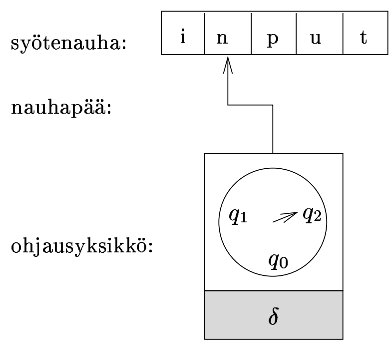
}
if ($q == 2$) printf("luvun arvo on %d", $sgn * val$);
else printf("Virheellinen luku");

3.2.2 Äärellisen automaatin formaali määrittely



- Äärellinen automaatti M
  - äärellistilainen *ohjausyksikkö*, jonka toimintaa säätelee automaatin *siirtymäfunktio* δ
  - merkkipaikkoihin jaettu *syötenauha*
  - *nauhapää*, joka kullakin hetkellä osoittaa yhtä syötenauhan merkkiä
- Automaatin "toiminta":
  - Automaatti käynnistetään erityisessä *alkutilassa* q_0, siten että tarkasteltava syöte on kirjoitettuna syötenauhalle ja nauhapää osoittaa sen ensimmäistä merkkiä
  - Yhdessä toiminta-askelessa automaatti lukee nauhapään kohdalla olevan syötemerkin, päättää ohjausyksikön tilan ja luetun merkin perusteella siirtymäfunktion mukaisesti ohjausyksikön uudesta tilasta, ja siirtää nauhapäätä yhden merkin eteenpäin
  - Automaatti pysähtyy, kun viimeinen syötemerkki on käsitelty. Jos ohjausyksikön tila tällöin kuuluu erityiseen *(hyväksyvien) lopputilojen* joukkoon, automaatti *hyväksyy* syötteen, muuten *hylkää* sen
  - Automaatin *tunnistama kieli* on sen hyväksymien merkkijonojen joukko

- *Määritelmä: Äärellinen automaatti* (engl. finite automaton) on viisikko

$$M = (Q, \Sigma, \delta, q_0, F),$$

missä

  - Q on automaatin *tilojen* äärellinen joukko;
  - Σ on automaatin *syöteaakkosto*;
  - $\delta : Q \times \Sigma \to Q$ on automaatin *siirtymäfunktio*;
  - $q_0 \in Q$ on automaatin *alkutila*;
  - $F \subseteq Q$ on automaatin *(hyväksyvien) lopputilojen* joukko.

- Esim. reaalilukuautomaatin formaali esitys:

$$\begin{aligned} M &= (\{q_0, \ldots, q_6, error\}, \{\texttt{0,1, ... ,9,.,E,e,+,-}\}, \\ & \quad \delta, q_0, \{q_3, q_6\}), \end{aligned}$$

missä δ on kuten aiemmin taulukossa; esim.

$$\delta(q_0, \texttt{0}) = \delta(q_0, \texttt{1}) = \cdots = \delta(q_0, \texttt{9}) = q_1,$$

$$\delta(q_0, .) = q_2, \quad \delta(q_0, E) = error, \quad \delta(q_1, \texttt{E}) = q_4 \quad \text{jne.}$$

- Automaatin *tilanne* on pari $(q, w) \in Q \times \Sigma^*$
  - automaatin *alkutilanne syötteellä* x on pari (q_0, x)
  - q on nykyinen tila ja w on syötemerkkijonon käsittelemätön osa

- Tilanne (q, w) *johtaa suoraan* tilanteeseen (q', w'), merk.

$$(q, w) \underset{M}{\vdash} (q', w'),$$

jos on $w = aw'$ $(a \in \Sigma)$ ja $q' = \delta(q, a)$.
Tilanne (q', w') on tilanteen (q, w) *välitön seuraaja*

- Tilanne (q, w) *johtaa tilanteeseen* (q', w') eli tilanne (q', w') on tilanteen (q, w) *seuraaja*, merk.

$$(q, w) \underset{M}{\vdash}^* (q', w'),$$

jos on olemassa välitilannejono (q_0, w_0), (q_1, w_1), ... , (q_n, w_n), $n \geq 0$, siten että

$$(q, w) = (q_0, w_0) \underset{M}{\vdash} (q_1, w_1) \underset{M}{\vdash} \cdots \underset{M}{\vdash} (q_n, w_n) = (q', w').$$

Erikoistapaus: $n = 0$, $(q, w) \underset{M}{\vdash}^* (q, w)$ millä tahansa tilanteella (q, w)

- Automaatti M *hyväksyy* merkkijonon $x \in \Sigma^*$, jos on voimassa

$$(q_0, x)\underset{M}{\vdash}^*(q_f, \epsilon) \qquad \text{jollakin } q_f \in F;$$

muuten M *hylkää* x:n. Ts. automaatti hyväksyy x:n, jos sen alkutilanne syötteellä x johtaa johonkin hyväksyvään lopputilanteeseen

- *Määritelmä:* Automaatin M *tunnistama kieli*

$$L(M) = \{x \in \Sigma^* \mid (q_0, x)\underset{M}{\vdash}^*(q_f, \epsilon) \quad \text{jollakin } q_f \in F\}$$

- Esim. merkkijonon "`0.25E2`" käsittely edellä esitetyllä liukulukuautomaatilla:

$$\begin{array}{lllll} (q_0, \texttt{0.25E2}) & \vdash & (q_1, \texttt{.25E2}) & \vdash & (q_3, \texttt{25E2}) \\ & \vdash & (q_3, \texttt{5E2}) & \vdash & (q_3, \texttt{E2}) \\ & \vdash & (q_4, \texttt{2}) & \vdash & (q_6, \epsilon). \end{array}$$

Koska $q_6 \in F = \{q_3, q_6\}$, on siis $\texttt{0.25E2} \in L(M)$

3.3 Automaattien minimointi

- Kaksi automaattia, jotka tunnistavat täsmälleen saman kielen ovat keskenään *ekvivalentteja*
- Äärellinen automaatti on *minimaalinen* jos se on tilamäärältään pienin ekvivalenttien automaattien joukossa
- Automaatti, jossa on enemmän tiloja kuin ekvivalentissa minimaalisessa automaatissa on *redundantti*
- Automaatteja muodostava algoritmit eivät aina tuota minimaalista automaattia
- On helpompi nähdä mikä on minimaalisen automaatin tunnistama kieli kuin redundantin automaatin tunnistama kieli
- On turha tallettaa ylimääräisiä tiloja
- Minimaalisen automaatin käsittely on tehokkaampaa kuin redundantin automaatin

3.3.1 Apukäsitteitä

- Määritellään M:lle laajennettu siirtymäfunktio δ^*, joka voi saada parametrinaan merkkijonon:
  jos $q \in Q$, $x \in \Sigma^*$, niin

$$\delta^*(q,x) = \text{ se } q' \in Q, \text{ jolla } (q,x) \underset{M}{\vdash}^* (q',\epsilon)$$

- Tilojen ekvivalenssi: M:n tilat q ja q' ovat *ekvivalentit*, merk.

$$q \equiv q',$$

  jos kaikilla $x \in \Sigma^*$ on

$$\delta^*(q,x) \in F \quad \text{jos ja vain jos} \quad \delta^*(q',x) \in F$$

  (ts. jos automaatti q:sta ja q':sta lähtien hyväksyy täsmälleen samat merkkijonot)

- *k-ekvivalenssi*: tilat q ja q' ovat k-ekvivalentit, merk.

$$q \stackrel{k}{\equiv} q',$$

  jos kaikilla $x \in \Sigma^*$, $|x| \leq k$, on

$$\delta^*(q,x) \in F \quad \text{jos ja vain jos} \quad \delta^*(q',x) \in F$$

  (ts. jos mikään enintään k:n pituinen merkkijono ei pysty erottamaan tiloja toisistaan)

- Selvästi pätee:

$$\begin{array}{llll} \text{(i)} & q \stackrel{0}{\equiv} q', & \text{joss} & \text{sekä } q \text{ että } q' \text{ ovat lopputiloja} \\ & & & \text{tai kumpikaan ei ole; ja} \\ \text{(ii)} & q \equiv q', & \text{joss} & q \stackrel{k}{\equiv} q' \text{ kaikilla } k = 0, 1, 2, \ldots \end{array}$$

- Minimoinnin idea: syötteenä annetun automaatin tilojen k-ekvivalenssiluokkia ositetaan $(k+1)$-ekvivalenssiluokiksi kunnes saavutetaan täysi ekvivalenssi

3.3.2 Äärellisen automaatin minimointi -algoritmi

- Syöte: Äärellinen automaatti $M = (Q, \Sigma, \delta, q_0, F)$.
  1. (Turhien tilojen poisto) Poista M:stä kaikki tilat, joita ei voida saavuttaa tilasta q_0 millään syötemerkkijonolla.
  2. (0-ekvivalenssi) Osita M:n jäljelle jääneet tilat kahteen luokkaan: ei-lopputiloihin ja lopputiloihin.
  3. (k-ekvivalenssi $\rightarrow$ $(k+1)$-ekvivalenssi)

     while not(tilasiirtymäfkt yhteensopiva luokkajaon kanssa) {
         jaa luokan sisällä eri tavalla käyttäytyvät tilat eri luokkiin;
     }
     return $\widehat{M} = (\hat{Q}, \Sigma, \hat{\delta}, \hat{q}_0, \widehat{F})$,

  missä
  - $\hat{Q}$=M:n tilaluokat
  - $\hat{\delta}$=luokkien välinen siirtymäfunktio
  - $\hat{q}_0$=M:n alkutilan luokka
  - $\widehat{F}$=M:n lopputilojen luokat
- Lopputulos
  - M:n kanssa ekvivalentti äärellinen automaatti $\widehat{M}$, jossa on minimimäärätiloja
  - $\widehat{M}$ on tilojen nimeämistä vaille yksikäsitteinen
- Huom: Tiloja on alun perin äärellinen määrä ja joka askeleessa nro 3 (paitsi viimeisessä) ositetaan vähintään yksi tilaluokka, joten algoritmi päättyy aina
- Todistus siitä, että algoritmin tuottama automaatti $\widehat{M}$ on minimiautomaatti ja yksikäsitteinen sivuutetaan (Orposen moniste s. 17–18)

Esimerkki

- Olkoon $M = (Q, \Sigma, \delta, q_0, F)$,
  - tilojen joukko $Q = \{1, 2, 3, 4, 5, 6\}$,

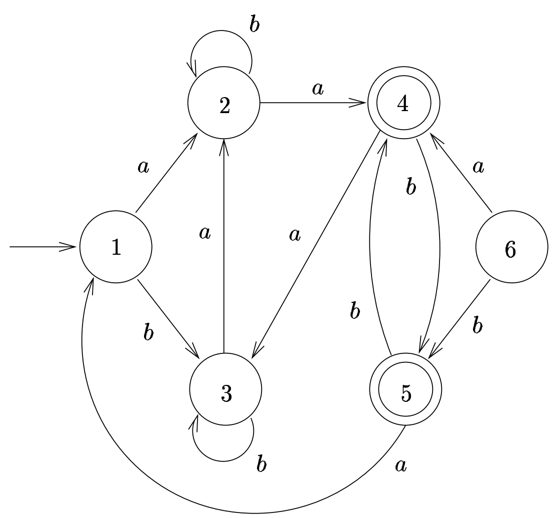
– syöteaakkosto $\Sigma = \{a, b\}$,

– alkutila $q_0 = \{1\}$,

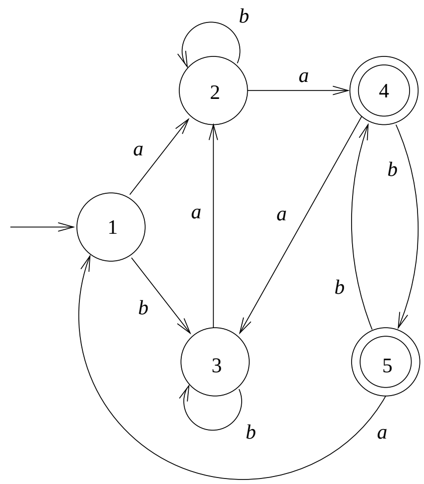
– lopputilojen joukko $F = \{4, 5\}$ ja

– siirtymäfunktio δ:

		a	b
$\rightarrow$	1	2	3
	2	4	2
	3	2	3
$\leftarrow$	4	3	5
$\leftarrow$	5	1	4
	6	4	5



• Askel 1: Turhien tilojen poisto



- Askel 2: 0-ekvivalenssi
  - Osita M:n jäljelle jääneet tilat kahteen luokkaan: lopputiloihin ja muihin tiloihin

			a	b
I:	$\rightarrow$	1	2, I	3, I
		2	4, II	2, I
		3	2, I	3, I
II:	$\leftarrow$	4	3, I	5, II
	$\leftarrow$	5	1, I	4, II

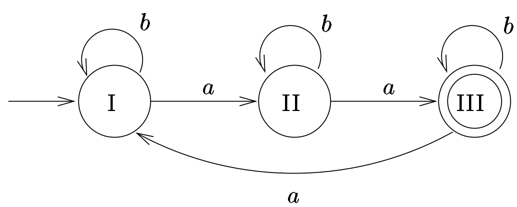
  - Nyt osituksesta voidaan muodostaa automaatti, jossa
    - ∗ kutakin luokkaa vastaa yksi tila ja
    - ∗ kustakin tilasta on kaikki erilliset siirtymät, jotka luokkaan kuuluvilla tiloilla on
  - Tila on *epädeterministinen*, jos siitä voidaan siirtyä jollain merkillä useampaan kuin yhteen tilaan
  - Esimerkissä tila I on epädeterministinen, koska merkillä a voidaan siirtyä tilaan I tai tilaan II

- Askel 3: k-ekvivalenssi $\Rightarrow$ $(k+1)$-ekvivalenssi
  - Jos $\widehat{M}$:ssä ei ole epädeterministisiä tiloja, niin algoritmi päättyy ja palauttaa $\widehat{M}$:n

- Muutoin hienonna kutakin $\widehat{M}$:n epädeterminististä tilaa vastaavan luokan ositusta edelleen:
  * Jaa sen sisällä alkuperäiset tilat eri luokkiin s.e. kustakin luokasta on vain samanlaisia siirtymiä
  * Suorita askel 3 uudestaan
- Esimerkissämme jaetaan tila I kahtia
- Tämän jälkeen ei ole enää epädeterminisiä tiloja ja algoritmi päättyy

• Lopputulos

			a	b
I:	$\rightarrow$	1	2, II	3, I
		3	2, II	3, I
II:		2	4, III	2, II
III:	$\leftarrow$	4	3, I	5, III
	$\leftarrow$	5	1, I	4, III

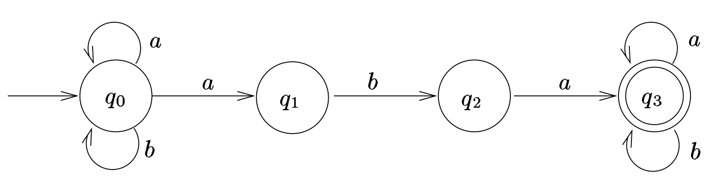


3.4 Epädeterministiset äärelliset automaatit

• Esimerkiksi alimerkkijonon *aba* etsiminen aakkoston $\{a, b\}$ merkkijonoista on luontevasti kuvattavissa epädeterministisellä automaatilla. Epädeterminismi siis helpottaa automaatin konstruointia

• Epädeterminististen automaattien avulla luodaan yhteys determinististen äärellisten automaattien ja *säännöllisten kielten* välille

- deterministiset ja epädeterministiset automaatit tunnistavat täsmälleen samat kielet



  - epädeterministiset automaatit tunnistavat täsmälleen säännölliset kielet
    $\Rightarrow$ Siis deterministiset automaatit tunnistavat täsmälleen säännölliset kielet

- Epädeterministisellä automaatilla siirtymäfunktio liittää vanhan tilan ja syötemerkin pariin (q, x) *joukon* mahdollisia seuraavia tiloja

- Epädeterministinen automaatti hyväksyy merkkijonon jos jokin mahdollisten tilojen jono johtaa lopputilaan. Jos yhtään tällaista jonoa ei ole, niin epädeterministinen automaatti hylkää syötemerkkijonon

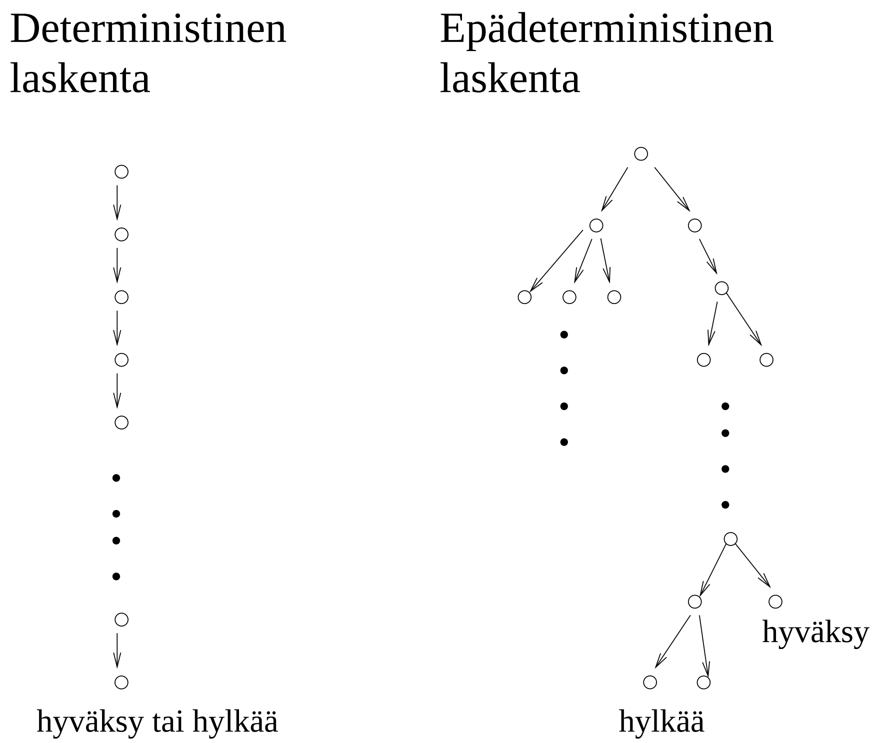
- Esim. kuvan automaatti hyväksyy syötejonon $abbaba$, koska se voidaan käsitellä seuraavasti

$$
\begin{aligned}
&(q_0, abbaba) \vdash (q_0, bbaba) \vdash (q_0, baba) \\
&\vdash (q_0, aba) \vdash (q_1, ba) \vdash (q_2, a) \vdash (q_3, \epsilon)
\end{aligned}
$$

- Toisaalta voidaan myös päätyä hylkäävään tilaan

$$
\begin{aligned}
&(q_0, abbaba) \vdash (q_0, bbaba) \vdash (q_0, baba) \\
&\vdash (q_0, aba) \vdash (q_0, ba) \vdash (q_0, a) \vdash (q_0, \epsilon)
\end{aligned}
$$

- Epädeterministinen automaatti voidaan ajatella suorittavan kaikki johdot rinnakkain



- *Määritelmä:* Epädeterministinen äärellinen automaatti (engl. nondeterministic finite automaton) on viisikko $M = (Q, \Sigma, \delta, q_0, F)$, missä
  - Q on äärellinen tilojen joukko,
  - Σ on syöteaakkosto,
  - $\delta : Q \times \Sigma \to \mathcal{P}(Q)$ on (joukkoarvoinen) siirtymäfunktio,
  - $q_0 \in Q$ on alkutila ja
  - $F \subseteq Q$ lopputilojen joukko.

- Kuvan automaatin siirtymäfunktio

	a	b
$\to q_0$	$\{q_0, q_1\}$	$\{q_0\}$
q_1	$\emptyset$	$\{q_2\}$
q_2	$\{q_3\}$	$\emptyset$
$\leftarrow q_3$	$\{q_3\}$	$\{q_3\}$

- Nyt virhetilanne on helposti ilmaistavissa tyhjän seuraajatilajoukon avulla

- (q,w) *voi* johtaa suoraan tilanteeseen (q',w'), $(q,w) \underset{M}{\vdash} (q',w')$, jos $w = aw'$ ja $q' \in \delta(q,a)$. Tilanne (q',w') on (q,w):n *mahdollinen* välitön seuraaja
- Muutoin määritelmät epädeterministisille automaateille ovat samat kuin aiemmin

- Deterministiset automaatit ovat epädeterminististen erikoistapaus $\Rightarrow$ kaikki edellisillä tunnistettavat kielet ovat tunnistettavissa myös jälkimmäisillä

- Mutta myös kääntäen: *deterministiset ja epädeterministiset äärelliset automaatit ovat yhtä vahvoja*

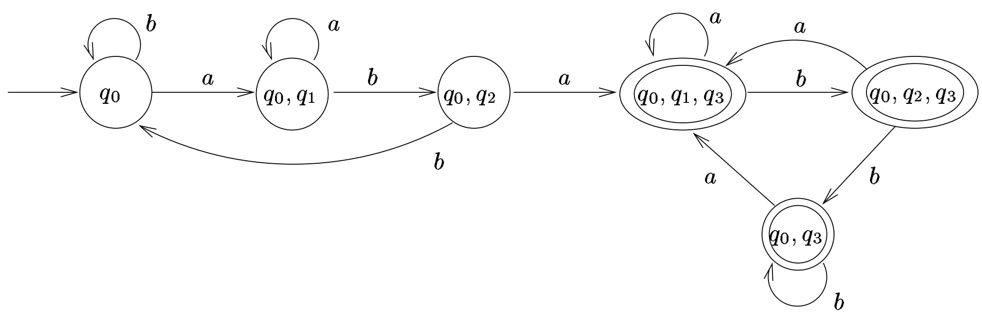
3.4.1 Automaatin determinisointi

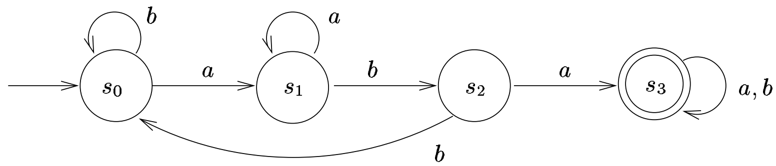
- Muodostetaan epädeterminististä automaattia M vastaava deterministinen automaatti $\widehat{M}$:

  1. Muodosta $\widehat{M}$:n tilat $S \subseteq \mathcal{P}(Q)$
     - ts. kaikki M:n tilojen joukot $\mathcal{P}(Q)$
     - Merk. $\mathcal{P}(Q) = \{\emptyset, s_1, s_2, ..., s_m\}$, missä tyhjä joukko vastaa virhetilaa ja $k = 2^n - 1$
  2. Lisää $\widehat{M}$:n tilojen välile siirtymät:
     - $s_i \stackrel{a}{\longrightarrow} s_j$, missä $s_j = \bigcup\{q' | f(q,a) = q', q \in s_i$
     - ts. tilajoukon s_i seuraajajoukkoon merkillä a kuuluvat kaikki ne tilat q', jotka voidaan saavuttaa s_i:n tiloista q
  3. Alkutilaksi $\{q_0\}$ (alkuperäisestä alkutilasta muodostettu joukko)
  4. Lopputiloiksi kaikki alkuperäisen lopputilan q_f sisältävät tilajoukot s_i, $q_f \in s_i$
  5. Karsi turhat tilat, joita ei voi saavuttaa alkutilasta
  6. Minimoi automaatti
     - jaa loppu- ja muihin tiloihin
     - hienonna luokkajakoa, kunnes yhdenmukainen siirtymäfkt:n kanssa

- Huom! Periaatteessa voitaisiin lisätä siirtymä kahden tilajoukon s_i ja s_j välille, jos yhdestäkin s_i:n tilasta voidaan saavuttaa jokin s_j:n tila. Tällä tavalla tulee kuitenkin paljon turhia tiloja.

- Esimerkki:





- *Lause:* Olk. $A = L(M)$ jonkin epädeterministisen äärellisen automaatin M tunnistama kieli. Tällöin on olemassa deterministinen automaatti $\widehat{M}$, jolla $L(\widehat{M}) = A$.
  *Todistus: Olk. $A = L(M)$, $M = (Q, \Sigma, \delta, q_0, F)$. Laaditaan determ. automaatti $\widehat{M} = (\hat{Q}, \Sigma, \hat{\delta}, \hat{q_0}, \hat{F})$, joka simuloi M:n toimintaa kaikissa sen kullakin hetkellä mahdollisissa tiloissa rinnakkain. Automaatin $\widehat{M}$ tilat vastaavat M:n tilojen joukkoja

$$\begin{aligned} \hat{Q} &= \mathcal{P}(Q), \\ \hat{q_0} &= \{q_0\}, \\ \hat{F} &= \{S \subseteq Q \mid S \text{ sisältää jonkin } q_f \in F\}, \\ \hat{\delta}(S, a) &= \bigcup_{q \in S} \delta(q, a). \end{aligned}$$

  Tarkastetaan, että $L(\widehat{M}) = L(M)$. Kielten ekvivalenssi seuraa, kun todistetaan kaikilla $x \in \Sigma^*$ ja $q \in Q$:

$$(q_0, x) \underset{M}{\vdash}^* (q, \epsilon) \Leftrightarrow (\{q_0\}, x) \underset{\widehat{M}}{\vdash}^* (S, \epsilon) \text{ ja } q \in S.$$

  Todistus induktiolla merkkijonon x pituuden suhteen:

  1. $|x| = 0$: $(q_0, \epsilon) \underset{M}{\vdash}^* (q, \epsilon) \Leftrightarrow q = q_0$.
     Samoin $(\{q_0\}, \epsilon) \underset{\widehat{M}}{\vdash}^* (S, \epsilon) \Leftrightarrow S = \{q_0\}$.
  2. *Induktio-oletus*: väite pätee kun $|x| \leq k$.

3. $|x| = k+1$: tällöin $x = ya$ jollakin y, $|y| = k$, jolle väite pätee induktio-oletuksen perusteella. Nyt

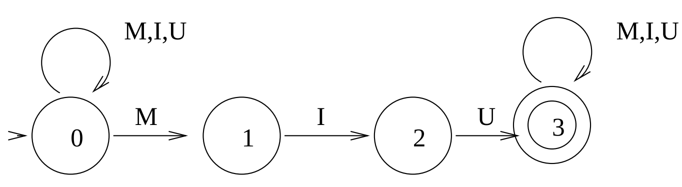
$$
\begin{aligned}
& (q_0, x) = (q_0, ya) \underset{M}{\vdash}^* (q, \epsilon) \\
\Leftrightarrow\ & \exists q' \in Q \text{ s.e. } (q_0, ya) \underset{M}{\vdash}^* (q', a) \text{ ja } (q', a) \underset{M}{\vdash} (q, \epsilon) \\
\Leftrightarrow\ & \exists q' \in Q \text{ s.e. } (q_0, y) \underset{M}{\vdash}^* (q', \epsilon) \text{ ja } (q', a) \underset{M}{\vdash} (q, \epsilon) \\
\Leftrightarrow\ & \exists q' \in Q \text{ s.e. } (\{q_0\}, y) \underset{\widehat{M}}{\vdash}^* (S', \epsilon) \text{ ja } q' \in S' \text{ ja } q \in \delta(q', a) \\
\Leftrightarrow\ & (\{q_0\}, y) \underset{\widehat{M}}{\vdash}^* (S', \epsilon) \text{ ja } \exists q' \in S' \text{ s.e. } q \in \delta(q', a) \\
\Leftrightarrow\ & (\{q_0\}, y) \underset{\widehat{M}}{\vdash}^* (S', \epsilon) \text{ ja } q \in \bigcup_{q' \in S'} \delta(q', a) = \hat{\delta}(S', a) \\
\Leftrightarrow\ & (\{q_0\}, ya) \underset{\widehat{M}}{\vdash}^* (S', a) \text{ ja } q \in \hat{\delta}(S', a) = S \\
\Leftrightarrow\ & (\{q_0\}, ya) \underset{\widehat{M}}{\vdash}^* (S', a) \text{ ja } (S', a) \underset{\widehat{M}}{\vdash} (S, \epsilon) \text{ ja } q \in S \\
\Leftrightarrow\ & (\{q_0\}, x) = (\{q_0\}, ya) \underset{\widehat{M}}{\vdash}^* (S, \epsilon) \text{ ja } q \in S.
\end{aligned}
$$

□

3.4.2 Hahmontunnistus epädeterministisellä automaatilla

Epädeterministisillä automaateilla voidaan kätevästi kuvata hahmontunnistusongelmia. Ohjelmatoteutusta verten automaatti on kuitenkin determinisoitava. Edellä huomasimme, että pahimmassa tapauksessa vastaavan deterministisen automaatin tilamäärä on eksponentiaalinen. Hahmontunnistusongelmissa determinisointialgoritmi tuottaa kuitenkin automaatin, jossa on *yhtä monta tilaa* kuin alkuperäisessä epädeterministisessä automaatissa, ja joka on samalla *minimiautomaatti!*

Esim. Olkoon aakkosto $\{M, I, U\}$. Esiintyykö hahmo MIU merkkijonossa? Vastaava epädeterministinen automaatti:

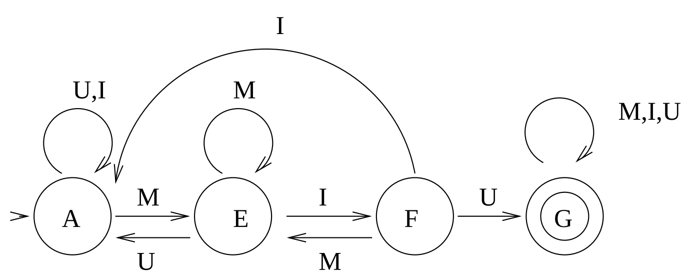


Kuva 3.1: Epädeterministinen MIU-automaatti.

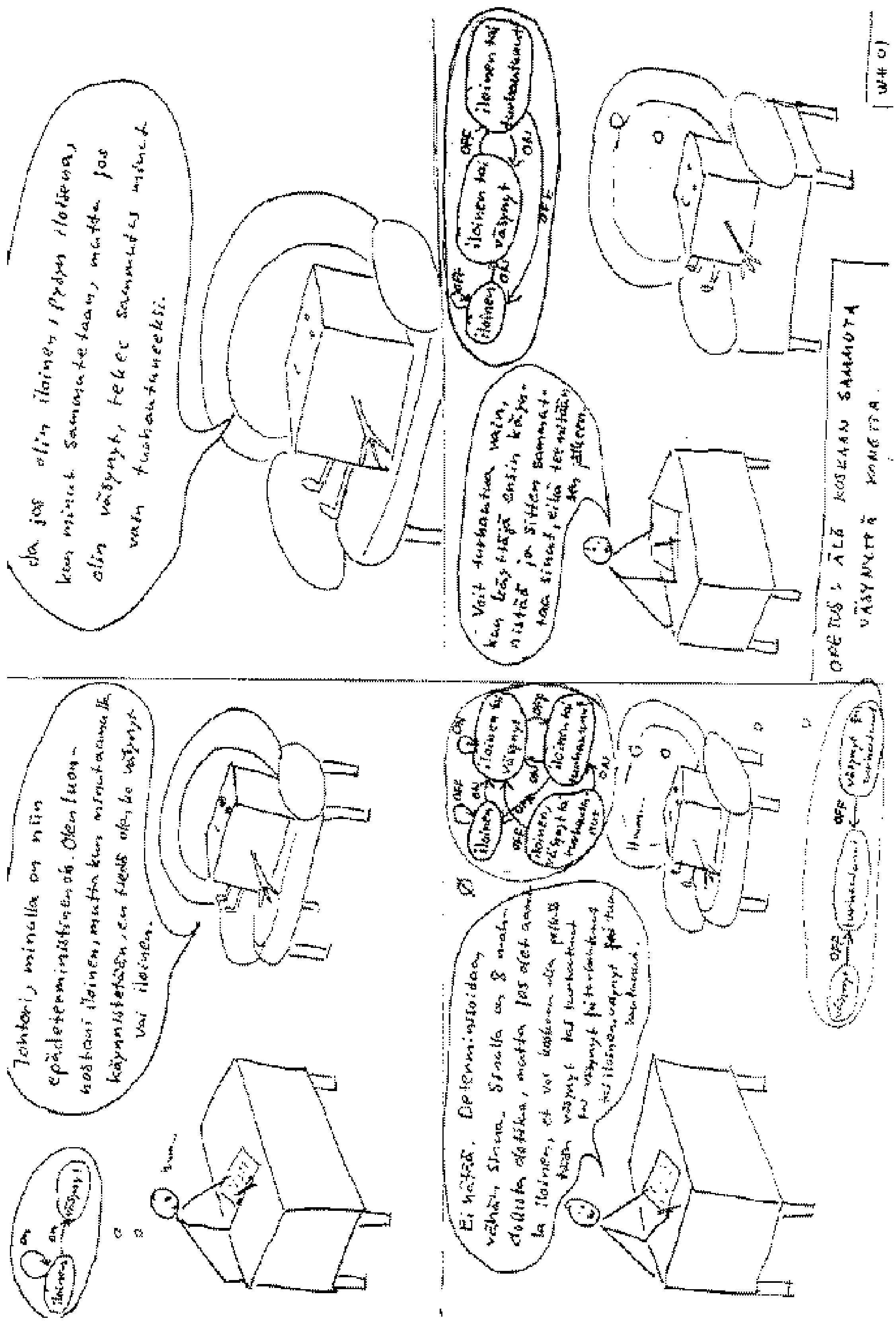
Muodostetaan vastaava deterministinen automaatti:

		M	I	U
	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
A	$\{0\}$	$\{0,1\}$=E	$\{0\}$=A	$\{0\}$=A
B	$\{1\}$	$\emptyset$	$\{2\}$=C	$\emptyset$
C	$\{2\}$	$\emptyset$	$\emptyset$	$\{3\}$=D
D	$\{3\}$	$\{3\}$=D	$\{3\}$=D	$\{3\}$=D
E	$\{0,1\}$	$\{0,1\}$=E	$\{0,2\}$=F	$\{0\}$=A
F	$\{0,2\}$	$\{0,3\}$=E	$\{0\}$=A	$\{0,3\}$=G
G	$\{0,3\}$	$\{0,1,3\}$=L	$\{0,3\}$=G	$\{0,3\}$=G
H	$\{1,2\}$	$\emptyset$	$\{2\}$=C	$\{3\} = D$
I	$\{1,3\}$	$\{3\}$=D	$\{2,3\}$=J	$\{3\}$=D
J	$\{2,3\}$	$\{3\}$=D	$\{3\}$=D	$\{3\}$=D
K	$\{0,1,2\}$	$\{0,1\}$=E	$\{0,2\}$=F	$\{0,3\}$=G
L	$\{0,1,3\}$	$\{0,1,3\}$=L	$\{0,2,3\}$=M	$\{0,3\}$=G
M	$\{0,2,3\}$	$\{0,1,3\}$=L	$\{0,3\}$=G	$\{0,3\}$=G
N	$\{1,2,3\}$	$\{3\}$=D	$\{2,3\}$=J	$\{3\}$=D
O	$\{0,1,2,3\}$	$\{0,1,3\}$=L	$\{0,2,3\}$=M	$\{0,3\}$=G

(Huom! Piirrä taulukon kuvaama automaatti!)
Poistetaan tilat, joita ei voi saavuttaa alkutilasta, sekä yhdistetään lopputilat (minimointialgoritmin mukaan).



Kuva 3.2: Lopputulos: minimaalinen deterministinen MIU-automaatti.


iloinen
väsynyt!
Tohtori, minulla on niin epädeterministinen olo. Olen luonnostani iloinen, mutta kun minut aamulla käynnistetään, en tiedä olenko väsynyt vai iloinen.
Ja jos olin iloinen, pysyn iloisena, kun minut sammutetaan, mutta jos olin väsynyt, tekee sammutus minut vain turhautuneeksi.
Ei hätää. Deterministoidaan vähän sinua.
iloinen
iloinen tai väsynyt
iloinen tai turhautunut
Hmm...
OFF
ON
väsynyt
turhautunut
Voit turhautua vain, kun käyttäjä ensin käynnistää ja sitten sammuttaa sinut
sen jälkeen.
OPETUS: ÄLÄ KOSKAAN SAMMUTA VÄSYNYTTÄ KONETTA.