äärelliseksi*:

Esimerkki. Vastaesimerkki lauseen $\forall x(P(x) \to Q(x))$ pätevyydelle.

1. $E\,\forall x(P(x) \to Q(x))$
2. $E(P(c) \to Q(c))$ $^{1.\ x/c}$
3. $T(P(c))$ $^{2.}$
4. $E(Q(c))$ $^{2.}$

1. Totuusarvovaatimukset ristiriidattomasta polusta: $T\,P(c)$ ja $E\,Q(c)$.
2. Riittää, että universumiin $A = \{1\}$ otetaan yksi alkio s.e. $c^{\mathcal{A}} = 1$.
3. Totuusarvovaatimusten nojalla: $1 \in P^{\mathcal{A}}$ ja $1 \notin Q^{\mathcal{A}}$.
4. Nämä vaatimukset toteutuvat valinnoilla $P^{\mathcal{A}} = \{1\}$ ja $Q^{\mathcal{A}} = \emptyset$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. $\{\forall x(P(x) \wedge Q(x) \to R(x))\} \not\models \forall x(Q(x) \to R(x))$.

1. $T\,\forall x(P(x) \wedge Q(x) \to R(x))$
2. $E\,\forall x(Q(x) \to R(x))$
3. $E(Q(c) \to R(c))$ $^{2.\ x/c}$
4. $T\,Q(c)$ $^{3.}$
5. $E\,R(c)$ $^{3.}$
6. $T\,(P(c) \wedge Q(c) \to R(c))$ $^{1.\ x/c}$
7. $E(P(c) \wedge Q(c))$ | 7. $T\,R(c)$ ×
8. $E\,P(c)$ | 8. $E\,Q(c)$ ×

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Tarkastellaan taulun ainoaa ristiriidatonta polkua P.
  - Polulla esiintyy yksi vakiosymboli c muttei funktiosymboleja.
  - Voidaan muodostaa ainoastaan yksi muuttujaton termi eli c itse.
  - Täten solmu $T\,\forall x(P(x) \wedge Q(x) \to R(x))$ on hajoitettu polulla P, koska polulla P on solmu $T(P(t) \wedge Q(t) \to R(t))$ jokaista muuttujatonta termiä $t \in \{c\}$ kohti.
  - Polku P on siis valmis.
- Näin ollen taulu on kokonaisuutena myös valmis.
- Polulta P saadaan vaatimukset $E\,P(c)$, $T\,Q(c)$ ja $E\,R(c)$.
- Valitaan universumiksi $A = \{1\}$ ja symbolien tulkinnoiksi $c^{\mathcal{A}} = 1$, $P^{\mathcal{A}} = R^{\mathcal{A}} = \{\}$ ja $Q^{\mathcal{A}} = \{1\}$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Joskus äärettömästäkin ristiriidattomasta polusta voi onnistua muodostamaan *äärellisen vastamallin*.

1. $E\,\exists x(P(a) \vee P(f(x)))$
2. $E(P(a) \vee P(f(a)))$ $^{1.\ x/a}$
3. $E\,P(a)$ $^{2.}$
4. $E\,P(f(a))$ $^{2.}$
5. $E(P(a) \vee P(f(f(a))))$ $^{1.\ x/f(a)}$
6. $E\,P(a)$ $^{5.}$
7. $E\,P(f(f(a)))$ $^{5.}$
⋮

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Tarkastellaan taulun ainoata ristiriidatonta polkua P.
- Polku ei ole valmis, koska solmu $E\,\exists x(P(a) \vee P(f(x)))$ ei ole hajoitettu: polulla P ei ole solmua $E(P(a) \vee P(f(t)))$ esim. muuttujattomalle termille $t = f(f(a))$.
- Polulta P saadaan vaatimukset

$$E\,P(a),\ E\,P(f(a)),\ E\,P(f(f(a))),\ \ldots$$

- Näiden säännönmukaisuudesta johtuen valitaan universumiksi $A = \{1\}$ ja symbolien tulkinnoiksi $a^{\mathcal{A}} = 1$, $f^{\mathcal{A}} : 1 \mapsto 1$ ja $P^{\mathcal{A}} = \emptyset$.
- Koska taulu ei ollut valmis, lisäksi on syytä todeta totuusmääritelmästä lähtien, että kysymyksessä on todella vastaesimerkki eli $\mathcal{A} \not\models \exists x(P(a) \vee P(f(x)))$.
- Täten $\mathcal{A}$ on vastamalli lauseen $\exists x(P(a) \vee P(f(x)))$ pätevyydelle.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

4 Hilbertin järjestelmä

Aksiomat ovat seuraavan muotoisia *kaavoja*:

1. $\alpha \to (\beta \to \alpha)$
2. $(\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma))$
3. $(\neg\alpha) \to (\alpha \to \beta)$
4. $\forall x\alpha(x) \to \alpha(t)$ kaikille termeille t (sijoitettavissa α:aan)
5. $\forall x(\alpha \to \beta(x)) \to (\alpha \to \forall x\beta(x))$, jos x ei esiinny vapaana α:ssa.

Päättelysäännöt: $\dfrac{\alpha \to \beta \quad \alpha}{\beta}$ (MP) $\dfrac{\alpha}{\forall x\alpha}$ (Yleistys)

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä. Olkoon Σ joukko lauseita.

Todistus Σ:sta on jono *kaavoja* $\alpha_1, \ldots, \alpha_n$ siten, että kaikille $i \in \{1, \ldots, n\}$

- $\alpha_i \in \Sigma$,
- α_i on aksioma, tai
- α_i on saatu päättelysäännöllä aikaisemmista kaavoista.

Lause α on *johdettavissa* lausejoukosta Σ (merkitään $\Sigma \vdash_H \alpha$), jos on olemassa todistus $\alpha_1, \ldots, \alpha_n$ lausejoukosta Σ siten, että $\alpha = \alpha_n$.

Lause α on *teoreema/todistuva* (merkitään $\vdash_H \alpha$), jos se on johdettavissa tyhjästä lausejoukosta $\Sigma = \emptyset$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Todistetaan $\forall x(P(x) \to P(x))$ Hilbert-järjestelmällä.

1.	$P(x) \to ((P(x) \to P(x)) \to P(x))$	aksioma 1
2.	$(P(x) \to ((P(x) \to P(x)) \to P(x)))$ $\to ((P(x) \to (P(x) \to P(x))) \to (P(x) \to P(x)))$	aksioma 2
3.	$(P(x) \to (P(x) \to P(x))) \to (P(x) \to P(x))$	MP, 1, 2
4.	$P(x) \to (P(x) \to P(x))$	aksioma 1
5.	$P(x) \to P(x)$	MP, 3, 4
6.	$\forall x(P(x) \to P(x))$	Yleistys, 5

- Todista $\forall x(P(x) \to P(x))$ myös semanttisella taululla!
- Suppesin järjestelmä voidaan myös yleistää predikaattilogiikalle.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

5 Normaalimuodot

- Prenex-normaalimuoto
- Konjunktiivinen normaalimuoto
- Eksistenssikvanttorien eliminointi
- Lauseiden klausuulimuoto predikaattilogiikassa

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

5.1 Prenex-normaalimuoto

Lause α on *prenex*-normaalimuodossa, mikäli se on muotoa

$$\mathcal{Q}_1 x_1 \mathcal{Q}_2 x_2 \cdots \mathcal{Q}_n x_n \phi,$$

missä jokainen $\mathcal{Q}_i$ on jompikumpi kvanttoreista ($\forall$ tai $\exists$) ja alikaava ϕ ei sisällä kvanttoreita.

Esimerkki. Seuraavat lauseet ovat prenex-normaalimuodossa:
$P(a)$, $\forall x P(x)$, $\forall x \exists y\ P(x,y)$ ja
$\forall x \exists y \forall z \forall w (P(x,y,z) \rightarrow (Q(y,z,w) \rightarrow R(z,w,x)))$.

Väite. Jokainen predikaattilogiikan lause on loogisesti ekvivalentti jonkin prenex-normaalimuodossa olevan lauseen kanssa.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Lauseiden muuttaminen prenex-normaalimuotoon

Mikä tahansa predikaattilogiikan lause voidaan muuttaa prenex-normaalimuotoon seuraavalla menettelyllä:

1. Poistetaan konnektiivit $\rightarrow$ ja $\leftrightarrow$:

$$\phi \rightarrow \psi \rightsquigarrow \neg\phi \vee \psi$$
$$\phi \leftrightarrow \psi \rightsquigarrow (\neg\phi \vee \psi) \wedge (\phi \vee \neg\psi)$$

2. Viedään negaatiot lauserakenteen sisään (atomisten kaavojen eteen):

$$\neg\neg\varphi \rightsquigarrow \varphi$$
$$\neg(\varphi \wedge \psi) \rightsquigarrow \neg\varphi \vee \neg\psi \qquad \vec{\mathcal{Q}}x\neg\forall y\varphi \rightsquigarrow \vec{\mathcal{Q}}x\exists y\neg\varphi$$
$$\neg(\varphi \vee \psi) \rightsquigarrow \neg\varphi \wedge \neg\psi \qquad \vec{\mathcal{Q}}x\neg\exists y\varphi \rightsquigarrow \vec{\mathcal{Q}}x\forall y\neg\varphi$$

Yllä $\vec{\mathcal{Q}}x$ on mikä tahansa kvanttorien sekvenssi.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

3. Tuodaan kvanttorit ulos lauserakenteesta:

$$\vec{\mathcal{Q}}x(\forall y\varphi(y) \vee \psi) \rightsquigarrow \vec{\mathcal{Q}}x\forall z(\varphi(z) \vee \psi)$$
$$\vec{\mathcal{Q}}x(\forall y\varphi(y) \wedge \psi) \rightsquigarrow \vec{\mathcal{Q}}x\forall z(\varphi(z) \wedge \psi)$$
$$\vec{\mathcal{Q}}x(\psi \vee \forall y\varphi(y)) \rightsquigarrow \vec{\mathcal{Q}}x\forall z(\psi \vee \varphi(z))$$
$$\vec{\mathcal{Q}}x(\psi \wedge \forall y\varphi(y)) \rightsquigarrow \vec{\mathcal{Q}}x\forall z(\psi \wedge \varphi(z))$$

- Yllä y korvataan uudella muuttujalla z, mikäli y esiintyy vapaana alikaavassa ψ. Muussa tapauksessa z voi aivan hyvin olla y.
- Eksistenssikvanttorit $\exists y$ käsitellään samaan tapaan (saadaan 4 vastaavanmuotoista sääntöä lisää).

Esimerkki. Muuttujan korvaaminen uudella on olennaista:

$$\forall x P(x) \vee \forall x Q(x) \rightsquigarrow \forall x(P(x) \vee \forall x Q(x)) \rightsquigarrow \forall x \forall y(P(x) \vee Q(y)).$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Suoritetaan muunnos prenex-normaalimuotoon:

$$\begin{aligned}
& \forall x(P(x) \to \exists z R(z,x)) \to \exists x Q(x) \\
\leadsto\ & \neg\forall x(P(x) \to \exists z R(z,x)) \vee \exists x Q(x) \\
\leadsto\ & \neg\forall x(\neg P(x) \vee \exists z R(z,x)) \vee \exists x Q(x) \\
\leadsto\ & \exists x\neg(\neg P(x) \vee \exists z R(z,x)) \vee \exists x Q(x) \\
\leadsto\ & \exists x(\neg\neg P(x) \wedge \neg\exists z R(z,x)) \vee \exists x Q(x) \\
\leadsto\ & \exists x(P(x) \wedge \forall z\neg R(z,x)) \vee \exists x Q(x) \\
\leadsto\ & \exists x(\exists x(P(x) \wedge \forall z\neg R(z,x)) \vee Q(x)) \\
\leadsto\ & \exists x\exists y((P(y) \wedge \forall z\neg R(z,y)) \vee Q(x)) \\
\leadsto\ & \exists x\exists y(\forall z(P(y) \wedge \neg R(z,y)) \vee Q(x)) \\
\leadsto\ & \exists x\exists y\forall z((P(y) \wedge \neg R(z,y)) \vee Q(x)).
\end{aligned}$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

5.2 Konjunktiivinen normaalimuoto

Määritelmä. *Literaalit* ovat joko

1. atomikaavoja $P(\vec{t})$ (eli *positiivisia literaaleja*) tai
2. atomikaavojen negaatioita $\neg P(\vec{t})$ (eli *negatiivisia literaaleja*).

Määritelmä. Lause α on konjunktiivisessa normaalimuodossa, mikäli se on on prenex-normaalimuodossa $\mathcal{Q}_1 x_1 \mathcal{Q}_2 x_2 \cdots \mathcal{Q}_n x_n \phi$, missä kvanttoreita sisältämätön osa $\phi = \phi_1 \wedge \cdots \wedge \phi_n$ ja jokainen konjunktion jäsen ϕ_i on muodoltaan literaalien disjunktio.

Esimerkki. Lause

$$\forall x\exists y\forall z((\neg P(x,y) \vee Q(y,x)) \wedge R(z) \wedge (\neg R(x) \vee P(y,z) \vee Q(x,z)))$$

on konjunktiivisessa normaalimuodossa.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Tarvittaessa prenex-normaalimuodon kvanttoreita sisältämätön osa voidaan järjestää konjunktiiviseen normaalimuotoon seuraavasti:

$$\begin{aligned}
\varphi \vee (\phi \wedge \psi) &\quad\leadsto\quad (\varphi \vee \phi) \wedge (\varphi \vee \psi) \\
(\phi \wedge \psi) \vee \varphi &\quad\leadsto\quad (\phi \vee \varphi) \wedge (\psi \vee \varphi)
\end{aligned}$$

- Näin ollen voimme todeta seuraavan tuloksen:

Väite. Jokainen predikaattilogiikan lause on loogisesti ekvivalentti jonkin konjunktiivisessa normaalimuodossa olevan lauseen kanssa.

Esimerkki. Muunnetaan edellä johdettu prenex-normaalimuoto edelleen konjunktiiviseksi normaalimuodoksi:

$$\begin{aligned}
& \exists x\exists y\forall z((P(y) \wedge \neg R(z,y)) \vee Q(x)) \\
\leadsto\ & \exists x\exists y\forall z((P(y) \vee Q(x)) \wedge (\neg R(z,y) \vee Q(x))).
\end{aligned}$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

5.3 Eksistenssikvanttorien eliminointi

Esimerkki. Tarkastellaan kahta kokonaislukuja koskevaa väittämää:

1. Summafunktiolla on vasen identiteetti:

$$\exists x\forall y(x + y = y).$$

   Identiteettialkio voidaan nimetä vakiosymbolilla 0:

$$\forall y(0 + y = y).$$

2. Jokaisella kokonaisluvulla on vastaluku:

$$\forall x\exists y(x + y = 0).$$

   Vastalukufunktio voidaan nimetä funktiosymbolilla $-$:

$$\forall x(x + -(x) = 0).$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Eksistenssikvanttorien eliminointi yleisessä tapauksessa

Olkoon $\vec{\mathcal{Q}}x\phi$ prenex-normaalimuodossa ja $\exists x$ kvanttorisekvenssin $\vec{\mathcal{Q}}x$ ensimmäinen eksistenssikvanttori.

1. Jos kvanttori $\exists x$ on sekvenssissä $\vec{\mathcal{Q}}x$ vieläpä ensimmäisenä, poistetaan $\exists x$ sekvenssistä ja korvataan näin syntyvät muuttujan x vapaat esintymät jollain uudella vakiolla c.
2. Jos kvanttoria $\exists x$ esiintyy sekvenssissä $\vec{\mathcal{Q}}x$ universaalikvanttorien $\forall y_1 \cdots \forall y_n$ jälkeen, poistetaan kvanttori $\exists x$ ja korvataan näin syntyvät muuttujan x vapaat esiintymät termillä $f(y_1, \ldots, y_n)$, missä f on uusi funktiosymboli.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Eksistenssikvanttorien eliminointia kutsutaan kehittäjänsä mukaan Skolemoinniksi ja vastaavasti ko. prosessissa valittavia uusia vakio- ja funktiosymboleita Skolem-vakioiksi ja -funktioiksi.

Esimerkki. Suoritetaan Skolemointi seuraaville lauseille:

$$\begin{aligned}
\exists x P(x) &\rightsquigarrow P(c) \\
\exists x \forall y \exists x P(x,y) &\rightsquigarrow \forall y P(f(y), y) \\
\forall x \exists y (P(x,y) \wedge Q(y,x)) &\rightsquigarrow \forall x\ (P(x, f(x)) \wedge Q(f(x), x)) \\
\forall x \exists y \forall z \exists w P(w,z,y,x) &\rightsquigarrow \forall x \forall z P(g(x,z), z, f(x), x) \\
\exists x \exists y \forall z P(x,y,z,x) &\rightsquigarrow \forall z P(c_1, c_2, z, c_1) \\
\forall x \exists y \exists z P(x,y,z) &\rightsquigarrow \forall x P(x, f_1(x), f_2(x))
\end{aligned}$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Skolemoinnin loogiset ominaisuudet

Väite. Prenex-normaalimuodossa oleva lause φ on toteutuva $\iff$ lauseen φ skolemoitu muoto φ' on toteutuva.

Huomio. Prenex-normaalimuodossa olevan lauseen φ skolemoitu muoto φ' **ei välttämättä ole** loogisesti ekvivalentti lauseen φ kanssa.

Esimerkki. Lause $\exists x P(x)$ ja sen skolemoitu muoto $P(c)$.

Nyt $\models P(c) \rightarrow \exists x P(x)$, mutta $\not\models \exists x P(x) \rightarrow P(c)$.

Vastamalli $\mathcal{A}$: universumi $A = \{1, 2\}$, $c^{\mathcal{A}} = 1$ ja $P^{\mathcal{A}} = \{2\}$.
Nyt $\mathcal{A} \models \exists x P(x)$, mutta $\mathcal{A} \not\models P(c)$.

Täten myös $\not\models \exists x\ P(x) \leftrightarrow P(c)$ ja edelleen $\exists x P(x) \not\equiv P(c)$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

5.4 Lauseiden klausuulimuoto predikaattilogiikassa

Mille tahansa lauseelle voidaan hakea klausuulimuoto seuraavasti:

1. Haetaan prenex-normaalimuoto.
2. Muunnetaan tämä konjunktiiviseen normaalimuotoon.
3. Tarvittaessa poistetaan eksistenssikvanttorit Skolemoimalla.
4. Kirjoitetaan klausuuliesitys (joukko literaalien joukkoja).

Esimerkki. Klausuuliesitys lauseelle $\forall x(\neg(P(x) \rightarrow \forall y Q(x,y)) \vee R(x))$:

$$\begin{aligned}
&\rightsquigarrow \forall x \exists z((P(x) \wedge \neg Q(x,z)) \vee R(x)) && (1) \\
&\rightsquigarrow \forall x \exists z((P(x) \vee R(x)) \wedge (\neg Q(x,z) \vee R(x))) && (2) \\
&\rightsquigarrow \forall x((P(x) \vee R(x)) \wedge (\neg Q(x, f(x)) \vee R(x))) && (3) \\
&\rightsquigarrow \{\{P(x), R(x)\}, \{\neg Q(x, f(x)), R(x)\}\}. && (4)
\end{aligned}$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Huomio. Jos lause on muodoltaan *konjunktio* $\phi_1 \wedge \cdots \wedge \phi_n$, on mahdollista saattaa konjunktion jäsenet $\phi_1, \ldots, \phi_n$ klausuulimuotoon *erikseen*. Muista myös, että $\neg(\phi_1 \wedge \cdots \wedge \phi_n \rightarrow \phi) \equiv \phi_1 \wedge \cdots \wedge \phi_n \wedge \neg\phi$.

Esimerkki. $\forall x P(x) \wedge \exists x Q(x)$. Konjunktion jäsenille saadaan klausuuliesitykset $\{\{P(x)\}\}$ ja $\{\{Q(c)\}\}$ ja näinollen koko lauseen klausuuliesitykseksi näiden unioni $\{\{P(x)\}, \{Q(c)\}\}$.

Huomio. Existenssikvanttorit kannattaa tuoda ulos lauserakenteesta ennen universaalikvanttoreita (mikäli mahdollista).

Esimerkki. Siis $\forall x P(x) \vee \exists x Q(x)$ kirjoitetaan muotoon $\exists x \forall y (P(y) \vee Q(x))$ eikä muotoon $\forall x \exists y (P(x) \vee Q(y))$.

Näin saatetaan välttää Skolem-funktioiden käyttöönotto tai ainakin vähennetään Skolem-funktioiden argumenttien lukumäärää $\Longrightarrow$ klausuulimuodosta tulee rakenteeltaan yksinkertaisempi.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

6 Tietämyksen esittämisestä

- Tietämyksen esittäminen predikaattilogiikalla
- Ohjeita predikaattien määrittelemiseen
- Nimien yksikäsitteisyys ja kattavuus
- Negatiiviset ehdot ja johtopäätökset

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

6.1 Tietämyksen esittäminen predikaattilogiikalla

Annettuun järjestelmään liittyvää tietämystä voidaan esittää valitsemalla

- sopiva predikaattilogiikan aakkosto (joukot $\mathcal{P}$, $\mathcal{C}$ ja $\mathcal{F}$) ja
- vastaavaan kieleen $\mathcal{L}$ perustuva lausejoukko $\Sigma \subseteq \mathcal{L}$, jonka lauseet määrittelevät järjestelmän ominaisuudet.

Tarkastellaan määritelmäjoukon Σ loogisten seurausten joukkoa

$$\mathsf{Cn}(\Sigma) = \{\phi \in \mathcal{L} \mid \Sigma \models \phi\}.$$

Nyt

- Σ muodostaa järjestelmää koskevan *eksplisiittisen tietämyksen* ja
- joukon $\mathsf{Cn}(\Sigma) - \Sigma$ lauseet ovat *implisiittistä tietämystä* eli väittämiä, jotka voidaan päätellä eksplisiittisestä tietämyksestä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Kuvataan sähkölinjaa predikaattilogiikan lausein:

$$\begin{aligned}\Sigma \;=\; \{\, & \forall x \forall y \forall z(\mathsf{johto}(x,y) \wedge \mathsf{jännite}(x,z) \rightarrow \mathsf{jännite}(y,z)),\\ & \mathsf{johto}(\mathsf{p1},\mathsf{p2}),\ \mathsf{johto}(\mathsf{p2},\mathsf{p3}),\ \mathsf{jännite}(\mathsf{p1},220)\,\}.\end{aligned}$$

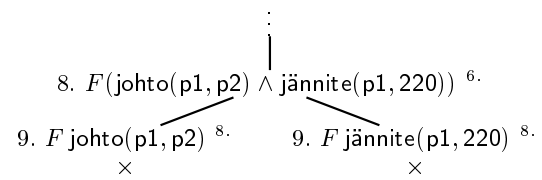
Nyt esim. lause $\mathsf{jännite}(\mathsf{p3}, 220)$ on implisiittistä tietämystä:

1. $T\ \forall x \forall y \forall z(\mathsf{johto}(x,y) \wedge \mathsf{jännite}(x,z) \rightarrow \mathsf{jännite}(y,z))$
2. $T\ \mathsf{johto}(\mathsf{p1},\mathsf{p2})$
3. $T\ \mathsf{johto}(\mathsf{p2},\mathsf{p3})$
4. $T\ \mathsf{jännite}(\mathsf{p1},220)$
5. $F\ \mathsf{jännite}(\mathsf{p3},220)$
6. $T\ \mathsf{johto}(\mathsf{p1},\mathsf{p2}) \wedge \mathsf{jännite}(\mathsf{p1},220) \rightarrow \mathsf{jännite}(\mathsf{p2},220)$ $^{1.\ x/\mathsf{p1},y/\mathsf{p2},z/220}$
7. $T\ \mathsf{johto}(\mathsf{p2},\mathsf{p3}) \wedge \mathsf{jännite}(\mathsf{p2},220) \rightarrow \mathsf{jännite}(\mathsf{p3},220)$ $^{1.\ x/\mathsf{p2},y/\mathsf{p3},z/220}$

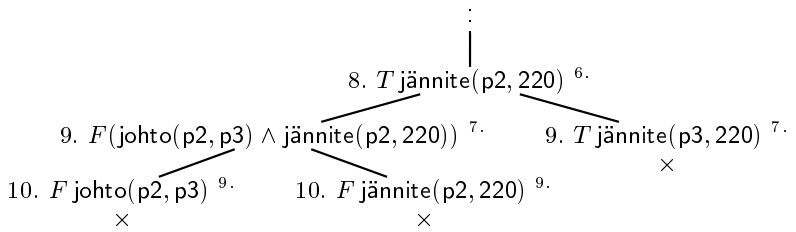
© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Hajoitettaessa polun 6. solmu taulu jakautuu kahteen haaraan:

- Vasen haara (τ_1):



- Oikea haara (τ_2):



© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Tietämyksen esittämisen problematiikkaa

- Kaikki struktuurit ovat tyhjän lausejoukon $\emptyset$ malleja, joten $\mathsf{Cn}(\emptyset)$ on itse asiassa pätevien lauseiden joukko.
- Enemmän lauseita $\Rightarrow$ vähemmän malleja $\Rightarrow$ enemmän loogisia seurauksia: siis $\Sigma_1 \subseteq \Sigma_2 \Rightarrow \mathsf{Cn}(\Sigma_1) \subseteq \mathsf{Cn}(\Sigma_2)$ (*monotonisuus*).
- Jos lausejoukko tulee ristiriitaiseksi, sillä ei ole yhtään mallia ja kaikki lauseet ovat tällöin lausejoukon loogisia seurauksia.

$\Longrightarrow$ Tavoitteena rajata predikaattilogiikan lausejoukolla $\Sigma \subseteq \mathcal{L}$ mallien joukko siten, että saadaan halutut loogiset seuraukset.

Esimerkki. Jos $\Sigma = \{\forall x(P(x) \rightarrow Q(x))\}$, niin $\Sigma \not\models Q(a)$.

- Vastamalli $\mathcal{A}$: universumi $A = \{1\}$, $a^{\mathcal{A}} = 1$ ja $P^{\mathcal{A}} = Q^{\mathcal{A}} = \emptyset$.
- Esim. lisäämällä $P(a)$ saadaan $\Sigma' = \Sigma \cup \{P(a)\}$, jolle $\Sigma' \models Q(a)$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

6.2 Ohjeita predikaattien määrittelemiseen

- Tavoitteena kirjoittaa predikaatille P määritelmä joidenkin muiden predikaattien avulla.
- Mielivaltainen predikaattilogiikan kaava ϕ voidaan saattaa muotoon

$$\mathcal{Q}_1 x_1 \mathcal{Q}_2 x_2 \cdots \mathcal{Q}_n x_n \psi$$

  missä kukin kvanttori $\mathcal{Q}_i$ on joko $\forall$ tai $\exists$, ja kaava ψ on konjunktiivisessa normaalimuodossa eikä sisällä kvanttoreita.
- Yllä $\psi = \psi_1 \wedge \cdots \wedge \psi_m$, missä kukin ψ_i on literaalien disjunktio

$$\begin{aligned} & \neg Q_1(\vec{t_1}) \vee \cdots \vee \neg Q_k(\vec{t_k}) \vee P_1(\vec{s_1}) \vee \cdots \vee P_l(\vec{s_l}) \\ \equiv \quad & Q_1(\vec{t_1}) \wedge \cdots \wedge Q_k(\vec{t_k}) \rightarrow P_1(\vec{s_1}) \vee \cdots \vee P_l(\vec{s_l}). \end{aligned}$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Predikaattilogiikalla annetut määritelmät ovat korkeintaan yhtä monimutkaisia kuin edellä kuvattu normaalimuoto.
- Jos jokainen kvanttori $\mathcal{Q}_i$ on universaalikvanttori $\forall$, $m = 1$ ja $l = 1$, saamme erikoistapauksena muotoa

$$\forall x_1 \forall x_2 \cdots \forall x_n (Q_1(\vec{t_1}) \wedge \cdots \wedge Q_k(\vec{t_k}) \rightarrow P(\vec{t}))$$

  olevia lauseita, missä predikaattien $Q_1, \ldots, Q_k$ ja P argumentteina on vakiosymboleista, muuttujista $x_1, \ldots, x_n$ ja funktiosymboleista rakentuvien termien jonoja $\vec{t_1}, \ldots, \vec{t_k}$ ja $\vec{t}$.
- Määritelmiä voidaan usein kirjoittaa tähän muotoon: mietitään millä ehdoilla $Q_1(\vec{t_1}), \ldots, Q_k(\vec{t_k})$ voidaan päätellä predikaattia P koskeva väittämä $P(\vec{t})$. Ko. muotoa olevia lauseita saatetaan tarvita useita.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Olkoon annettuna predikaatit

1. sairastaa(x) = "henkilö x on sairas" ja
2. tapaa(x, y) = "henkilö x tapaa henkilön y".

Tarkoituksena on määritellä näiden avulla predikaatti

$$\text{tartuntavaarassa}(x) = \text{"henkilö } x \text{ on tartuntavaarassa"}.$$

Kysymys: millä ehdoilla jonkin henkilö on tartuntavaarassa?

1. Jos henkilö tapaa jonkun sairaan henkilön.
2. Jos henkilö tapaa jonkun toisen tartuntavaarassa olevan henkilön.

Yritetään kirjoittaa nämä edellä esitetyn mukaisesti muotoon

$$\forall x \forall y \ldots (Q_1(\vec{t_1}) \wedge \ldots \wedge Q_k(\vec{t_k}) \rightarrow \text{tartuntavaarassa}(x)).$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Näin saadaan muodostettua lauseet

  $\forall x \forall y(\text{tapaa}(x, y) \wedge \text{sairastaa}(y) \rightarrow \text{tartuntavaarassa}(x))$ ja
  $\forall x \forall y(\text{tapaa}(x, y) \wedge \text{tartuntavaarassa}(y) \rightarrow \text{tartuntavaarassa}(x))$.

- Kysymyksessä on itse asiassa tartuntavaarassa-predikaatin induktiivinen (rekursiivinen) määritelmä. Lauseista ensimmäinen vastaa perustapausta ja jälkimmäinen induktioaskelta.
- Lisäksi voidaan todeta tapaamiset symmetrisiksi:

  $$\forall x \forall y(\text{tapaa}(x, y) \rightarrow \text{tapaa}(y, x)).$$

- Universumin voidaan ajatella koostuvan pelkästään henkilöistä (eli edellä annetut lauseet puhuvat kaikista henkilöistä).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmien käyttö konkreettisessa päättelyssä

Esimerkki. Lisätään edellä johdettuun tartuntavaarassa-predikaatin määritelmään tietokanta, jossa kuvataan tapaamiset ja sairastamiset:

Näin saadaan lausejoukko

$$\begin{aligned}\Sigma = \{\ & \forall x \forall y(\text{tapaa}(x, y) \wedge \text{sairastaa}(y) \rightarrow \text{tartuntavaarassa}(x)),\\ & \forall x \forall y(\text{tapaa}(x, y) \wedge \text{tartuntavaarassa}(y) \rightarrow \text{tartuntavaarassa}(x)),\\ & \forall x \forall y(\text{tapaa}(x, y) \rightarrow \text{tapaa}(y, x)),\\ & \text{tapaa}(\text{Lyyli}, \text{Hemmo}),\ \text{tapaa}(\text{Lyyli}, \text{Erkki}),\ \text{sairastaa}(\text{Erkki})\ \}.\end{aligned}$$

- Kyseisessä asetelmassa saadaan
  $\Sigma \models \text{tartuntavaarassa}(\text{Lyyli}) \wedge \text{tartuntavaarassa}(\text{Hemmo})$.
- Kokeile tämän osoittamista semanttisella taululla!

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Suoritetaan vastaava päättely OTTER-teoreemantodistimella. Tarvittava syötetiedosto:

```
set(auto).
formula_list(usable).

% Section A: database
tapaa(lyyli,hemmo). tapaa(lyyli,erkki). sairastaa(erkki).

% Section B: definitions
all x y (tapaa(x,y) -> tapaa(y,x)).
all x y (tapaa(x,y) & sairastaa(y) -> tartuntavaarassa(x)).
all x y (tapaa(x,y) & tartuntavaarassa(y) -> tartuntavaarassa(x)).

% Section C: negation of the query
-(tartuntavaarassa(lyyli) & tartuntavaarassa(hemmo)).

end_of_list.
```

- OTTER pystyy osoittamaan lausejoukon helposti ristiriitaiseksi.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Tyypitetyt kvanttorit

- Usein on mielekästä ajatella universumin koostuva tyypiltään erilaisista alkioista.
- Tällöin syntyy tarve rajata kvantifiointia koskemaan ainoastaan tiettyä tyyppiä T olevia alkioita seuraavaan tapaan:

$$\forall x \in T : \phi(x) \text{ ja } \exists x \in T : \phi(x).$$

- Tyyppi T voidaan esittää yksipaikkaisen predikaatin avulla:

$$T(x) = \text{“alkio } x \text{ on tyyppiä } T\text{”}.$$

- Tyypitetyt kvanttorit ilmaistaan predikaattilogiikassa seuraavasti:

$$\forall x(T(x) \to \phi(x)) \text{ ja } \exists x(T(x) \wedge \phi(x)).$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Lisätään edellisen esimerkkiin tyyppipredikaatteja.

- Määritellään predikaatit henkilöiden ja tautien erottelemiseksi: henkilö$(x) =$ “x on henkilö” ja tauti$(x) =$ “x on tauti”.
- Määritellään predikaatit ilman tyyppi-informaatiota:
  – tapaa$(x, y) =$ “x tapaa y:n”,
  – sairastaa$(x, y) =$ “x sairastaa y:tä” ja
  – tartuntavaarassa$(x, y) =$ “x on vaarassa sairastua y:hyn”.
- Lauseet saadaan nyt seuraavaan muotoon:

$$\forall x \forall y \forall z(\text{henkilö}(x) \wedge \text{henkilö}(y) \wedge \text{tapaa}(x, y) \wedge \text{tauti}(z) \wedge \text{sairastaa}(y, z) \to \text{tartuntavaarassa}(x, z)) \text{ ja}$$

$$\forall x \forall y \forall z(\text{henkilö}(x) \wedge \text{henkilö}(y) \wedge \text{tapaa}(x, y) \wedge \text{tauti}(z) \wedge \text{tartuntavaarassa}(y, z) \to \text{tartuntavaarassa}(x, z)).$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Tyypeillä voi olla erilaisia suhteita:
  – Erillisyys: $\forall x \neg(\text{henkilö}(x) \wedge \text{tauti}(x))$.
  – Kattavuus: $\forall x(\text{henkilö}(x) \vee \text{tauti}(x))$.
  – Alityyppi: $\forall x(\text{rokko}(x) \to \text{tauti}(x))$.
- Yksi mahdollisuus on tyypittää predikaatit erikseen:

$$\forall x \forall y(\text{tapaa}(x, y) \to \text{henkilö}(x) \wedge \text{henkilö}(y))$$

$$\forall x \forall y(\text{sairastaa}(x, y) \to \text{henkilö}(x) \wedge \text{tauti}(y))$$

- Tällöin varsinainen tartuntavaarassa-predikaatin määritelmä voidaan kirjoittaa yksinkertaisemmin ilman tyyppi-informaatiota:

$\forall x \forall y \forall z(\text{tapaa}(x, y) \wedge \text{sairastaa}(y, z) \to \text{tartuntavaarassa}(x, z))$ ja
$\forall x \forall y \forall z(\text{tapaa}(x, y) \wedge \text{tartuntavaarassa}(y) \to \text{tartuntavaarassa}(x, z))$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Muodoltaan monimutkaisempia määritelmiä

- Edellä otettiin lähtökohdaksi muotoa

$$\forall x_1 \forall x_2 \cdots \forall x_n(Q_1(\vec{t_1}) \wedge \cdots \wedge Q_k(\vec{t_k}) \to P(\vec{t}))$$

  olevat määritelmät. Näiden ilmaisuvoima ei ole aina riittävä.
- Joissain tilanteissa tarvitaan eksistentiaalista kvantifiointia:

$$\forall x(\text{solmu}(x) \to \exists y(\text{väri}(y) \wedge \text{väritetty}(x, y)))$$
$$\equiv \quad \forall x \exists y(\text{solmu}(x) \to \text{väri}(y) \wedge \text{väritetty}(x, y)).$$

- Implikaation seurauksena voi olla myös atomien disjunktio $P_1(\vec{s_1}) \vee \cdots \vee P_l(\vec{s_l})$ pelkän atomin $P(\vec{t})$ sijaan:

$$\forall x(\text{bitti}(x) \to \text{nolla}(x) \vee \text{yksi}(x)).$$

Huomio. Edellä oli keskeistä vaihtoehtoisuuden ilmaiseminen.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

6.3 Nimien yksikäsitteisyys ja kattavuus

- Rajoitutaan jatkossa predikaattilogiikan kieliin $\mathcal{L}$, joissa ei ole funktiosymboleita ja ainoastaan äärellinen määrä vakiosymboleita.
- Predikaattilogiikassa struktuurin $\mathcal{A}$ määritelmä ja tapa jolla vakiosymbolit tulkitaan $\mathcal{A}$:ssa mahdollistavat, että
  1. jokin universumin alkio $a \in A$ on useamman vakion $c_1, \ldots, c_n$ $(n > 1)$ nimeämä: $c_1^{\mathcal{A}} = \ldots = c_n^{\mathcal{A}} = a$.
  2. jokin universumin alkio $a \in A$ ei ole minkään vakion nimeämä (eli kaikille vakiosymboleille c pätee $c^{\mathcal{A}} \neq a$).
- Tietämyksen esittämisen kannalta tällainen mahdollisuus muodostuu usein jopa turhaksi vapausasteeksi.
- Nimeäminen voidaan pakottaa yksikäsitteiseksi lauseita lisäämällä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Nimien yksikäsitteisyys

- Vastaava käsite englanniksi on *unique names assumption* (UNA).
- Kun kielessä on äärellinen määrä vakiosymboleita $c_1, \ldots, c_n$, riittää lisätä muotoa

  $$\neg(c_i = c_j)$$

  olevat lauseet, missä $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, n\}$ ja $i < j$.
- Lauseita tarvitaan neliöllinen määrä (yhteensä $\frac{n^2-n}{2}$ kappaletta).

Esimerkki. Olkoon kielessä $\mathcal{L}$ vakiosymbolit Lyyli, Hemmo ja Erkki. Yksikäsitteisten nimien oletus ilmaistaan seuraavasti:

$\neg(\text{Lyyli} = \text{Hemmo})$, $\neg(\text{Lyyli} = \text{Erkki})$ ja $\neg(\text{Hemmo} = \text{Erkki})$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Tarkastellaan lausejoukon

$$\Sigma_{\text{UNA}} = \{\neg(\text{Lyyli} = \text{Hemmo}), \neg(\text{Lyyli} = \text{Erkki}), \neg(\text{Hemmo} = \text{Erkki})\}$$

malleja, kun universumina A_i on joukko henkilöitä $h_1, h_2, \ldots$.

A_i	$\text{Lyyli}^{\mathcal{A}_i}$	$\text{Hemmo}^{\mathcal{A}_i}$	$\text{Erkki}^{\mathcal{A}_i}$
$\{h_1, h_2, h_3\}$	h_1	h_2	h_3
$\{h_1, h_2, h_3\}$	h_1	h_3	h_2
$\{h_1, h_2, h_3\}$	h_2	h_1	h_3
$\{h_1, h_2, h_3\}$	h_2	h_3	h_1
$\{h_1, h_2, h_3\}$	h_3	h_1	h_2
$\{h_1, h_2, h_3\}$	h_3	h_2	h_1
$\{h_1, h_2, h_3, h_4\}$	h_1	h_2	h_3
$\vdots$	$\vdots$	$\vdots$	$\vdots$

$\Longrightarrow$ Universumissa oltava vähintään 3 henkilöä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Nimien kattavuus

- Vastaava käsite englanniksi on *domain closure assumption* (DCA).
- Kun kielessä on äärellinen määrä vakiosymboleita $c_1, \ldots, c_n$, riittää lisätä seuraavaa muotoa oleva lause:

  $$\forall x(x = c_1 \vee \cdots \vee x = c_n).$$

- Tarvittavan lauseen pituus riippuu lineaarisesti vakioiden lukumäärästä n.

Esimerkki. Edellisen esimerkin mukaisessa kielessä tarvitaan lause

$$\forall x(x = \text{Lyyli} \vee x = \text{Hemmo} \vee x = \text{Erkki}).$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Tarkastellaan lausejoukon

$$\Sigma_{\mathrm{DCA}} = \{\forall x(x = \mathsf{Lyyli} \vee x = \mathsf{Hemmo} \vee x = \mathsf{Erkki})\}$$

malleja, kun universumina A_i on joukko henkilöitä $h_1, h_2, \ldots$.

A_i	$\mathsf{Lyyli}^{\mathcal{A}_i}$	$\mathsf{Hemmo}^{\mathcal{A}_i}$	$\mathsf{Erkki}^{\mathcal{A}_i}$
$\{h_1\}$	h_1	h_1	h_1
$\{h_1, h_2\}$	h_1	h_1	h_2
$\{h_1, h_2\}$	h_1	h_2	h_1
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\{h_1, h_2, h_3\}$	h_1	h_2	h_3
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\{h_1, h_2, h_3\}$	h_3	h_2	h_1

$\Longrightarrow$ Universumissa voi olla korkeintaan 3 henkilöä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Tarkastellaan vielä edeltävien lausejoukkojen unionin

$$\begin{aligned}\Sigma_{\mathrm{UNA}} \cup \Sigma_{\mathrm{DCA}} = \{\ & \neg(\mathsf{Lyyli} = \mathsf{Hemmo}), \neg(\mathsf{Lyyli} = \mathsf{Erkki}),\\ & \neg(\mathsf{Hemmo} = \mathsf{Erkki}),\\ & \forall x(x = \mathsf{Lyyli} \vee x = \mathsf{Hemmo} \vee x = \mathsf{Erkki})\ \}\end{aligned}$$

malleja, kun universumina on joukko henkilöitä $h_1, h_2, \ldots$.

A_i	$\mathsf{Lyyli}^{\mathcal{A}_i}$	$\mathsf{Hemmo}^{\mathcal{A}_i}$	$\mathsf{Erkki}^{\mathcal{A}_i}$
$\{h_1, h_2, h_3\}$	h_1	h_2	h_3
$\{h_1, h_2, h_3\}$	h_1	h_3	h_2
$\{h_1, h_2, h_3\}$	h_2	h_1	h_3
$\{h_1, h_2, h_3\}$	h_2	h_3	h_1
$\{h_1, h_2, h_3\}$	h_3	h_1	h_2
$\{h_1, h_2, h_3\}$	h_3	h_2	h_1

$\Longrightarrow$ Universumissa on oltava täsmälleen 3 henkilöä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

6.4 Negatiiviset ehdot ja johtopäätökset

- Tarkastellaan muotoa

  $$\forall x_1 \forall x_2 \cdots \forall x_n (Q_1(\vec{t_1}) \wedge \cdots \wedge Q_k(\vec{t_k}) \rightarrow P(\vec{t}))$$

  olevien määritelmien yleistämistä tapaukseen, missä sallitaan atomien $Q_i(\vec{t_i})$ lisäksi myös negatiivisia literaaleja $\neg Q_i(\vec{t_i})$.

- Negatiivinen ehto $\neg Q_i(\vec{t_i})$ voidaan muuntaa positiiviseksi vaihtoehdoksi $Q_i(\vec{t_i})$ seuraukselle $P(\vec{t})$.

  Esimerkki.

  $$\begin{aligned} & \forall x(\neg \mathsf{sairastaa}(x) \wedge \neg \mathsf{tartuntavaarassa}(x) \rightarrow \mathsf{turvassa}(x))\\ \equiv\ & \forall x(\mathsf{sairastaa}(x) \vee \mathsf{tartuntavaarassa}(x) \vee \mathsf{turvassa}(x)).\end{aligned}$$

- Jotta negatiiviset ehdot tulisivat määritellyiksi, määritelmistä tulisi seurata loogisesti $\neg Q_i(\vec{t_i})$ mikäli $Q_i(\vec{t_i})$ ei ole looginen seuraus.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmien täydellisyys

Määritelmä. Predikaatin $P \in \mathcal{P}_n$ määritelmä Σ (kielen $\mathcal{L}$ lausejoukko) on *täydellinen*, mikäli kaikille kielen $\mathcal{L}$ muuttujattomille termeille $t_1, \ldots, t_n$ pätee joko $\Sigma \models P(t_1, \ldots, t_n)$ tai $\Sigma \models \neg P(t_1, \ldots, t_n)$.

Huomioita.

- Jos predikaatin $P \in \mathcal{P}_n$ määritelmä Σ on ristiriitainen, se on triviaalisti täydellinen: kaikille muuttujattomille termeille $t_1, \ldots, t_n$ pätee tällöin $\Sigma \models P(t_1, \ldots, t_n)$.
- Jos predikaatin $P \in \mathcal{P}_n$ määritelmä Σ on sekä ristiriidaton että täydellinen ja $\Sigma \not\models P(t_1, \ldots, t_n)$ joillekin muuttujattomille termeille $t_1, \ldots, t_n$, niin $\Sigma \models \neg P(t_1, \ldots, t_n)$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Tarkastellaan muunnelmaa tartuntavaara-esimerkistä:

$$\Sigma = \{\ \forall x \forall y(\text{tapaa}(x,y) \wedge \text{sairastaa}(y) \rightarrow \text{tartuntavaarassa}(x)),$$
$$\forall x(\neg\text{sairastaa}(x) \wedge \neg\text{tartuntavaarassa}(x) \rightarrow \text{turvassa}(x)),$$
$$\text{tapaa}(\text{Lyyli}, \text{Hemmo}),\ \text{tapaa}(\text{Lyyli}, \text{Erkki}),\ \text{sairastaa}(\text{Erkki})\ \}.$$

- Nyt $\Sigma \models \text{tartuntavaarassa}(\text{Lyyli})$, $\Sigma \not\models \text{tartuntavaarassa}(\text{Hemmo})$ ja $\Sigma \not\models \neg\text{tartuntavaarassa}(\text{Hemmo})$.
- Täten Σ ei ole täydellinen määritelmä tartuntavaarassa-predikaatille.
- Jotta näin olisi, määritelmästä tulisi seurata loogisesti $\neg\text{tartuntavaarassa}(\text{Hemmo})$ ja $\neg\text{tartuntavaarassa}(\text{Erkki})$.
- Määritelmäjoukko Σ ei ole täydellinen myöskään muille ko. kielen predikaateille (tapaa, sairastaa, turvassa ja $=$).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Predikaatin määritelmän täydentäminen

Esimerkki. Täydennetään edellisen esimerkin predikaattien määritelmät.

1. Yhtäsuuruuspredikaatin osalta riittää todeta nimien yksikäsitteisyys:

   $\neg(\text{Lyyli} = \text{Hemmo})$, $\neg(\text{Lyyli} = \text{Erkki})$, $\neg(\text{Hemmo} = \text{Erkki})$ ja
   $\forall x(x = \text{Lyyli} \vee x = \text{Hemmo} \vee x = \text{Erkki})$.

2. Predikaateille tapaa ja sairastaa saadaan tiiviit esitykset yhtäsuuruuspredikaatin avulla:

   $$\forall x(\text{sairastaa}(x) \leftrightarrow x = \text{Erkki}) \text{ ja}$$
   $$\forall x(\text{tapaa}(x,y) \leftrightarrow (x = \text{Lyyli} \wedge y = \text{Hemmo}) \vee (x = \text{Lyyli} \wedge y = \text{Erkki})).$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

3. Predikaattien tartuntavaarassa ja turvassa määritelmät voidaan kirjoittaa vastaavasti ekvivalensseiksi:

   $$\forall x(\text{tartuntavaarassa}(x) \leftrightarrow \exists y(\text{tapaa}(x,y) \wedge \text{sairastaa}(y))) \text{ ja}$$
   $$\forall x(\text{turvassa}(x) \leftrightarrow \neg\text{sairastaa}(x) \wedge \neg\text{tartuntavaarassa}(x)).$$

- Täydennetyillä määritelmillä on haluamamme loogiset seuraukset:

sairastaa	tartuntavaarassa	turvassa
$\neg$sairastaa(Lyyli)	tartuntavaarassa(Lyyli)	$\neg$turvassa(Lyyli)
$\neg$sairastaa(Hemmo)	$\neg$tartuntavaarassa(Hemmo)	turvassa(Hemmo)
sairastaa(Erkki)	$\neg$tartuntavaarassa(Erkki)	$\neg$turvassa(Erkki)

- Predikaattien täydentäminen ei valitettavasti tuota haluttua lopputulosta *rekursiivisten* määritelmien tapauksessa, kuten seuraavassa esimerkissä osoitetaan.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Tarkastellaan vastaavaa konstruktiota lausejoukolle

$$\Sigma = \{\ \text{tuntee1}(\text{Lyyli}, \text{Lyyli}), \text{tuntee1}(\text{Hemmo}, \text{Hemmo}),$$
$$\forall x \forall y(\text{tuntee1}(x,y) \rightarrow \text{tuntee2}(x,y)),$$
$$\forall x \forall y(\text{tuntee2}(y,x) \rightarrow \text{tuntee2}(x,y))\ \}.$$

- Rekursiivisesti määritellyn predikaatin tuntee2 tarkoituksena on täydentää predikaatti tuntee1 symmetriseksi.
- Täydennettynä määritelmät saadaan muotoon

$$\Sigma' = \{\ \neg(\text{Lyyli} = \text{Hemmo}), \quad \forall x(x = \text{Lyyli} \vee x = \text{Hemmo}),$$
$$\forall x \forall y(\text{tuntee1}(x,y) \leftrightarrow (x = \text{Lyyli} \wedge y = \text{Lyyli}) \vee (x = \text{Hemmo} \wedge y = \text{Hemmo})),$$
$$\forall x \forall y(\text{tuntee2}(x,y) \leftrightarrow \text{tuntee1}(x,y) \vee \text{tuntee2}(y,x))\ \}.$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Yllättäen täydennetyistä määritelmistä ei seuraa loogisesti $\neg$tuntee2(Lyyli, Hemmo) eikä $\neg$tuntee2(Hemmo, Lyyli).
- Lausejoukolla Σ' on seuraava ei-toivottu malli $\mathcal{A}$:
  Universumi $A = \{1, 2\}$,
  Lyyli$^{\mathcal{A}} = 1$, Hemmo$^{\mathcal{A}} = 2$,
  tuntee1$^{\mathcal{A}} = \{\langle 1,1\rangle, \langle 2,2\rangle\}$ ja
  tuntee2$^{\mathcal{A}} = \{\langle 1,1\rangle, \langle 1,2\rangle, \langle 2,1\rangle, \langle 2,2\rangle\}$.
- Kyseiselle struktuurille $\mathcal{A}$ pätee:

  $\mathcal{A} \models$ tuntee2(Lyyli, Hemmo) ja $\mathcal{A} \models$ tuntee2(Hemmo, Lyyli).

Huomio. Tentissä eikä myöskään 3. kotitehtävässä ei edellytetä täydellisten määritelmien kirjoittamista predikaateille (ellei tätä sitten erikseen jossain yksinkertaisessa tapauksessa pyydetä).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

7 Herbrandin teoreema

- Herbrand-universumit
- Herbrand-struktuurit ja -mallit
- Herbrandin teoreema
- Lause- ja predikaattilogiikan suhteesta

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

7.1 Herbrand-universumit

Määritelmä. Predikaattilogiikan kielen $\mathcal{L}$ *Herbrand-universumi* on niiden muuttujattomien termien t joukko, jotka ovat muodostettavissa kielen vakio- ja funktiosymboleista.

Esimerkki. Olkoon kielessä $\mathcal{L}$ ainoastaan yksi vakiosymboli c ja yksi funktiosymboli $f \in \mathcal{F}_2$.

Herbrand-universumiksi saadaan muuttujattomien termien joukko

$$H = \{c, f(c,c), f(f(c,c),c), f(c,f(c,c)), f(f(c,c),f(c,c)), \ldots\}.$$

Huomio. Jos kielessä $\mathcal{L}$ ei ole funktiosymboleita ja ainoastaan äärellinen määrä vakioita, Herbrand-universumi H jää tällöin äärelliseksi.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Herbrand-universumi voidaan määritellä myös annetusta lausejoukosta Σ lähtien.
- Jos lausejoukossa Σ ei esiinny yhtään vakiosymbolia, Herbrand-universumiin valitaan ainakin yksi vakiosymboli c (struktuurien määritelmän mukaan universumit ovat aina ei-tyhjiä).

Määritelmä. Lausejoukon Σ Herbrand-universumi H on niiden muuttujattomien termien t joukko, jotka ovat muodostettavissa lausejoukossa Σ esiintyvistä vakio- ja funktiosymboleista.

Esimerkki. Lausejoukon $\Sigma = \{\forall x P(x, f(x))\}$ Herbrand-universumi on

$$H = \{c, f(c), f(f(c)), \ldots\} = \{f^n(c) \mid n \geq 0\}.$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

7.2 Herbrand-struktuurit ja -mallit

Määritelmä. Kielen $\mathcal{L}$ *Herbrand-struktuuri* on struktuuri $\mathcal{H}$, jonka

1. universumina on kielen $\mathcal{L}$ Herbrand-universumi H,
2. jokaisen vakiosymbolin $c \in \mathcal{C}$ tulkintana $c^{\mathcal{H}}$ on c itse,
3. jokaisen funktiosymbolin $f \in \mathcal{F}_n$ tulkintana on funktio $f^{\mathcal{H}}$, joka kuvaa muuttujattomat termit $t_1, \ldots, t_n$ muuttujattomaksi termiksi $f(t_1, \ldots, t_n)$, ja
4. jokaisen predikaattisymbolin $P \in \mathcal{P}_n$ tulkintana on $P^{\mathcal{H}} \subseteq H^n$ (yhtäsuuruuspredikaatille "$=$" tulkinta $=^{\mathcal{A}}$ on $\{\langle t, t\rangle \mid t \in H\}$).

- Lauseen $\phi \in \mathcal{L}$ totuusarvo Herbrand-struktuurissa $\mathcal{H}$ lasketaan predikaattilogiikan totuusmääritelmän mukaisesti.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä. Kielen $\mathcal{L}$ Herbrand-struktuuri $\mathcal{H}$ on

1. lauseen $\phi \in \mathcal{L}$ Herbrand-malli $\iff \mathcal{H} \models \phi$, ja
2. lausejoukon $\Sigma \subseteq \mathcal{L}$ Herbrand-malli $\iff \mathcal{H} \models \sigma$ kaikille $\sigma \in \Sigma$.

Esimerkki. Tarkastellaan lausejoukkoa

$$\Sigma = \{P(a),\ \forall x(P(x) \to Q(x)),\ \forall x(Q(x) \to Q(f(x)) \wedge R(x, f(x)))\}.$$

- Lausejoukon Σ Herbrand-universumi on $H = \{f^n(a) \mid n \geq 0\}$.
- Muodostetaan Herbrand-struktuuri on $\mathcal{H}$, jonka universumina on H siten, että jokainen muuttujaton termi $t \in H$ tulkitaan $t^{\mathcal{H}} = t$, ja

$$P^{\mathcal{H}} = \{a\},\ Q^{\mathcal{H}} = H \text{ ja } R^{\mathcal{H}} = \{\langle f^n(a), f^{n+1}(a)\rangle \mid n \geq 0\}.$$

- Kyseinen struktuuri $\mathcal{H}$ on lausejoukon Σ Herbrand-malli.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä. Lausejoukon Σ (kielen $\mathcal{L}$) *Herbrand-kanta* B on niiden *atomisten lauseiden* joukko, jotka voidaan muodostaa lausejoukossa Σ esiintyvistä (kielen $\mathcal{L}$) predikaattisymboleista ja vastaavan Herbrand-universumin H muuttujattomista termeistä.

Esimerkki. Edellisen esimerkin tapauksessa Herbrand-kantana on $B = \{P(f^n(a)), Q(f^n(a)) \mid n \geq 0\} \cup \{R(f^n(a), f^m(a)) \mid n \geq 0, m \geq 0\}$.

- Tämä mahdollistaa Herbrand-struktuurien $\mathcal{H}$ määrittelemisen Herbrand-kannan B osajoukkoina: jokaiselle $P \in \mathcal{P}_n$ pätee

$$P(t_1, \ldots, t_n) \in \mathcal{H} \iff \langle t_1, \ldots, t_n\rangle \in P^{\mathcal{H}}.$$

- Herbrand-struktuurille $\mathcal{H}$ voidaan antaa myös literaaliesitys:

$$lit(\mathcal{H}) = \{\ P(a), Q(a), \neg R(a,a), \neg P(f(a)), Q(f(a)),\\ R(a, f(a)), \neg R(f(a), a), \neg R(f(a), f(a)), \ldots\}.$$

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

7.3 Herbrandin teoreema

- Rajoitutaan tarkastelemaan klausuulijoukkoja.
- Merkintä $C(x_1, \ldots, x_n)$ tarkoittaa muuttujat $x_1, \ldots, x_n$ sisältävää klausuulia $\{P_1(\vec{t_1}), \ldots, P_k(\vec{t_k}), \neg Q_1(\vec{s_1}), \ldots, \neg Q_l(\vec{s_l})\}$.
- Kausuuli $C(x_1, \ldots, x_n)$ vastaa universaalisti kvantifioitua lausetta $\forall x_1 \cdots \forall x_n \phi_C(x_1, \ldots, x_n)$, missä $\phi_C(x_1, \ldots, x_n)$ on klausuulin $C(x_1, \ldots, x_n)$ esitys literaalien disjunktiona.
- Klausuulijoukolle S voidaan määritellä Herbrand-struktuurit samaan tapaan kuin lausejoukoillekin.
- Klausuulijoukko S voidaan *instantioida* vastaavan Herbrand-universumin H_S suhteen seuraavasti.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä. Klausuulijoukon S *Herbrand-instanssien joukko* S' koostuu muuttujattomista klausuuleista $C(t_1, \ldots, t_n)$, missä $C(x_1, \ldots, x_n) \in S$ ja muuttujattomat termit $t_1 \in H_S, \ldots, t_n \in H_S$.

- Mikäli S ja H_S ovat äärelliset, myös S' on äärellinen.
- Joukko S' voidaan tulkita lauselogiikan klausuulijoukoksi (atomisina lauseina Herbrand-kannan B atomiset lauseet).

Esimerkki. Tarkastellaan klausuulijoukkoa
$S = \{\{P(a)\}, \{\neg P(x), P(f(x))\}\}$.

1. Herbrand-universumi $H_S = \{a, f(a), f(f(a)), \ldots\}$.
2. Herbrand-instanssien joukko
   $S' = \{\{P(a)\}, \{\neg P(a), P(f(a))\}, \{\neg P(f(a)), P(f(f(a)))\}, \ldots\}$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Väite. (Hebrandin teoreema). Olkoon S joukko klausuuleita ja S' sen Herbrand-instanssien joukko. Tällöin

1. S on toteutumaton $\iff$ S' on toteutumaton, ja
2. S on toteutumaton $\iff$
   $\exists$ joukon S' äärellinen osajoukko S'', joka on toteutumaton.

Huomioita. Predikaattilogiikan tapauksessa voidaan täten rajoittua *syntaktisiin malleihin* (Herbrand-malleihin) mielivaltaisten mallien sijaan.

Herbrandin teoreema johtaa myös naiiviin proseduuriin klausuulijoukon S toteutuvuusongelman ratkaisemiseksi:

(i) tuotetaan äärellinen Herbrand-instanssien osajoukko S'' ja

(ii) testataan, onko S'' on toteutumaton. Jos on, lopetetaan ja todetaan S toteutumattomaksi. Muutoin jatketaan kohdasta (i).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

7.4 Lauselogiikan ja predikaattilogiikan suhteesta

- Lauselogiikka on osa predikaattilogiikkaa:
  - Kaikki lauselogiikan konnektiivit ovat käytettävissä predikaattilogiikassa.
  - 0-paikkaiset predikaatit vastaavat atomisia lauseita.
- Lauselogiikan päätelmät ja loogiset ongelmat voidaan suorittaa/ratkoa sellaisenaan predikaattilogiikan puitteissa.
- Herbrandin teoreeman nojalla predikaattilogiikan päättely voidaan palauttaa lauselogiikan päättelyksi.
- Lauselogiikan ja predikaattilogiikan *ilmaisuvoimassa* (eli kyvyssä esittää tietämystä) on kuitenkin huomattava ero.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Ilmaisuvoimaeron ilmentyminen:
  - Äärellistä predikaattilogiikan lausejoukkoa saattaa vastata ääretön lauselogiikan lausejoukko.
  - Lauselogiikan ratkeavuus vs. predikaattilogiikan puoliratkeavuus.
- Rajoittamalla syntaksia sopivasti saadaan predikaattilogiikallekin ratkeavia (ja ilmaisuvoimaltaan heikompia) osajoukkoja.
  - Esim. jos S on äärellinen ja siinä ei esiinny funktiosymboleja, sen Herbrand-instanssien joukko S' jää äärelliseksi.
  - Tällöin S':n toteutuvuus on selvitettävissä äärellisessä ajassa.

Esimerkki. Klausuulijoukon $S = \{\{P(a)\}, \{\neg P(x), P(b)\}\}$
Herbrand-universumi $H = \{a, b\}$ ja Herbrand-instanssien joukko $S' = \{\{P(a)\}, \{\neg P(a), P(b)\}, \{\neg P(b), P(b)\}\}$, joka voidaan nähdä lauselogiikan klausuulijoukkona $S' = \{\{P\}, \{\neg P, Q\}, \{\neg Q, Q\}\}$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

8 Unifikaatio

- Substituutiot
- Yleisimmät unifioijat
- Unifikaatioalgoritmi

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

8.1 Substituutiot

Määritelmä. *Substituutio* (tai *korvaus*) θ on äärellinen joukko

$$\{x_1/t_1, x_2/t_2, \ldots, x_n/t_n\},$$

missä x_i:t ovat muuttujia ja t_i:t korvaavia termejä siten, että

1. korvattavat muuttujat $x_1, \ldots, x_n$ ovat toisistaan eriävät ja
2. mikään korvaava termi t_i ei ole muuttuja x_i itse eli $t_i \neq x_i$.

Lisäksi erotetaan seuraavat erikoistapaukset:

- Jos korvaavat termit t_i ovat muuttujattomia, θ on *muuttujaton*.
- Jos korvaavat termit t_i ovat muuttujia, θ on *nimeämissubstituutio*.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Esimerkkeinä todettakoon

- tyhjä substituutio $\epsilon = \{\}$,
- substituutio $\theta_1 = \{x/y, y/a, z/f(w)\}$,
- muuttujaton substituutio $\theta_2 = \{x/a, y/g(c, c)\}$ ja
- nimeämissubstituutio $\theta_3 = \{x/y, y/z, z/x\}$.

Määritelmä. Olkoon E jokin *lauseke* (eli termi, atomikaava, literaali, klausuuli tms.) ja $\theta = \{x_1/t_1, \ldots, x_n/t_n\}$ substituutio.

Lauseke $E\theta$ on muutoin rakenteeltaan kuten E, paitsi että jokainen muuttujan x_i esiintymä lausekkeessa E on korvattu termillä t_i.

Jos lausekkeessa $E\theta$ ei esiinny muuttujia, kutsutaan lauseketta $E\theta$ lausekkeen E *muuttujattomaksi instanssiksi*.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Olkoon lauseke $E = P(x, y, f(z), v, w)$ ja substituutio $\theta = \{x/y, y/x, z/x, v/f(z), w/g(f(y), c)\}$.

Tällöin $E\theta$ on $P(y, x, f(x), f(z), g(f(y), c))$.

Määritelmä. Olkoot $\theta = \{x_1/t_1, \ldots, x_n/t_n\}$ ja $\lambda = \{y_1/u_1, \ldots, y_m/u_m\}$ kaksi substituutiota.

Substituutioiden θ ja λ *kompositio* $\theta\lambda$ määritellään joukkona

$$\{x_i/t_i\lambda \mid i \in \{1, \ldots, n\} \text{ ja } x_i \neq t_i\lambda\} \cup$$
$$\{y_i/u_i \mid i \in \{1, \ldots, m\} \text{ ja } y_i \notin \{x_1, \ldots, x_n\}\}.$$

Huomio. Komposition kokonaisvaikutus $E(\theta\lambda) = (E\theta)\lambda$.

Esimerkki. Substituutioiden $\theta = \{x/f(y), y/z\}$ ja $\lambda = \{x/a, y/b, z/y\}$. kompositio on $\{x/f(b), z/y\}$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

8.2 Yleisimmät unifioijat

Määritelmä. Olkoon $S = \{E_1, \ldots, E_n\}$ joukko lausekkeita. Substituutio θ on lausekejoukon S *unifioija*, jos $E_1\theta = E_2\theta = \cdots = E_n\theta$.

Lausekejoukko S on *unifioituva*, mikäli sillä on ainakin yksi unifioija.

Esimerkki. Tarkastellaan seuraavien joukkojen unifioituvuutta.

Joukko S:	Unifioija θ:
$\{P(x, f(a)), P(y, z)\}$	$\{y/x, z/f(a)\}$ tai $\{x/y, z/f(a)\}$
$\{P(x, f(x)), P(f(a), y)\}$	$\{x/f(a), y/f(f(a))\}$
$\{P(a), P(f(x))\}$	ei unifioijaa
$\{P(x), P(f(x))\}$	ei unifioijaa (termit aina äärellisiä)

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Määritelmä. Olkoon σ lausekejoukon $S = \{E_1, \ldots, E_n\}$ unifioija.

Substituutiota σ kutsutaan lausekejoukon S *yleisimmäksi unifioijaksi*, mikäli jokainen S:n unifioija $\theta = \sigma\lambda$ jollekin substituutiolle λ.

- Vastaava käsite englanniksi on *most general unifier* (MGU).

Esimerkki. Joukon $L = \{P(x, f(y)), P(a, z)\}$ unifioijia ovat mm. $\theta = \{x/a, z/f(b), y/b\}$ ja $\sigma = \{x/a, z/f(y)\}$.

Näistä σ on L:n yleisin unifioija, koska esim. $\theta = \sigma\{y/b\}$.

- Yleisimmät unifioijat ovat yksikäsitteisiä seuraavaan tapaan:

Väite. Olkoot θ ja σ joukon S yleisimpiä unifioijia. Tällöin on olemassa nimeämissubstituutio λ siten että $S\theta\lambda = S\sigma$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

8.3 Unifikaatioalgoritmi

- Tavoitteena laskea atomikaavojen joukolle $S \neq \emptyset$ yleisin unifioija σ.

Määritelmä. Olkoon S ei-tyhjä joukko johonkin predikaattisymboliin P perustuvia atomikaavoja $\{P(\vec{t_1}), \ldots, P(\vec{t_n})\}$.

1. Joukon S *erokohta* on järjestyksessä ensimmäinen kohta (vasemmalta oikealle siirryttäessä), jossa joukon S atomikaavojen merkkijonoesityksissä on jokin eroavaisuus.
2. Joukon S *erojoukkoon* $D(S)$ kuuluvat atomikaavojen $P(\vec{t_1}), \ldots, P(\vec{t_n})$ erokohdasta alkavat termit $u_1, \ldots, u_n$.

Esimerkki. Joukon $S_1 = \{P(x, a), P(x, y)\}$ erojoukko $D(S_1) = \{a, y\}$. Joukon $S_2 = \{Q(g(x, y), y), Q(g(x, f(z)), x), Q(g(x, x), f(a))\}$ erojoukko $D(S_2) = \{y, f(z), x\}$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Unifikaatioalgoritmi: syötteenä ei-tyhjä atomikaavojen joukko S:

1. Jos joukon S atomikaavojen predikaattisymbolit eivät ole samat, totea, ettei S unifioidu ja lopeta algoritmin suoritus.
2. Aseta $k := 0$, $S_k := S$ ja $\sigma_k := \epsilon$.
3. Jos S_k on yksialkioinen (ja siten jo unifioitunut) joukko, totea S unifioituvaksi ja lopeta algoritmin suoritus.
4. Laske joukon S_k erojoukko $D(S_k)$.
5. Jos $D(S_k)$:ssa on muuttuja v_k ja termi t_k siten, että v_k ei esiinny t_k:ssa, jatka algoritmin suoritusta kohdasta 7.
6. Muutoin totea, ettei S ole unifioituva, ja lopeta algoritmin suoritus.
7. Aseta $\sigma_{k+1} := \{v_k/t_k\}$ ja laske $S_{k+1} := S_k\{v_k/t_k\}$.
8. Aseta $k := k + 1$ ja jatka algoritmin suorittamista kohdasta 3.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Väite. Olkoon S äärellinen ei-tyhjä joukko atomikaavoja.

- Jos S on unifioituva, niin unifikaatioalgoritmin suoritus päättyy askeleen 3 kohdalla ja substituutioiden $\sigma_0, \sigma_1, \ldots, \sigma_k$ kompositio $\sigma = \sigma_0\sigma_1 \ldots \sigma_k$ on joukon S yleisin unifioija
- Jos S ei ole unifioituva, niin unifikaatioalgoritimin laskenta päättyy askeleessa 1 tai askeleessa 6.

Esimerkki. Lasketaan unifikaatioalgoritmilla joukon $S = \{P(x, f(x)), P(g(a), z)\}$ yleisin unifioija:

1. Predikaattisymbolit ovat samat, jatketaan.
2. $k = 0, S_0 = \{P(x, f(x)), P(g(a), z)\}, \sigma_0 = \epsilon$.
3. S_0 ei ole yksialkoinen, jatketaan.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

4. $D(S_0) = \{x, g(a)\}$.
5. Valitaan muuttuja $v_0 = x$ ja termi $t_0 = g(a)$.
7. $\sigma_1 = \{x/g(a)\}, S_1 = \{P(g(a), f(g(a))), P(g(a), z)\}$.
8. $k = 1$.
3. S_1 ei ole yksialkioinen, jatketaan.
4. $D(S_1) = \{f(g(a)), z\}$.
5. Valitaan muuttuja $v_1 = z$ ja termi $t_1 = f(g(a))$.
7. $\sigma_2 = \{z/f(g(a))\}$, $S_2 = \{P(g(a), f(g(a)))\}$.
8. $k = 2$.
3. S_2 on yksialkioinen, joten S on unifioituva.

Yleisin unifioija on $\sigma = \sigma_0\sigma_1\sigma_2 = \epsilon\{x/g(a)\}\{z/f(g(a))\} = \{x/g(a), z/f(g(a))\}$ ja $S\sigma = \{P(g(a), f(g(a)))\} = S_2$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

9 Resoluutiosääntö ja -todistukset

- Resoluutiosääntö predikaattilogiikan tapauksessa
- Resoluutiotodistukset
- Ohjeita resoluutiotodistusten kirjoittamiseen
- Resoluutiostrategioita

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

9.1 Resoluutiosääntö predikaattilogiikan tapauksessa

Määritelmä. Olkoot

$$C_1 = C_1' \sqcup \{P(\vec{t_1}), \ldots, P(\vec{t_n})\} \text{ ja } C_2 = C_2' \sqcup \{\neg P(\vec{u_1}), \ldots, \neg P(\vec{u_m})\}$$

kaksi klausuulia,

1. joissa *ei esiinny yhteisiä* muuttujia ja
2. joissa esiintyvien atomikaavojen joukko
$$\{P(\vec{t_1}), \ldots, P(\vec{t_n}), P(\vec{u_1}), \ldots, P(\vec{u_m})\}$$
on unifioituva (yleisimpänä unifioijana σ).

Klausuulien C_1 ja C_2 *yhdistelmä* on klausuuli $C_1'\sigma \cup C_2'\sigma$.

Huomio. Yllä käytetty merkintä $A \sqcup B$ tarkoittaa keskenään alkiovieraiden ($A \cap B = \emptyset$) joukkojen A ja B unionia $A \cup B$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Tarkastellaan seuraavia klausuuleja:

$$C_1 = \{Q(x), \neg R(y), P(x,y), P(f(z), f(z))\} \text{ ja}$$
$$C_2 = \{\neg N(u), \neg R(w), \neg P(f(a), f(a)), \neg P(f(w), f(w))\}.$$

Klausuuleissa ei esiinny yhteisiä muuttujia ja joukon

$$\{P(x,y), P(f(z), f(z)), P(f(a), f(a)), P(f(w), f(w))\}$$

yleisin unifioija on $\sigma = \{x/f(a), y/f(a), z/a, w/a\}$. Klausuulien yhdistelmäksi saadaan $\{Q(f(a)), \neg R(f(a)), \neg N(u), \neg R(a)\}$.

Esimerkki. (Faktorointi) Klausuleilla voi olla useita eri yhdistelmiä. Klausulien $\{P(x_1), P(y_1)\}$ ja $\{\neg P(x_2), \neg P(y_2)\}$ yhdistelmiä ovat mm.

- $\{P(x_1), \neg P(x_2)\}$ (joukolle $\{P(y_1), P(y_2)\}$ MGU $\sigma = \{y_2/y_1\}$) ja
- tyhjä klausuuli $\square$ (joukolle $\{P(x_1), P(y_1), P(x_2), P(y_2)\}$ MGU $\sigma = \{y_1/x_1, x_2/x_1, y_2/x_1\}$).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

9.2 Resoluutiotodistukset

- Lähtökohtana on joukko klausuuleita S, jonka klausuuleista johdetaan uusia klausuuleita resoluutiosäännöllä.
- *Johtojen* ja *hylkäyksen* määritelmät säilyvät ennallaan, mutta resoluutioaskeleiden tulee täyttää resoluutiosäännön vaatimukset.
- Tarvittaessa klausuulien muuttujat tulee nimetä uudelleen.
- Resoluutio on myös predikaattilogiikan tapauksessa virheetön ja täydellinen menettely klausuulijoukon toteutuvuuden tutkimiseen.

Väite. Klausuulijoukolle S löytyy hylkäys (eli klausuulijoukosta S on johto $C_1, \ldots, C_n$ tyhjälle klausuulille $C_n = \square$) $\iff$ S on toteutumaton.

Todistus. Sivuutetaan.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Osoitetaan predikaattilogiikan lausejoukko

$$\Sigma = \{\forall x \exists y(P(x) \wedge P(y)),\ \forall x \forall y(P(x) \rightarrow \neg P(y))\}$$

toteutumattomaksi. Haetaan lauseille ensin klausuuliesitykset:

- $\forall x \exists y(P(x) \wedge P(y)) \rightsquigarrow \forall x(P(x) \wedge P(f(x))) \rightsquigarrow$ $S_1 = \{\{P(x)\}, \{P(f(x))\}\}$.
- $\forall x \forall y(P(x) \rightarrow \neg P(y))\} \rightsquigarrow \forall x \forall y(\neg P(x) \vee \neg P(y))\} \rightsquigarrow$ $S_2 = \{\{\neg P(x), \neg P(y)\}\}$.

Hylkäys:		
	1. $\{P(x)\}$	S_1
	2. $\{\neg P(z), \neg P(y)\}$	$S_2\{x/z\}$
	3. $\square$	1,2,MGU $\{x/y, z/y\}$

$\Longrightarrow$ $S_1 \cup S_2$ on toteutumaton $\Longrightarrow$ Σ on toteutumaton.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Muiden loogisten ongelmien ratkominen

- Resoluutiolla voidaan selvittää lauseiden pätevyyttä ja loogista ekvivalenssia sekä tutkia lausejoukon loogisia seuraavuuksia.
- Koska Skolemointi ei säilytä loogista ekvivalenssia vaan toteutuvuuden, nämä tulee muuntaa toteutuvuusongelmiksi.

Väite. Olkoon ϕ ja ψ lauseita ja Σ lausejoukko.

1. Pätevyys: $\models \phi \iff \mathrm{KM}(\{\neg\phi\})$:lle löytyy hylkäys.
2. Ekvivalenssi: $\phi \equiv \psi \iff \mathrm{KM}(\{\neg(\phi \leftrightarrow \psi)\})$:lle löytyy hylkäys.
3. Looginen seuraavuus: $\Sigma \models \phi$ $\iff$ klausuulijoukolle $\mathrm{KM}(\Sigma \cup \{\neg\phi\})$ löytyy hylkäys.

Yllä $\mathrm{KM}(\Gamma)$ tarkoitaa lausejoukon Γ klausuulimuotoa, mikä saadaan ottamalla yksittäisten lauseiden $\gamma \in \Gamma$ klausuulimuotojen unioni.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Osoitetaan resoluutiolla, että $\models \forall x P(x) \to \exists x P(x)$.

- Haetaan lauseen negaatiolle klausuulimuoto:

$$\begin{aligned} \neg(\forall x P(x) \to \exists x P(x)) &\rightsquigarrow \forall x P(x) \wedge \neg\exists x P(x) \\ &\rightsquigarrow \forall x P(x) \wedge \forall x \neg P(x) \\ &\rightsquigarrow \forall x \forall y (P(x) \wedge \neg P(y)). \\ &\rightsquigarrow S = \{\{P(x)\}, \{\neg P(y)\}\}. \end{aligned}$$

- Klausuuleista $\{P(x)\}$ ja $\{\neg P(y)\}$ saadaan tyhjä klausuuli $\square$ (MGU $\{x/y\}$) yhdellä resoluutioaskelella.
- Täten klausuulijoukolle S on hylkäys
  $\Longrightarrow S$ on toteutumaton
  $\Longrightarrow \neg(\forall x P(x) \to \exists x P(x))$ on toteutumaton
  $\Longrightarrow \forall x P(x) \to \exists x P(x)$ on pätevä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Osoitetaan lause $\exists x(E(x) \wedge K(x))$ lausejoukon

$$\Sigma = \{\forall x(I(x) \to E(x)), \exists x(I(x) \wedge K(x))\}$$

loogiseksi seuraukseksi. Haetaan tarvittavat klausuulimuodot:

$$\begin{aligned} \forall x(I(x) \to E(x)) &\rightsquigarrow \forall x(\neg I(x) \vee E(x)) \\ &\rightsquigarrow S_1 = \{\{\neg I(x), E(x)\}\}. \\ \exists x(I(x) \wedge K(x)) &\rightsquigarrow I(c) \wedge K(c) \\ &\rightsquigarrow S_2 = \{\{I(c)\}, \{K(c)\}\} \\ \neg\exists x(E(x) \wedge K(x)) &\rightsquigarrow \forall x \neg(E(x) \wedge K(x)) \\ &\rightsquigarrow \forall x(\neg E(x) \vee \neg K(x)) \\ &\rightsquigarrow S_3 = \{\{\neg E(x), \neg K(x)\}\} \end{aligned}$$

Kokonaisuutena saadaan siis klausuulijoukko $S = S_1 \cup S_2 \cup S_3$
$= \{\{\neg I(x), E(x)\}, \{I(c)\}, \{K(c)\}, \{\neg E(x), \neg K(x)\}\}$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Hylkäys löydetään esimerkiksi seuraavasti:

1.	$\{\neg I(x), E(x)\}$	S
2.	$\{I(c)\}$	S
3.	$\{K(c)\}$	S
4.	$\{\neg E(y), \neg K(y)\}$	$S\{x/y\}$
5.	$\{\neg I(y), \neg K(y)\}$	1,4,MGU $\{x/y\}$
6.	$\{\neg K(c)\}$	2,5,MGU $\{y/c\}$
7.	$\square$	3,6,MGU ϵ

- Yleisimpien unifioijien kompositio $\{x/y\}\{y/c\}\epsilon = \{x/c, y/c\}$.
- Täten S on toteutumaton
  $\Longrightarrow \Sigma \cup \{\neg\exists x(E(x) \wedge K(x))\}$ on toteutumaton
  $\Longrightarrow \exists x(E(x) \wedge K(x))$ on joukon Σ looginen seuraus.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

9.3 Ohjeita resoluutiotodistusten kirjoittamiseen

- Muuttujien uudelleennimeäminen on hyvä suorittaa systemaattisesti (esimerkiksi alaindeksien avulla).
- Yksittäistä klausuulijoukon klausuulia saatetaan tarvita useita kertoja resoluutiotodistuksessa (jolloin muuttujien uudelleennimeäminen on välttämätöntä).
- Kirjoita yleisimmät unifioijat (MGU:t) näkyviin.
- Ellet kirjoita todistusta binääripuun muotoon, numeroi klausuulit ja ilmoita, mistä klausuuleista mikin klausuuli on johdettu.
- Laske yleisimpien unifioijien kompositio.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Esitetään listat vakion e (tyhjä lista) ja kaksipaikkaisen funktiosymbolin c avulla (näin lista $[1,2]$ saa esityksen $c(1,c(2,e))$).

Määritellään seuraavat listoja koskevat predikaatit

- $K(x,y) =$
  "listan x alkioina ovat listan y alkiot käänteisessä järjestyksessä":
  1. $K(e,e)$ ja
  2. $\forall x\forall y\forall z\forall v(K(x,y) \wedge L(y,v,z) \rightarrow K(c(v,x),z))$.
- $L(y,v,z) =$ "lista z on lista y, jonka perään on liitetty alkio v":
  3. $\forall x L(e,x,c(x,e))$ ja
  4. $\forall y\forall v\forall z\forall x(L(y,v,z) \rightarrow L(c(x,y),v,c(x,z)))$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

- Johdetaan lauseille klausuulimuodot:

(1) $\rightsquigarrow \{\{K(e,e)\}\}$

(2) $\rightsquigarrow \forall x\forall y\forall z\forall v(\neg K(x,y) \vee \neg L(y,v,z) \vee K(c(v,x),z))$
$\rightsquigarrow \{\{\neg K(x,y), \neg L(y,v,z), K(c(v,x),z)\}\}$

(3) $\rightsquigarrow \{\{L(e,x,c(x,e))\}\}$

(4) $\rightsquigarrow \forall y\forall v\forall z\forall x(\neg L(y,v,z) \vee L(c(x,y),v,c(x,z))$
$\rightsquigarrow \{\{\neg L(y,v,z), L(c(x,y),v,c(x,z))\}\}$

(5) Todistettavan lauseen negaatio $\neg\exists x K(c(1,c(2,e)),x)$
$\rightsquigarrow \forall x\neg K(c(1,c(2,e)),x) \rightsquigarrow \{\{\neg K(c(1,c(2,e)),x)\}\}$

- Haluamme siis selvitää, mikä on lista $[1,2]$ käännettynä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Resoluutiotodistus:

1. $\{\neg K(c(1,c(2,e)),x_0)\}$ P5
2. $\{\neg K(x_1,y_1), \neg L(y_1,v_1,z_1), K(c(v_1,x_1),z_1)\}$ P2
3. $\{\neg K(c(2,e),y_1), \neg L(y_1,1,z_1)\}$ 1,2,MGU $\{v_1/1, x_1/c(2,e), x_0/z_1\}$
4. $\{\neg K(x_2,y_2), \neg L(y_2,v_2,z_2), K(c(v_2,x_2),z_2)\}$ P2
5. $\{\neg K(e,y_2), \neg L(y_2,2,y_1), \neg L(y_1,1,z_1)\}$
   3,4,MGU $\{v_2/2, x_2/e, z_2/y_1\}$
6. $\{K(e,e)\}$ P1
7. $\{\neg L(e,2,y_1), \neg L(y_1,1,z_1)\}$ 5,6,MGU $\{y_2/e\}$
8. $\{L(e,x_3,c(x_3,e))\}$ P3

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

9. $\{\neg L(c(2,e),1,z_1)\}$ 7,8,MGU $\{x_3/2, y_1/c(2,e)\}$
10. $\{\neg L(y_4,v_4,z_4), L(c(x_4,y_4),v_4,c(x_4,z_4))\}$ P4
11. $\{\neg L(e,1,z_4)\}$ 9,10,MGU $\{x_4/2, y_4/e, v_4/1, z_1/c(2,z_4)\}$
12. $\{L(e,x_5,c(x_5,e)))\}$ P3
13. $\square$ 11,12,MGU $\{x_5/1, z_4/c(1,e)\}$

- Unifioijien kompositio: $\{\ v_1/1,\ x_1/c(2,e),\ x_0/c(2,c(1,e)),$
  $v_2/2,\ x_2/e,\ z_2/c(2,e),$
  $y_2/e,$
  $x_3/2,\ y_1/c(2,e)$
  $x_4/2,\ y_4/e,\ v_4/1,\ z_1/c(2,c(1,e))$
  $x_5/1,\ z_4/c(1,e)\ \}.$
- Rajaus kyselyyn: $\{x_0/c(2,c(1,e))\}$ (ns. *vastaussubstituutio*).

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

9.4 Resoluutiostrategioita

- Tavoitteena äärellisen klausuulijoukon S refutointi.
- *Kerrossaturaatiostrategiassa* resoluutioaskeleita suoritetaan kerroksittain seuraavasti.
  1. Joukko S muodostaa kerroksen S_0.
  2. Kerros S_i $(i > 0)$ muodostuu klausuuleista C, jotka saadaan resoluutiosäännöllä joistain klausuuleista $C_1 \in S_{i-1}$ ja $C_2 \in S_0 \cup \cdots \cup S_{i-1}$.

Esimerkki. Olkoon alimpana kerroksena
$S_0 = S = \{\{\neg I(x), E(x)\}, \{I(c)\}, \{K(c)\}, \{\neg E(y), \neg K(y)\}\}$.

Tällöin $S_1 = \{\{E(c)\}, \{\neg E(c)\}, \{\neg I(x), \neg K(x)\}\}$
$S_2 = \{\square, \{\neg I(c)\}, \{\neg K(c)\}\}$.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Tukijoukkostrategia

Määritelmä. Klausuulijoukon S osajoukko T on *tukijoukko* (engl. *set of support*), jos $S - T$ on toteutuva.

- *Tukijoukkostrategiassa* ei milloinkaan suoriteta resoluutiota joukon $S - T$ klausuuleille keskenään.
- Tutkittaessa loogista seuraavuutta $\Sigma \models \phi$ lausejoukko Σ (olettamukset) on tyypillisesti toteutuva. Tällöin voidaan ajatella:
  1. T muodostuu lauseesta $\neg\phi$ saatavien klausuulien joukosta ja
  2. joukkoon $S - T$ kuuluvat lausejoukosta Σ saatavat klausuulit.
- Ristiriita aiheutuu siis konkreettisesti tukijoukon T klausuuleista, mikäli $\Sigma \models \phi$ (eli $\Sigma \cup \{\neg\phi\}$ on toteutumaton).

Väite. Tukijoukkostrategiaan perustuva resoluutio on täydellistä.

© 2002 TKK / Tietojenkäsittelyteorian laboratorio

Esimerkki. Tutkitaan seuraako $\exists y(E(y) \wedge K(y))$ loogisesti lausejoukosta $\Sigma = \{\forall x(I(x) \rightarrow E(x)),\ \exists x(I(x) \wedge K(x))\}$.

Lausejoukosta Σ saadaan klausuulijoukko
$S - T = \{\{\neg I(x), E(x)\}, \{I(c)\}, \{K(c)\}\}$.

Lauseen negaatiosta saadaan tukijoukko $T = \{\{\neg E(y), \neg K(y)\}\}$.

1. $\{\neg I(x), E(x)\}$	$S - T$
2. $\{I(c)\}$	$S - T$
3. $\{K(c)\}$	$S - T$
4. $\{\neg E(y), \neg K(y)\}$	T
5. $\{\neg E(c)\}$	3, 4, MGU $\{y/c\}$
6. $\{\neg I(c)\}$	1, 5, MGU $\{x/c\}$
7. $\square$	2, 6, MGU ϵ

© 2002 TKK / Tietojenkäsittelyteorian laboratorio