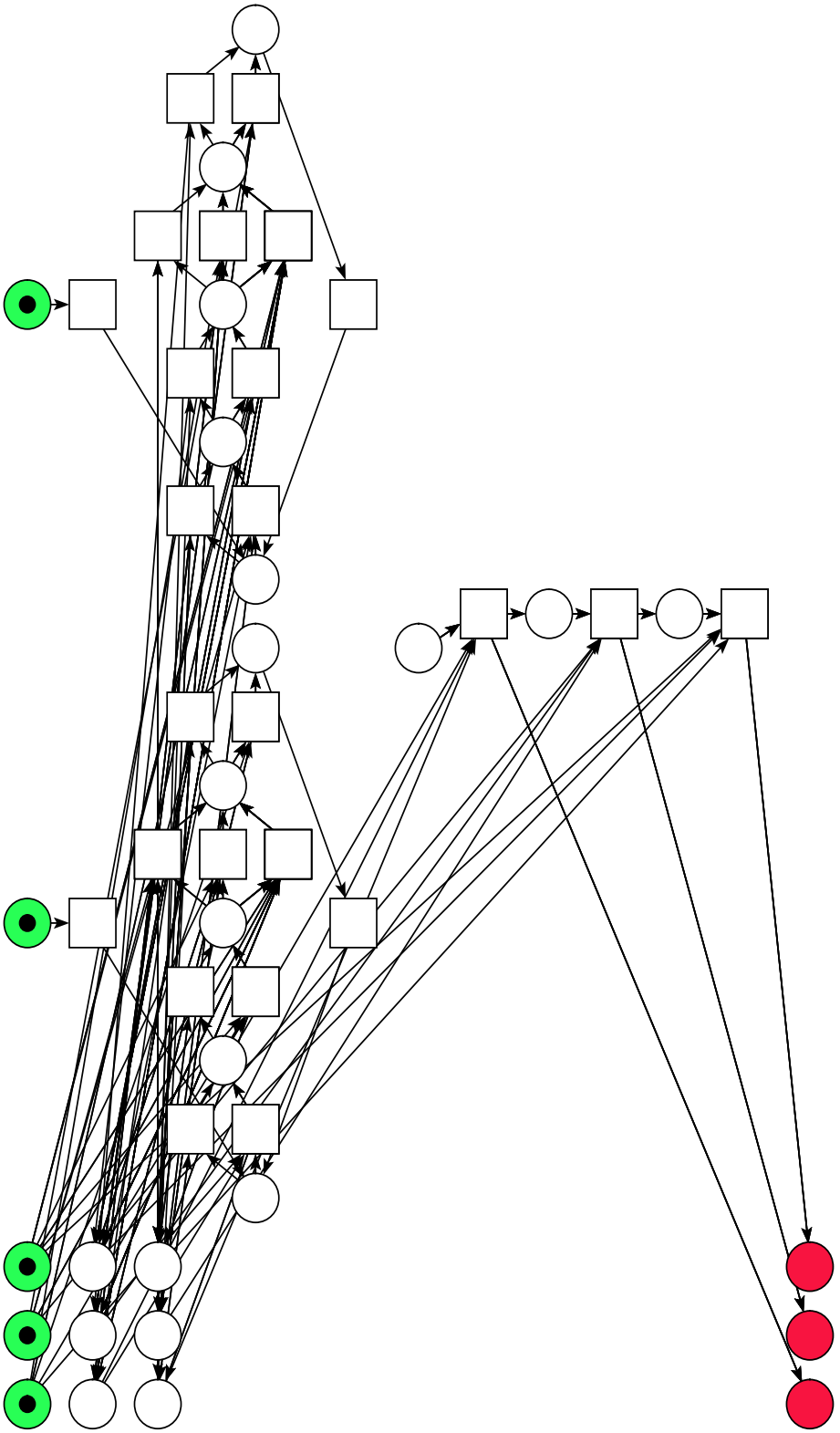
# 1 KORKEAN-TASON VERKOT

- P/T-verkot eivät sovellu laajojen järjestelmien kuvaamiseen:

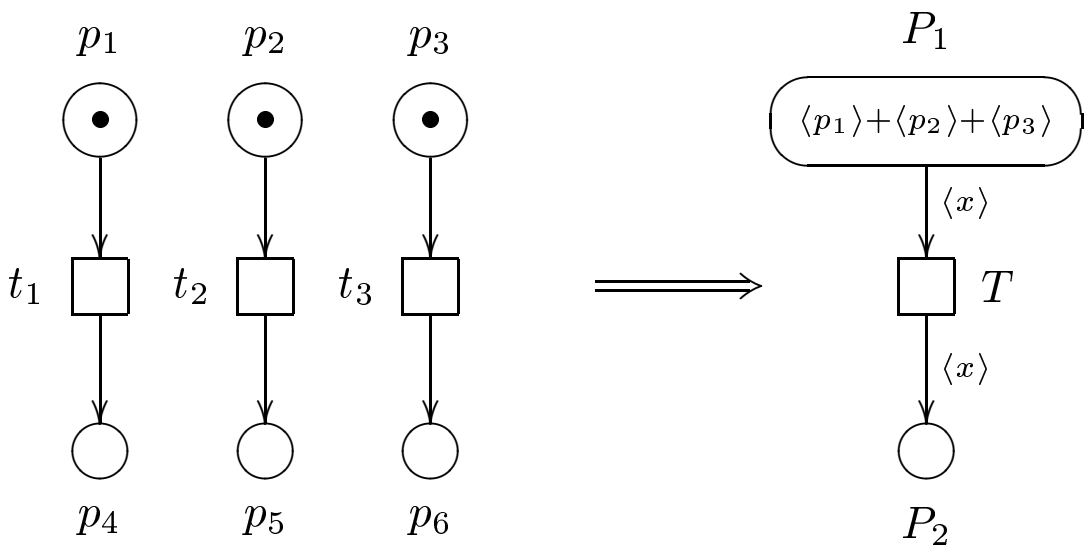


- Tarvitaan kompaktimpi notaatio.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## JOHDANTO

- P/T-verkoilla on myös toinen ongelma: esim. moniprosessorijärjestelmän mallissa ei voitu erottaa prosessoreita toisistaan.
- Idea: entä jos paikka ei sisälläkkään enää nimetöntä täplää, vaan informaatiota:




- $t_1, t_2, t_3$ ovat saman toiminnan $T$ "instansseja" eri prosesseille.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

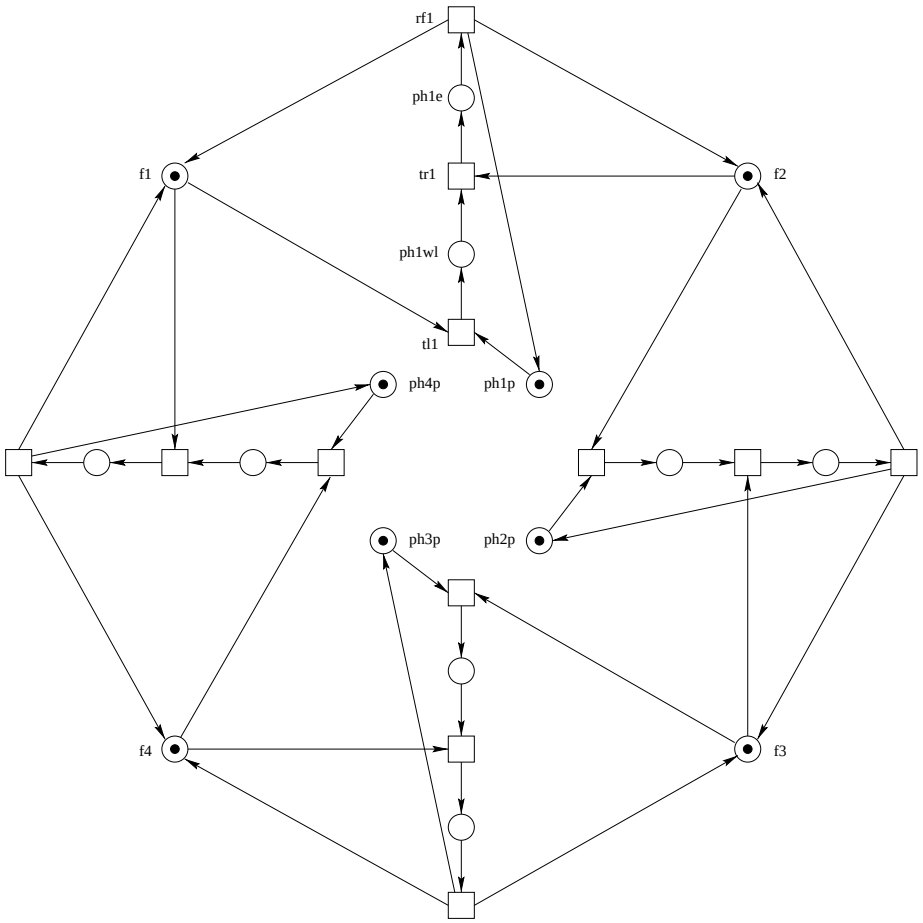
## Johdanto

- Lounastavat filosofit:
  Klassinen hajautetun resurssijaon esimerkkiongelma. 5 filosofia istuu pöydän ääressä ja joko mietiskelle tai syö. Filosofien väliin pöydän ympärille on sijoitettu 5 haarukkaa. Filosofien ruokana on spagettia, joka on niin sotkuista, että he tarvitsevat molemmat heidän vieressä olevat haarukat. Filosofit eivät kuitenkaan voi ottaa molempia haarukoita haltuunsa samanaikaisesti.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

# JOHDANTO

- Eräs ratkaisu kuvattuna 1-turvallisena P/T-verkkona on seuraava:

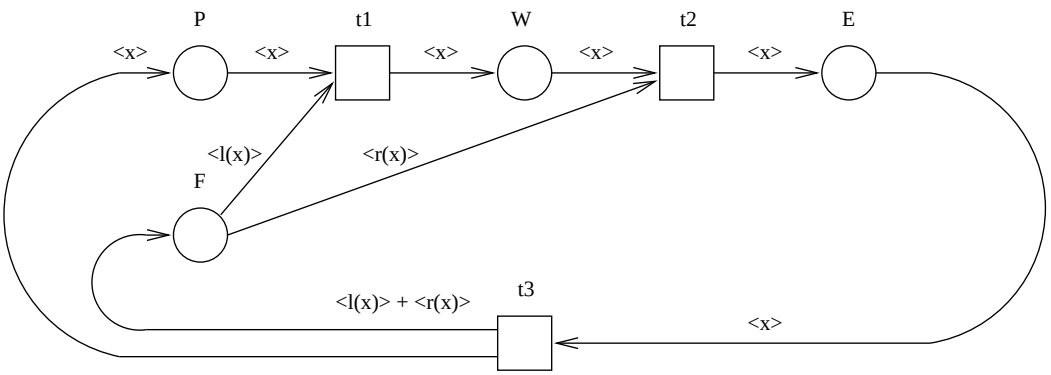



- Verkossa on lukkiuma, mutta se on seuraavan tarkastelun kannalta epäoleellista.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## JOHDANTO

- Rakennetaan korkeantason verkko seuraavasti:
  - Tehdään filosofeista ja haarukoista tokeneita, ja
  - käytetään kaarilausekkeita kuvaamaan mitkä filosofit syövät milläkin haarukalla

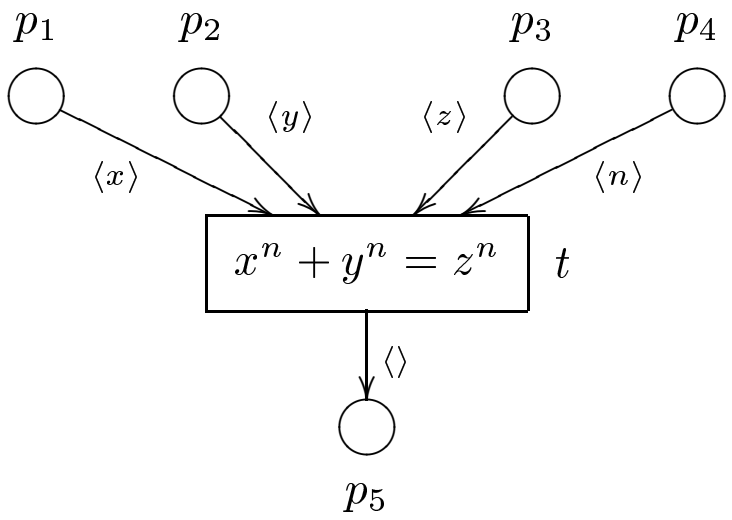



- Alkumerkintä:
  $P \mapsto \{\langle ph_1 \rangle + \langle ph_2 \rangle + \langle ph_3 \rangle + \langle ph_4 \rangle + \langle ph_5 \rangle\}$,
  $F \mapsto \{\langle f_1 \rangle + \langle f_2 \rangle + \langle f_3 \rangle + \langle f_4 \rangle + \langle f_5 \rangle\}$
- Funktiot: $l(ph_i) = f_i$, $r(ph_i) = f_{(i \bmod 5)+1}$

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## JOHDANTO

- Nyt esikaarille kirjoitettavat *annotaatiot* kertovat mitkä tuppelit paikasta tulee löytää, jotta transitio olisi vireessä.
- Transitioon on myös mahdollista liittää ehtoja, jotka rajoittavat tuppeleita vielä enemmän:




- Jotta annotaatiota voidaan tulkita täytyy ne *kiinnittää* johonkin rakenteeseen.
- Tämän kurssi puitteissa oletetaan, että ko. struktuuri on kokonaislukujen struktuuri:

$$Z = \langle \mathbb{Z}; \{+, \times, \dots\}, \{=, \neq, \leq, \geq, \dots\} \rangle.$$

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

- Annotaatioita ovat siis:

  $L_f$ hyvinmuodostettujen kaavojen joukko (totuusarvolliset lausekkeet)

  $$\text{Esim.:} \quad (x < 5 \wedge x > 1), x = y + 3, \exists z : P(x, z), \ldots$$

  $L_t$ hyvinmuodostettujen termien joukko (kokonaislukuarvoiset lausekkeet)

  $$\text{Esim.:} \quad f(x), x + 5, \ldots$$

- Lopuksi tarvitaan vielä:
  - *monikko*: "tuppeli" $\langle x, y\rangle, \langle x + 2, y\rangle, \langle x + y, 3\rangle, \langle x\rangle, \langle\rangle$
  - *formaali summa*: tapa esittää äärellistä koria peruslukukertoimisena lausekkeena:

    $$2\langle x\rangle + 3\langle y\rangle \equiv \{\langle x\rangle + \langle x\rangle + \langle y\rangle + \langle y\rangle + \langle y\rangle\}$$

  - $L_\Sigma(A)$ on joukon $A$ kaikkien formaalien summien joukko.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## Pr/T-verkko

**Definition 1** Olkoon $L$ jonkin ensimmäisen kertaluvun predikaatikalkyylin kieli. *Predikaati/Transitio- (Pr/T-)* verkko (engl. Predicate/Transition net) on kolmikko: $N_L = \langle N, A, M_0 \rangle$ missä:

1. $N = \langle S, T; F \rangle$ on niin sanottu *pohjana oleva* verkko (engl. underlying net),
2. $A = \langle A_N, A_S, A_T, A_F \rangle$ on sen *selostus* (engl. annotation), jossa
   (a) $A_N$ on ensimmäisen kertaluvun predikaattikalkyylin $L$ malli (tässä kokonaisluvut $Z$), niin sanottu $N_L$:n *tuki* support,
   (b) $A_S : S \to \mathbb{N}$ liittää kuhunkin paikkaan sen *ariteetin* (engl. arity), sen merkinnöissä sisältämien monikkojen koon (leveyden),
   (c) $A_T : T \to L_f$ liittää kuhunkin transitioon hyvinmuodostetun kaavan, niin sanotun *laukeamisehdon* (engl. firing condition, transition selector); lausekkeen vapaiden muuttujien on kunkin esiinnyttävä jossain kaariselityksen $A_F$ kyseiseen transitioon osuvaan kaareen liittämässä formaalissa summassa, ja

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

(d) $A_F$ liittää kuhunkin kaareen $\langle a, a' \rangle$ kielen $L_t$ termien monikkojen *formaalin summan* (engl. formal sum); kaikkien summan lausekemonikkojen tulee olla pituudeltaan samoja, ja pituuden tulee olla sama kuin sen paikan ariteetti, johon kaari osuu;

3. $M_0 : S \to L_\Sigma([A_N]^*)$ on Pr/T-verkon $N_L$ alkumerkintä, joka liittää kuhunkin paikkaan $s$ jonkin määrän struktuurin $A_N$ kantajajoukon alkioiden monikkoja (monijoukon kokonaisluku tuppeleita); näiden monikkojen koon tulee olla sama kuin kyseisen paikan ariteeti $A_s(s)$.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

# Pr/T-verkko

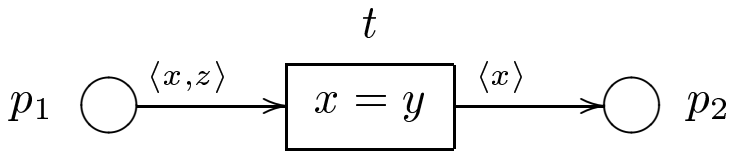



- $\langle S, T; F \rangle =$
  $\langle \{p_1, p_2, p_3\}, \{t\}, \{\langle p_1, t \rangle, \langle p_2, t \rangle, \langle t, p_3 \rangle\} \rangle$
- $A_N = Z$
- $A_S = \{p_1 \mapsto 1, p_2 \mapsto 1, p_3 \mapsto 1\}$
- $A_T = \{t \mapsto z = x + 1\}$
- $A_F = \{\langle p_1, t \rangle \mapsto \langle x \rangle, \langle p_2, t \rangle \mapsto \langle z \rangle, \langle t, p_3 \rangle \mapsto \langle x \rangle\}$
- $M_0 = \{p_1 \mapsto \langle 1 \rangle, p_2 \mapsto \langle 2 \rangle\}$

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## PR/T-VERKKO

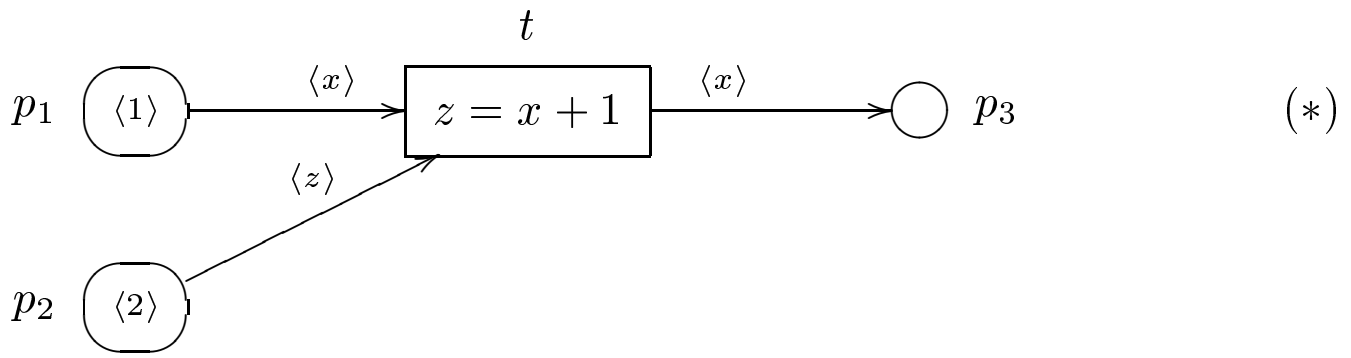
- HUOM:
  - Kohta 2c:




    ei sallittua
  - ei kapasiteetirajoituksia
  - ariteetti jätetään usein merkitsemättä, koska se voidaan päätellä kaarilausekkiden muodosta

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

# Laukeamissääntö




- P/T-verkkojen laukeamisehtoa pitää muuttaa siten, että se puhuu "nimettömien" tokeneiden sijaan struktuurin $A_N$ alkioista.
- Intuitio:
  - Tarkastellaan kaikkia mahdollisia transition $t$ kaarilausekkeiden muuttujien arvojakeluita (sijoituksia: $x \mapsto a, z \mapsto b$, jossa $a, b \in \mathbb{Z}$).
  - Jos sijoituksen määrämät tuppelit löytyvät esipaikoista voi transition laueta.
  - Esimerkissä transitio $t$ on vireessä, koska sijoitusta $x \mapsto 1, z \mapsto 2$ vastaavat tuppelit $(\langle 1 \rangle, \langle 2 \rangle)$ löytyvät paikoista $p_1$, ja $p_2$.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## Laukeamissääntö

**Definition 2** Olkoon $N_L$ Pr/T-verkko, ja $t$ jokin sen transitio.

- Transition $t$ *indeksi*, merkitään $I(t)$, on transition $t$ tulo- tai lähtökaarien kaarilausekkeissa esiintyvien muuttujien joukko.
- Funktio $\sigma_t : I(t) \to \mathbb{Z}$ on transition $t$ *arvonjakelu* (engl. substitution).
- Transition $t$ arvonjakelu on *kelvollinen* (engl. feasible), joss $Z \models (A_T(t)) : \sigma_t$, eli lauseke joka saadaan korvaamalla kaikki muuttujat arvonjakelun $\sigma$ antamilla kokonaisluvuilla transition $t$ laukeamisehdossa, on tosi.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## Laukeamissääntö

- Esimerkissä $(*)$:

$$\begin{aligned} I(t) &= \{x, z\}, \\ \sigma_t &= \{x \mapsto 2, z \mapsto 3\}, \\ \sigma'_t &= \{x \mapsto 2, z \mapsto 1\}, \end{aligned}$$

- $\sigma_t$ on kelvollinen, koska $Z \models (z = x + 1) : \sigma_t$ eli $Z \models 3 = 2 + 1$, mutta
- $\sigma'_t$ ei kelvollinen, koska $Z \not\models (z = x + 1) : \sigma'_t$ eli $Z \not\models 1 = 2 + 1$.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## Laukeamissääntö

**Definition 3** Olkoon $N_L$ Pr/T-verkko, $M$ sen merkintä, ja $t$ sen transitio, ja $\sigma$ transition $t$ arvonjakelu. Transition $t$ $\sigma$-*ilmentymä* (engl. $\sigma$-instance) on vireessä merkinnässä $M$ (merkitään $M[t:\sigma>$, joss

1. $\sigma$ on kelvollinen, ja
2. jos $sFt$, niin $M(s) \geq A_F((s,t)):\sigma$, missä $\geq$ tarkoittaa kokonaislukukorien pisteittäin määriteltyä järjestysrelaatiota.

- Esimerkissä $(*)$:
  - $M(p_1) = \langle 1 \rangle, M(p_2) = \langle 2 \rangle$,
  - sekä $A_F((p_1,t)):\sigma = \langle 1 \rangle$ ja $A_F((p_2,t)):\sigma = \langle 2 \rangle$.
  - Nyt siis sekä $M(p_1) \geq A_F((p_1,t)):\sigma = \langle 1 \rangle$ että $M(p_2) \geq A_F((p_2,t)):\sigma = \langle 2 \rangle$,
  - eli $M[t:\sigma>$.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## Laukeamissääntö

**Definition 4** Olkoon $N_L$ Pr/T-verkko, $M$ ja $M'$ sen merkintöjä, sekä $t$ sen transitio, ja $\sigma$ transition $t$ arvonjakelu. Transition $t$ *$\sigma$-ilmentymä* (engl. $\sigma$-instance) vie merkinnästä $M$ merkintään $M'$ (merkitään $M[t:\sigma>M'$), joss

1. $M[t:\sigma>$
2. kaikille verkon $N_L$ paikoille $s$ pätee $M'(s) = M(s) - (A_F((s,t)):\sigma) + (A_F((t,s)):\sigma)$.

- Esimerkissä saadaan:

$$\begin{aligned}
M'(p_1) &= M(p_1) - (A_F((p_1,t)):\sigma) + \\
&\quad (A_F((t,p_1)):\sigma) \\
&= \langle 1\rangle - \langle 1\rangle + \emptyset \\
M'(p_2) &= M(p_2) - (A_F((p_2,t)):\sigma) + \\
&\quad (A_F((t,p_2)):\sigma) \\
&= \langle 2\rangle - \langle 2\rangle + \emptyset \\
M'(p_3) &= M(p_3) - (A_F(<p_3,t>):\sigma) + \\
&\quad (A_F(<t,p_3>):\sigma) \\
&= \emptyset - \emptyset + \langle 1\rangle
\end{aligned}$$

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## Mallintaminen Pr/T-verkoilla

- Seuraavassa tarkastellaan erilaisia tapoja mallintaa AB-protokollaa.
- Mallitettaessa Pr/T-verkoilla joudutaan tekemään valinta:
  1. mikä osa järjestelmästä mallitetaan verkolla, ja
  2. mikä osa annotaatiolla
- Koska jokainen P/T-verkko on myös triviaalisti Pr/T-verkko esitä kalvolla 20 esitetty AB-protokollan malli yhtä ääripäätä: kaikki on mallinettu verkolla.

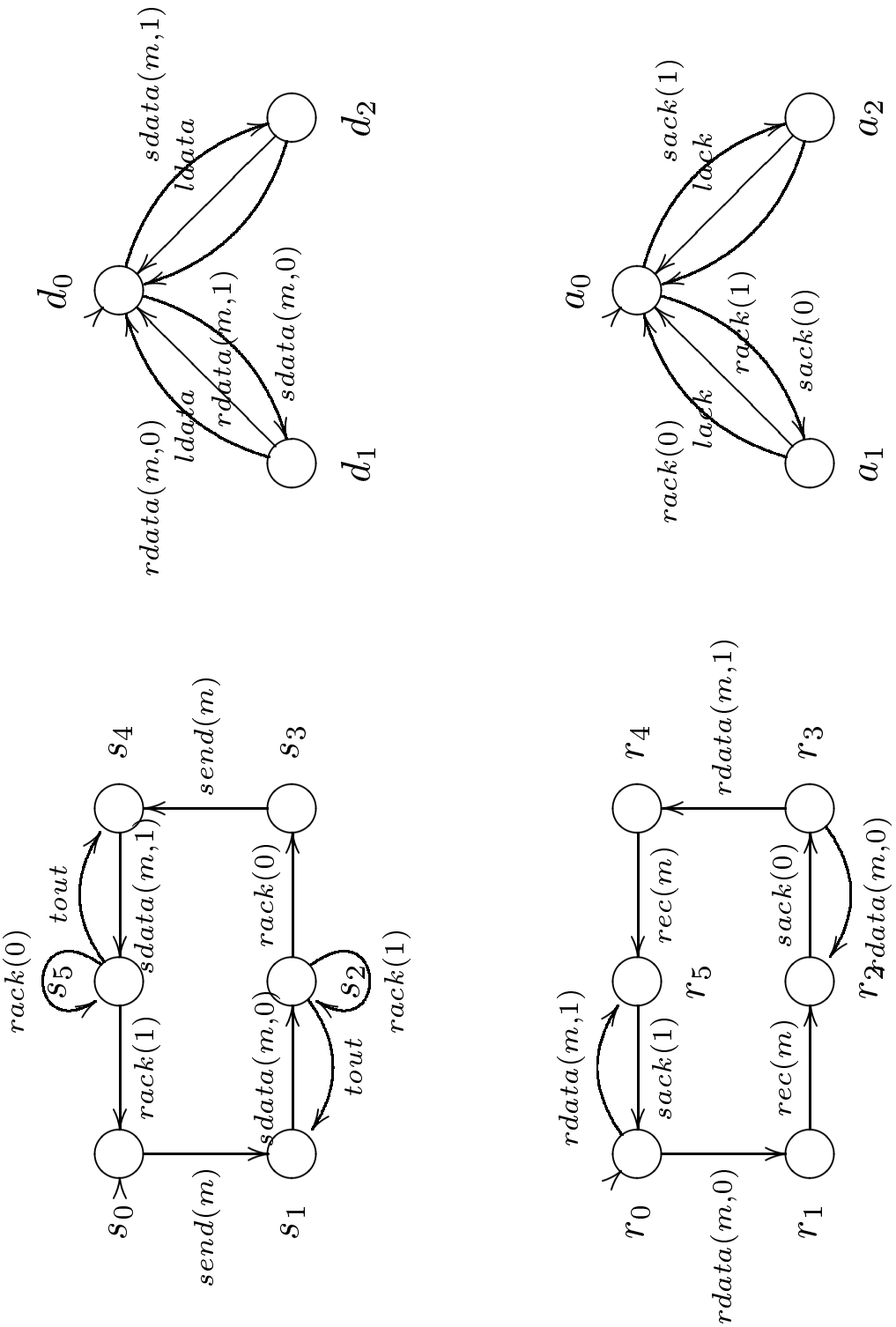
©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## AB-protokolla (korjattu)

- AB protokolla on yksinkertainen tietoliikenneprotokolla,
- Viestit lähetetään biteillä 0 ja 1 "taggattuna".
  - Lähettäjäprosessi: Saatuaan toimeksiannon, viesti lähetetään yhdessä bitin $b$ kanssa.
    - Jos viesti on hävinnyt saa prosessi time-outin ja viesti lähetetään uudestaan bitin $b$ kanssa.
    - Jos prosessi vastaanottaa bitin $b$ kättelynä se lähettää uuden viestin bitin $\hat{b} = 1 - b$ kanssa.
    - Jos prosessi vastaanottaa väärän bitin käättelynä se unohdetaan.
  - Vastaanottajaprosessi: prosessi vastaanottaa viestin yhdessä bitin $b$ kanssa. Bitti $b$ lähetetään kättelynä takaisin.
    - Jos prosessi vastaanottaa uuden viestin ja bitin $\hat{b}$ tämä kuitataan.
    - Jos prosessi vastaanottaa viestin ja bitin $b$ kättely uusitaan.
  - Protokolla siis lähettää ja vastaanottaa viestejä vuorotelleen.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

# SIIRTYMÄKAAVIOT (KOMMUNIKOIVAT AUTOMAATIT)




©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

# ABP:n Pr/T-Malli

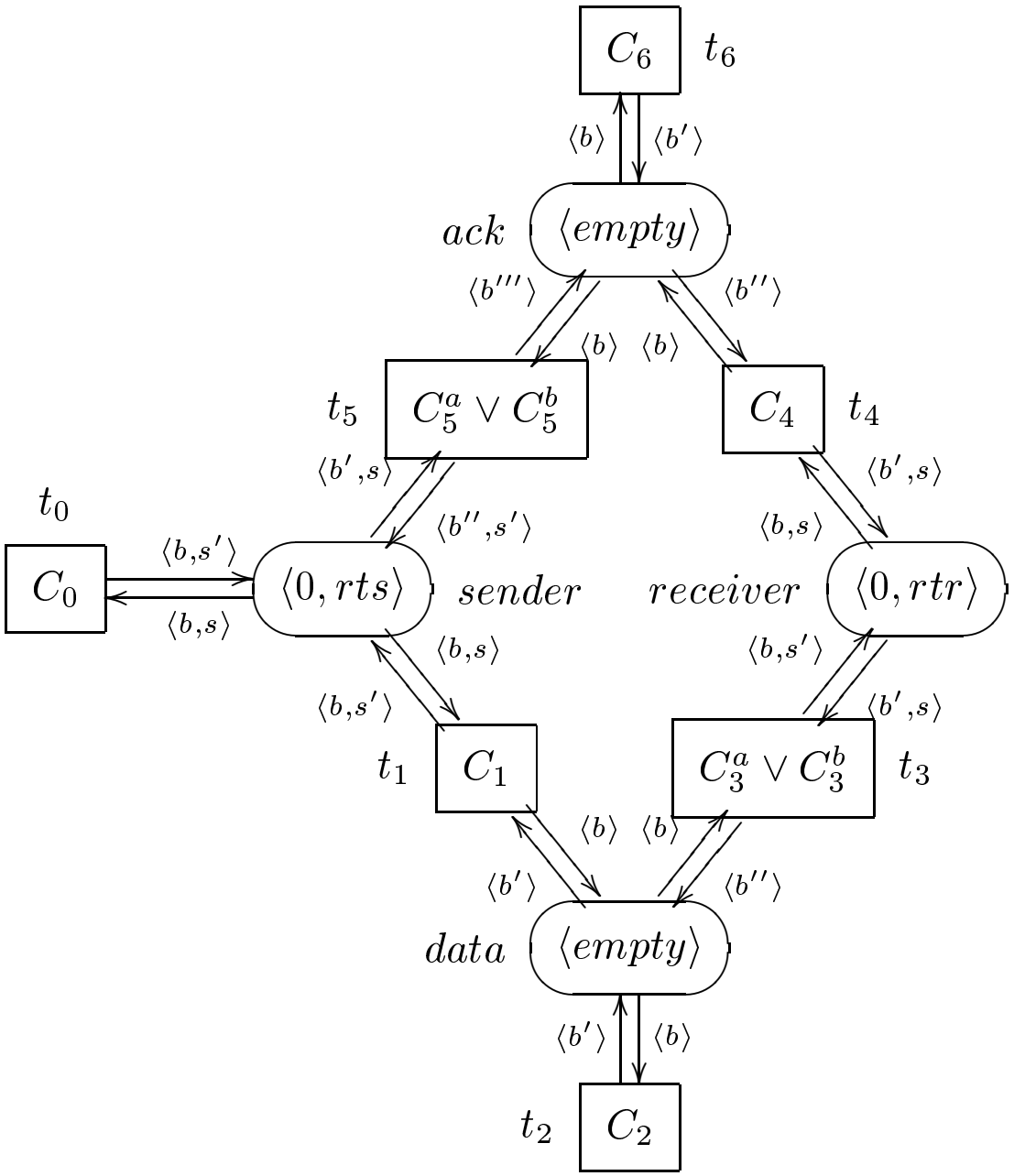
- Tehdään joitakin yksinkertaistuksia:
  - Järjestelmän oikeellisuuden kannalta on yhdentekevää mitä lähetettävä viesti sisältää,
  - samalla voidaan transitiot $send$ ja $rec$ poistaa,
  - käytetään yhtä paikka jokaiselle automaatille: automaatin tila koodataan tokeniin:

| lähettäjä | | | vastaanottaja | | |
|---|---|---|---|---|---|
| $\langle 0, rts \rangle$ | $=$ | $s_1$ | $\langle 0, rtr \rangle$ | $=$ | $r_0$ |
| $\langle 0, wack \rangle$ | $=$ | $s_2$ | $\langle 0, sack \rangle$ | $=$ | $r_2$ |
| $\langle 1, rts \rangle$ | $=$ | $s_4$ | $\langle 1, rtr \rangle$ | $=$ | $r_3$ |
| $\langle 1, wack \rangle$ | $=$ | $s_5$ | $\langle 1, sack \rangle$ | $=$ | $r_5$ |

  - kanavat ovat tiloissa $0, 1, empty$ (koska Pr/T-verkoille ei ole määritelty kapasiteettirajoituksia).

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

# ABP:n Pr/T-Malli (versio 1)




©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

# ABP:n Pr/T-Malli (versio 1)

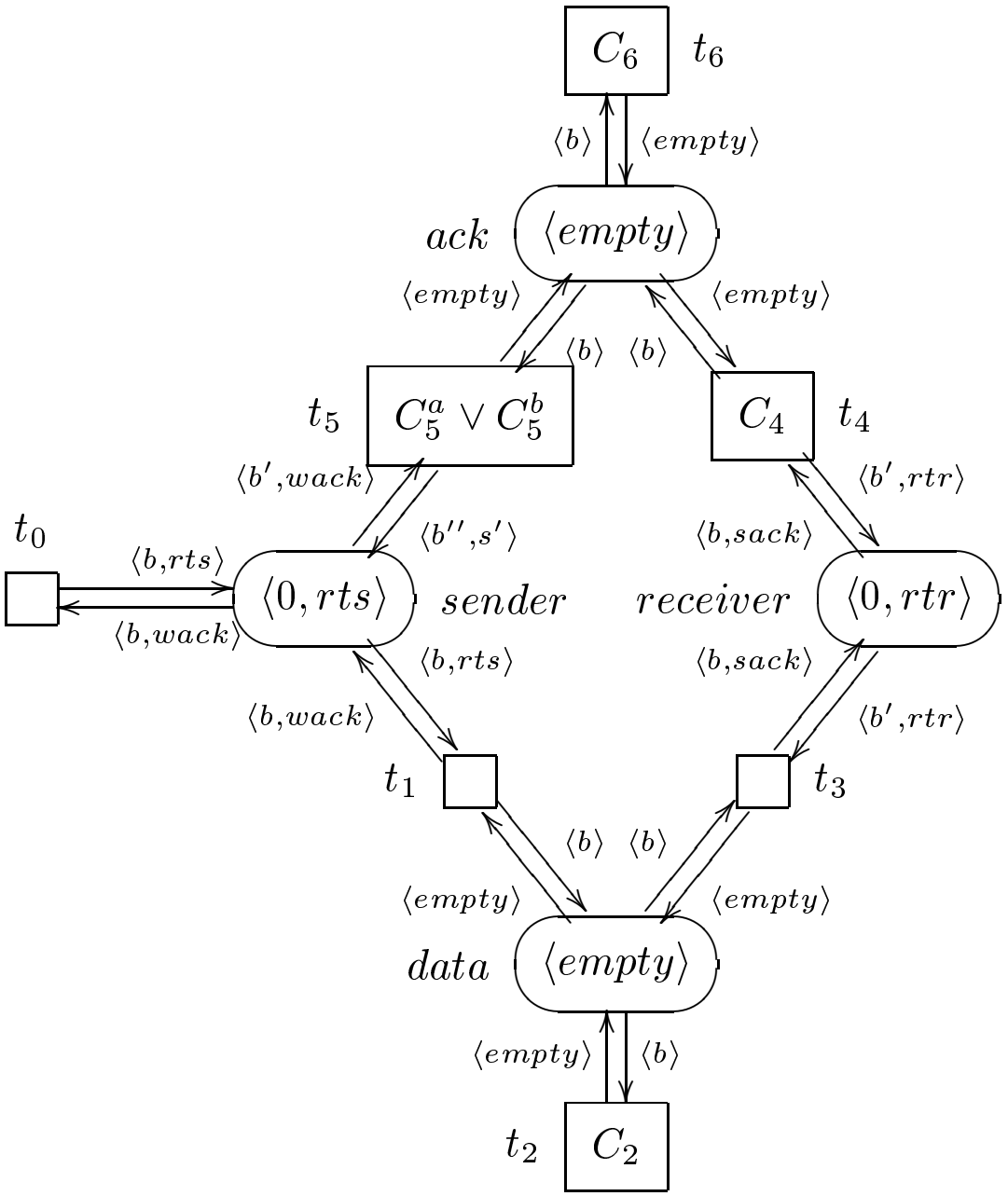
- Missä:

$$
\begin{aligned}
\text{tout: } C_0 &\equiv s = wack \wedge s' = rts \\
\text{sdata: } C_1 &\equiv s = rts \wedge b' = empty \wedge s' = wack \\
\text{ldata: } C_2 &\equiv b \neq empty \wedge b' = empty \\
\text{rdata OK: } C_3^a &\equiv s = rtr \wedge b = b' \wedge s' = sack \wedge b'' = empty \\
\text{rdata NOK: } C_3^b &\equiv s = rtr \wedge b = \neg b' \wedge s' = sack \wedge b'' = empty \\
\text{sack: } C_4 &\equiv s = sack \wedge b' = \neg b \wedge s' = rtr \wedge b'' = empty \\
\text{ack OK: } C_5^a &\equiv s = wack \wedge b' = b \wedge s' = rts \wedge b'' = \neg b \wedge b''' = empty \\
\text{ack NOK: } C_5^b &\equiv s = wack \wedge b' = \neg b \wedge s' = wack \wedge b'' = \neg b \wedge b''' = empty \\
\text{lack: } C_6 &\equiv b \neq empty \wedge b' = empty
\end{aligned}
$$

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

# ABP:n Pr/T-Malli (versio 2)

- Siirretään vakiot kaarille ja yksinkertaistetaan ehtoja:




©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## ABP:n Pr/T-Malli (versio 2)

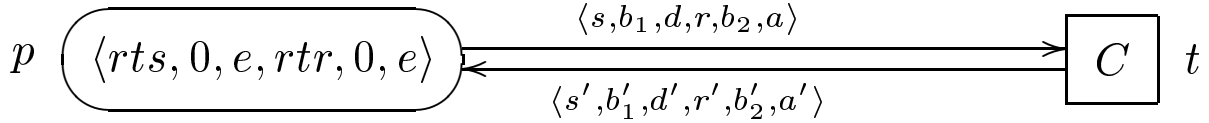
- Missä:

$$
\begin{aligned}
C_2 &\equiv b \neq empty \\
C_4 &\equiv b' = \neg b \\
C_5^a &\equiv b' = b \wedge s' = rts \wedge b'' = \neg b \\
C_5^b &\equiv b' = \neg b \wedge s' = wack \wedge b'' = \neg b \\
C_6 &\equiv b \neq empty
\end{aligned}
$$

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

# ABP:n Pr/T-Malli (versio 3)

- Transition ehtoja voidaan käyttää koko ABP:n saavutettavuusgraafin koodaamiseen:




missä,

| | $s$ | $b_1$ | $d$ | $r$ | $b_2$ | $a$ | $s'$ | $b'_1$ | $d'$ | $r'$ | $b'_2$ | $a'$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $send(m,0)$ | $rts$ | 0 | $e$ | $rtr$ | 0 | $e$ | $wack$ | 0 | 0 | $rtr$ | 0 | $e$ |
| $ldata$ | $wack$ | 0 | 0 | $rtr$ | 0 | $e$ | $wack$ | 0 | $e$ | $rtr$ | 0 | $e$ |
| $tout$ | $wack$ | 0 | 0 | $rtr$ | 0 | $e$ | $rts$ | 0 | $e$ | $rtr$ | 0 | $e$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |

Mikä tahansa järjestelmä on esitettävissä Pr/T-verkon avulla jossa on yksi transitio ja yksi paikka.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## ABP:n Pr/T-Malli

- ABP-n Pr/T-malli versio 3 edustaa ääripäätä: kaikki oleellinen on mallinnettu annotaatioilla. Verkkoa tarvitaan enää vain kuvaamaan dynamiikkaa.
- ABP-n Pr/T-malli versio 2 edustaa tasapainoista kuvaustapaa: verkosta näkee nopeasti informaation kulun, mutta informaation käsittely on koodattu transitioiden laukeamisehtoihin.
- Versiot 1 ja 2 ovat periaatteessa ekvivalentit. Versio 1 on huonompi: annotaatioihin on koodattu liikaa informaatiota.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## Pr/T-verkkojen ja P/T-verkkojen suhteesta

- Jos Pr/T-verkon jokaiselle transitiolle löytyy vain äärellinen määrä kelvollisia arvojakeluja voidaan Pr/T-verkko nähdä P/T-järjestelmän lyhennysmerkintänä.
- Jokainen Pr/T-verkko $N$ voidaan nimittäin tällöin *avata* P/T-verkoksi $U(N)$.
- Verkojen $N$ ja $U(N)$ saavutettavuusgraafit ovat isomorfiset.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

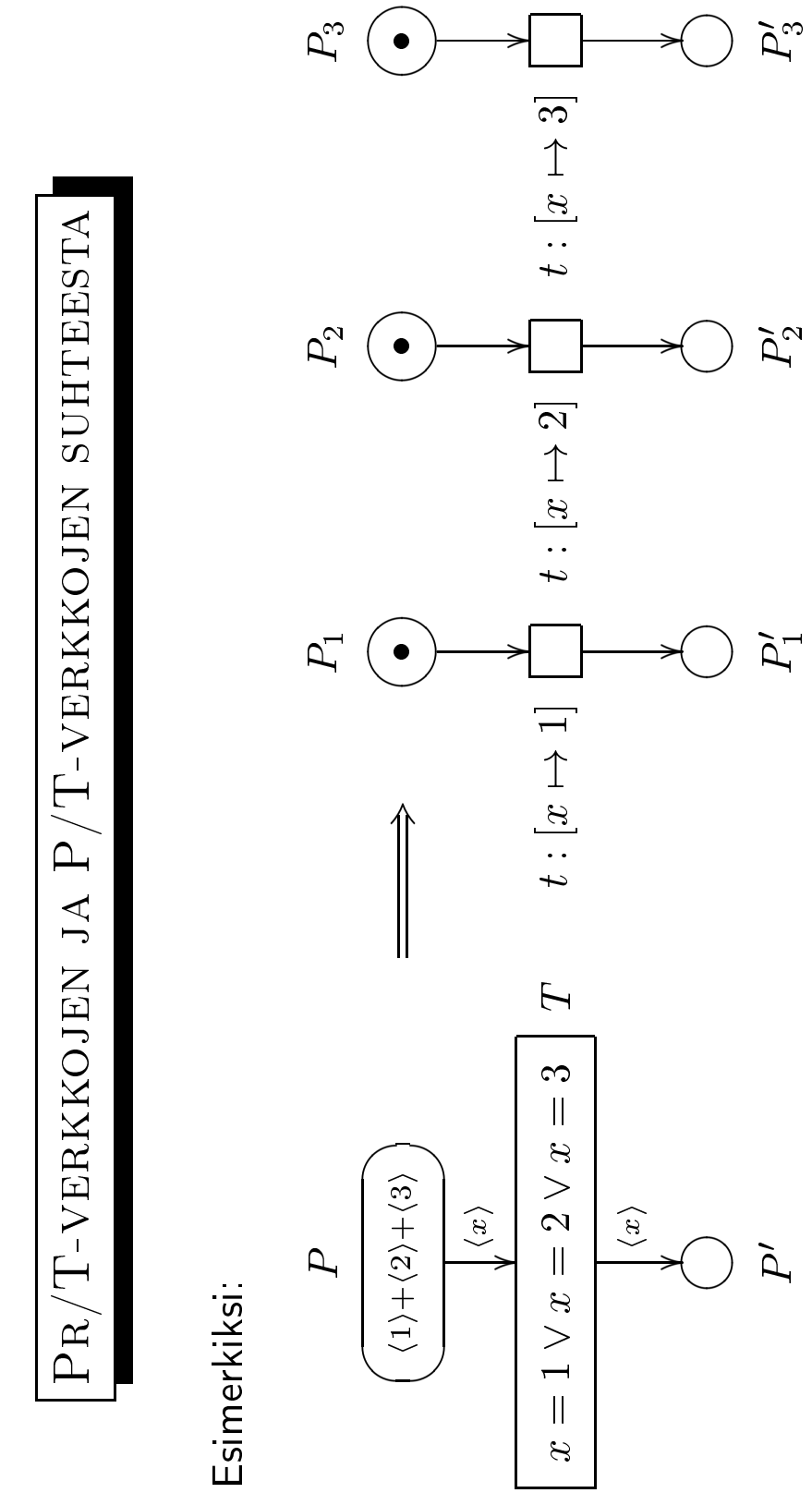
## Pr/T-verkkojen ja P/T-verkkojen suhteesta

- Avaus tapahtuu seuraavasti:
  1. Jokaista Pr/T-verkon paikan $p$ mahdollista monikkoa $m$ kohden luodaan yksi P/T-verkon paikka $p_m$.
  2. Jokaista Pr/T-verkon transition $t$ kelvollista arvojakelua $\sigma$ kohden luodaan yksi P/T-verkon transitio $t : \sigma$.
  3. Vuorelaatio avataan kytkemällä paikka $p_m$ transitioon $t : \sigma$ joss $A_F(p, t) : \sigma = m$ tai päinvastoin.
  4. Poistetaan kaikki kytkemättömät paikat.
  5. Avataan alkumerkintä.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## PR/T-VERKKOJEN JA P/T-VERKKOJEN SUHTEESTA

- Esimerkiksi:




©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

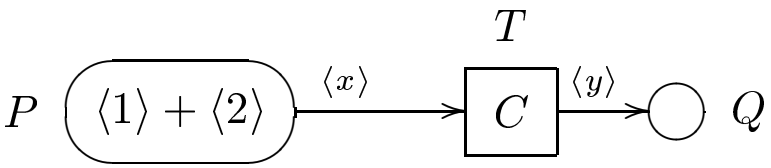
## Pr/T-verkkojen ja P/T-verkkojen suhteesta

- Avatussa verkossa saattaa olla *kuolleita* transitioita, jotka eivät ole koskaan vireessä. Nämä voidaan myös poistaa, mutta tämä edellyttää saavutettavuusgraafin laskentaa.
- P/T-verkon koko:
  - Olkoon paikan $p$ mahdollisten monikkojen joukko $M_p$.
  - Verkossa U(M) on paikkoja $\Sigma_{p \in P} |M_p| \approx O(max|M_p| \times |P|)$.
  - Olkoon transition $t$ kelvollisten arvonjakelujen joukko $S_t$
  - Transitioita syntyy $\Sigma_{t \in T} |S_t| \approx O(max|S_t| \times |T|)$.
  - Esim. jos transition $t$ indeksissä on $n$ muuttujaa joiden arvoalue on $\{0, 1\}$ on generoitujen transitioiden lkm. $2^n$.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## Pr/T-verkkojen ja P/T-verkkojen suhteesta

- Korkean-tason verkkojen alkuperäinen idea oli käyttää samaa allaolevaa verkkoa kuvamaan useita erillisiä rinnakkaisia prosesseja.
- Annotaatioiden avulla voidaan kuitenkin verkossa eksplisiittisesti kuvattu käyttäytyminen piiloittaa:




Missä $C \equiv (x = 1 \vee x = 2) \wedge (y = 1 \vee y = 2)$.

©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001

## Pr/T-verkkojen ja P/T-verkkojen suhteesta

- Tällöin avattu verkko on seuraava:




©Johan Lilius, Keijo Heljanko, and Esa Kettunen 1997–2001