T-79.148 **Kevät 2002**
Tietojenkäsittelyteorian perusteet
Harjoitus 12
Demonstraatiotehtävien ratkaisut

4. Tehtävässä halutaan todistaa seuraava ongelma ratkeamattomaksi:

   Hyväksyykö annettu Turingin kone M syötteen ε?

   Määritellään aluksi kieli $L = \{M \mid M \text{ pysähtyy syötteellä } \varepsilon\}$. Nyt L on rekursiivinen jos ja vain jos tehtävänannon ongelma on ratkeava. Seuraavaksi osoitetaan, että kieli $H = \{Mw \mid M \text{ pysähtyy syötteellä } w\}$ voidaan palauttaa rekursiivisesti (*recursively reduce*) kieleen L (merkitään $H \leq_m L$), joten L on vähintään yhtä vaikea kuin H. Koska H ei ole rekursiivinen (monisteen lause 6.9), ei L voi myöskään olla rekursiivinen.

   Rekursiivinen palautus määritellään seuraavasti: Olkoon $A \subseteq \Sigma^*$ ja $B \subseteq \Gamma^*$ kieliä. Nyt $A \leq_m B$ jos ja vain jos on olemassa rekursiivinen funktio $f : \Sigma^* \to \Gamma^*$ siten, että

   $$\forall w \in \Sigma^* : w \in A \Leftrightarrow f(w) \in B \ .$$

   Tässä tapauksessa halutaan löytää funktio f siten, että $f(Mw) \in L$ jos ja vain jos $Mw \in H$. Käytännössä tämä tarkoittaa sitä, että halutaan löytää systemaattinen tapa rakentaa Turingin kone M', joka pysähtyy tyhjällä syöttellä täsmälleen silloin, kun kone M pysähtyy syötteellä $w = w_1 w_2 \cdots w_n$.

   Onneksi tämä on helppo tehtävä. Kone M' kirjoittaa ensin nauhalle sanan w, siirtyy takaisin nauhan alkuun, ja alkaa simuloida konetta M tekemällä samat siirtymät kuin M:kin tekisi. Nyt M' pysähtyy jos ja vain jos M pysähtyy.

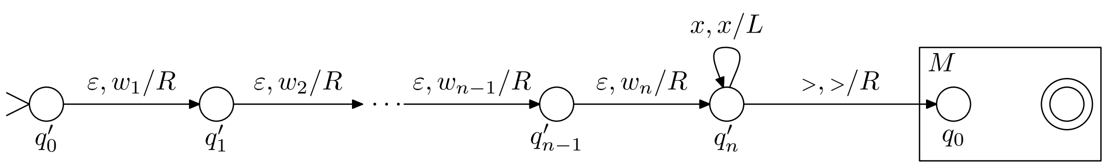
   Formaalisti f voidaan määritellä seuraavasti:

   $$f(\langle Q, \Sigma, \Gamma, \delta, q_0, q_{\text{acc}}, q_{\text{rej}} \rangle, w_1 w_2 \cdots w_n) = \langle Q', \Sigma, \Gamma, \delta', q'_0, q_{\text{acc}}, q_{\text{rej}} \rangle,$$

   missä

   $$\begin{aligned} Q' &= Q \cup \{q'_i \mid 0 \leq i \leq n\} \\ \delta' &= \delta \cup \{\langle q'_i, \varepsilon, q'_{i+1}, w_{i+1}, R\rangle \mid 0 \leq i < n\} \\ &\cup \{\langle q_n, x, q_n, x, L\rangle \mid x \in \Gamma \cup \{<\}\} \\ &\cup \{\langle q_n, >, q_0, >, R\rangle\} \end{aligned}$$

   Koska koneeseen M lisätään vain äärellinen määrä uusia tiloja ja siirtymiä (n ei voi olla ääretön), on f selvästikin rekursiivinen funktio.



5. Määritellään aluksi kolme yksinkertaista apukonetta:

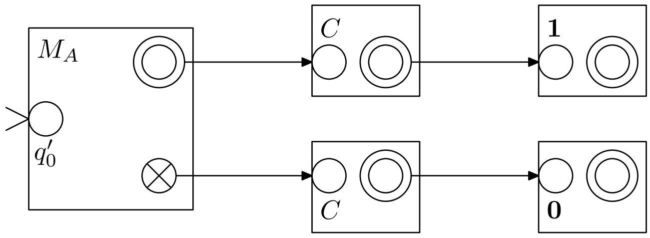
   - **1** kirjoittaa nauhalle merkin 1, siirtää lukupäätä askeleen oikealle ja pysähtyy.
   - **0** kirjoittaa nauhalle merkin 0, siirtää lukupäätä askeleen oikealle ja pysähtyy ja pysähtyy.
   - C tyhjentää koneen syötenauhan, palauttaa lukupään nauhan alkuun ja pysähtyy.

Koneet ovat yksinkertaisia, joten niitä ei esitetä tässä.

(i) [$\Rightarrow$] Olkoon kieli $A \subseteq \Sigma^*$ rekursiivinen. Tällöin on olemassa Turingin kone $M_A = \langle Q, \Sigma, \Gamma, \delta, q_0, q_{\text{acc}}, q_{\text{rej}} \rangle$ siten, että:

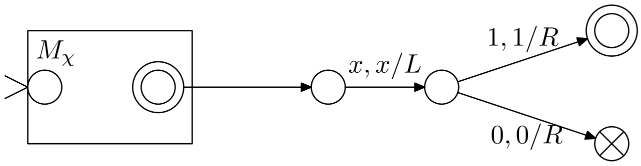
$$\forall w \in \Sigma^* : w \in L \Leftrightarrow (q_0, w) \vdash^*_{M_A} (q_{\text{acc}}, \alpha)$$
$$w \notin L \Leftrightarrow (q_0, w) \vdash^*_{M_A} (q_{\text{rej}}, \alpha)$$

Muodostetaan Turingin kone M yhdistämällä kone M_A koneisiin $\mathbf{1}$, $\mathbf{0}$ ja C seuraavaan tapaan:



Mikäli $w \in L$, niin kone M_A päätyy hyväksyvään tilaan, minkä jälkeen M tyhjentää nauhan ja kirjoittaa nauhalle luvun 1. Muussa tapauksessa nauhalle kirjoitetaan 0. Koska A on rekursiivinen, M_A pysähtyy aina, joten myös M pysähtyy aina. Näin ollen M laskee funktion $\chi(w) = \begin{cases} 1, w \in A \\ 0, w \notin A \end{cases}$, joka on kielen A karakteristinen funktio.

[$\Leftarrow$] Seuraavaksi oletetaan, että funktio $\chi(w)$ on rekursiivinen. Tällöin on olemassa Turingin kone M_χ, joka laskee sen arvon. Määritelmän mukaan M_χ pysähtyy aina, ja laskennan tulos kirjoitetaan nauhalle lukupään vasemmalle puolelle. Muodostetaan Turingin kone M seuraavaan tapaan:



Nyt kone M hyväksyy sanan w aina, kun $\chi(w) = 1$, ja hylkää sen, kun $\chi(w) = 0$. Näin ollen M tunnistaa kielen A totaalisesti, joten A on rekursiivinen.

(ii) Todistuksessa käytetään apuna yksinkertaisia Turingin koneita:

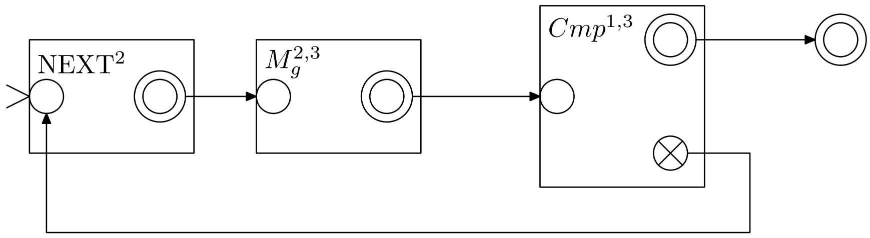
- NEXT — lukee nauhalla olevan merkkijonon $x \in \Sigma^*$ ja korvaa sen leksikografisesti seuraavalla merkkijonolla (tehtävän 9.5 tapaan).
- $Cmp^{i,j}$ — vertailee moninauhaisen Turingin koneen nauhoja i ja j keskenään ja hyväksyy syötteen mikäli niiden sisältö on sama.

Halutaan todistaa väite:

> Kieli $A \subseteq \Sigma^*$ on rekursiivisesti numeroituva ($A \in RE$), jos ja vain jos $A = \emptyset$ tai on olemassa rekursiivinen funktio $g : \{0,1\}^* \to \Sigma^*$, jolla
>
> $$A = \{g(x) \mid x \in \{0,1\}^*\}$$

Mikäli $A = \emptyset$, niin triviaalisti $A \in RE$ ja $g(x) = 0$ vastaava rekursiivinen funktio.

Jos annetut ehdot täyttävä funktio g on olemassa, niin on olemassa Turingin kone M_g, joka laskee g:n. Tästä voidaan triviaalisti muodostaa kaksinauhainen kone $M_g^{1,2}$, joka laskee g:n mutta tallettaa tuloksen toiselle nauhalle ensimmäisen asemasta. Muodostetaan 3-nauhainen Turingin kone M_A seuraavasti:



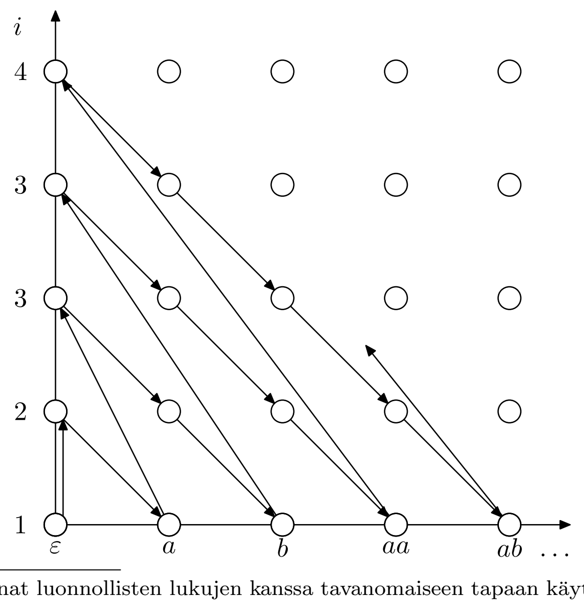
Kone saa syötteensä ensimmäisellä nauhalla, ja sitä ei enää missään vaiheessa muuteta. Varsinaisessa laskentasilmukassa M_A korvaa 2. nauhalla olevan bittijonon x leksikografisesti seuraavalla bittijonolla y, minkä jälkeen lasketaan $g(y)$ ja talletetaan tulos 3. nauhalle. Lopuksi vertaillaan 1. ja 3. nauhojen sisältö keskenään. Mikäli nauhojen sisältö on sama, sana hyväksytään, muuten palataan alkuun.

[$\Leftarrow$] Tarkastellaan sanaa $w \in A$. Oletetaan, että annetut ehdot täyttävä rekursiivinen funktio g on olemassa. Tällöin $w = g(x)$ jollain $x = x_1 x_2 \ldots x_n$, missä n on äärellinen. Koska millä tahansa äärellisen mittaisella merkkijonolla on vain äärellinen määrä edeltäjiä leksikografisessa järjestyksessä, generoi NEXT lopulta x:n, jolloin $M_g^{2,3}$ laskee kolmannelle nauhalle sanan w, ja kone hyväksyy sanan. Näin ollen M_A tunnistaa kielen A, joten $A \in RE$.

[$\Rightarrow$] Seuraavaksi oletetaan, että $A \in RE - \{\emptyset\}$. Tällöin on olemassa Turingin kone M_A, joka tunnistaa A:n. Määritellään apukone $M_{A,i}$, joka simuloi koneen M_A toimintaa i:n askeleen ajan. Kone $M_{A,i}$ hyväksyy syötteen x mikäli M_A hyväksyy sen käyttäen korkeintaan i askelta, muuten se hylkää sen. Huomataan, että kone $M_{A,i}$ on totaalinen, eli se pysähtyy aina.

Funktio g rakennetaan koneen $M_{A,i}$ avulla. Kaikki syötteet x ja rajat i koodataan funktion $c(x, y) = 0^x 10^y$ avulla bittijonoiksi pareittain[1], ja $g(c(x, y)) = x$, mikäli $M_{A,y}$ hyväksyy sanan x. Funktiota g määriteltäessä valitaan jokin kiinnitetty sana $x_0 \in A$, joka palautetaan aina kun $M_{A,y}$ ei pysähdy tai jos g:n syöte on väärää muotoa.

Konstruktion perusajatuksena on käydä läpi kaikki mahdolliset M_A:n syötteet lomitettuina: ensin ajetaan M_A:ta yhden askeleen verran ensimmäisellä syötteellä, sitten kaksi askelta ensimmäisellä syötteellä, yksi askel toisella syötteellä, jne. Alla olevassa kuvassa on tämä havainnollistettu aakkostolle $\Sigma = \{a, b\}$ (huomaa samankaltaisuus tehtävän 2.5 kanssa):



[1] Tässä samaistetaan Σ^*:n sanat luonnollisten lukujen kanssa tavanomaiseen tapaan käyttäen leksikografista järjestystä.

Esitetään vielä lopuksi todistus formaalimmin: Määritellään funktio $g' : \{0,1\}^* \to \{0,1\}^*$ seuraavasti:

$$g'(w) = \begin{cases} x, & w = 0^x 10^y \text{ ja } M_{A,y}(x) \text{ pysähtyy} \\ x_0, & \text{muulloin} \end{cases},$$

missä $x_0 \in A$. Lopuksi määritellään funktio $g(x) = d(g'(x))$, missä d on funktio, joka kuvaa bittijonon o^x joukon Σ^* x:nteen alkioon leksikografisessa järjestyksessä. Funktion $g'(x)$ arvo voidaan laskea äärellisessä ajassa, koska $M_{A,y}(x)$ ei voi jäädä ikuiseen silmukkaan. Näin ollen g' on rekursiivinen, joten myös g:kin on.

Tässä vaiheessa on huomattava, että vaikka g on aina olemassa, niin sitä ei ole välttämättä mahdollista löytää, sillä sopivan alkion $x_0 \in A$ etsiminen on yleisessä tapauksessa ratkeamaton ongelma.