1 Deskriptiivistä vaativuusteoriaa

Tämän osion esitys seuraa H.-D. Ebbinghausin ja J. Flumin kirjaa *Finite Model Theory.*

1.1 Turingin kone ja vaativuusluokat

Edellisessä luvussa nähtiin, että äärellisen automaatin kielten tunnistukyky on varsin rajoittunut. Äärellisen automaatin kielten tunnistusvoimaa rajoittaa se, että automaatti joutuu hyväksymään tai hylkäämään syötteen yhden lukukerran jälkeen ja toisaalta se, että automaatti ei voi hyödyntää työmuistia laskentansa aikana.

Tässä luvussa määritellään Turingin kone, joka laajentaa äärellistä automaattia molemmissa edellä mainituissa suhteissa. Turingin kone on äärellinen kone, joka suorittaa laskentaa lukien soluista (yhden merkin yksikkö) koostuvaa nauhaa. Turingin koneen nauha on rajoitettu vasemmalta ja rajoittamaton oikealta. Laskennan aikana jokainen nauhan solu sisältää joko merkin merkistöstä Σ tai tyhjää solua indikoivan merkin $\Box$. Turingin koneen M laskenta muodostuu yksittäisistä peräkkäisistä laskenta-askeleista. Yhden laskenta-askeleen aikana kone M suorittaa seuraavat toiminnot sisäisen tilansa s ja lukupään lukeman merkin a perusteella,

1. korvaa merkin a jollakin a',
2. siirtää lukupäätä yhden solun verran oikealle (1), vasemmalle (-1), tai pitää lukupään paikallaan (0),
3. siirtyy tilasta s tilaan s'.

Määritelmä 1.1. Olkoon Σ äärellinen merkistö. Turingin kone M on

$$M = (Q, \Sigma, \delta, s_0, s_+, s_-),$$

missä

- Q on äärellinen tilajoukko,
- $s_0, s_+, s_- \in Q$, missä s_0 on koneen M alkutila, s_+ M:n hyväksyvä lopputila ja s_0 on M:n hylkäävä lopputila,
- Kuvaus δ on M:n tilasiirtymä (osittainen) funktio:

$$\delta\colon (Q \setminus \{s_+, s_-\} \times \Sigma \cup \{\alpha, \Box\}) \longrightarrow (Q \times \Sigma \cup \{\alpha, \Box\} \times \{-1, 0, 1\}),$$

missä $\alpha \notin \Sigma$ on syötenauhan vasemman puolen alkumerkki ja $\square \notin \Sigma$ on tyhjää solua indikoiva merkki. Kun $\delta(q, a) = (q', a', i)$, niin kuvauksen δ oletetaan toteuttavan ehdot: $a = \alpha$ joss $a' = \alpha$ eli alkumerkkiä α ei voi korvata toisella ja jos $a = \alpha$, niin $i \neq -1$ eli lukupää ei voi tippua nauhan vasemmalta reunalta.

Edellisen määritelmän mukaista Turingin konetta kutsutaan *deterministiseksi.* Nimitys tulee siitä, että missä tahansa laskennan vaiheessa koneella on korkeintaan yksi mahdollinen seuraava laskenta-askel, eli jokaisella $s \in Q \setminus \{s_+, s_-\}$ ja $a \in \Sigma \cup \{\alpha, \square\}$, $\delta(s, a)$ on yksikäsitteinen, jos se on määritelty.

Jos Turingin koneen määritelmässä luovutaan relaation δ funktionaalisuudesta, niin tällöin vastaavaa Turingin konetta kutsutaan *epädeterministiseksi.* Tällöin paria (s, a) kohti voi olemassa enemmän kuin yksi kolmikko (s', a', i) siten, että $((s, a), (s', a', i)) \in \delta$.

Määritelmä 1.2. Olkoon $w = w_0 \dots w_{k-1} \in \Sigma^*$. Turingin koneen M laskenta syötteellä w tapahtuu seuraavasti:

1. Sana w kirjoitetaan M:n nauhalle alkaen merkistä w_0 siten, että α on vasemmalta katsoen nauhan ensimmäisessä solussa, w_0 sitä seuraavassa jne. Merkin w_{k-1} jälkeen tulevissa soluissa on merkki $\square$.

2. Koneen M laskenta syötteellä w alkaa tilanteesta, jossa M:n lukupää lukee merkkiä α ja M on tilassa s_0. Alkutilanteesta lähtien laskenta etenee askel kerrallaan kuvauksen δ mukaisesti.

3. Koneen M laskenta pysähtyy, jos M päätyy laskentansa aikana tilanteeseen (tilassa s lukiessaan merkin a), jossa arvoa $\delta(a, s)$ ei ole määritelty, tai vastaavasti $((s, a), (s', a', i)) \notin \delta$ kaikilla (s', a', i). Deterministisen Turingin koneen M laskennan sanotaan hyväksyvän syötesanan w, jos sen laskenta pysähtyy ja se on pysähtyessään tilassa s_+ ja hylkäävän w:n jos M pysähtyy tilaan s_-.

Epädeterministinen Turingin kone ei kuvaa yhtä laskentaa, vaan paremminkin perheen erilaisia deterministisiä laskentoja, joista osa voi pysähtyä hyväksyvään tilaan, osa hylätä syötteen ja osa olla pysähtymättä. Epädeterministisen koneen M laskentaa syötteellä w voidaan kuvata puulla, jonka juuren muodostaa kone M alkutilanteessa syötteellä w. Tässä puussa haarautuminen vastaa sitä, että kyseisessä laskennan vaiheessa koneella M on enemmän kuin yksi seuraava laskenta-askel.

Määritelmä 1.3. Olkoot Σ merkistö ja $L \subseteq \Sigma^*$ kieli.

- Deterministinen Turingin kone M *tunnistaa* kielen L, jos jokaisella $w \in \Sigma^*$: $w \in L$ joss M hyväksyy sanan w ja $w \notin L$ joss M hylkää sanan w.
- Epädeterministinen Turingin kone M *tunnistaa* kielen L, jos jokaisella $w \in \Sigma^*$: $w \in L$ joss ainakin yksi M:n laskennoista syötteellä w pysähtyy tilaan s_+.
- Kun $f\colon \mathbb{N} \to \mathbb{N}$, niin Turingin koneen M sanotaan tunnistavan kielen L ajassa f, jos M tunnistaa kielen L ja jokaisella $w \in \Sigma^*$ M:n laskenta (laskennat) pysähtyy korkeintaan $f(|w|)$:n laskenta-askeleen jälkeen.
- Kun $f\colon \mathbb{N} \to \mathbb{N}$, niin Turingin koneen M sanotaan tunnistavan kielen L tilassa f, jos M tunnistaa kielen L ja jokaisella $w \in \Sigma^*$ M:n laskenta käyttää korkeintaan $f(|w|)$ solua (eli lukupää pysyy etäisyydellä $f(|w|)$ nauhan vasemmasta reunasta).

Luokka PTIME (NPTIME) sisältää ne kielet L, joille on olemassa deterministinen (epädeterministinen) Turingin kone M ja polynomi $p \in \mathbb{N}[x]$ siten, että M tunnistaa kielen L ajassa p. Vastaavasti luokka PSPACE (NPSPACE) sisältää ne kielet L, joille on olemassa polynomi p ja deterministinen (epädeterministinen) Turingin kone M, joka tunnistaa kielen L tilassa p. Koska deterministinen Turingin kone on erikoistapaus epädeterministisestä, saadaan suoraan, että PTIME $\subseteq$ NPTIME. Vastaavasti Turingin kone joka tunnistaa kielen L ajassa f, tunnistaa selvästi kielen L myös tilassa f, PTIME $\subseteq$ PSPACE. Voidaan myös näyttää, että NPTIME $\subseteq$ PSPACE ja NPSPACE=PSPACE.

Huomautus 1.4. Epädeterministisen Turingin koneen M_0 tapauksessa on täysin mahdollista, että M_0 tunnistaa kielen $L \in$ NPTIME ajassa f, mutta silti joillakin syötteillä osa M:n laskennoista ei pysähdy ollenkaan. Laskennan vaativuusteorian kontekstissa voidaan huoletta rajata tällaiset Turingin koneet tarkastelun ulkopuolelle. Toisin sanoen voidaan oletetaan, että tarkastellaan ainoastaan sellaisia Turingin koneita, joiden kaikki laskennat ovat äärellisiä ja päätyvät aina joko tilaan s_+ tai s_-. Tämä oletus ei muuta vaativuusluokkia, sillä esimerkiksi koneesta M_0 saadaan oletukset täyttävä polynomiaikainen kone M_1, joka syötteellä w simuloi konetta M_0, mutta hylkää syötteen w, jos koneen M_0 simulointi ei ole pysähtynyt tilaan s_+ $f(|w|)$ laskenta-askeleen jälkeen.

Huomautus 1.5. Tässä luvussa esitetystä Turingin koneen määritelmästä on olemassa paljon erilaisia versioita. Voidaan esimerkiksi sallia, että koneella on käytössään erikseen syötenauha(t) ja erillisiä työnauhoja. *Moninauhaisia* Turingin koneita voidaan simuloida yksinauhalisella koneella siten, että simuloiva kone käyttää aikaa $p(f(|w|))$ syötteellä w, missä $p(x) \in \mathbb{N}[x]$ on polynomi ja $f\colon \mathbb{N} \to \mathbb{N}$ on moninauhaisen koneen laskenta-aikaa rajoittava funktio.

Erityisesti tästä seuraa, että luokat PTIME, NPTIME ja PSPACE ovat suljettuja myös polynomisessa ajassa toimivien moninauhaisten Turingin koneiden suhteen.

1.2 Kiintopistelogiikat IFP ja PFP

Tässä luvussa määritellään Inflatorinen IFP ja Partiaalinen PFP kiintopistelogiikka.

Olkoon M äärellinen ja epätyhjä joukko ja F kuvaus

$$F\colon \mathcal{P}(M) \to \mathcal{P}(M).$$

Kuvaus F indusoi jonon joukon M osajoukkoja

$$\emptyset, F(\emptyset), F(F(\emptyset)), \ldots.$$

Merkitään näitä joukkoja $F_0, F_1, \ldots$ eli $F_0 = \emptyset$ ja $F_{n+1} = F(F_n)$. Huomataan, että jos on olemassa $n_0 \geq 0$, jolle pätee $F_{n_0+1} = F_{n_0}$, niin tällöin $F_m = F_n$ kaikilla $m \geq n_0$. Tällaista joukkoa F_{n_0} merkitään F_∞ ja kutsutaan kuvauksen F *kiintopisteeksi*. Jos tälläistä n_0 ei ole olemassa, eli kuvauksella F ei ole kiintopistettä, asetetaan $F_\infty = \emptyset$. Kuvaus F on *inflatorinen*, jos kaikilla $X \subseteq M$ pätee $X \subseteq F(X)$.

Lemma 1.6. *Olkoon M äärellinen joukko ja $F\colon \mathcal{P}(M) \to \mathcal{P}(M)$.*

a) Jono $(F_n)_{n\geq 0}$ on periodinen, eli on olemassa $m < 2^{|M|}$ ja $l \geq 1$ siten, että $F_k = F_{k+l}$ kaikilla $k \geq m$.

b) Jos kuvauksella F on kiintopiste, niin tällöin $F_\infty = F_{2^{|M|}-1}$.

c) Jos F on inflatorinen, niin kuvauksella F on kiintopiste ja $F_\infty = F_{|M|}$.

Todistus. Koska $|\mathcal{P}(M)| = 2^{|M|}$, niin on olemassa $m < 2^{|M|}$ ja $l \geq 1$ siten, että $F_m = F_{m+l}$. Tällöin $F_{m+1} = F(F_m) = F(F_{m+l}) = F_{m+1+l}$ ja yleisesti siis $F_{m+i} = F_{m+i+l}$, mistä väite a) seuraa.

Osoitetaan sitten väite b). Olkoot m ja l kuten kohdassa a). Jos $F_m = F_{m+1}$, niin tällöin $F_m = F_{2^{|M|-1}} = F_\infty$. Jos taas $F_m \neq F_{m+1}$, niin tällöin kohdan a) perusteella kaikilla $s \geq 0$ pätee

$$F_{m+sl} = F_m \neq F_{m+1} = F_{m+sl+1},$$

joten kuvauksella F ei ole kiintopistettä.

Tarkastellaan lopuksi väitettä c). Nyt $F_0 \subseteq F_1 \subseteq F_2 \subseteq \cdots \subseteq M$, joten $F_i \neq F_{i+1}$ jos F_i on joukon F_{i+1} aito osajoukko. Jonossa voi siis olla korkeintaan $|M|$ eri jäsentä. $\square$

Määritelmä 1.7. Olkoot $\varphi(x_1,\dots,x_k,R)$ $\tau \cup \{x_1,\dots,x_k,R\}$-lause (esim $\varphi \in \mathrm{FO}$), missä $\sharp(R) = k$ ja $\mathfrak{A}$ τ-malli. Tällöin lause φ ja malli $\mathfrak{A}$ indusoivat kuvauksen

$$F^{\varphi} \colon \mathcal{P}(\mathrm{Dom}(\mathfrak{A})^k) \to \mathcal{P}(\mathrm{Dom}(\mathfrak{A})^k),$$

missä $F^{\varphi}(X) = \{(a_1,\dots,a_k) \in \mathrm{Dom}(\mathfrak{A})^k \mid \mathfrak{A} \models \varphi[a_1/x_1,\dots,a_k/x_k,X/R]\}$.

Esimerkki 1.8. Olkoon $\mathcal{G} = (G, E^{\mathcal{G}})$ verkko ja $\varphi_0(x,y,X)$ lause

$$\varphi_0(x,y,X) = (E(x,y) \vee \exists z(X(x,z) \wedge E(z,y))),$$

missä siis x,y vastaa edellisen määritelmän muuttujia x_1 ja x_2 ja X relaatiosymbolia R. Koska $F_0^{\varphi_0} = \emptyset$. niin helposti nähdään, että $F_1^{\varphi_0} = E^{\mathcal{G}}$ ja

$$F_2^{\varphi_0} = F(F_1^{\varphi_0}) = E^{\mathcal{G}} \cup \{(a,b) \mid (E^{\mathcal{G}}(a,c) \text{ ja } E^{\mathcal{G}}(c,b)) \text{ jollakin } c\}.$$

Induktiolla voidaan näyttää, että

$$F_n^{\varphi_0} = \{(a,b) \mid a\text{:sta on} \leq n \text{ mittainen polku } b\text{:hen}\},$$

joten

$$F_\infty^{\varphi_0} = \{(a,b) \mid a\text{:sta on polku } b\text{:hen}\}.$$

Määritellään nyt Inflatorinen IFP ja Partiaalinen PFP kiintopistelogiikka.

Määritelmä 1.9. Logiikoiden IFP ja PFP lauseiden joukko saadaan laajentamalla ensimmäisen kertaluvun logiikan lauseiden muodostussääntöjä ehdolla:

- Jos φ on logiikan IFP aakkoston $\tau \cup \{x_1,\dots,x_k,R\}$ lause, missä $\sharp(R) = k$ ja $\bar{t} = t_1,\dots t_k$ ovat termejä, niin $[\mathrm{IFP}_{\bar{x},R}\,\varphi]\bar{t}$ on aakkoston $(\cup_{1\leq i\leq k}\tau(t_i)) \cup \tau$ lause.
- Jos φ on logiikan PFP aakkoston $\tau \cup \{x_1,\dots,x_k,R\}$ lause, missä $\sharp(R) = k$ ja $\bar{t} = t_1,\dots t_k$ ovat termejä, niin $[\mathrm{PFP}_{\bar{x},R}\,\varphi]\bar{t}$ on aakkoston $(\cup_{1\leq i\leq k}\tau(t_i)) \cup \tau$ lause.

Lauseiden $[\mathrm{IFP}_{\bar{x},R}\,\varphi]\bar{t}$ ja $[\mathrm{PFP}_{\bar{x},R}\,\varphi]\bar{t}$ tulkinta σ-mallissa $\mathfrak{A}$, jolle $(\cup_{1\leq i\leq k}\tau(t_i)) \cup \tau \subseteq \sigma$, määritellään seuraavasti:

$$\mathfrak{A} \models [\mathrm{IFP}_{\bar{x},R}\,\varphi]\bar{t} \text{ joss } (t_1^{\mathfrak{A}},\dots,t_k^{\mathfrak{A}}) \in F_\infty^{R(\bar{x})\vee\varphi}.$$

On tärkeää huomata, että kuvaus $F^{R(\bar{x})\vee\varphi}$ on aina inflatorinen, joten kiintopiste on olemassa. Toisaalta

$$\mathfrak{A} \models [\mathrm{PFP}_{\bar{x},R}\,\varphi]\bar{t} \text{ joss } (t_1^{\mathfrak{A}},\dots,t_k^{\mathfrak{A}}) \in F_\infty^{\varphi},$$

eli ei ole takeita siitä, että kuvauksella F on kiintopistettä. Tässä tapauksessahan asetettiin $F_\infty^{\varphi} = \emptyset$.

Esimerkki 1.10. Verkoilla $\{E, x, y\}$-lause ψ

$$\psi = [\mathrm{IFP}_{x,y,X}\,(E(x,y) \vee \exists z(X(x,z) \wedge E(z,y)))]xy,$$

ilmaisee, että solmujen x ja y välillä on polku. Siispä yhtenäisten verkkojen luokka voidaan määritellä logiikassa IFP lauseella $\forall x \forall y((x \neq y) \to \psi)$ (ja verkon aksioomilla).

Esimerkissä 1.10 kiintopisteoperaattoria käytettiin relaation $E^{\mathcal{G}}$ transitiivisen sulkeuman määrittelemiseen. Yleisesti kun $\mathfrak{A}$ on τ-malli ja $R \subseteq \mathrm{Dom}(\mathfrak{A})^k$, niin relaatiota R sanotaan *määriteltäväksi* (logiikan $\mathcal{L}$ lauseella) (parametrein $\overline{b}$), jos on olemassa $\varphi(\overline{x}, \overline{y}) \in \mathcal{L}[\tau \cup \{\overline{x}, \overline{y}\}]$, jolle

$$R = \{\overline{a} \in \mathrm{Dom}(\mathfrak{A})^k \mid \langle \mathfrak{A}, \overline{a}, \overline{b} \rangle \models \varphi\}.$$

Seuraava esimerkki osoittaa, että logiikassa IFP määriteltävien kaksipaikkaisten relaatioiden joukko (mallissa $\mathfrak{A}$) on suljettu transitiivisen sulkeuman suhteen:

Esimerkki 1.11. Olkoon $\mathfrak{A}$ malli, $\varphi(x,y) \in \mathrm{IFP}$ ja R kaavan $\varphi(x,y)$ määrittelemä relaatio mallissa $\mathfrak{A}$. Tällöin

$$\psi(x,y) = [\mathrm{IFP}_{x,yX}\,(\varphi(x,y) \vee \exists z(X(x,z) \wedge E(z,y)))]xy,$$

määrittelee relaation R transitiivisen sulkeuman $TC(R)$.

Esimerkki 1.12. Kun $\tau = \{<, \mathrm{Suc}, \min, \max\}$ niin lause

$$\neg[\mathrm{IFP}_{x,X}(x = \min \vee \exists y \exists z(X(y) \wedge \mathrm{Suc}(y,z) \wedge \mathrm{Suc}(z,x)))]\max$$

on totta täsmälleen parillisen kokoisissa järjestyksissä. Tässä siis kiintopiste tulee sisältämään ne alkiot, joilla on parillinen määrä edeltäjiä mallin järjestyksen suhteen.

Määritelmä 1.13. Olkoot τ aakkosto, K luokka τ-malleja ja $\mathcal{L}$ ja $\mathcal{L}'$ logiikoita.

- Luokan K sanotaan olevan aksiomatisoituva (tai määriteltävä) logiikassa $\mathcal{L}$ jos on olemassa lause $\varphi \in \mathcal{L}[\tau]$, jolle $K = \{\mathfrak{A} \in \mathrm{Str}(\tau) \mid \mathfrak{A} \models \varphi\}$. Tällöin voidaan käyttää myös merkintää $K = \mathrm{Mod}(\varphi)$.
- Merkitään $\mathcal{L} \leq \mathcal{L}$, jos kaikilla τ ja lauseilla $\varphi \in \mathcal{L}[\tau]$ on olemassa $\varphi' \in \mathcal{L}'[\tau]$ siten, että $\mathrm{Mod}(\varphi) = \mathrm{Mod}(\varphi')$. Logiikat $\mathcal{L}$ ja $\mathcal{L}'$ ovat ekvivalentit $\mathcal{L}' \equiv \mathcal{L}$ jos $\mathcal{L}' \leq \mathcal{L}$ ja $\mathcal{L} \leq \mathcal{L}'$. Merkitään $\mathcal{L} < \mathcal{L}'$, jos $\mathcal{L} \leq \mathcal{L}'$, mutta $\mathcal{L}' \not\leq \mathcal{L}$.

Lause 1.14. *1.* IFP $\leq$ PFP.

2. FO $<$ IFP.

3. PFP $< \mathcal{L}^{\omega}_{\infty\omega}$

Todistus. Ensimmäinen väite seuraa siitä, että lauseet $[\mathrm{IFP}_{\overline{x},R}\,\varphi]\overline{t}$ ja $[\mathrm{PFP}_{\overline{x},R}\,(R(\overline{x})\vee\varphi)]\overline{t}$ ovat loogisesti ekvivalentteja. Toinen väite seuraa Esimerkistä 1.10, sillä verkkojen yhtenäisyys ei ole määriteltävä logiikassa FO. Viimeinen väite on harjoitustehtävä. □

1.3 Mallit Turingin koneen syötteenä

Olkoon τ äärellinen aakkosto, jossa ei esiinny funktiosymboleja. Büchin lauseessa tarkastelu rajoitettiin logiikan puolella sanamalleille, jolloin yhteys sanojen ja mallien on ongelmaton. Kun $\mathfrak{A}$ on mielivaltainen τ-malli, niin ei ole olemassa kanonista tapaa koodata mallia $\mathfrak{A}$ Turingin koneen syötteeksi. Toisaalta jos oletetaan, että $\mathfrak{A}$ on järjestetty, niin tällöin mallin relaatiot voidaan koodata binäärisanoiksi mallin $\mathfrak{A}$ järjestyksen indusoiman sanakirjajärjestyksen suhteen.

Tästä eteenpäin oletetaan, että kaikki tarkasteltavat aakkostot τ sisältävät symbolit $\tau_0 = \{<, S, \min, \max\}$, jotka on tulkittu kaikissa τ-malleissa $\mathfrak{A}$ siten, että $<^{\mathfrak{A}}$ on $\mathrm{Dom}(\mathfrak{A})$: n lineaarijärjestys, $S^{\mathfrak{A}}$ tämän järjestyksen seuraajarelaatio ja $\min^{\mathfrak{A}}$ ja $\max^{\mathfrak{A}}$ järjestyksen $<^{\mathfrak{A}}$ pienin ja suurin alkio. Järjestettyjen äärellisten τ-mallien luokkaa merkitään $\mathcal{O}[\tau]$. Olkoon $\mathfrak{A} \in \mathcal{O}[\tau]$ ja $\mathrm{Card}(\mathfrak{A}) = n \in \mathbb{N}^*$. Voidaan olettaa (isomorfiaa vaille), että $\mathrm{Dom}(\mathfrak{A}) = \{0, \ldots, n-1\}$ ja, että $<^{\mathfrak{A}}$ on tulkittu luonnollisesti. Nyt siis $\min^{\mathfrak{A}} = 0$ ja $\max^{\mathfrak{A}} = n-1$. Merkitään $\tau = \tau_0 \cup \tau_1$, missä $\tau_1 = \{R_1, \ldots, R_k, c_1, \ldots, c_l\}$ sisältää siis ne aakkoston τ relaatio- ja vakiosymbolit, joiden tulkinta voidaan valita vapaasti.

Määritellään seuraavasti moninauhainen versio Turingin koneesta, joka on räätälöity hyväksymään syötteikseen τ-malleja. Kun $\tau = \tau_0 \cup \tau_1$, missä $\tau_1 = \{R_1, \ldots, R_k, c_1, \ldots, c_l\}$, niin aakkostoa τ vastaa moninauhainen Turingin kone jolla on $k+l+1$ syötenauhaa ja m työnauhaa jollakin $m \in \mathbb{N}$. Jokainen nauha on rajoitettu vasemmalta ja rajoittamaton oikealta puolelta. Nauhojen solut numeroidaan siten, että ensimmäinen solu, jossa on siis merkki α, numeroidaan luvulla -1, ja tätä seuraavat luvuilla 0,1,2 jne. Jokainen syötenauhoista sisältää sanan, jonka loppumista indikoi merkki ω syötettä välittömästi seuraavassa solussa. Jokaisella nauhalla on oma lukupää, jonka toiminta on riippumatonta toisten lukupäiden toiminnasta. Syötenauhojen lukupäät voivat ainoastaan lukea syötettä, kun taas työnauhoilla toimivat lukupäät voivat myös korvata lukemiansa merkkeja toisilla (kuten Turingin koneen alkuperäisessä määritelmässä). Turingin koneen M aakkosto on $\Sigma = \{0,1\}$ ja tyhjää solua tarkoittava merkki $\square$ samastetaan merkin 0 kanssa.

Aakkostoa τ vastaavan Turingin koneen M laskentaa ohjaa relaatio δ joka sisältää alkioinaan pareja (t, t'), missä

$$t = (s, a_0, \ldots, a_{k+l}, c_1, \ldots, c_m)$$

ja

$$t' = (s', c'_1, \ldots, c'_m, i_0, \ldots, i_{k+l+m}).$$

Pari (t, t') määrittää koneen M yhden mahdollisen laskenta-askeleen seuraavasti: jos M on tilassa s, syötenauhojen lukupäiden alla on merkit $a_0, \ldots, a_{k+l}$ ja työnauhojen lukupäät lukevat merkit $c_1, \ldots, c_m$, niin tällöin M korvataan merkit $c_1, \ldots, c_m$ merkeillä $c'_1, \ldots, c'_m$, siirtää lukupäätä nauhalla j luvun i_j mukaisesti ($i_j \in \{-1, 0, 1\}$ aivan kuten Turingin koneen edellisessä määritelmässä) ja lopuksi M siirtyy tilaan s'. Relaation δ oletetaan toteuttavan seuraavat ehdot:

- $a_0, \ldots, a_{k+l} \in \{0, 1, \alpha, \omega\}$ ja $c_1, \ldots, c_m, c'_1, \ldots, c'_m \in \{0, 1, \alpha\}$.
- $i_0, \ldots, i_{k+l+m} \in \{-1, 0, 1\}$.
- Jos $a_j = \alpha$, niin $i_j \neq -1$, eli lukupää ei voi tippua nauhan vasemmalta reunalta.
- Jos $a_j = \omega$, niin $i_j \neq 1$, eli syötenauhojen lukupäät voivat liikkua vain syötesanan alueella nauhalla.
- Jos $c_j = \alpha$, niin $i_{k+l+j} \neq -1$ ja $c'_j = \alpha$, eli työnauhojen lukupäät eivät voi tippua nauhan vasemmalta reunalta, eikä merkkiä α voi korvata toisella.
- Jos $c_j \in \{0, 1\}$, niin $c'_j \in \{0, 1\}$.
- $s \neq s_+$ ja $s \neq s_-$, eli laskenta pysähtyy jos kone siirtyy tilaan s_+ tai s_-.

Seuraavaksi määritellään miten malli $\mathfrak{A}$ syötetään Turingin koneelle M. Olkoon $\mathfrak{A}$ τ-malli, missä siis $\tau = \tau_0 \cup \tau_1$ ja $\tau = \{R_1, \ldots, R_k, c_1, \ldots, c_l\}$. Malli $\mathfrak{A}$ syötetään koneelle M kirjoittamalla seuraavat syötesanat koneen M $k + l$ syötenauhoille (nauhat numeroidaan luvuilla $0, 1, \ldots, k + l$). Nauhalle 0 kirjoitetaan sana

$$1^n = \underbrace{1 \cdots 1}_{n},$$

kun $n = \mathrm{Card}(\mathfrak{A})$. Nauhat $1 \leq i \leq k$ sisältävät informaation relaatiosta $R_1, \ldots, R_k$ koodattuna seuraavalla tavalla. Oletetaan, että $\sharp(R) = r$ and $R^{\mathfrak{A}} \subseteq \{0, 1, \ldots, n-1\}^r$. Kun $j < n^r$, niin soluun j kirjoitetaan merkki 1 joss j:s r-jono (merkitään $|j|_r$) joukon $\{0, 1, \ldots, n-1\}^r$ sanakirjajärjestyksessä kuuluu relaatioon $R^{\mathfrak{A}}$. Tosin sanoen, kun $j < n^r$, etsitään luvun j esitys muodossa:

$$j = j_1 n^{r-1} + j_2 n^{r-2} + \cdots + j_r,$$

missä $0 \leq j_i < n$ ja asetetaan $(j_1, j_2, \ldots, j_t) = |j|_r$.

Kun $1 \leq i \leq l$, nauhalle $(k+i)$ kirjoitetaan luvun $c_i^{\mathfrak{A}}$ binääriesitys ilman merkitsemättömiä nollia.

Määritelmä 1.15.

- Turingin kone M käynnistetään syötteellä $\mathfrak{A}$ seuraavasti: koneen M syötenauhoille kirjoitetaan informaatio mallin $\mathfrak{A}$ relatiosta ja vakiosta, työnauhat ovat tyhjiä eli jokaisessa solussa on tyhjää solua indikoiva merkki 0 ja jokaisen nauhan lukupää asetetaan solun 0 kohdalle.

- Syötteen $\mathfrak{A}$ hyväksyminen ja hylkääminen, kielen tunnistaminen jne. määritellään kuten luvussa 1.1.

- Kun $K \subseteq \mathcal{O}[\tau]$, niin koneen M sanotaan tunnistavan luokan K ajassa f, jos M tunnistaa luokan K ja jokaisella $\mathfrak{A} \in \mathcal{O}[\tau]$, koneen M laskenta syötteellä $\mathfrak{A}$ pysähtyy korkeintaan $f(\mathrm{Card}(\mathfrak{A}))$ laskenta-askeleen jälkeen.

- Koneen M sanotaan tunnistavan kielen $K \subseteq \mathcal{O}[\tau]$ tilassa f, jos M tunnistaa tunnistaa luokan K jos jokaisella $\mathfrak{A} \in \mathcal{O}[\tau]$, koneen M laskenta syötteellä $\mathfrak{A}$ käyttää korkeintaan $f(\mathrm{Card}(\mathfrak{A}))$ ensimmäistä solua jokaisella työnauhalla.

Määritelmä 1.16. Olkoon $K \subseteq \mathcal{Q}[\tau]$.

- K kuuluu vaativuusluokkaan PTIME (NPTIME) joss on olemassa deterministinen (epädeterministinen) moninauhainen Turingin kone M ja polynomi $p \in \mathbb{N}[x]$ siten että M hyväksyy luokan K ajassa p.

- K kuuluu vaativuusluokkaan PSPACE joss on olemassa deterministinen moninauhainen Turingin kone M ja polynomi $p \in \mathbb{N}[x]$ siten että M hyväksyy luokan K tilassa p.

1.4 Vaativuusluokkien PTIME, NPTIME ja PSPACE looginen karakterisointi

Palautetaan mieleen, että Σ_1^1 on toisen kertaluvun alilogiikka, joka sisältää lauseet muotoa

$$\exists R_0 \ldots, \exists R_{l-1}\varphi,$$

missä R_i ovat mielivaltaisia relaatiosymboleita (paikkalukua ei ole rajoitettu) ja $\varphi \in \mathrm{FO}$. Kun K on luokka järjestettyjä τ-malleja, eli $K \subseteq \mathcal{O}[\tau]$, ja $\mathcal{L}$ on logiikka, niin merkitään $K \in \mathcal{L}$ jos on olemassa lause $\varphi \in \mathcal{L}[\tau]$ jolle $K = \mathrm{Mod}(\varphi)$. Tämän luvun päätulos voidaan nyt muotoilla seuraavasti:

Lause 1.17. *Olkoot τ äärellinen aakkosto joka ei sisällä funktiosymboleja ja $K \subseteq \mathcal{Q}[\tau]$. Tällöin*

$$\begin{aligned} K \in \mathrm{PTIME} \quad &\textit{joss} \quad K \in \mathrm{IFP}, \\ K \in \mathrm{NPTIME} \quad &\textit{joss} \quad K \in \Sigma_1^1, \\ K \in \mathrm{PSPACE} \quad &\textit{joss} \quad K \in \mathrm{PFP}. \end{aligned}$$

1.4.1 Totuusrelaation laskennallinen vaativuus

Tässä luvussa osoitetaan toinen puoli lauseen 1.17 väitteestä.

Lause 1.18. *Olkoon $K \subseteq \mathcal{O}[\tau]$.*

a) Jos $K \in \mathrm{IFP}$, niin $K \in \mathrm{PTIME}$,

b) Jos $K \in \Sigma_1^1$, niin $K \in \mathrm{NPTIME}$,

c) Jos $K \in \mathrm{PFP}$, niin $K \in \mathrm{PSPACE}$.

Todistus. Aloitetaan väitteestä a). Väite todistetaan induktiolla lauseen $\varphi \in$ IFP rakenteen suhteen. Osoitetaan väite samanaikaisesti lauseille φ, joiden aakkostolle σ pätee, että

$$\sigma \setminus \tau = \{x_1, \ldots, x_l, P_1, \ldots, P_r\}, \tag{1}$$

eli σ sisältää τ-symbolien lisäksi äärellisen monta uutta vakio- ja relaatiosymbolia. Oletetaan, että φ on atomilause ja muotoa $R(x, y)$. Osoitetaan, että on olemassa deterministinen polynomiaikainen Turingin kone M joka tunnistaa luokan

$$\{\langle \mathfrak{A}, i, j\rangle \mid \mathfrak{A} \in \mathcal{O}[\tau],\ (i, j) \in R^{\mathfrak{A}}\}.$$

Olkoon $\langle \mathfrak{A}, i, j\rangle \in \mathcal{O}[\tau \cup \{x, y\}]$, missä $\mathrm{Dom}(\mathfrak{A}) = \{0, 1, \ldots, n-1\}$. Tieto siitä, että kuuluuko pari (i, j) relaatioon $R^{\mathfrak{A}}$ löytyy $(in + j)$:nnestä solusta relaatiota R vastaavalta syötenauhalta ja luvut i ja j ovat esitettyinä binäärimuodossa vakioita x ja y vastaavilta syötenauhoilta. Nyt on selvää, että kysymys kuuluuko (i,j) relaatioon $R^{\mathfrak{A}}$ voidaan selvittää polynomisessa ajassa ja siksi luokka (1) voidaan tunnistaa polynomisessa ajassa. Muut atomilauseet voidaan käsitellä vastaavasti.

Oletetaan, että $\varphi = \neg\psi$ ja induktio-oletus pätee lauseelle ψ koneella M. Tällöinhän $\mathrm{Mod}(\varphi) = \mathcal{O}[\sigma] \setminus \mathrm{Mod}(\psi)$ ja luokan $\mathrm{Mod}(\varphi)$ tunnistama Turingin kone saadaan koneesta M vaihtamalla tilat s_+ ja s_- keskenään.

Oletetaan, että $\varphi = \psi \vee \theta$ ja väite pätee lauseille ψ ja θ. Induktio-oletuksen mukaan on olemassa koneet M_ψ ja M_θ, jotka tunnistavat luokat $\mathrm{Mod}(\psi)$ ja $\mathrm{Mod}(\theta)$. Määritellään kone M siten, että M simuloi ensin konetta M_ψ ja sen jälkeen konetta M_θ ja hyväksyy syötteen, jos ainakin toinen simuloitavista koneista hyväksyy syötteen. Simulointien välillä M palauttaa lukupäät alkutilanteeseen ja korvaa merkit 1 merkillä 0 työnauhoilla (eli tyhjäntää työnauhat). Nämä operaatiot voidaan tehdä polynomisessa ajassa käyttämällä apuna ylimääräisiä työnauhoja. Koska koneet M_ψ ja M_θ ovat polynomiaikaisia on myös M polynomiaikainen.

Oletetaan, että $\varphi = \exists x\psi$. Induktio-oletuksen perusteella on olemassa kone M_ψ, joka tunnistaa luokan $\mathcal{O}[\sigma \cup \{x\}]$, missä φ on σ-lause. Määritellään kone M joka tunnistaa luokan $\mathrm{Mod}(\varphi)$. Oletetaan, että syöte on σ-malli $\mathfrak{A}$, missä $\mathrm{Dom}(\mathfrak{A}) = \{0,\ldots,n-1\}$. Kun $i = 0,\ldots,n-1$ kone M kirjoittaa i:n binääriesityksen työnauhalle W ja simuloi konetta M_ψ syötteellä $\langle \mathfrak{A}, i\rangle$. Nyt kone M hyväksyy mallin $\mathfrak{A}$ jos jollakin i koneen M_ψ simulointi hyväksyy syöteen $\langle \mathfrak{A}, i\rangle$. Selvästi koneen M laskenta on polynomiaikainen induktio-oletuksen perusteella. Koska syöte nauhalla W ei päädy merkkiin ω, kone M joutuu pitämään i:n binääriesityksen pituuden muistissaan simuloidessaan konetta M_ψ. Tämä voidaan tehdä käyttämällä uutta työnauhaa W', jolle M kirjoittaa sanan 1^l, missä siis l vastaa i:n binääriesityksen pituutta.

Oletetaan, että $\varphi = [\mathrm{IFP}_{\overline{x},R}\,\psi]\overline{t}$, missä R on r-paikkainen ja, että M_ψ tunnistaa luokan

$$\{\langle \mathfrak{A}, \overline{a}, R^{\mathfrak{A}}\rangle \mid \mathfrak{A} \in \mathcal{O}[\sigma],\ \langle \mathfrak{A}, \overline{a}, R^{\mathfrak{A}}\rangle \models \psi\}.$$

Luokan $\mathrm{Mod}(\varphi)$ tunnistama kone M sisältää alirutiinin I, joka käyttää työnauhoja W ja W': Jos I käynnistetään n^r:n mittainen sana nauhalla W, joka vastaa yksikäsitteistä r-paikkaista relaatiota R, ja nauha W' "tyhjänä", I kirjoittaa nauhalle W' sanan, joka vastaa relaatiota R'

$$R' = \{\overline{a} \mid \langle \mathfrak{A}, \overline{a}, R\rangle \models (R(\overline{x}) \vee \psi)\}$$

simuloimalla konetta M_ψ. Kone M toimii nyt seuraavasti: M asettaa aluksi $R = \emptyset$ ja laskee relaation R' alirutinin I avulla. Jos $R = R'$, M tarkastaa, että päteekö $R(\overline{t})$ vai ei ja hyväksyy tai hylkää syötteen vastaavasti. Muutoin M

asettaa $R = R'$, tyhjentää nauhan W' ja käynnistää uudelleen I:n. Tiedetään, että $R = R'$ saavutetaan korkeintaan n^r:n kutsun jälkeen alirutiinille I, joten M on polynomiaikainen.

Tarkastellaan seuraavaksi väitettä c). Riittää tarkastella tapausta $\varphi = [\mathrm{PFP}_{\overline{x},R}\,\psi]\overline{t}$, missä R on r-paikkainen ja, missä M_ψ tunnistaa luokan

$$\{\langle \mathfrak{A}, \overline{a}, R^{\mathfrak{A}}\rangle \mid \mathfrak{A} \in \mathcal{O}[\sigma],\ \langle \mathfrak{A}, \overline{a}, R^{\mathfrak{A}}\rangle \models \psi\}$$

polynomisessa tilassa. Palautetaan mieleen, että kaikilla malleilla $\mathfrak{A}$ pätee, että jos kuvauksella F^ψ on kiintopiste, niin tällöin $F^\psi_{2^{n^r}-1} = F^\psi_\infty$ ja, että muutoin asetettiin $F^\psi_\infty = \emptyset$. Luokan $\mathrm{Mod}(\varphi)$ tunnistava kone M toimii seuraavasti. Kone M asettaa laskentansa aluksi laskurin lukuun $2^{n^r} - 1$, eli kirjoittaa sanan 1^{n^r} työnauhalle. Sitten M toimii kuten logiikan IFP tapauksessa, eli laskee relaation R' relaatiosta R alirutiinin I' avulla

$$R' = \{\overline{a} \mid \langle \mathfrak{A}, \overline{a}, R\rangle \models \psi\}.$$

Kone M pitää kirjaa laskurin avulla, että alirutiinia I kutsutaan korkeintaan $2^{n^r} - 1$ kertaa. Tämän jälkeen M tarkastaa, että päteekö $R = R'$ ja $R(\overline{t})$ ja hyväksyy syötteen, jos molemmat ehdot ovat voimassa ja muuten hylkää syötteen. Selvästi kone M toimii polynomisessa tilassa induktio-oletuksen perusteella.

Tarkastellaan lopuksi väitettä b). Oletetaan, että $K = \mathrm{Mod}(\varphi)$, missä φ on muotoa

$$\exists R_1 \ldots, \exists R_l\, \psi,$$

$\sharp(R_i) = r_i$ ja $\psi \in \mathrm{FO}$. Logiikan IFP $\geq$ FO käsittelyn perusteella tiedetään, että luokka

$$\{\langle \mathfrak{A}, R_1^{\mathfrak{A}}, \ldots, R_l^{\mathfrak{A}}\rangle \mid \mathfrak{A} \in \mathcal{O}[\sigma],\ \langle \mathfrak{A}, R_1^{\mathfrak{A}}, \ldots, R_l^{\mathfrak{A}}\rangle \models \psi\}.$$

voidaan tunnistaa polynomisessa ajassa deterministisellä Turingin koneella M_ψ. Nyt luokka $\mathrm{Mod}(\varphi)$ voidaan tunnistaa epädeterministisesti seuraavasti. Kone M kirjoittaa epädeterministisesti $n^{r_1}, \ldots, n^{r_l}$:n mittaiset binäärisanat työnauhoille, jotka vastaavat siis yksikäsitteisiä relaatiota $A_1, \ldots, A_l$, missä $A_i \subseteq \{0, \ldots, n-1\}^{r_i}$. Tämän jälkeen M simuloi konetta M_ψ (deterministisesti) ja hyväksyy mallin $\mathfrak{A}$ jos ja vain jos $\langle \mathfrak{A}, A_1, \ldots, A_l\rangle \models \psi$. Epädeterministisen Turingin koneen määritelmän perusteella saadaan, että M hyväksyy syötteen $\mathfrak{A}$ joss ainakin yksi koneen M laskennoista syötteellä $\mathfrak{A}$ pysähtyy tilaan s_+ joss ainakin yhdellä jonolla $A_1, \ldots, A_l$ pätee

$$\langle \mathfrak{A}, A_1, \ldots, A_l\rangle \models \psi,$$

joss $\mathfrak{A} \models \exists R_1 \ldots, \exists R_l \psi$. $\square$

1.4.2 Turingin koneen laskennan looginen kuvailu

Tässä luvussa saatetaan loppuun lauseen 1.17 todistus.

Palautetaan mieleen, että aakkostojen τ oletetaan olevan muotoa $\tau = \tau_0 \cup \tau_1$, missä $\tau_0 = \{<, S, \min, \max\}$ ja $\tau_1 = \{R_1, \ldots, R_k, c_1, \ldots, c_l\}$. Tässä luvussa oletetaan, että aakkostot eivät sisällä vakiosymboleja, eli $\tau_1 = \{R_1, \ldots, R_k\}$ jollakin $k \in \mathbb{N}$, missä $\sharp(R_i) = r_i$ kun $i \in \{1, \ldots, k\}$. Aakkoston τ malleja vastaa nyt siis moninauhainen Turingin kone, jolla on $k+1$ syötenauhaa ja m työnauhaa jollakin $m \in \mathbb{N}$. Jatkossa luvulla n tarkoitetaan aina, jos ei toisin eksplisiittisesti määritellä, Turingin koneen syötteen $\mathfrak{A}$ mahtavuutta $n = \mathrm{Card}(\mathfrak{A})$.

Seuraavaksi määritellään ns. konfiguraation käsite, joka on täydellinen kuvaus Turing koneen M laskennan tilanteesta jossakin laskennan vaiheessa. Konfiguraatio CONF sisältää tiedot:

- Koneen M tila s,
- Koneen M työnauhojen sisältö,
- Koneen M lukupäiden paikka työ- ja syötenauhoilla.

Konfiguraatiota CONF kutsutaan hyväksyväksi, jos $s = s_+$. Konfiguraatio CONF on konfiguraation CONF$'$ seuraaja, jos M voi siirtyä konfiguraatiosta CONF konfiguraation CONF$'$ tekemällä yhden relaation δ mukaisen laskenta-askeleen. Teknisistä syistä sovitaan myös, että hyväksyvä konfiguraatio on itsensä seuraaja. Konfiguraatiota CONF kutsutaan n^d-rajoitetuksi, jos CONF kuvaa M:n laskennan tilannetta, jossa jokaisella työnauhalla W ainoastaan nauhan ensimmäiset n^d solua ovat epätyhjiä ja W:n lukupää on etäisyydellä $\leq n^d - 1$ alkumerkistä α. Erityisesti, jos koneen M tunnistaa luokan K tilassa x^d, niin tällöin kaikki M:n konfiguraatiot ovat x^d rajoitettuja.

Olkoon M (epädeterministinen) Turingin kone, joka hyväksyy tai hylkää jokaisen syötteen tilassa x^d, missä $d \in \mathbb{N}$. Nyt siis koneen M kaikki mahdolliset konfiguraatiot ovat x^d rajoitettuja. Koneen M laskennan loogisen kuvailun idea on, että M:n konfiguraatiot koodataan uusien relaatiomuuttujien avulla. Koodaamisesta selvitään teknisesti helpommin, jos syötteenä olevan $\mathfrak{A}$ koon $n = \mathrm{Card}(\mathfrak{A})$ oletetaan olevan riittävän suuri. Tehdään seuraavaksi muutamia teknisiä oletuksia:

- Oletetaan, että $d \geq r_i$ kaikilla $i \in \{1, \ldots, k\}$.
- Oletetaan, että $n > k+m$ ja että n on suurempi kuin koneen M tilojen joukko Q.

- Edellisen oletuksen perusteella voidaan olettaa, että koneen M tilojen joukko Q on muotoa $\{0,\ldots,q\}$ ja $s_0 = 0$.

Nämä oletukset eivät rajoita seuraavien tulosten yleisyyttä. Erityisesti rajoittuminen riittävän suuriin malleihin on harmiton oletus, koska jokaisella $n \in \mathbb{N}$ on olemassa isomorfiaa vaille vain äärellinen määrä τ-malleja $\mathfrak{A}$, joilla $\mathrm{Card}(\mathfrak{A}) \leq n$. Koska jokainen äärellinen malli $\mathfrak{A}$ voidaan määritellä lauseella $\varphi_{\mathfrak{A}} \in \mathrm{FO}$, voidaan mikä tahansa kokoelma K_0 τ-malleja, joilla $\mathrm{Card}(\mathfrak{A}) \leq n$, määritellä FO:n lauseella. Jos nyt

$$K \subseteq \mathcal{O}[\tau]_{\geq n} = \{\mathfrak{A} \in \mathcal{O}[\tau] \mid \mathrm{Card}(\mathfrak{A}) \geq n\}$$

on määriteltävissä logiikassa $\mathcal{L} \geq \mathrm{FO}$, niin jokainen $K^* \subseteq \mathcal{O}[\tau]$, jolla $K^* \cap \mathcal{O}[\tau]_{\geq n} = K$ voidaan määritellä logiikassa $\mathcal{L}$.

Määritellään seuraavaksi koneen M konfiguraatioiden koodaus uusien relaatiomuuttujien avulla. Kiinnitetään relaatiot ST^{CONF}, E_j^{CONF}, H_j^{CONF} ja I_j^{CONF}, joihin konfiguraation CONF informaatio koodataan seuraavasti:

$$ST^{\mathrm{CONF}} = \{s\},\ \text{missä } s \text{ on konfiguraation CONF tila.}$$

Kun $0 \leq j \leq k+m$, niin $E_j^{\mathrm{CONF}} = \{0\}$, jos nauhan j lukupää lukee merkkiä α, $E_j^{\mathrm{CONF}} = \{n-1\}$, jos nauhan j lukupää lukee merkkiä ω, ja $E_j^{\mathrm{CONF}} = \emptyset$ muissa tapauksissa. Kun $0 \leq j \leq k$, relaation H_j^{CONF} paikkaluku on r_j ja

$$H_j^{\mathrm{CONF}} = \{|e|_{r_j} \mid e \geq 0,\ \text{lukupää } j \text{ lukee } e\text{:ttä solua, jossa ei merkkiä } \omega\}.$$

Kun $k \leq j \leq k+m$, relaation H_j^{CONF} paikkaluku on d ja

$$H_j^{\mathrm{CONF}} = \{|e|_d \mid e \geq 0,\ \text{lukupää } j \text{ lukee } e\text{:ttä solua}\}.$$

Kun $k \leq j \leq k+m$, relaation I_j^{CONF} paikkaluku on d ja

$$I_j^{\mathrm{CONF}} = \{|e|_d \mid 0 \leq e \leq n^d \text{ ja työnauhan } j \text{ solussa } e \text{ on merkki } 1\}.$$

Selvästi konfiguraatio CONF määräytyy yksikäsitteisesti yllä olevista relaatioista.

Esimerkki 1.19. Koneen M alkutilannetta syötteellä $\mathfrak{A}$ vastaa konfiguraatio CONF_0, missä kaikki lukupäät ovat solun 0 kohdalla (ensimmäinen solu jossa on siis merkki α numeroitiin luvulla -1), työnauhat ovat tyhjiä ja M on tilassa s_0. Konfiguraatiota CONF_0 vastaa relaatiot

$$ST^{\mathrm{CONF}_0} = \{0\},\ E_j^{\mathrm{CONF}_0} = \emptyset,\ H_j^{\mathrm{CONF}_0} = \{(0,\ldots,0)\},\ \text{and } I_j^{\mathrm{CONF}_0} = \emptyset.$$

Koodataan konfiguraatio CONF lopuksi yhdeksi $d+2$ paikkaiseksi relaatioksi C^{CONF} liittämällä edellä määritellyt realaatiot yhteen. Kahta ensimmäistä koordinaattia käytetään eri relaatioiden erotteluun ja jonot laajennetaan pituuteen $d+2$ luvulla 0:

$$\begin{aligned}
C^{\mathrm{CONF}} &= \{(0,0)\} \times \{\overline{0}\} \times ST^{\mathrm{CONF}} \\
&\cup \bigcup_{0 \leq j \leq k+m} \{(1,j)\} \times \{\overline{0}\} \times E_j^{\mathrm{CONF}} \\
&\cup \bigcup_{0 \leq j \leq k+m} \{(2,j)\} \times \{\overline{0}\} \times H_j^{\mathrm{CONF}} \\
&\cup \bigcup_{k+1 \leq j \leq k+m} \{(3,j)\} \times I_j^{\mathrm{CONF}}.
\end{aligned}$$

Kun $C \subseteq \{0,\ldots,n-1\}^{d+2}$ voidaan helposti tarkastaa, että vastaako C jotain n^d-rajoitettua konfiguraatiota.

Lemma 1.20. *Olkoon M Turingin kone, joka toimii tilassa n^d. Tällöin on olemassa ensimmäisen kertaluvun logiikan lauseet $\varphi_{start}(\overline{x})$, $\varphi_{succ}(\overline{x}, X)$ ja $\psi_{succ}(X,Y)$ siten, että kaikille rittävän suurille $\mathfrak{A} \in \mathcal{O}[\tau]$ ja $\overline{a} \in \mathrm{Dom}(\mathfrak{A})^{d+2}$:*

a) Lause $\varphi_{start}(\overline{x})$ määrittelee konfiguraation CONF_0, eli kun relaatio C_0 vastaa konfiguraatiota CONF_0, niin

$$\langle \mathfrak{A}, \overline{a} \rangle \models \varphi_{start} \text{ joss } \overline{a} \in C_0.$$

b) Lause $\varphi_{succ}(\overline{x}, X)$ määrittelee X:n seuraajan, eli jos M on deterministinen ja C on M:n konfiguraatio, niin

$$\langle \mathfrak{A}, \overline{a}, C \rangle \models \varphi_{succ} \text{ joss } \overline{a} \in C', \text{ missä } C' \text{ on } C\text{:n seuraaja.}$$

c) Lause $\psi_{succ}(X,Y)$ ilmaisee, että Y on X:n seuraaja, eli jos C_1 on M:n konfiguraatio ja C_2 $d+2$-paikkainen joukon $\mathrm{Dom}(\mathfrak{A})$ relaatio, niin

$$\langle \mathfrak{A}, C_1, C_2 \rangle \models \psi_{succ} \text{ joss } C_2 \text{ } M\text{:n konfiguraatio ja } C_1\text{:n seuraaja.}$$

Todistus. Kohdassa a) määritellään lause $\varphi_{start}(\overline{x})$ seuraavasti:

$$xy\overline{x} = \overline{0} \lor (x = 2 \land \bigvee_{0 \leq j \leq k+m} (y = j \land x_1 \ldots x_d = \overline{0})),$$

missä siis $\overline{x} = x_1 \ldots x_d$.

Tarkastellaan seuraavaksi väitteitä $b)$ ja $c)$. Olkoon $(t, t') \in \delta$, missä siis

$$t = (s, a_0, \dots, a_k, c_1, \dots, c_m)$$

ja

$$t' = (s', c'_1, \dots, c'_m, i_0, \dots, i_{k+m}).$$

Paria (t, t') kohti voidaan määritellä lause $\varphi_{(t,t')}(\overline{x}, X)$ siten, että jos X:n tulkinta vastaa n^d-rajoitettua konfiguraatiota CONF, missä CONF on t:n mukainen, niin tällöin

$$\{\overline{a} \mid \langle \mathfrak{A}, \overline{a}, X^{\mathfrak{A}} \rangle \models \varphi_{(t,t')}\}, \tag{2}$$

vastaa konfiguraatiota CONF$'$, johon M siirtyy tekemällä parin (t, t') mukaisen laskenta-askeleen. Kun relaation X tulkinta vastaa n^d-rajoitettua konfiguraatiota CONF, niin voidaan helposti muodostaa lause $\varphi_{acc}(X)$, joka sanoo konfiguraation CONF olevan hyväksyvä konfiguraatio:

$$\varphi_{acc}(X) = X(00\overline{0}, s_+).$$

Nyt voidaan määritellä

$$\varphi_{succ}(\overline{x}, X) = (\varphi_{acc}(X) \wedge X(\overline{x})) \vee \bigvee_{(t,t') \in \delta} \varphi_{(t,t')}(\overline{x}, X))$$

ja

$$\begin{aligned} \psi_{succ}(X, Y) \quad = \quad & (\varphi_{acc}(X) \wedge \forall \overline{x}(Y(\overline{x}) \leftrightarrow X(\overline{x})) \vee \\ & \bigvee_{(t,t') \in \delta} (\exists \overline{x} \varphi_{(t,t')}(\overline{x}, X) \wedge \forall \overline{x}(Y(\overline{x}) \leftrightarrow \varphi_{(t,t')}(\overline{x}, X))) \end{aligned}$$

□

Lemman 1.20 avulla voidaan nyt saattaa loppuun Lauseen 1.17 todistus.

Lause 1.21. *Olkoon* $K \subseteq \mathcal{O}[\tau]$. *Jos* $K \in$ PSPACE, *niin* $K \in$ PFP.

Todistus. Olkoon M deterministinen Turingin kone joka tunnistaa luokan K tilassa x^d ja $\mathfrak{A} \in \mathcal{O}[\tau]$. Määritellään

$$\varphi(\overline{x}, X) = (\neg \exists \overline{y} X(\overline{y}) \wedge \varphi_{start}(\overline{x})) \vee (\exists \overline{y} X(\overline{y}) \wedge \varphi_{succ}(\overline{x}, X))).$$

Edellisen lemman nojalla $F_0^{\varphi}, F_1^{\varphi}, \dots$ on nyt jono $\emptyset, C_0, C_1, \dots$, missä

- C_0 on koneen M alkutilanteen konfiguraatio syötteellä $\mathfrak{A}$,

- Konfiguraatio C_{i+1} on konfiguraation C_i seuraaja. Erityisesti, jos C_i on hyväksyvä, niin tällöin $C_i = C_{i+1} = C_{i+2} \ldots$.
- Jos konfiguraatiolla C_i ei ole seuraajaa, niin C_i:n tilan täytyy olla s_-. Tällöin $C_{i+1} = \emptyset$ ja $C_{i+2} = C_0 \ldots$, eli jonolla ei ole kiintopistettä.

Yllä olevan perusteella kone M hyväksyy mallin $\mathfrak{A}$, joss F_∞^φ on hyväksyvä konfiguraatio, joss F_∞^φ on konfiguraatio, jonka tila on s_+. Ehto "F_∞^φ on konfiguraatio, jonka tila on s_+" voidaan ilmaista PFP:n lauseella

$$\exists y(y \text{ on } s_+\text{:s alkio järjestyksessä } <^{\mathfrak{A}} \wedge [\mathrm{PFP}_{\overline{x},X}\, \varphi]\min \min \overline{\min}\, y).$$

Kirjotetaan yllä oleva lause lyhyemmin muotoon $[\mathrm{PFP}_{\overline{x},X}\, \varphi]\min \min \overline{\min}\, s_+$. Nyt siis

$$\begin{aligned} \mathfrak{A} \in K, \quad & \text{joss} \quad M \text{ hyväksyy mallin } \mathfrak{A}, \\ & \text{joss} \quad \mathfrak{A} \models [\mathrm{PFP}_{\overline{x},X}\, \varphi]\min \min \overline{\min}\, s_+ \end{aligned}$$

Eli $K = \mathrm{Mod}([\mathrm{PFP}_{\overline{x},X}\, \varphi]\min \min \overline{\min}\, s_+)$. □

Tarkastellaan seuraavaksi logiikkaa IFP ja polynomisessa ajassa tunnistettavia luokkia. Polynomisessa tilassa toimivan Turingin koneen laskenta-aika voi olla eksponentiaalinen suhteessa syötteen kokoon. Polynomisessa ajassa toimivan koneen M laskenta taas pysähtyy oletuksen mukaan x^d laskenta-askelen jälkeen jollakin $d \in \mathbb{N}$, eli se on myös x^d tilarajoitettu. Tämä mahdollistaa Turingin koneen laskennan koodaamisen kokonaisuudessaan inflatorisesti yhdellä uudella relaatiomuuttujalla. Olkoon M ajassa x^d toimiva Turingin kone ja $C_0, C_1, \ldots,$ yksi koneen M laskenta syötteellä $\mathfrak{A}$. Jos M on deterministinen ja M hyväksyy mallin $\mathfrak{A}$, niin konfiguraation C_{n^d-1} täytyy olla hyväksyvä. Yllä kuvattu laskenta voidaan määritellä lauseella $\varphi(\overline{v}, \overline{x}, Z)$, missä

$$F_i^{Z(\overline{v},\overline{x})\vee\varphi} = \bigcup_{m<i,\ C_m \text{ määritelty}} \{|m|_d\} \times C_m,$$

ja

$$F_\infty^{Z(\overline{v},\overline{x})\vee\varphi} = \bigcup_{m<n^d,\ C_m \text{ määritelty}} \{|m|_d\} \times C_m.$$

Ideana on siis käyttää d:tä ensimmäistä koordinaattia kellona laskennalle.

Lause 1.22. *Olkoon $K \subseteq \mathcal{O}[\tau]$. Jos $K \in$ PTIME, niin $K \in$ IFP.*

Todistus. Olkoon M deterministinen Turingin kone, joka tunnistaa luokan K ajassa x^d. Kun $\overline{v} = v_0 \ldots v_{d-1}$, määritellään

$$\varphi(\overline{v}, \overline{x}, Z) = (\overline{v} = \overline{\min} \wedge \varphi_{start}(\overline{x})) \vee (\exists \overline{u}(S^d(\overline{u}, \overline{v}) \wedge \varphi_{succ}(\overline{x}, Z(\overline{u}, \cdot)))),$$

missä $S^d(\overline{u},\overline{v})$ määrittelee seuraajarelaation sanakirjajärjestyksen suhteen ja lause $\varphi_{succ}(\overline{x}, Z(\overline{u},\cdot))$ saadaan lauseesta $\varphi_{succ}(\overline{x}, X)$ korvaamalla alikaavat muotoa $X(\overline{t})$ kaavalla $Z(\overline{u},\overline{t})$. Nyt edellisen perusteella kaikilla tarpeeksi suurilla $\mathfrak{A}$

$$\begin{aligned}
\mathfrak{A} \in K, \quad & \text{joss} \quad M \text{ hyväksyy mallin } \mathfrak{A}, \\
& \text{joss} \quad n^d - 1\text{:s } M\text{:n konfiguraatio syötteellä } \mathfrak{A} \text{ on hyväksyvä}, \\
& \text{joss} \quad \mathfrak{A} \models [\mathrm{IFP}_{\overline{v},\overline{x},Z}\,\varphi]\overline{\max}\min\min, \overline{\min}\, s_+.
\end{aligned}$$

□

Logiikan IFP tapaus voidaan nyt helposti ylistää logiikalle Σ_1^1 ja luokalle NPTIME.

Lause 1.23. *Olkoon $K \subseteq \mathcal{O}[\tau]$. Jos $K \in$ NPTIME, niin $K \in \Sigma_1^1$.*

Todistus. Olkoon M epädeterministinen Turingin kone joka tunnistaa luokan K ajassa x^d. Tällöin

$$\begin{aligned}
\mathfrak{A} \in K, \quad & \text{joss} \quad M \text{ ainakin yksi } M\text{:n laskennoista syötteellä } \mathfrak{A} \text{ pysähtyy tilaan } s_+ \\
& \qquad\quad \text{ajassa } n^d, \\
& \text{joss} \quad \text{on olemassa } C_0, C_1, \ldots, C_{n^d-1},\ M\text{:n konfiguraatioita, missä} \\
& \qquad\quad C_0 \text{ vastaa } M\text{:n alkutilaa syötteellä } \mathfrak{A},\ C_{i+1} \text{ on } C_i\text{:n seuraaja} \\
& \qquad\quad \text{ja } C_{n^d-1} \text{ on hyväksyvä}, \\
& \text{joss} \quad \mathfrak{A} \models \varphi,
\end{aligned}$$

missä φ on lause

$$\begin{aligned}
\exists Z(\forall \overline{x}(Z(\overline{\min}\overline{x}) \leftrightarrow \varphi_{start}(\overline{x})) \wedge \forall \overline{u} \forall \overline{v}(S^d(\overline{u},\overline{v}) \rightarrow \psi_{succ}(Z(\overline{u},\cdot), Z(\overline{v},\cdot)) \\
\wedge Z(\overline{\max}\min\min, \overline{\min}\, s_+).
\end{aligned}$$

□

Seuraavat väiteet saadaan välittömästi edellä osoitetusta.

Korollaari 1.24. *Olkoon $K \subseteq \mathcal{O}[\tau]$.*

1. *Jos $K \in$ PTIME (PSPACE), niin K voidaan määritellä logikassa* IFP *(*PFP*) lauseella, jossa on vain yksi* IFP *(*PFP*) operaattorin esiintymä.*

2. PTIME = NPTIME, *joss* IFP $\equiv \Sigma_1^1$ *järjestetyillä malleilla.*

3. PTIME = PSPACE, *joss* IFP $\equiv$ PFP *järjestetyillä malleilla.*