Jos sekaannuksen vaaraa ei ole, samastamme säännöllisen lausekkeen ja sen esittämän kielen (eli kirjoitamme R vaikka tarkoitammekin $L(R)$).

Esimerkkejä:

- Σ^* koostuu kaikista aakkoston Σ merkkijonoista ja Σ^+ kaikista aakkoston Σ ei-tyhjistä merkkijonoista.
- $1\Sigma^*$ esittää kaikkia ykkösellä alkavia merkkijonoja.
- $(0(0 \cup 1)^*1) \cup (1(0 \cup 1)^*0)$ esittää kaikkia aakkoston $\{0, 1\}$ merkkijonoja, joiden viimeinen merkki on eri kuin ensimmäinen.
- $R\emptyset = \emptyset$ kaikilla R, ja $\emptyset^* = \varepsilon$.

Operaatioilla $\cup$ ja $\circ$ on neutraalialkiot $\emptyset$ ja ε: kaikilla R pätee

$$R \cup \emptyset = \emptyset \cup R = R$$
$$R \circ \varepsilon = \varepsilon \circ R = R.$$

Säännöllisiä lausekkeita käytetään etsittävän hahmon kuvaamiseen erilaisissa tekstinkäsittelytyökaluissa kuten Unixin `grep`.

Monissa sovelluksissa tiettyjen syötteiden kuten päivämäärän tai henkilötunnuksen oikea muotoilu määritellään säännöllisenä lausekkeena.

Myös ohjelmointikielten syntaktiset perusalkiot määritellään usein säännöllisenä lausekkeena. Esim. jossakin ohjelmointikielessä sallitut tunnukset voisivat olla muotoa

$$\text{kirjain}(\text{kirjain} \cup \text{numero} \cup \text{alaviiva})^* \quad ,$$

kokonaislukuvakiot muotoa

$$(\varepsilon \cup + \cup -)\text{numero}^+$$

ja liukulukuvakiot muotoa

$$(\varepsilon \cup + \cup -)(\text{numero}^*.\text{numero}^+ \cup \text{numero}^+.\text{numero}^*) \quad ,$$

missä "kirjain" tarkoittaa $\text{a} \cup \ldots \cup \text{z}$ ja "numero" tarkoittaa $0 \cup \ldots \cup 9$.

Käytännön sovelluksissa säännöllinen lauseke hyvin usein muunnetaan äärelliseksi automaatiksi (kohta nähtävän kaltaisella konstruktiolla).

Äärellisten automaattien ja säännöllisten lausekkeiden ekvivalenssi [Sipser s. 66–76]

Osoitamme seuraavan keskeisen tuloksen:

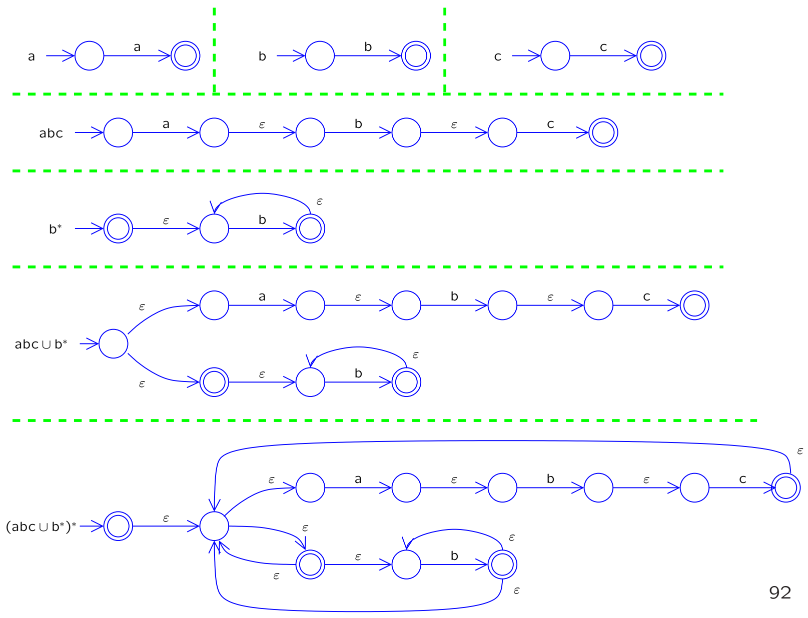
Lause 1.9: [Sipser Thm. 1.54] Kieli on säännöllinen, jos ja vain jos jokin säännöllinen lauseke esittää sitä.

Siis säännöllisillä lausekkeilla voidaan esittää tasan ne kielet, jotka voidaan tunnistaa äärellisillä automaateilla.

Todistuksen helpompi suunta on seuraava:

Lemma 1.10: [Sipser Lemma 1.55] Säännöllisen lausekkeen kuvaama kieli voidaan tunnistaa äärellisellä automaatilla.

Seuraavalla sivulla on esimerkkinä muodostettu säännöllistä lauseketta $(\text{abc} \cup \text{b}^*)^*$ vastaava NFA. Tämä perustuu edellä esitettyihin muunnoksiin, jotka esimerkin jälkeen kokoamme Lemman 1.10 todistukseksi.



Todistus: Ennen yksityiskohtiin menemistä tarkastellaan yleisesti tällaisen todistuksen luonnetta. Säännöllisen lausekeen määritelmä on *induktiivinen* (eli rekursiivinen).

Vertaa induktiiviseen luonnollisen luvun määritelmään:

1. 0 on luonnollinen luku ja

2. jos n on luonnollinen luku, niin $n+1$ on luonnollinen luku.

Luonnollisten lukujen induktioperiaatteesta seuraa, että jos

1. luvulla 0 on ominaisuus P ja

2. jos luvulla n on ominaisuus P niin luvulla $n+1$ on ominaisuus P

niin kaikilla luonnollisilla luvuilla on ominaisuus P.

Todistamme lemman suorittamalla analogisen "induktion säännöllisten lausekkeiden yli".

Haluamme siis osoittaa, että kaikilla säännöllisillä lausekkeilla R kieli $L(R)$ on säännöllinen. Teemme tämän osoittamalla, että

1. $L(\emptyset)$ ja $L(\varepsilon)$ ovat säännöllisiä,

2. $L(a)$ on säännöllinen kaikilla $a \in \Sigma$,

3. jos $L(R_1)$ ja $L(R_2)$ ovat säännöllisiä, niin $L(R_1 \cup R_2)$ ja $L(R_1 \circ R_2)$ ovat säännöllisiä ja

4. jos $L(R)$ on säännöllinen, niin $L(R^*)$ on säännöllinen.

Kohdat (1) ja (2) muodostavat perustapauksen ja niiden todistus on ilmeinen: on helppoa muodostaa äärellinen automaatti tunnistamaan kieli $\emptyset$, kieli $\{\,\varepsilon\,\}$ ja kieli $\{\,a\,\}$, missä $a \in \Sigma$.

Kohdat (3) ja (4) muodostavat induktioaskeleen. Koska

$$L(R_1 \cup R_2) = L(R_1) \cup L(R_2) \quad \text{ja} \quad L(R_1 \circ R_2) = L(R_1) \circ L(R_2),$$

niin kohta (3) seuraa lauseista 1.6 ja 1.7. Samoin kohta (4) seuraa lauseesta 1.8, koska $L(R^*) = (L(R))^*$. $\square$

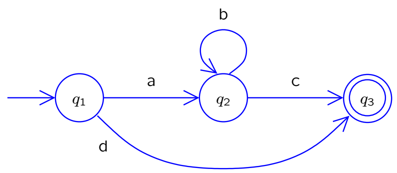
Säännöllisten kielten luokka on suljettu myös leikkauksen ja komplementin suhteen, joten edellä nähty todistus toimisi edelleen, jos säännöllisten lausekkeiden määritelmää laajennettaisiin *leikkaus- ja komplementtioperaatioilla*. Näin ei kuitenkaan tehdä paristakin syystä:

- Säännöllinen lauseke muunnetaan usein NFA:ksi edellä nähdyn kaltaisella konstruktiolla, joka on yksinkertainen ja jolla on useita hyödyllisiä ominaisuuksia. Esim. jokaista säännöllisen lausekkeen itsenäistä osaa kohti on automaatissa erotettavissa vastaava osa-automaatti. Leikkaukselle ja komplementille ei vastaavanlaista konstruktiota ole. (Tätä käsitellään vähän tarkemmin harjoituksissa.)

- Leikkaus ja komplementti ovat myös "turhia" operaatioita siinä mielessä, että niiden salliminen ei laajentaisi esitettävissä olevien kielten joukkoa. Jo nykyisillä operaatioilla voidaan esittää kaikki säännölliset kielet, kuten seuraavaksi osoitetaan.

Osoitamme nyt, että mille tahansa äärelliselle automaatille M voidaan muodostaa säännöllinen lauseke R, jolle $L(R) = L(M)$.

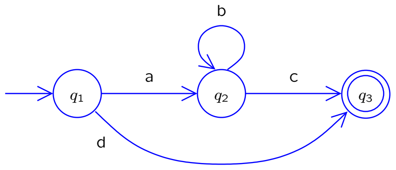
Jos M on automaatti



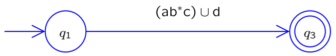
niin selvästi voidaan valita $R = (ab^*c) \cup d$.

Jotta voisimme hyödyntää tätä havaintoa osana monimutkaisemman automaatin analysoimista, otamme käyttöön *yleistetyt* NFA:t (generalized NFA, GNFA), joissa kaareen voi liittyä yksittäisen symbolin tai symbolijoukon sijasta kokonainen säännöllinen lauseke.

Ajatuksena on, että esim. edellä esitetystä automaatista



voidaan eliminoida tila q_2 ja saada ekvivalentti GNFA



Meidän pitää nyt sopia tarkemmin, mitä GNFA tarkalleen on.

Lähtökohtana on, että GNFA on kuten NFA, mutta kuhunkin kaareen voi liittyä mielivaltainen säännöllinen lauseke. Siirtyminen kaarta pitkin "kuluttaa" syötteestä jonkin osamerkkijonon, joka kuuluu kyseisen säännöllisen lausekkeen kuvaamaan kieleen.

Yksinkertaisuuden vuoksi sovimme, että GNFA noudattaa seuraavia rajoitteita:

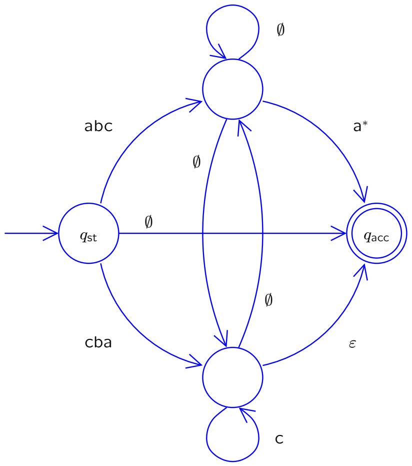
1. Siinä on tasan yksi hyväksyvä tila ja yksi alkutila, ja nämä eivät ole sama tila.

2. Alkutilaan ei tule siirtymiä.

3. Hyväksyvästä tilasta ei lähde siirtymiä.

4. Muuten minkä tahansa kahden tilan välillä on siirtymä, johon liittyy tasan yksi säännöllinen lauseke.

Esimerkki 1.11: Alla on eräs GNFA, joka tunnistaa kielen (abca*) ∪ (cbac*):



□

Muodollisesti GNFA on siis viisikko $(Q, \Sigma, \delta, q_{\text{start}}, q_{\text{accept}})$, missä

1. Q on äärellinen tilajoukko,

2. Σ on aakkosto,

3. δ on funktio $(Q - \{\, q_{\text{accept}} \,\}) \times (Q - \{\, q_{\text{start}} \,\}) \to \mathcal{R}$, missä $\mathcal{R}$ on aakkoston Σ säännöllisten lausekkeiden joukko ja

4. $q_{\text{start}} \in Q$, $q_{\text{accept}} \in Q$ ja $q_{\text{start}} \neq q_{\text{accept}}$.

Huomaa siirtymäfunktion δ aiemmasta poikkeava määrittely. Kun aiemmin olisi kirjoitettu esim. $\delta(q_1, a) = q_2$, nyt kirjoitetaan sen sijaan $\delta(q_1, q_2) = a$.

GNFA $(Q, \Sigma, \delta, q_{\text{start}}, q_{\text{accept}})$ hyväksyy merkkijonon $w \in \Sigma^*$, jos on olemassa esitys $w = w_1 \ldots w_k$, missä $w_i \in \Sigma^*$ kaikilla $1 \le i \le k$, ja tilajono $(q_0, \ldots, q_k) \in Q^{k+1}$, joilla

1. $q_0 = q_{\text{start}}$,

2. $w_i \in L(R_i)$ kaikilla $1 \le i \le k$, missä $R_i = \delta(q_{i-1}, q_i)$ ja

3. $q_k = q_{\text{accept}}$.

Huom. Jos $\delta(r, s) = \emptyset$, niin r ja s eivät koskaan voi esiintyä peräkkäin yllä mainitussa tilajonossa, sillä edes ε ei kuulu kieleen $L(\emptyset)$. Jätämme jatkossa GNFA:ta piirrettäessä tällaiset kaaret kuvasta pois.

Mistä tahansa DFA:sta voidaan helposti muodostaa saman kielen tunnistava GNFA:

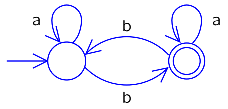
1. Lisää uudet tilat q_{start} ja q_{accept}.

2. Lisää ε-siirtymät tilasta q_{start} vanhaan alkutilaan ja vanhoista hyväksyvistä tiloista tilaan q_{accept}.

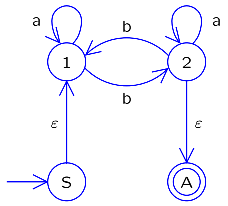
3. Kaikilla alkuperäisillä tiloilla q_1 ja q_2 aseta $\delta(q_1, q_2) = a_1 \cup \ldots \cup a_n$, missä $\{ a_1, \ldots, a_n \}$ on niiden merkkien joukko, joilla vanhassa automaatissa on siirtymä tilasta q_1 tilaan q_2.

4. Niillä q_1 ja q_2, joilla $\delta(q_1, q_2)$ ei vielä tullut määritellyksi, aseta $\delta(q_1, q_2) = \emptyset$.

Esimerkki 1.12: Seuraava DFA hyväksyy merkkijonot, joissa on pariton määrä b-merkkiä:



Vastaava GNFA on seuraava:



Tilat on nimetty S, A, 1 ja 2 jatkoa ajatellen.

□

Korkean tason algoritmi DFA:ta M vastaavan säännöllisen lausekkeen muodostamiseksi on nyt seuraava:

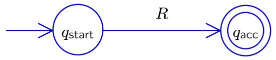
1. Olkoon automaatin M tilojen lukumäärä k. Muodosta $k+2$-tilainen GNFA G_{k+2}, joka tunnistaa saman kielen.

2. Muodosta 2-tilainen GNFA G_2, joka tunnistaa saman kielen kuin G_{k+2}.

3. Päättele kaksitilaisesta GNFA:sta G_2, mikä säännöllinen lauseke esittää sen tunnistamaa kieltä.

Totesimme juuri, miten askel 1 toteutetaan.

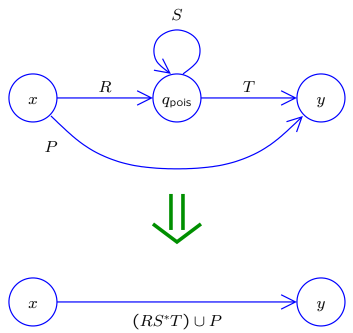
Askel 3 on triviaali: kaksitilaisen GNFA:n



tunnistamaa kieltä esittää tietysti säännöllinen lauseke R.

Jäljelle jää askeleen 2 toteutus.

GNFA:n G_{k+2} muuntaminen ekvivalentiksi kaksitilaiseksi GNFA:ksi G_2 tapahtuu eliminoimalla tiloja yksi kerrallaan. Tilan eliminoiminen perustuu muunnokseen



Huomaa, että tämä pitää tehdä samanaikaisesti kaikilla pareilla (x, y), joiden välillä on yhteys tilan q_{pois} kautta. Erityisesti pitää muistaa tapaus $x = y$.

Saamme seuraavan proseduurin:

EliminoiTila(G, q_{pois})

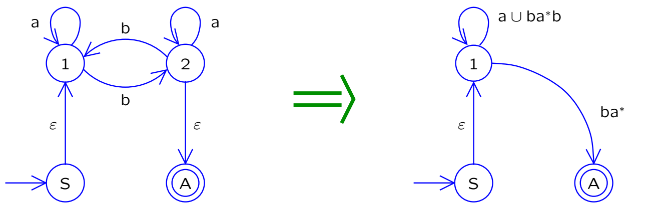
$\triangleright$ oletus: $G = (Q, \Sigma, \delta, q_{\text{start}}, q_{\text{accept}})$ on GNFA
$\triangleright$ $|Q| > 2$ ja $q_{\text{pois}} \in Q - \{ q_{\text{start}}, q_{\text{accept}} \}$
$Q' \leftarrow Q - \{ q_{\text{pois}} \}$
for all $x \in Q' - \{ q_{\text{accept}} \}$ ja $y \in Q' - \{ q_{\text{start}} \}$
do $R \leftarrow \delta(x, q_{\text{pois}})$
$S \leftarrow \delta(q_{\text{pois}}, q_{\text{pois}})$
$T \leftarrow \delta(q_{\text{pois}}, y)$
$P \leftarrow \delta(x, y)$
$\delta'(x, y) \leftarrow ((R)(S)^*(T)) \cup (P)$
return $G' = (Q', \Sigma, \delta', q_{\text{start}}, q_{\text{accept}})$.

Tämä pitää automaatin tunnistaman kielen ennallaan:

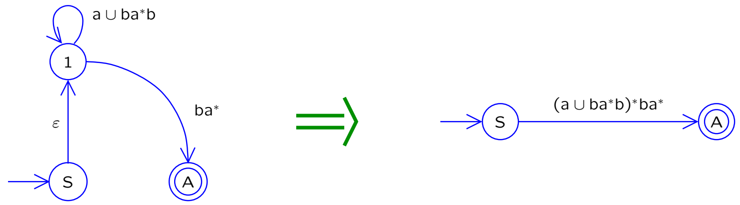
Lemma 1.13: Jos $G' = $ EliminoiTila(G, q_{pois}), niin G ja G' tunnistavat saman kielen.

Ennen todistusta katsotaan

Esimerkki 1.14: Parittoman määrän b-merkkiä hyväksyvästä GNFA:sta eliminoidaan ensin tila 2:



Toistetaan proseduuri vielä kerran ja tehdään kolmitilaisesta GNFA:sta kaksitilainen:



Kaksitilaisesta automaatista nähdään, että automaatin hyväksymää kieltä esittää säännöllinen lauseke

$$(\mathsf{a} \cup \mathsf{ba}^*\mathsf{b})^*\mathsf{ba}^*.$$

Tämä tosiaan esittää kieltä, jossa on parittoman määrän b:tä sisältävät merkkijonot.

□

Todistus lemmalle 1.13: Väitämme siis, että G ja G' hyväksyvät tasan samat merkkijonot.

Olkoon ensin w merkkijono, jonka G hyväksyy. Määritelmän mukaan on olemassa tilat $q_0, \ldots, q_k$ ja merkkijonot $w_1, \ldots, w_k \in \Sigma^*$, missä $q_0 = q_{\text{start}}$, $q_k = q_{\text{accept}}$, $w = w_1 \ldots w_k$ ja $w_i \in L(\delta(q_{i-1}, q_i))$ kun $i = 1, \ldots, k$. Merkitsemme tätä

$$q_{\text{start}} \xrightarrow{w_1} q_1 \xrightarrow{w_2} q_2 \xrightarrow{w_3} q_3 \ldots \xrightarrow{w_{k-1}} q_{k-1} \xrightarrow{w_k} q_{\text{accept}}.$$

Tarkastellaan ensin tilannetta $q_{i-1} \xrightarrow{w_i} q_i$, missä kumpikaan tiloista q_{i-1} ja q_i ei ole q_{pois}. Oletuksen mukaan $w_i \in L(\delta(q_{i-1}, q_i))$. Konstruktion mukaan $\delta'(q_{i-1}, q_i) = A \cup \delta(q_{i-1}, q_i)$ jollekin A. Siis pätee myös $w_i \in L(\delta'(q_{i-1}, q_i))$, eli

$$q_{i-1} \xrightarrow{w_i} q_i$$

on kelvollinen siirtymä myös automaatissa G'.

Tarkastellaan nyt tilannetta, että $q_i \neq q_{\text{pois}}$ ja $q_j \neq q_{\text{pois}}$, mutta $q_{i+1} = q_{i+2} = \ldots = q_{j-1} = q_{\text{pois}}$, joillakin $i \leq j - 2$. Siis

$$q_i \xrightarrow{w_{i+1}} q_{\text{pois}} \xrightarrow{w_{i+2}} q_{\text{pois}} \xrightarrow{w_{i+3}} \ldots \xrightarrow{w_{j-1}} q_{\text{pois}} \xrightarrow{w_j} q_j.$$

Nyt oletuksen mukaan

$$\begin{aligned} w_{i+1} &\in L(\delta(q_i, q_{\text{pois}})) \\ w_k &\in L(\delta(q_{\text{pois}}, q_{\text{pois}})) \qquad \text{kun } i+2 \leq k \leq j-1 \\ w_j &\in L(\delta(q_{\text{pois}}, q_j)), \end{aligned}$$

joten funktion δ' määritelmän mukaan

$$\begin{aligned} w_{i+1} \ldots w_j &\in L(\delta(q_i, q_{\text{pois}})) \circ L(\delta(q_{\text{pois}}, q_{\text{pois}}))^* \circ L(\delta(q_{\text{pois}}, q_j)) \\ &\subseteq L(\delta(q_i, q_{\text{pois}})) \circ L(\delta(q_{\text{pois}}, q_{\text{pois}}))^* \circ L(\delta(q_{\text{pois}}, q_j)) \cup L(\delta(q_i, q_j)) \\ &= L(\delta'(q_i, q_j)). \end{aligned}$$

Siis automaatissa G' voidaan laillisesti siirtyä

$$q_i \xrightarrow{w_{i+1} \ldots w_j} q_j.$$

Siis automaatin G siirtymäketjusta

$$q_{\text{start}} \xrightarrow{w_1} q_1 \xrightarrow{w_2} q_2 \xrightarrow{w_3} q_3 \ldots \xrightarrow{w_{k-1}} q_{k-1} \xrightarrow{w_k} q_{\text{accept}}$$

voidaan ensin ohittaa kaikki tilan q_{pois} esiintymät. Syntyvä siirtymäketju on kelvollinen tapa päästä tilasta q_{start} tilaan q_{accept} automaatissa G' merkkijonolla w. Siis G' hyväksyy merkkijonon w.

Pitää vielä osoittaa kääntäen, että jos G' hyväksyy merkkijonon w, niin myös G hyväksyy sen. Tämä jätetään harjoitustehtäväksi. □

Kootaan nyt edelläesitetty yhteen.

Lemma 1.15: [Sipser Lemma 1.60] Jos kieli on säännöllinen, se voidaan esittää säännöllisellä lausekkeella.

Todistus: Olkoon A säännöllinen kieli. Siis jollakin k on olemassa k-tilainen DFA M, joka tunnistaa kielen A. Muodostetaan säännöllinen lauseke R seuraavasti:

1. Muodosta DFA:sta M GNFA G_{k+2}, jossa on $k+2$ tilaa ja joka tunnistaa kielen A.
2. Toista seuraava arvoilla $n = k+1, k, k-1, \ldots, 2$:
   (a) Valitse automaatista G_{n+1} jokin tila $q_{\text{pois}} \notin \{ q_{\text{start}}, q_{\text{accept}} \}$.
   (b) Aseta $G_n \leftarrow \text{EliminoiTila}(G_{n+1}, q_{\text{pois}})$.
3. Palauta $R = \delta'(q_{\text{start}}, q_{\text{accept}})$, missä δ' on kaksitilaisen GNFA:n G_2 siirtymäfunktio.

Lemman 1.13 perusteella kaikki automaatit $G_{k+2}, G_{k+1}, \ldots, G_2$ tunnistavat saman kielen A. Siis $L(R) = A$. □

Lause 1.9 seuraa nyt suoraan lemmoista 1.10 ja 1.15.