T-79.5102 **Syksy 2005**

Laskennalllisen logiikan erikoiskurssi

Laskuharjoitus 8

Ratkaisut

1. a) Koska $\Gamma(\emptyset) = \{a, b, c, d\}$, WF-malli on tyhjä. Lasketaan stabiilit mallit:

not a	oletus
b	$b \leftarrow$ not a
c	$c \leftarrow b$
not d	koska $d \leftarrow$ not c, a ei voida käyttää (not a)

   Atomilauseiden totuusarvot ovat nyt määritelty. Tarkastetaan onko $\{b, c\}$ stabiili malli:

$$\Gamma(\{b, c\}) = \{b, c\}$$

   implikoi että $\{b, c\}$ on ensimmäinen malli. Peruutetaan nyt oletus not a:

a	oletus, peruutettu
not b	a saadaan ainoastaan jos not b ja not d
not d	
c	$c \leftarrow$ not d

   Kun

$$\Gamma(\{a, c\}) = \{a, c\}$$

   niin $\{a, c\}$ on toinen stabiili malli. Muita malleja ei ole, koska oletuksia ei ole enää jäljellä.

   b) Käytetään lyhennysmerkintää $s^n(0) = n$. Lisäksi ratkaisussa esiintyy joukko-opillinen komplementaatio $\overline{S}$.

$$\begin{aligned}
\Gamma(\emptyset) &= \overline{\{p(0,i)\}}_{i=0}^{\infty} \\
\Gamma^2(\emptyset) &= \{q(0,0), r(0), dom(i)\}_{i=0}^{\infty} \\
\Gamma^3(\emptyset) &= \overline{\{p(0,i), p(1,i)\}}_{i=0}^{\infty} \\
\Gamma^4(\emptyset) &= \{q(0,0), q(1,0), r(0), dom(i)\}_{i=0}^{\infty} \\
&\vdots \\
\Gamma^{2n+1}(\emptyset) &= \overline{\{p(0,i), \ldots, p(n,i)\}}_{i=0}^{\infty} \\
\Gamma^{2n+2}(\emptyset) &= \{q(0,0), \ldots, q(n,0), r(0), dom(i)\}_{i=0}^{\infty} \\
&\vdots \\
\left(\Gamma^2\right)^{\omega}(\emptyset) &= \{q(i,0), dom(i), r(0)\}_{i=0}^{\infty} \\
\left(\Gamma^2\right)^{\omega+1}(\emptyset) &= \{q(i,0), dom(i), r(0), r(1), q(0,1)\}_{i=0}^{\infty} \\
&\vdots \\
\left(\Gamma^2\right)^{\omega\cdot 2}(\emptyset) &= \{q(i,0), q(i,1), dom(i), r(0), r(1)\}_{i=0}^{\infty} \\
&\vdots \\
\left(\Gamma^2\right)^{\omega\cdot k}(\emptyset) &= \{q(i,j), dom(i), r(j)\}_{\substack{0\leq i<\infty \\ 0\leq j<k}} \\
&\vdots \\
\left(\Gamma^2\right)^{\omega^2}(\emptyset) &= \{q(i,j), dom(i), r(j)\}_{\substack{0\leq i<\infty \\ 0\leq j<\infty}} = M
\end{aligned}$$

Koska $\Gamma(M) = M$, on ohjelman WF-malli $M \cup \{a \mid a \notin M\}$, eli:

$$\{q(i,j), dom(i), r(j), \neg u(i), \neg p(i,j)\}_{\substack{0\leq i<\infty \\ 0\leq j<\infty}}$$

Koska WF-malli antaa totuusarvon kaikille ohjelmassa esiintyville atomeille, on sen positiivinen osa samalla ohjelman yksikäsitteinen stabiili malli.

2. Yleistetään aluksi ohjelman koodausta siirtämällä atomien nimissä esiintyvät alaindeksit predikaattien argumenteiksi. Tällöin voidaan koodata kaksi n-bittistä laskuria säännöillä:

$$\begin{aligned}
a(X) &\leftarrow d(X), \text{not } \overline{a}(X) \\
\overline{a}(X) &\leftarrow d(X), \text{not } a(X) \\
b(X) &\leftarrow d(X), \text{not } \overline{b}(X) \\
\overline{b}(X) &\leftarrow d(X), \text{not } b(X),
\end{aligned}$$

missä d on apupredikaatti, joka on määritelty n:llä faktalla $d(1), \ldots, d(n)$.

Predikaatin c voi määritellä useammallakin eri tavalla. Ehkä yksinkertaisin lähtökohta on olettaa, että $c(i)$ on tosi kaikilla $1 \leq i \leq n$:

$$c(X) \leftarrow d(X), \text{not } \overline{c}(X).$$

Yllä oleva sääntö on *oletussääntö*: jos ei toisin todisteta, $c(i)$ on tosi. Tämän oletuksen jälkeen täytyy ainoastaan käydä läpi tapaukset, joissa $c(i)$ on epätosi, eli laskurin $a_i \cdots a_n$ arvo on suurempi kuin laskurin $b_i \cdots b_n$. Näitä tapauksia on kaksi erilaista:

(a) Bitti $a_i > b_i$:

$$\overline{c}(X) \leftarrow d(X), a(X), \text{not } b(X) \ .$$

(b) Bitti $a_i = b_i$ ja laskuri $a_{i+1} \cdots a_n > b_{i+1} \cdots b_n$. Tämän ehdon mallintamiseksi tarvitaan kaksi eri sääntöä:

$$\begin{aligned} &\overline{c}(X) \leftarrow d(X), a(X), b(X), \overline{c}(X+1) \\ &\overline{c}(X) \leftarrow d(X), \text{not } a(X), \text{not } b(X), \overline{c}(X+1). \end{aligned}$$

Huom. Jälkimmäisessä säännössä ei voi korvata atomia $\overline{c}(X+1)$ literaalilla not $c(X+1)$, sillä tällöin laskurin viimeinen bitti toimisi väärin.

Vaihtoehtoisessa ratkaisutavassa oletetaan $c(i)$ epätodeksi säännöllä:

$$\overline{c}(X) \leftarrow d(X), \text{not } c(X),$$

ja määritellään lisäksi säännöt tapauksille, joissa $c(i)$ on tosi.