



## PREDIKAATTOLOGIikka

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



### 1 Predikaattilogiikan kieli

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



## 1.1 Motivaatio

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

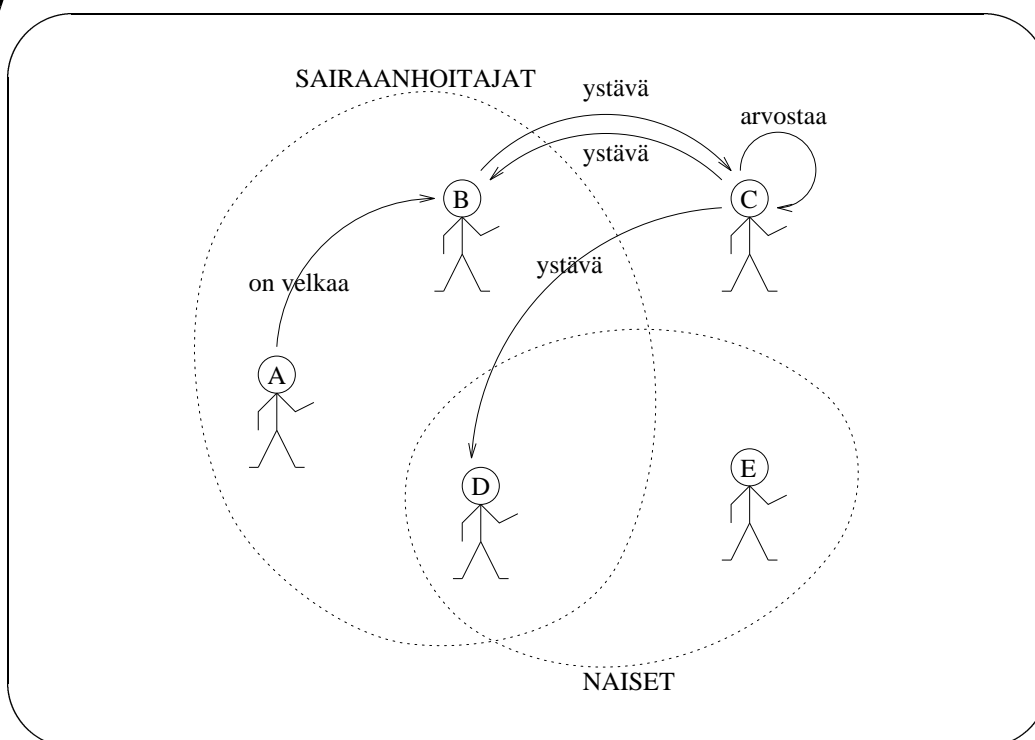
### Esimerkki.

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

$C_d = \text{"c on isompi kuin d"}$ ,  $D_e = \text{"d on isompi kuin e"}$ , ...

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



## 1.2 Predikaattilogiikan aakkosto

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

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



### 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  $\mathcal{P}_n = \{P \in \mathcal{P} \mid P\text{:n paikkaluku on } n\}$ .
- Yhtäsuuruspredikaatti  $= \in \mathcal{P}$  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.



### 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  $t_1, \dots, t_n$  ovat termejä, niin myös  $f(t_1, \dots, t_n)$  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*).



#### 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  $t_1, \dots, t_n$  ovat termejä, niin  $P(t_1, \dots, t_n)$  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).

**Määritelmä.**

1. Jokainen atomikaava  $\phi$  on *kaava*.
2. Jos  $\phi$  ja  $\psi$  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:  $P(c)$ ,  $(\forall x(P(x) \rightarrow Q(x)))$ ,  $(\forall x(P(x) \vee (\exists yQ(x, y))))$ ,  $(\exists x(\forall y(\forall zP(x, y, z))))$ .

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

**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

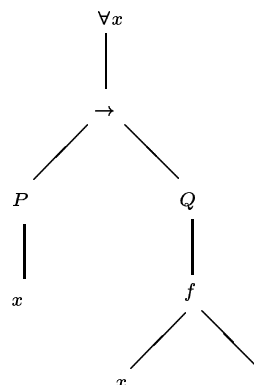
$$\exists x\forall y(\exists z(P(x, z) \wedge P(z, y)) \rightarrow Q(x) \vee Q(y) \vee R(x, y)).$$



## Kaavojen jäsenyspuut

Predikaattilogiikan kaavoilla on yksikäsitteinen jäsenyspuu.

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



Kyseinen kaava on muodoltaan *universaalisti kvantifioitu kaava*.

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



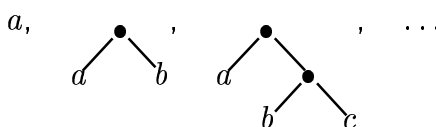
## 1.4 Lauseiden muodostaminen

1. Tunnistetaan kuvattavaan järjestelmään liittyvät objektit:
  - otetaan käyttöön vakiosymboli jokaiselle objektille, johon on tarve viitata erikseen, ja
  - mikäli objektien välillä on funktionaalisia suhteita, otetaan käyttöön vastaavat funktiosymbolit.
2. Tutkitaan millaisia suhteita objektien välillä on ja otetaan käyttöön näitä vastaavat predikaattisymbolit.
3. Kuvataan suhteiden väliset riippuvuudet predikaattilogiikan lausein.



**Esimerkki.** Funktiosymbolit tarjoavat tavan esittää induktiivisia tietorakenteita termien avulla.

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



**Esimerkki.**

Olkoon  $t = \text{“tuoli”}$ ,  $h = \text{“hattu”}$ ,  $s = \text{“sateenvarjo”}$  vakioita ja  $P(x, y) = \text{“}x \text{ on } y\text{:n päällä”}$  kaksipaikkainen predikaatti. Tällöin:

$\neg P(s, t) = \text{“sateenvarjo ei ole tuolin päällä”}$ .

$\exists x P(x, h) =$

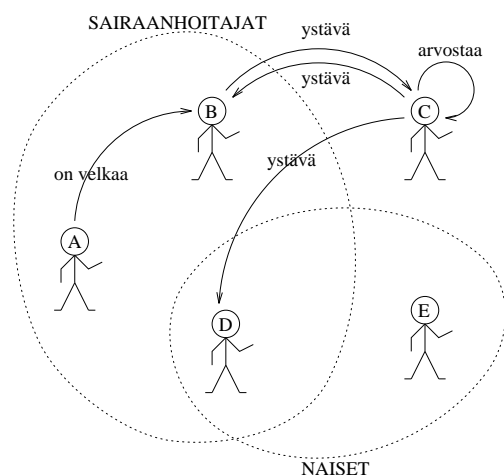
“on olemassa  $x$ , joka on hatun päällä”,  
eli “hatun päällä on jotakin”.

$\exists x \forall y \neg P(y, x) =$

“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”.

**Esimerkki.**

$$N(e) \rightarrow A(c, c)$$

$$\exists x \exists y (Y(x, y) \wedge Y(y, x))$$

$$\exists x \exists y (S(x) \wedge S(y) \wedge V(x, y))$$

$$\exists x A(x, x) \wedge \exists x (S(x) \wedge N(x))$$

$$\neg \forall x (S(x) \rightarrow N(x))$$

$$\forall x (Y(x, c) \rightarrow V(a, x))$$

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio

**1.5 Kvanttoreihin liittyviä määritelmiä**

Kaavan **alikaavat** määräytyvät seuraavasti:

- Atomisen kaavan  $\psi$  ainoa alikaava on  $\psi$  itse.
- Kaavan  $\exists x \psi$  ( $\forall x \psi$ ) alikaavat ovat  $\exists x \psi$  ( $\forall x \psi$ ) ja  $\psi$ :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  $\phi$ ,  $\exists x A(x, x)$ ,  $\exists x (S(x) \wedge N(x))$ ,  $A(x, x)$ ,  $S(x) \wedge N(x)$ ,  $S(x)$  ja  $N(x)$ .

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



**Vapaat ja sidotut muuttujat kaavassa**

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

**Esimerkki.**

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

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



**Määritelmä.** Muuttujan  $x$  esiintymä kaavassa  $\psi$  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*  $\phi$ :ssä, jos sillä on vapaa esiintymä  $\phi$ :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  $\phi$  on *lause*, jos siinä ei ole vapaita muuttujaesiintymiä.

© 2001 Teknillinen korkeakoulu, 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  $\phi$ , 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.



### 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  $\psi(x) = \exists xP(x) \wedge Q(x)$ .
  - Sijoittamalla  $c$  saadaan  $\psi(c) = \exists xP(x) \wedge Q(c)$ .
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  $\xi(f(z)) = \exists x(P(x) \wedge Q(f(z)))$ .



## 2 Predikaattilogiikan semantiikka

- Struktuurit
- Predikaattilogiikan totuusmääritelmä
- Semanttiset peruskäsitteet



### 2.1 Struktuurit

Predikaattilogiikassa totuusjaketut korvataan struktuureilla.

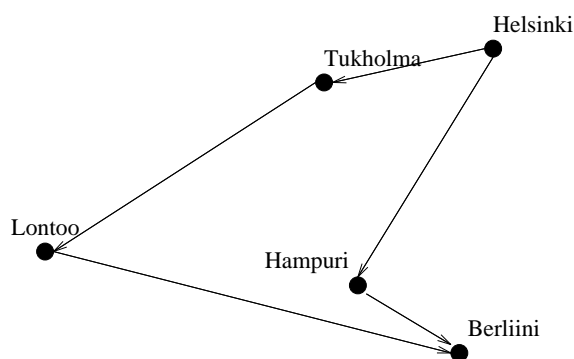
**Määritelmä.** *Strukturi* (rakenne)  $\mathcal{A}$  kielelle  $\mathcal{L}$  koostuu

- 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  $f^{\mathcal{A}}: A^n \rightarrow A$ .
  4. Predikaattisymbolin  $P \in \mathcal{P}_n$  tulkintana on relaatio  $P^{\mathcal{A}} \subseteq A^n$ .

**Huomioita.**

- Joukon  $A^n = \overbrace{A \times \cdots \times A}^{n \text{ kpl}}$  alkiot ovat monikkoja (tai jonoja)  $\langle a_1, \dots, a_n \rangle$ , missä alkiot  $a_1 \in A, \dots, a_n \in A$ .
- 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).

**Esimerkki.**

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



## 2.2 Predikaattilogiikan totuusmääritelmä

### Termien tulkinta strukturissa

**Määritelmä.** Olkoon  $\mathcal{A}$  strukturi 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  $t_1, \dots, t_n$  nimeävät universumin alkion  $t_1^{\mathcal{A}}, \dots, t_n^{\mathcal{A}}$  ja  $f \in \mathcal{F}_n$ , niin termi  $f(t_1, \dots, t_n)$  nimeää universumin alkion  $f^{\mathcal{A}}(t_1^{\mathcal{A}}, \dots, t_n^{\mathcal{A}})$ .

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



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

$$c^{\mathcal{A}} = 0,$$

$$f^{\mathcal{A}} = \text{seuraajafunktio, eli } f^{\mathcal{A}}(n) = n + 1, \text{ ja}$$

$$g^{\mathcal{A}} = \text{summafunktio, eli } g^{\mathcal{A}}(n, m) = n + m.$$

Tällöin  $c$  nimeää alkion 0,

$f(c)$  nimeää alkion 1,

$f^n(c) = \underbrace{f(f(\dots f(c)\dots))}_{n \text{ kpl}}$  nimeää alkion  $n$  ja

$g(f(c), f(f(c)))$  nimeää alkion 3.

**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.  $\mathcal{A} \models P(t_1, \dots, t_n) \iff \langle t_1, \dots, t_n \rangle^{\mathcal{A}}$  (eli jono  $\langle t_1^{\mathcal{A}}, \dots, t_n^{\mathcal{A}} \rangle$ ) kuuluu tulkintaan  $P^{\mathcal{A}}$  ( $t_1, \dots, t_n$  ovat mitä tahansa termejä ja  $P \in \mathcal{P}_n$ ).
3.  $\mathcal{A} \models \neg \alpha \iff \mathcal{A} \not\models \alpha$ .
4.  $\mathcal{A} \models \alpha \wedge \beta \iff \mathcal{A} \models \alpha$  ja  $\mathcal{A} \models \beta$ .
5.  $\mathcal{A} \models \alpha \vee \beta \iff \mathcal{A} \models \alpha$  tai  $\mathcal{A} \models \beta$ .

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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

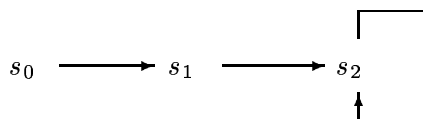
**Väite.** Kaikille kaavoille  $\phi \in \mathcal{L}$  pätee joko  $\mathcal{A} \models \phi$  tai  $\mathcal{A} \not\models \phi$ .

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

© 2001 Teknillinen korkeakoulu, 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  $\langle s_2, s_2 \rangle \in K^{\mathcal{A}} \implies \mathcal{A}[x \mapsto s_2] \models \exists y K(x, y).$



**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.

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



## 2.3 Semanttiset peruskäsitteet

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

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

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



**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  $\mathcal{A} \models \exists x \forall y P(x, y)$ .





**Määritelmä.** Kielen  $\mathcal{L}$  lause  $\alpha$  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:

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



**Esimerkki.**  $\models \forall x P(x) \vee \neg \forall x P(x)$

Olkoon  $\mathcal{A}$  mielivaltainen  $\mathcal{L}$ :n strukturi.

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

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

Olkoon  $\mathcal{A}$  strukturi siten, että  $\mathcal{A} \models \forall x P(x)$ . Tällöin

- $$\begin{aligned} \implies & \text{ kaikille } a \in A \text{ pätee } \mathcal{A}[x \mapsto a] \models P(x) \\ \implies & \text{ jollekin } a \in A \text{ pätee } \mathcal{A}[x \mapsto a] \models P(x) \\ & \text{(universumi } A \text{ on ei-tyhjä struktuurin määritelmän perusteella)} \\ \implies & \mathcal{A} \models \exists x P(x). \end{aligned}$$

**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.
- $\models \alpha \iff \emptyset \models \alpha$ .
- $\{\phi_1, \dots, \phi_n\} \models \phi \iff \models \phi_1 \wedge \dots \wedge \phi_n \rightarrow \phi$ .
- **Kompaktius:** jos  $\Sigma \models \phi$ , niin on olemassa äärellinen osajoukko  $\Sigma' \subseteq \Sigma$  siten, että  $\Sigma' \models \phi$ .
- $\Sigma \subseteq \text{Cn}(\Sigma)$ .
- **Monotonisuus:**  $\Sigma_1 \subseteq \Sigma_2 \implies \text{Cn}(\Sigma_1) \subseteq \text{Cn}(\Sigma_2)$ .
- $\text{Cn}(\text{Cn}(\Sigma)) = \text{Cn}(\Sigma)$ .

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio

**Esimerkki.**

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

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

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

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



### 3 Semanttiset taulut predikaattilogiikalle

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



#### 3.1 Taulusäännöt kvanttorien käsittelyyn

- Vakio  $c$  on *uusi*, mikäli se ei esiinny muualla ko. polulla.
- Muotoa  $T \exists x \varphi(x)$  (tai  $E \forall x \varphi(x)$ ) oleva solmu tulee hajottaa *kertaalleen* käyttäen jotain (hajoitushetkellä) *uutta vakiota*  $c$ .

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

#### Huomio.

- Uuden vakion käyttöönotto johtuu siitä, ettemme tiedä, millä universumin alkiolla on ominaisuus  $\varphi$  (tai ei ole ominaisuutta  $\varphi$ ).

© 2001 Teknillinen korkeakoulu, 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).

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

**Esimerkki.**

$$\begin{array}{l}
 1. T \exists x \forall y \exists z P(x, y, z) \\
 | \\
 2. T \forall y \exists z P(a, y, z) \quad 1. x/a \quad (a \text{ uusi vakio}) \\
 | \\
 3. T \exists z P(a, a, z) \quad 2. y/a \\
 | \\
 4. TP(a, a, b) \quad 3. z/b \quad (b \text{ uusi vakio}) \\
 | \\
 5. T \exists z P(a, b, z) \quad 2. y/b \\
 | \\
 6. TP(a, b, c) \quad 5. z/c \quad (c \text{ uusi vakio}) \\
 | \\
 \text{jne.}
 \end{array}$$



### 3.2 Semanttiset taulut predikaattilogiikalle

- Semanttisten taulujen määritelmä säilyy ennallaan.
- Ehtoja, millä polun solmu on *hajoitettu*, joudutaan täydentämään: Olkoon  $\tau$  semanttinen taulu ja  $P$  polku juurisolmusta lehtisolmuun  $\tau$ :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,  $T\forall xP(x)$  tulee hajoittaa käyttäen jotain uutta vakiosymbolia  $c$ .
- Muilta osin semanttisen taulun (ja sen polkujen) ristiriitaisuuden ja valmiuden määritelmät säilyvät ennallaan.

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



- Taulumenetelmää voidaan käyttää loogisten ongelmien ratkomiseen kuten lauselogiikan tapauksessa.

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

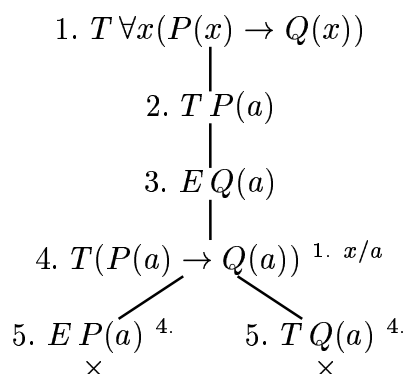
**Määritelmä.** Lause  $\phi$  on *johdettavissa* äärellisestä lausejoukosta  $\Sigma = \{\phi_1, \dots, \phi_n\}$  (merkitään  $\Sigma \vdash \phi$ ), joss juurialkiosta  $E(\phi_1 \wedge \dots \wedge \phi_n \rightarrow \phi)$  muodostettu valmis semanttinen taulu on ristiriitainen.

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



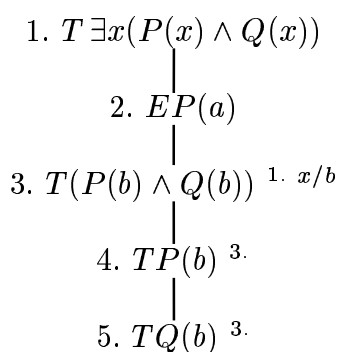
**Esimerkki.** Onko  $\{P(a), \forall x(P(x) \rightarrow Q(x))\} \vdash Q(a)$  ?



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



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



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



### 3.3 Ohjeita taulutodistusten laadintaan

- Lauseen rakenne määrää edelleen, mitä taulusääntöä tulee käyttää (jäsenyspuun 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.

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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

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

$$\begin{array}{c}
 1. \ T \forall xP(x) \\
 | \\
 2. \ E \exists xP(x) \\
 | \\
 3. \ TP(c)^1. \ x/c \\
 | \\
 4. \ EP(c)^2. \ x/c \\
 \times
 \end{array}$$

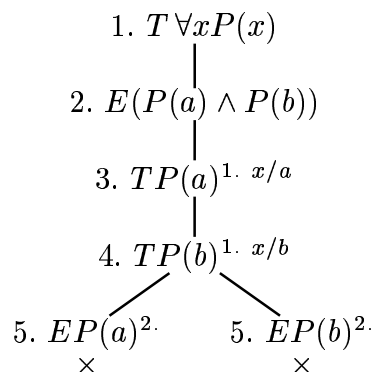
**Huomio.** Tämä on perusteltua, koska universumissa  $A$  on aina vähintään yksi alkio  $a \in A$ , joka voidaan nimetä (eli  $c^A = a$ ).

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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

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



© 2001 Teknillinen korkeakoulu, 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 kvanttorisäännöllä 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.

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio







### Kvanttorisekvenssien käsittely

Jatkossa sallimme seuraavien johdettujen taulusääntöjen käytön kvanttorisekvenssien käsittelyssä:

$T \forall x_1 \cdots \forall x_n \varphi(x_1, \dots, x_n)$ $\quad \quad \quad \downarrow$ $T \varphi(t_1, \dots, t_n)$	$E \exists x_1 \cdots \exists x_n \varphi(x_1, \dots, x_n)$ $\quad \quad \quad \downarrow$ $E \varphi(t_1, \dots, t_n)$
$T \exists x_1 \cdots \exists x_n \varphi(x_1, \dots, x_n)$ $\quad \quad \quad \downarrow$ $T \varphi(c_1, \dots, c_n)$	$E \forall x_1 \cdots \forall x_n \varphi(x_1, \dots, x_n)$ $\quad \quad \quad \downarrow$ $E \varphi(c_1, \dots, c_n)$

Yllä  $c_1, \dots, c_n$  ovat taulusääntöjen edellyttämiä *uusvia vakioita* ja vastaavasti  $t_1, \dots, t_n$  ovat valittuja *muuttujattomia termejä*.



### 3.4 Systemaattinen taulu

- Lauselogiikan keskeiset päättelyongelmat ovat *ratkeavia*.

**Esimerkki.** Voidaan konstruoida deterministinen Turing-kone  $T$ , jonka suoritus pysähtyy syötteen annettulla lauselogiikan lauseella  $\varphi$

1. hyväksyvään tilaan  $k$  (kyllä), jos syöte  $\varphi$  on pätevä, ja
2. hylkäävään tilaan  $e$  (ei), jos syöte  $\varphi$  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.



- Predikaattilogiikka ei ole ratkeava, vaan *puoliratkeava*.

**Esimerkki.** Lauseen  $\varphi$  pätevyyden tarkastamista varten voidaan konstruoida seuraavanlainen deterministinen Turing-kone  $T$ :

1. Jos syöte  $\varphi$  on pätevä,  $T$  pysähtyy hyväksyvään tilaan  $k$  (kyllä).
2. Jos syöte  $\varphi$  ei ole pätevä,  $T$  pysähtyy *joskus* hylkävään 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  $\varphi$  on pätevä.



### Systemaattisen taulun periaatteita

- Tuotetaan indeksoimalla riittävä määrä uusia vakioita  $c_1, c_2, c_3, \dots$ , kun hajoitetaan muotoa  $T \exists x \psi(x)$  tai  $E \forall x \psi(x)$  olevia solmuja.
- Tuotetaan tarpeen mukaan muuttujattomia termejä  $t_1, t_2, t_3, \dots$ , jotka rakentuvat  $E\varphi$ :ssä esiintyvistä vakio- ja funktiosymboleista sekä mahdollisesti käyttöönotetuista uusista vakioista  $c_1, c_2, c_3, \dots$
- Sekvenssin  $t_1, t_2, t_3, \dots$  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  $t_1, t_2, t_3, \dots$



**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:  $0, s(0), s(s(0)), s(s(s(0))), \dots$



- 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:

$$\begin{array}{c}
 1. T \forall x G(s(x), x) \\
 \quad \vdots \\
 2. T \forall x \forall y (G(x, y) \rightarrow G(s(x), y)) \\
 \quad \vdots \\
 3. E G(s(s(s(0))), s(0)) \\
 \quad \vdots \\
 4. T G(s(0), 0) \quad 1. x/0 \\
 \quad \vdots \\
 4. T G(s(s(0)), s(0)) \quad 1. x/s(0) \\
 \quad \vdots \\
 4. T G(s(s(s(0))), s(s(0))) \quad 1. x/s(s(0)) \\
 \quad \vdots
 \end{array}$$





### 3.5 Vastamallien konstruointi

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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

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

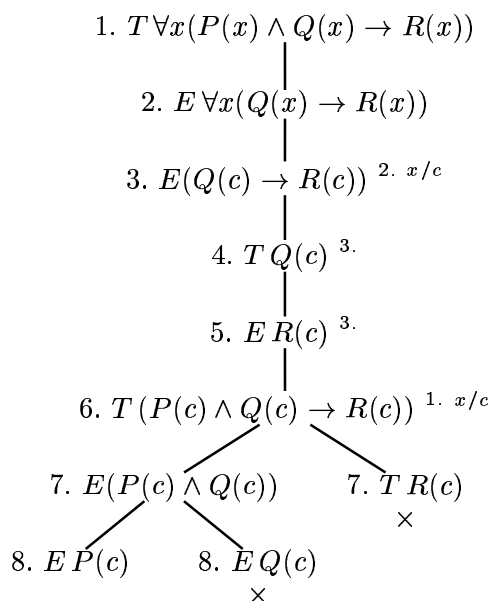
1.  $E \forall x(P(x) \rightarrow Q(x))$
2.  $E(P(c) \rightarrow Q(c))$  <sup>1.  $x/c$</sup>
3.  $T(P(c))$  <sup>2.</sup>
4.  $E(Q(c))$  <sup>2.</sup>

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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



© 2001 Teknillinen korkeakoulu, 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  $T \forall x(P(x) \wedge Q(x) \rightarrow R(x))$  on hajoitettu polulla  $P$ , koska polulla  $P$  on solmu  $T(P(t) \wedge Q(t) \rightarrow R(t))$  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  $c^A = 1$ ,  $P^A = R^A = \{\}$  ja  $Q^A = \{1\}$ .

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio







## 4 Hilbertin järjestelmä

Aksiomat ovat seuraavan muotoisia kaavoja:

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

Päätelysäännöt:  $\frac{\alpha \rightarrow \beta \quad \alpha}{\beta}$  (MP)  $\frac{\alpha}{\forall x\alpha}$  (Yleistys)



**Määritelmä.** Olkoon  $\Sigma$  joukko lauseita.

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

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

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

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



**Esimerkki.** Todistetaan  $\forall x(P(x) \rightarrow P(x))$ .

1.  $P(x) \rightarrow ((P(x) \rightarrow P(x)) \rightarrow P(x))$  Ax1
2.  $(P(x) \rightarrow ((P(x) \rightarrow P(x)) \rightarrow P(x)))$   
 $\rightarrow ((P(x) \rightarrow (P(x) \rightarrow P(x))) \rightarrow (P(x) \rightarrow P(x)))$  Ax2
3.  $(P(x) \rightarrow (P(x) \rightarrow P(x))) \rightarrow (P(x) \rightarrow P(x))$  MP,1,2
4.  $P(x) \rightarrow (P(x) \rightarrow P(x))$  Ax1
5.  $P(x) \rightarrow P(x)$  MP,3,4
6.  $\forall x(P(x) \rightarrow P(x))$  Yleistys,5

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



## 5 Normaalimuodot

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



## 5.1 Prenex-normaali muoto

Lause  $\alpha$  on *prenex*-normaalimuodossa, mikäli se on muotoa

$$Q_1x_1Q_2x_2 \cdots Q_nx_n\phi,$$

missä jokainen  $Q_i$  on jompikumpi kvanttoreista ( $\forall$  tai  $\exists$ ) ja alikaava  $\phi$  ei sisällä kvanttoreita.

**Esimerkki.** Seuraavat lauseet ovat prenex-normaalimuodossa:

$$P(a), \forall xP(x), \forall x\exists y P(x, y) \text{ ja}$$

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

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



## 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):

$$\neg\neg\varphi \rightsquigarrow \varphi$$

$$\neg(\varphi \wedge \psi) \rightsquigarrow \neg\varphi \vee \neg\psi$$

$$\vec{Q}x\neg\forall y\varphi \rightsquigarrow \vec{Q}x\exists y\neg\varphi$$

$$\neg(\varphi \vee \psi) \rightsquigarrow \neg\varphi \wedge \neg\psi$$

$$\vec{Q}x\neg\exists y\varphi \rightsquigarrow \vec{Q}x\forall y\neg\varphi$$

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



3. Tuodaan kvanttorit ulos lauserakenteesta:

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

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

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

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

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

**Esimerkki.** Muuttujan korvaaminen on olennaista:

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



**Esimerkki.** Suoritetaan muunnos prenex-normaaliinmuotoon:

$$\forall x(P(x) \rightarrow \exists zR(z, x)) \rightarrow \exists xQ(x)$$

$$\rightsquigarrow \neg\forall x(P(x) \rightarrow \exists zR(z, x)) \vee \exists xQ(x)$$

$$\rightsquigarrow \neg\forall x(\neg P(x) \vee \exists zR(z, x)) \vee \exists xQ(x)$$

$$\rightsquigarrow \exists x\neg(\neg P(x) \vee \exists zR(z, x)) \vee \exists xQ(x)$$

$$\rightsquigarrow \exists x(\neg\neg P(x) \wedge \neg\exists zR(z, x)) \vee \exists xQ(x)$$

$$\rightsquigarrow \exists x(P(x) \wedge \forall z\neg R(z, x)) \vee \exists xQ(x)$$

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

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

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

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



## 5.2 Konjunkttiivinen 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  $\alpha$  on konjunkttiivisessa normaalimuodossa, mikäli se on on prenex-normaalimuodossa  $Q_1x_1Q_2x_2\cdots Q_nx_n\phi$ , missä kvanttoreita sisältämätön osa  $\phi = \phi_1 \wedge \cdots \wedge \phi_n$  ja jokainen konjunktion jäsen  $\phi_i$  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 konjunkttiivisessa normaalimuodossa.



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

$$\varphi \vee (\phi \wedge \psi) \quad \rightsquigarrow \quad (\varphi \vee \phi) \wedge (\varphi \vee \psi)$$

$$(\phi \wedge \psi) \vee \varphi \quad \rightsquigarrow \quad (\phi \vee \varphi) \wedge (\psi \vee \varphi)$$

- Näin ollen voimme todeta seuraavan tuloksen:

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

**Esimerkki.** Muunnetaan edellä johdettu prenex-normaalimuoto edelleen konjunkttiiviseksi normaalimuodoksi:

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



### 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  $-$ :

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



### Eksistenssikvanttorien eliminointi yleisessä tapauksessa

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

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



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

**Esimerkki.** Suoritetaan Skolemointi seuraaville lauseille:

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



### Skolemoinnin loogiset ominaisuudet

**Väite.** Prenex-normaali muodossa oleva lause  $\varphi$  on toteutuva  $\iff$  lauseen  $\varphi$  skolemoitu muoto  $\varphi'$  on toteutuva.

**Huomio.** Prenex-normaali muodossa olevan lauseen  $\varphi$  skolemoitu muoto  $\varphi'$  ei välttämättä ole loogisesti ekvivalentti lauseen  $\varphi$  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  $\exists x P(x) \not\equiv P(c)$ .



## 5.4 Lauseiden klausuulimuoto predikaattilogiikassa

Mille tahansa lauseelle voidaan hakea klausuulimuoto seuraavasti:

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

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

$$\rightsquigarrow \forall x\exists z((P(x) \wedge \neg Q(x, z)) \vee R(x)) \quad (1)$$

$$\rightsquigarrow \forall x\exists z((P(x) \vee R(x)) \wedge (\neg Q(x, z) \vee R(x))) \quad (2)$$

$$\rightsquigarrow \forall x((P(x) \vee R(x)) \wedge (\neg Q(x, f(x)) \vee R(x))) \quad (3)$$

$$\rightsquigarrow \{\{P(x), R(x)\}, \{\neg Q(x, f(x)), R(x)\}\}. \quad (4)$$

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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

**Esimerkki.**  $\forall xP(x) \wedge \exists xQ(x)$ . Konjunktin 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  $\forall xP(x) \vee \exists xQ(x)$  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  $\implies$  klausuulimuodosta tulee rakenteeltaan yksinkertaisempi.

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio





## 6 Tietämyksen esittämisestä

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

© 2001 Teknillinen korkeakoulu, 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  $\Sigma$  loogisten seurausten joukkoa

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

Nyt

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



**Esimerkki.** Kuvataan sähkölinjaa predikaattilogiikan lausein:

$$\Sigma = \{ \forall x \forall y \forall z (\text{johto}(x, y) \wedge \text{jännite}(x, z) \rightarrow \text{jännite}(y, z)), \\ \text{johto}(p1, p2), \text{johto}(p2, p3), \text{jännite}(p1, 220) \}.$$

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

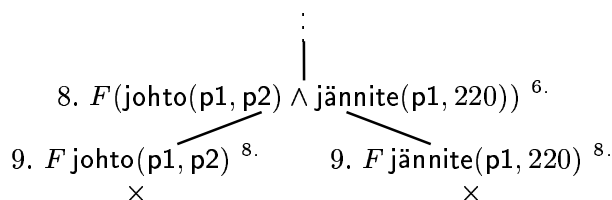
1.  $T \forall x \forall y \forall z (\text{johto}(x, y) \wedge \text{jännite}(x, z) \rightarrow \text{jännite}(y, z))$
2.  $T \text{johto}(p1, p2)$
3.  $T \text{johto}(p2, p3)$
4.  $T \text{jännite}(p1, 220)$
5.  $F \text{jännite}(p3, 220)$
6.  $T \text{johto}(p1, p2) \wedge \text{jännite}(p1, 220)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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio

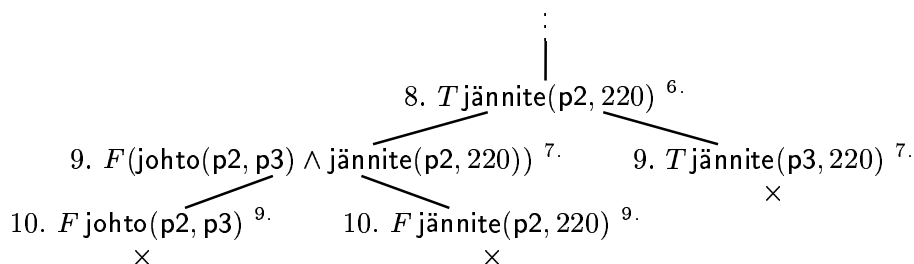


Hajoitettaessa polun 6. solmu taulu jakautuu kahteen haaraan:

- Vasen haara ( $\tau_1$ ):



- Oikea haara ( $\tau_2$ ):



© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio

**Tietämyksen esittämisen problematiikkaa**

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

$\Rightarrow$  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  $\Sigma' \models Q(a)$ .

**6.2 Ohjeita predikaattien määrittelemiseen**

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

$$Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \psi$$

missä kukin kvanttori  $Q_i$  on joko  $\forall$  tai  $\exists$ , ja kaava  $\psi$  on konjunkttiivisessa normaalimuodossa eikä sisällä kvanttoreita.

- Yllä  $\psi = \psi_1 \wedge \cdots \wedge \psi_m$ , missä kukin  $\psi_i$  on literaalien disjunktio

$$\begin{aligned} & \neg Q_1(\vec{t}_1) \vee \cdots \vee \neg Q_k(\vec{t}_k) \vee P_1(\vec{s}_1) \vee \cdots \vee P_l(\vec{s}_l) \\ \equiv & Q_1(\vec{t}_1) \wedge \cdots \wedge Q_k(\vec{t}_k) \rightarrow P_1(\vec{s}_1) \vee \cdots \vee P_l(\vec{s}_l). \end{aligned}$$



- Predikaattilogiikalla annetut määritelmät ovat korkeintaan yhtä monimutkaisia kuin edellä kuvattu normaalimuoto.
- Jos jokainen kvanttori  $Q_i$  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  $Q_1, \dots, Q_k$  ja  $P$  argumentteina on vakiosymboleista, muuttujista  $x_1, \dots, x_n$  ja funktiosymboleista rakentuvien termien jonoja  $\vec{t}_1, \dots, \vec{t}_k$  ja  $\vec{t}$ .

- Määritelmiä voidaan usein kirjoittaa tähän muotoon: mietitään millä ehdoilla  $Q_1(\vec{t}_1), \dots, Q_k(\vec{t}_k)$  voidaan päätellä predikaattia  $P$  koskeva väittämä  $P(\vec{t})$ . Ko. muotoa olevia lauseita saatetaan tarvita useita.



**Esimerkki.** Olkoon annettuna predikaatit

1.  $sairastaa(x) = \text{“henkilö } x \text{ on sairas”}$  ja
2.  $tapaa(x, y) = \text{“henkilö } x \text{ tapaa henkilön } y\text{”}$ .

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

$$\forall x \forall y \dots (Q_1(\vec{t}_1) \wedge \dots \wedge Q_k(\vec{t}_k) \rightarrow \text{tartuntavaarassa}(x)).$$



- Näin saadaan muodostettua lauseet

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

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

- 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).



### Määritelmien käyttö konkreettисessa päättelyssä

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

Näin saadaan lausejoukko

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

- Kyseisessä asetelmassa saadaan

$$\Sigma \models \text{tartuntavaarassa}(\text{Lyyli}) \wedge \text{tartuntavaarassa}(\text{Hemmo}).$$

- Kokeile tämän osoittamista semanttisella taululla!

**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.

**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:

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



**Esimerkki.** Lisätään edellisen esimerkkiin tyyppipredikaatteja.

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

$$\begin{aligned} & \forall x \forall y \forall z (\text{henkilö}(x) \wedge \text{henkilö}(y) \wedge \text{tapaa}(x, y) \wedge \\ & \quad \text{tauti}(z) \wedge \text{sairastaa}(y, z) \rightarrow \text{tartuntavaarassa}(x, z)) \text{ ja} \\ & \forall x \forall y \forall z (\text{henkilö}(x) \wedge \text{henkilö}(y) \wedge \text{tapaa}(x, y) \wedge \text{tauti}(z) \wedge \\ & \quad \text{tartuntavaarassa}(y, z) \rightarrow \text{tartuntavaarassa}(x, z)). \end{aligned}$$

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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

$$\begin{aligned} & \forall x \forall y (\text{tapaa}(x, y) \rightarrow \text{henkilö}(x) \wedge \text{henkilö}(y)) \\ & \forall x \forall y (\text{sairastaa}(x, y) \rightarrow \text{henkilö}(x) \wedge \text{tauti}(y)) \end{aligned}$$

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

$$\begin{aligned} & \forall x \forall y \forall z (\text{tapaa}(x, y) \wedge \text{sairastaa}(y, z) \rightarrow \text{tartuntavaarassa}(x, z)) \text{ ja} \\ & \forall x \forall y \forall z (\text{tapaa}(x, y) \wedge \text{tartuntavaarassa}(y) \rightarrow \text{tartuntavaarassa}(x, z)). \end{aligned}$$

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



### Muodoltaan monimutkaisempia määritelmiä

- Edellä otettiin lähtökohdaksi muotoa

$$\forall x_1 \forall x_2 \cdots \forall x_n (Q_1(\vec{t}_1) \wedge \cdots \wedge Q_k(\vec{t}_k) \rightarrow P(\vec{t}))$$

olevat määritelmät. Näiden ilmaisuvoima ei ole aina riittävä.

- Joissain tilanteissa tarvitaan eksistentiaalista kvantifiointia:

$$\begin{aligned} & \forall x (\text{solmu}(x) \rightarrow \exists y (\text{väri}(y) \wedge \text{väritetty}(x, y))) \\ \equiv & \quad \forall x \exists y (\text{solmu}(x) \rightarrow \text{väri}(y) \wedge \text{väritetty}(x, y)). \end{aligned}$$

- 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:

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

**Huomio.** Edellä oli keskeistä vaihtoehtoisuuden ilmaiseminen.



### 6.3 Nimien yksikäsitteisyys ja kattavuus

- Rajoitutaan jatkossa predikaattilogiikan kielisiin  $\mathcal{L}$ , 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  $c_1, \dots, c_n$  ( $n > 1$ ) nimeämä:  $c_1^{\mathcal{A}} = \dots = c_n^{\mathcal{A}} = a$ .
  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.



**Nimien yksikäsitteisyys**

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

$$\neg(c_i = c_j)$$

olevat lauseet, missä  $i \in \{1, \dots, n\}$ ,  $j \in \{1, \dots, n\}$  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:

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

**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  $h_1, h_2, \dots$ .

$A_i$	Lyyli <sup><math>A_i</math></sup>	Hemmo <sup><math>A_i</math></sup>	Erkki <sup><math>A_i</math></sup>
$\{h_1, h_2, h_3\}$	$h_1$	$h_2$	$h_3$
$\{h_1, h_2, h_3\}$	$h_1$	$h_3$	$h_2$
$\{h_1, h_2, h_3\}$	$h_2$	$h_1$	$h_3$
$\{h_1, h_2, h_3\}$	$h_2$	$h_3$	$h_1$
$\{h_1, h_2, h_3\}$	$h_3$	$h_1$	$h_2$
$\{h_1, h_2, h_3\}$	$h_3$	$h_2$	$h_1$
$\{h_1, h_2, h_3, h_4\}$	$h_1$	$h_2$	$h_3$
$\vdots$	$\vdots$	$\vdots$	$\vdots$

$\implies$  Universumissa oltava vähintään 3 henkilöä.

**Nimien kattavuus**

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

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

- Tarvittavan lauseen pituus riippuu lineaarisesti vakioiden lukumäärästä  $n$ .

**Esimerkki.** Edellisen esimerkin mukaisessa kielessä tarvitaan lause

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



**Esimerkki.** Tarkastellaan lausejoukon

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

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

$A_i$	Lyyli <sup><math>A_i</math></sup>	Hemmo <sup><math>A_i</math></sup>	Erkki <sup><math>A_i</math></sup>
$\{h_1\}$	$h_1$	$h_1$	$h_1$
$\{h_1, h_2\}$	$h_1$	$h_1$	$h_2$
$\{h_1, h_2\}$	$h_1$	$h_2$	$h_1$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\{h_1, h_2, h_3\}$	$h_1$	$h_2$	$h_3$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\{h_1, h_2, h_3\}$	$h_3$	$h_2$	$h_1$

$\implies$  Universumissa voi olla korkeintaan 3 henkilöä.



**Esimerkki.** Tarkastellaan vielä edeltävien lausejoukkojen unionin

$$\Sigma_{\text{UNA}} \cup \Sigma_{\text{DCA}} = \{ \neg(\text{Lyyli} = \text{Hemmo}), \neg(\text{Lyyli} = \text{Erkki}), \\ \neg(\text{Hemmo} = \text{Erkki}), \\ \forall x(x = \text{Lyyli} \vee x = \text{Hemmo} \vee x = \text{Erkki}) \}$$

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

$A_i$	$\text{Lyyli}^{A_i}$	$\text{Hemmo}^{A_i}$	$\text{Erkki}^{A_i}$
$\{h_1, h_2, h_3\}$	$h_1$	$h_2$	$h_3$
$\{h_1, h_2, h_3\}$	$h_1$	$h_3$	$h_2$
$\{h_1, h_2, h_3\}$	$h_2$	$h_1$	$h_3$
$\{h_1, h_2, h_3\}$	$h_2$	$h_3$	$h_1$
$\{h_1, h_2, h_3\}$	$h_3$	$h_1$	$h_2$
$\{h_1, h_2, h_3\}$	$h_3$	$h_2$	$h_1$

$\implies$  Universumissa on oltava täsmälleen 3 henkilöä.



## 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.**

$$\forall x (\neg \text{sairastaa}(x) \wedge \neg \text{tartuntavaarassa}(x) \rightarrow \text{turvassa}(x))$$

$$\equiv \forall x (\text{sairastaa}(x) \vee \text{tartuntavaarassa}(x) \vee \text{turvassa}(x)).$$

- 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.



### Määritelmien täydellisyys

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

### Huomioita.

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



**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  $\Sigma$  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  $\Sigma$  ei ole täydellinen myöskään muille ko. kielen predikaateille (tapaa, sairastaa, turvassa ja =).

**Predikaatin määritelmän täydentäminen****Esimerkki.** Täydennetään edellisen esimerkin predikaattien määritelmät.

1. Yhtäsuuruuspredikaatin osalta riittää todeta nimien yksikäsitteisyys:

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

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

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

© 2001 Teknillinen korkeakoulu, 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
$\neg\text{sairastaa}(\text{Lyyli})$	$\text{tartuntavaarassa}(\text{Lyyli})$	$\neg\text{turvassa}(\text{Lyyli})$
$\neg\text{sairastaa}(\text{Hemmo})$	$\neg\text{tartuntavaarassa}(\text{Hemmo})$	$\text{turvassa}(\text{Hemmo})$
$\text{sairastaa}(\text{Erkki})$	$\neg\text{tartuntavaarassa}(\text{Erkki})$	$\neg\text{turvassa}(\text{Erkki})$

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



**Esimerkki.** Tarkastellaan vastaavaa konstruktiota lausejoukolla

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

- Rekursiivisesti määritellyn predikaatin  $\text{tuntee2}$  tarkoituksena on täydentää predikaatti  $\text{tuntee1}$  symmetriseksi.
- Täydennettynä määritelmät saadaan muotoon

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



- Yllättäen täydennetyistä määritelmistä ei seuraa loogisesti  $\neg \text{tuntee2}(\text{Lyyli}, \text{Hemmo})$  eikä  $\neg \text{tuntee2}(\text{Hemmo}, \text{Lyyli})$ .

- Lausejoukolla  $\Sigma'$  on seuraava ei-toivottu malli  $\mathcal{A}$ :

$$\text{Universumi } A = \{1, 2\},$$

$$\text{Lyyli}^{\mathcal{A}} = 1, \text{Hemmo}^{\mathcal{A}} = 2,$$

$$\text{tuntee1}^{\mathcal{A}} = \{\langle 1, 1 \rangle, \langle 2, 2 \rangle\} \text{ ja}$$

$$\text{tuntee2}^{\mathcal{A}} = \{\langle 1, 1 \rangle, \langle 1, 2 \rangle, \langle 2, 1 \rangle, \langle 2, 2 \rangle\}.$$

- Kyseiselle struktuurille  $\mathcal{A}$  pätee:

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

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



## 7 Herbrandin teoreema

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



### 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

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

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



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

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

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

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



## 7.2 Herbrand-struktuurit ja -mallit

**Määritelmä.** Kielen  $\mathcal{L}$  Herbrand-strukturi on strukturi  $\mathcal{H}$ , 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  $t_1, \dots, t_n$  muuttujattomaksi termiksi  $f(t_1, \dots, t_n)$ , ja
  4. jokaisen predikaattisymbolin  $P \in \mathcal{P}_n$  tulkintana on jokin relaatio  $P^{\mathcal{H}} \subseteq H^n$ .
- Lauseen  $\phi \in \mathcal{L}$  totuusarvo Herbrand-strukturissa  $\mathcal{H}$  lasketaan predikaattilogiikan totuusmääritelmän mukaisesti.





**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

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

- Lausejoukon  $\Sigma$  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  $\Sigma$  Herbrand-malli.



**Määritelmä.** Lausejoukon  $\Sigma$  (kielen  $\mathcal{L}$ ) *Herbrand-kanta*  $B$  on niiden *atomisten lauseiden* joukko, jotka voidaan muodostaa lausejoukossa  $\Sigma$  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

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

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

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



### 7.3 Herbrandin teoreema

- Rajoitutaan tarkastelemaan klausuulijoukkoja.
- Merkintä  $C(x_1, \dots, x_n)$  tarkoittaa muuttujat  $x_1, \dots, x_n$  sisältävää klausuulia  $\{P_1(\vec{t}_1), \dots, P_k(\vec{t}_k), \neg Q_1(\vec{s}_1), \dots, \neg Q_l(\vec{s}_l)\}$ .
- Klausuuli  $C(x_1, \dots, x_n)$  vastaa universaalisti kvantifioitua lausetta  $\forall x_1 \cdots \forall x_n \phi_C(x_1, \dots, x_n)$ , missä  $\phi_C(x_1, \dots, x_n)$  on klausuulin  $C(x_1, \dots, x_n)$  esitys literaalien disjunktiona.
- Klausuulijoukolle  $S$  voidaan määritellä Herbrand-struktuurit samaan tapaan kuin lausejoukoillekin.
- Klausuulijoukko  $S$  voidaan *instantioida* vastaavan Herbrand-universumin  $H_S$  suhteen seuraavasti.

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



**Väite.** (Herbrandin 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 rajoittaa syntaktisiin malleihin (Herbrand-malleihin) mielivaltaisten mallien sijaan. Herbrandin teoreema johtaa myös naiiviin proseduriin 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).



## 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.



- 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 selvitetävissä 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  $S' = \{\{P\}, \{\neg P, Q\}, \{\neg Q, Q\}\}$ .

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



## 8 Unifikaatio

- Substituutiot
- Yleisimmät unifioijat
- Unifikaatioalgoritmi

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



## 8.1 Substituutiot

**Määritelmä.** *Substituutio* (tai *korvaus*)  $\theta$  on äärellinen joukko

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

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

1. korvattavat muuttujat  $x_1, \dots, x_n$  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,  $\theta$  on *muuttujaton*.
- Jos korvaavat termit  $t_i$  ovat muuttujia,  $\theta$  on *nimeämssubstituutio*.



**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
- nimeämssubstituutio  $\theta_3 = \{x/y, y/z, z/x\}$ .

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



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

Substituutioiden  $\theta$  ja  $\lambda$  kompositio  $\theta\lambda$  määritellään joukkona

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

**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  $\{x/f(b), z/y\}$ .



## 8.2 Yleisimmät unifioijat

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

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

**Esimerkki.** Tarkastellaan seuraavien joukkojen unifioituvuutta.

Joukko $S$ :	Unifioija $\theta$ :
$\{P(x, f(a)), P(y, z)\}$	$\{y/x, z/f(a)\}$ 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)



**Määritelmä.** Olkoon  $\sigma$  lausekejoukon  $S = \{E_1, \dots, E_n\}$  unifioija.

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

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

- Yleisimmät unifioijat ovat yksikäsitteisiä seuraavaan tapaan:

**Väite.** Olkoot  $\theta$  ja  $\sigma$  joukon  $S$  yleisimpiä unifioijia. Tällöin on olemassa nimeämism substituutio  $\lambda$  siten että  $S\theta\lambda = S\sigma$ .



### 8.3 Unifikaatioalgoritmi

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

**Määritelmä.** Olkoon  $S$  ei-tyhjä joukko johonkin predikaattisymboliin  $P$  perustuvia atomikaavoja  $\{P(\vec{t}_1), \dots, P(\vec{t}_n)\}$ .

1. Joukon  $S$  *erokohta* on järjestyksessä ensimmäinen kohta (vasemmalta oikealle siirryttäessä), jossa joukon  $S$  atomikaavojen merkkijonoesityksissä on jokin eroavaisuus.
2. Joukon  $S$  *eroujoukko*  $D(S)$  kuuluvat atomikaavojen  $P(\vec{t}_1), \dots, P(\vec{t}_n)$  erokohdasta alkavat termit  $u_1, \dots, u_n$ .

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



**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.

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



**Väite.** Olkoon  $S$  äärellinen ei-tyhjä joukko atomikaavoja.

- Jos  $S$  on unifioituva, niin unifikaatioalgoritmin suoritus päättyy askeleen 3 kohdalla ja substituutioiden  $\sigma_0, \sigma_1, \dots, \sigma_k$  kompositio  $\sigma = \sigma_0\sigma_1 \dots \sigma_k$  on joukon  $S$  yleisin unifioija
- Jos  $S$  ei ole unifioituva, niin unifikaatioalgoritmin 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 yksialkioinen, jatketaan.

© 2001 Teknillinen korkeakoulu, 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  $S\sigma = \{P(g(a), f(g(a)))\} = S_2$ .



## 9 Resoluutiosääntö ja -todistukset

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



## 9.1 Resoluutiosääntö predikaattilogiikan tapauksessa

**Määritelmä.** Olkoot

$$C_1 = C'_1 \sqcup \{P(\vec{t}_1), \dots, P(\vec{t}_n)\} \text{ ja } C_2 = C'_2 \sqcup \{\neg P(\vec{u}_1), \dots, \neg P(\vec{u}_m)\}$$

kaksi klausuulia,

1. joissa *ei esiinny yhteisiä* muuttujia ja
2. joissa esiintyvien atomikaavojen joukko

$$\{P(\vec{t}_1), \dots, P(\vec{t}_n), P(\vec{u}_1), \dots, P(\vec{u}_m)\}$$

on unifioituva (yleisimpänä unifioijana  $\sigma$ ).

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  $A \cup B$ .



**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  $\sigma = \{y_1/x_1, x_2/x_1, y_2/x_1\}$ ).



## 9.2 Resoluutiotodistukset

- Lähtökohtana on joukko klausuuleita  $S$ , jonka klausuuleista johdetaan uusia klausuuleita resoluutiosäännöllä.
- *Johtojen* ja *refutaation* 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.** Klausuulijoukko  $S$  on refutoituva (eli klausuulijoukosta  $S$  on johto  $C_1, \dots, C_n$  tyhjälle klausuulille  $C_n = \square$ )  $\iff S$  on toteutumaton.

*Todistus.* Sivutetaan.



**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:

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

Refutaatio:	1. $\{P(z)\}$	$S_1\{x/z\}$
	2. $\{P(f(z))\}$	$S_1\{x/z\}$
	3. $\{\neg P(x), \neg P(y)\}$	$S_2$
	4. $\{\neg P(y)\}$	1,3, MGU $\{x/z\}$
	5. $\square$	2,4, MGU $\{y/f(z)\}$

**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  $\phi$  ja  $\psi$  lauseita ja  $\Sigma$  lausejoukko.

1. Pätevyys:  $\models \phi \iff \text{KM}(\{\neg\phi\})$  on refutoituva.
2. Ekvivalenssi:  $\phi \equiv \psi \iff \text{KM}(\{\neg(\phi \leftrightarrow \psi)\})$  on refutoituva.
3. Looginen seuraavuus:  $\Sigma \models \phi \iff \text{KM}(\Sigma \cup \{\neg\phi\})$  on refutoituva.

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



**Esimerkki.** Osoitetaan resoluutiolla, että  $\models \forall xP(x) \rightarrow \exists xP(x)$ .

- Haetaan lauseen negaatiolle klausuulimuoto:

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

- Klausuuleista  $\{P(x)\}$  ja  $\{\neg P(y)\}$  saadaan tyhjä klausuuli  $\square$  (MGU  $\{x/y\}$ ) yhdellä resoluutioaskelella.
- Täten  $S$  on refutoituva
  - $\implies S$  on toteutumaton
  - $\implies \neg(\forall xP(x) \rightarrow \exists xP(x))$  on toteutumaton
  - $\implies \forall xP(x) \rightarrow \exists xP(x)$  on pätevä.



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

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

loogiseksi seuraukseksi. Haetaan tarvittavat klausuulimuodot:

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

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



Refutaatio onnistuu 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 $\epsilon$ |

• Yleisimpien unifioijien kompositio  $\{x/y\}\{y/c\}\epsilon = \{x/c, y/c\}$ .

• Täten  $S$  on toteutumaton

$\implies \Sigma \cup \{\neg \exists x(E(x) \wedge K(x))\}$  on toteutumaton

$\implies \exists x(E(x) \wedge K(x))$  on joukon  $\Sigma$  looginen seuraus.



### 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.

© 2001 Teknillinen korkeakoulu, 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.  $\forall y \forall v \forall z \forall x (L(y, v, z) \rightarrow L(c(x, y), v, c(x, z)))$ .

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



- Johdetaan lauseille klausuulimuodot:

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

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

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

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

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

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



### 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



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*).

© 2001 Teknillinen korkeakoulu, 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  $C_2 \in S_0 \cup \dots \cup S_{i-1}$ .

**Esimerkki.** Olkoon alimpana kerroksena

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

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

$$S_2 = \{\square, \{\neg I(c)\}, \{\neg K(c)\}\}.$$

© 2001 Teknillinen korkeakoulu, 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  $\Sigma$  (olettamukset) on tyypillisesti toteutuva. Tällöin voidaan ajatella:
  1.  $T$  muodostuu lauseesta  $\neg\phi$  saatavien klausuulien joukosta ja
  2. joukkoon  $S - T$  kuuluvat lausejoukosta  $\Sigma$  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.



**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  $\Sigma$  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 $\epsilon$ |