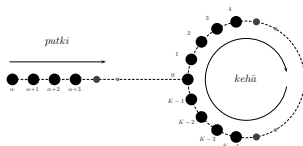


# Yleinen paikallinen vakautuva synkronointialgoritmi

Panu Luosto

23. marraskuuta 2007



## Lähdeartikkeli

Boulinier, C., Petit, F. ja Villain, V., When graph theory helps self-stabilization. *PODC '04: Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing*, New York, NY, USA, 2004, ACM Press, sivut 150–159.

# Sisältö

Määritelmät

Algoritmi

Lukkiutumattomuus ja parametri  $K$

Vakautuvuus ja parametri  $\alpha$

# Hajautettu järjestelmä

- äärellinen suuntaamaton yhtenäinen verkko  $G = (V, E)$
- jokaisella solmulla  $p$  kellorekisteri  $r_p$
- solmun  $p$  naapurien joukko on  $\mathcal{N}_p$
- naapurit näkevät toistensa muuttujat, ja laillisuusehdot ovat paikallisia
- ohjelma kokoelma ilmaisuja

*ehto*  $\longrightarrow$  *toimenpide*

- suoritusvuorot jakaa epäreilu demoni

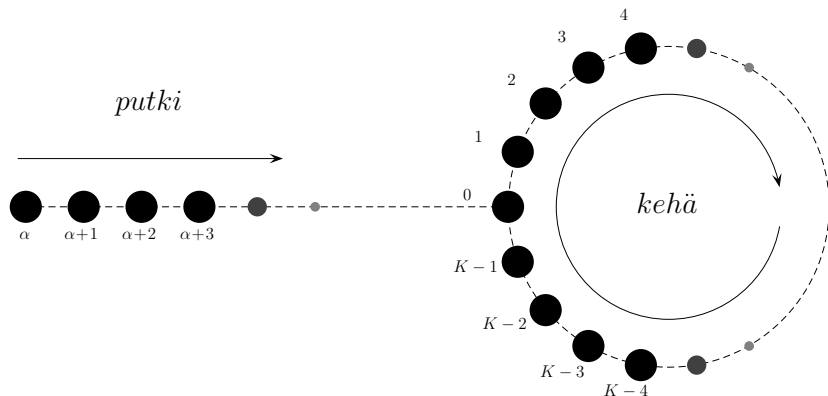
# Hajautettu järjestelmä

- äärellinen suuntaamaton yhtenäinen verkko  $G = (V, E)$
- jokaisella solmulla  $p$  kellorekisteri  $r_p$
- solmun  $p$  naapurien joukko on  $\mathcal{N}_p$
- naapurit näkevät toistensa muuttujat, ja laillisuusehdot ovat paikallisia
- ohjelma kokoelma ilmaisuja

*ehto*  $\longrightarrow$  *toimenpide*

- suoritusvuorot jakaa epäreilu demoni

# Äärellinen askeltava järjestelmä



# Äärellinen askeltava järjestelmä

- kellorekisterin arvot joukossa

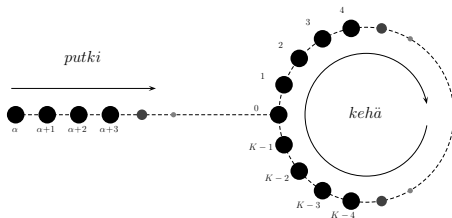
$$\mathcal{X} = \{\alpha, \alpha + 1, \dots, 0, \dots, K - 2, K - 1\}$$

- $putki_{\varphi} = \{\alpha, \alpha + 1, \dots, -1, 0\}$
- $putki_{\varphi}^* = putki_{\varphi} \setminus \{0\}$
- $kehä_{\varphi} = \{0, 1, \dots, K - 2, K - 1\}$

# Äärellinen askeltava järjestelmä

- kellorekisterin arvoa kasvattaa funktio  $\varphi : \mathcal{X} \rightarrow \mathcal{X}$

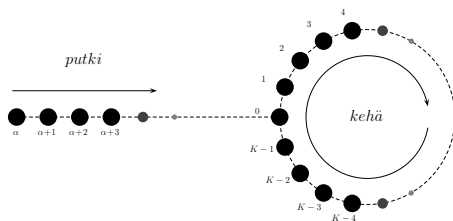
$$\varphi(x) = \begin{cases} x + 1, & \text{jos } x < K - 1 \\ 0, & \text{jos } x = K - 1. \end{cases}$$



# Äärellinen askeltava järjestelmä

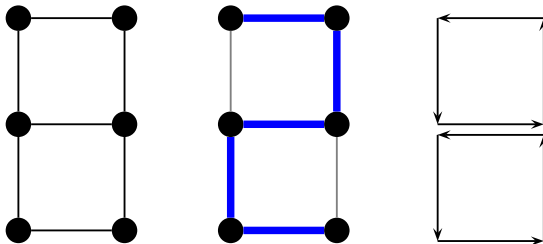
- paikallinen vähennysoperaattori

$$b \ominus_l a = \begin{cases} -1, & \text{jos } a \equiv b + 1 \pmod{K} \\ 0, & \text{jos } a = b \\ 1, & \text{jos } b \equiv a + 1 \pmod{K} \end{cases}$$



# Verkkoteoriaa

- syklikanta (Kirchhoff 1847)



- lyhin syklikanta
- jänteetön sykli eli reikä

# Proessin $p$ ohjelma

## Totuusarvoiset makrot

$ehto(p, q)$	prosessien $p$ ja $q$ synkronointiehto
$kehällä(p, q)$	$\equiv r_p \in kehä_\varphi \wedge r_q \in kehä_\varphi$
$askelehto(p, q)$	$\equiv kehällä(p, q) \wedge$ $((\varphi(r_p) = r_q) \vee ((r_p = r_q) \wedge ehto(p, q)))$
$pariehto(p, q)$	$\equiv askelehto(p, q) \vee askelehto(q, p)$
$naapurustoehto(p)$	$\equiv \forall q \in \mathcal{N}_p : pariehto(p, q)$

---

$kehäaskel(p)$	$\equiv \forall q \in \mathcal{N}_p : askelehto(p, q)$
$palautusaskel(p)$	$\equiv \neg naapurustoehto(p) \wedge (r_p \notin putki_\varphi)$
$putkiaskel(p)$	$\equiv r_p \in putki_\varphi^* \wedge$ $(\forall q \in \mathcal{N}_p : (r_q \in putki_\varphi) \wedge (r_p \leq r_q))$

## Ohjelma

$kehäaskel(p)$	$\longrightarrow$ <b>tee jotakin;</b> $r_p := \varphi(r_p)$
$palautusaskel(p)$	$\longrightarrow r_p := \alpha$
$putkiaskel(p)$	$\longrightarrow r_p := \varphi(r_p)$

## Välttämättömät lisäoletukset

Jos määriteltäisiin esimerkiksi  $ehto(p, q)$  identtisesti epätodeksi, algoritmissa ei olisi mitään mieltä.

$$\begin{aligned}\mathcal{O}_1 & : (askelehto(p, q) \Rightarrow askelehto(\Psi(p), \Psi(q))) \\ & \quad \wedge (askelehto(p, q) \Rightarrow pariehto(\Psi(p), q)) \\ \mathcal{O}_2 & : (r_p = r_q = 0) \Rightarrow pariehto(p, q)\end{aligned}$$

## Tärkeitä käsitteitä

- *palautusvaiheessa* oleva solmu
- algoritmin tarkastelussa keskeinen käsite: *kehälle vakiintunut*

$$\forall p \in V : \text{naapurustoehto}(p)$$

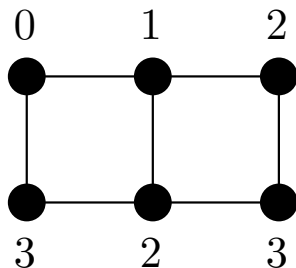
- kehälle vakiintuneessa järjestelmässä mikään prosessi ei ole palautusvaiheessa

## Muutama esimerkki

- asynkroninen unisono:  $ehto(p, q)$  aina tosi
- geneerinen algoritmi:  $ehto(p, q) \equiv v_p \triangleleft v_q$
- keskinäinen poissulkeminen:  $ehto(p, q) \equiv id_p \triangleleft id_q$
- ryhmien keskinäinen poissulkeminen (naapuriprosessit eivät saa käyttää yhtäaikaisesti eri resursseja):  
 $ehto(p, q) \equiv resurssi_p = resurssi_q \vee id_p \triangleleft id_q$

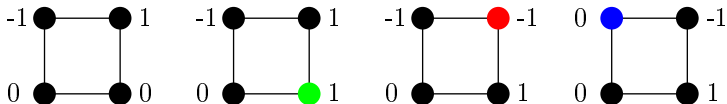
## Miten valitaan parametrit $\alpha$ ja $K$ ?

- jos parametrit valitaan väärin, voi käydä huonosti
- liian pieni  $K$ :n arvo voi johtaa lukkiutuneeseen tilanteeseen (alla  $K = 4$ )



## Miten valitaan parametri $\alpha$ ja $K$ ?

- jos  $\alpha$  on itseisarvoltaan liian pieni, toipuminen ei ehkä pääty koskaan (alla  $\alpha = -1$ )

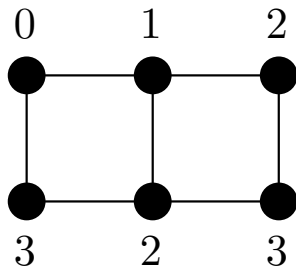


## Viiveen määritelmä

- oletuksena on, että järjestelmä on kehälle vakiintunut
- siis naapurisolmujen kellorekisterien arvot poikkeavat toisistaan enintään yhdellä pykälällä suuntaan tai toiseen
- polun  $\mu = p_0 p_1 \dots p_k$  viive  $\Delta_\mu$  määritellään:

$$\Delta_\mu = \sum_{i=0}^{k-1} (r_{p_{i+1}} \ominus_l r_{p_i})$$

## Esimerkki viiveen laskemisesta



- yllä pitkän syklin viive on kiertosuunnasta riippuen 4 tai  $-4$  ja  $K = 4$

## Viiveen ominaisuuksia

- $\Delta_\mu = -\Delta_{\tilde{\mu}}$ , missä  $\mu$  ja  $\tilde{\mu}$  ovat toistensa käänteispolkuja
- viiveen lineaarisuus:  $\Delta_{\mu_1\mu_2} = \Delta_{\mu_1} + \Delta_{\mu_2}$
- polulla  $\mu = p_0p_1 \dots p_k$  pätee  $r_{p_0} + \Delta_\mu \equiv r_{p_k} \pmod{K}$
- syklissä  $\mu = p_0p_1 \dots p_kp_0$  siis
$$r_{p_0} + \Delta_\mu \equiv r_{p_0} \pmod{K}$$
$$\Leftrightarrow \Delta_\mu \equiv 0 \pmod{K}$$
- on muistettava, että näissä tarkasteluissa järjestelmä on kehälle vakiintunut

## Kohti johtopäätöksiä

- viive on *rajoitettu*  $\iff$  viive on 0 jokaisessa syklissä  
→ nyt kahden prosessin välinen viive ei riipu polun valinnasta
- mielivaltaisen syklin viive on muuttumaton kehälle vakiintuneessa järjestelmässä
- erityisesti jos viive on jossakin vaiheessa rajoitettu, se myös pysyy rajoitettuna
- voidaan helposti todeta: jos viive on rajoitettu, kehälle vakiintunut järjestelmä ei voi lukkiutua

## Kohti johtopäätöksiä

- viive on *rajoitettu*  $\iff$  viive on 0 jokaisessa syklissä  
→ nyt kahden prosessin välinen viive ei riipu polun valinnasta
- mielivaltaisen syklin viive on muuttumaton kehälle vakiintuneessa järjestelmässä
- erityisesti jos viive on jossakin vaiheessa rajoitettu, se myös pysyy rajoitettuna
- voidaan helposti todeta: jos viive on rajoitettu, kehälle vakiintunut järjestelmä ei voi lukkiutua

## Syklikantaa tarvitaan nyt

- lausutaan syklin viive syklikannan avulla:

$$\Delta(C_k) = \Delta \left( \sum_{i=1}^{k_m} a_{k_i} C_{k_i} \right) = \sum_{i=1}^{k_m} a_{k_i} \Delta(C_{k_i})$$

- viive on rajoitettu järjestelmässä täsmälleen silloin, kun viive on 0 kaikissa verkon mielivaltaisen syklikannan sykleissä
- tarkastellaan seuraavaksi sitä syklikantaa, jonka pisin sykli on mahdollisimman lyhyt

# Lukkiutumattomuustulos

- oletetaan, että verkossa on ainakin yksi sykli
- olkoon verkon minimaalisen syklikannan pisimmän syklin pituus  $C_G$
- kuuluukoon sykli  $\mu$  minimaaliseen syklikantaan
- järjestelmä on kehälle vakiintunut, joten kahden naapurisolmun välisen viiveen itseisarvo on korkeintaan 1
- siis  $|\Delta_\mu| \leq C_G$
- muistetaan, että syklissä  $\Delta_\mu \equiv 0 \pmod{K}$
- jos valitaan  $K > C_G$ , pätee  $\Delta_\mu = 0$

# Lukkiutumattomuustulos

- oletetaan, että verkossa on ainakin yksi sykli
- olkoon verkon minimaalisen syklikannan pisimmän syklin pituus  $C_G$
- kuuluukoon sykli  $\mu$  minimaaliseen syklikantaan
- järjestelmä on kehälle vakiintunut, joten kahden naapurisolmun välisen viiveen itseisarvo on korkeintaan 1
- siis  $|\Delta_\mu| \leq C_G$
- muistetaan, että syklissä  $\Delta_\mu \equiv 0 \pmod{K}$
- jos valitaan  $K > C_G$ , pätee  $\Delta_\mu = 0$

# Lukkiutumattomuustulos

- oletetaan, että verkossa on ainakin yksi sykli
- olkoon verkon minimaalisen syklikannan pisimmän syklin pituus  $C_G$
- kuuluukoon sykli  $\mu$  minimaaliseen syklikantaan
- järjestelmä on kehälle vakiintunut, joten kahden naapurisolmun välisen viiveen itseisarvo on korkeintaan 1
- siis  $|\Delta_\mu| \leq C_G$
- muistetaan, että syklissä  $\Delta_\mu \equiv 0 \pmod{K}$
- jos valitaan  $K > C_G$ , pätee  $\Delta_\mu = 0$

# Lukkiutumattomuustulos

- oletetaan, että verkossa on ainakin yksi sykli
- olkoon verkon minimaalisen syklikannan pisimmän syklin pituus  $C_G$
- kuuluukoon sykli  $\mu$  minimaaliseen syklikantaan
- järjestelmä on kehälle vakiintunut, joten kahden naapurisolmun välisen viiveen itseisarvo on korkeintaan 1
- siis  $|\Delta_\mu| \leq C_G$
- muistetaan, että syklissä  $\Delta_\mu \equiv 0 \pmod{K}$
- jos valitaan  $K > C_G$ , pätee  $\Delta_\mu = 0$

# Lukkiutumattomuustulos

- oletetaan, että verkossa on ainakin yksi sykli
- olkoon verkon minimaalisen syklikannan pisimmän syklin pituus  $C_G$
- kuuluukoon sykli  $\mu$  minimaaliseen syklikantaan
- järjestelmä on kehälle vakiintunut, joten kahden naapurisolmun välisen viiveen itseisarvo on korkeintaan 1
- siis  $|\Delta_\mu| \leq C_G$
- muistetaan, että syklissä  $\Delta_\mu \equiv 0 \pmod{K}$
- jos valitaan  $K > C_G$ , pätee  $\Delta_\mu = 0$

# Lukkiutumattomuustulos

- oletetaan, että verkossa on ainakin yksi sykli
- olkoon verkon minimaalisen syklikannan pisimmän syklin pituus  $C_G$
- kuuluukoon sykli  $\mu$  minimaaliseen syklikantaan
- järjestelmä on kehälle vakiintunut, joten kahden naapurisolmun välisen viiveen itseisarvo on korkeintaan 1
- siis  $|\Delta_\mu| \leq C_G$
- muistetaan, että syklissä  $\Delta_\mu \equiv 0 \pmod{K}$
- jos valitaan  $K > C_G$ , pätee  $\Delta_\mu = 0$

# Lukkiutumattomuustulos

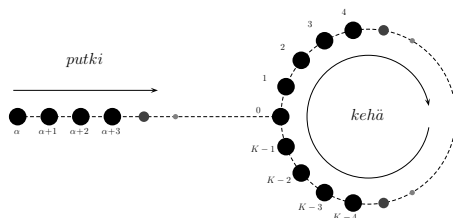
- oletetaan, että verkossa on ainakin yksi sykli
- olkoon verkon minimaalisen syklikannan pisimmän syklin pituus  $C_G$
- kuuluukoon sykli  $\mu$  minimaaliseen syklikantaan
- järjestelmä on kehälle vakiintunut, joten kahden naapurisolmun välisen viiveen itseisarvo on korkeintaan 1
- siis  $|\Delta_\mu| \leq C_G$
- muistetaan, että syklissä  $\Delta_\mu \equiv 0 \pmod{K}$
- jos valitaan  $K > C_G$ , pätee  $\Delta_\mu = 0$

## Mitä äsken näytettiin?

Jos valitaan  $K > C_G$ ,

- minimaalisen syklikannan kaikkien syklien viiveet ovat 0
- viive on 0 verkon kaikissa sykleissä
- koska viive on rajoitettu, järjestelmä ei voi lukkiutua

Parametrin  $K$  arvo määräytyy siis ainoastaan verkon minimaalisen syklikannan pisimmän syklin mukaan.



## Miten tästä eteenpäin?

- kun valitaan riittävän iso  $K$ , kehälle vakiintunut järjestelmä ei voi lukkiintua
- itse asiassa on helppo näyttää, että lukkiutunut järjestelmä on aina kehälle vakiintunut!
- tavoitteena on näyttää, että palautuksia ei olla äärettömän monta
- raapaistaan pintaa

## Miten tästä eteenpäin?

- kun valitaan riittävän iso  $K$ , kehälle vakiintunut järjestelmä ei voi lukkiintua
- itse asiassa on helppo näyttää, että lukkiutunut järjestelmä on aina kehälle vakiintunut!
- tavoitteena on näyttää, että palautuksia ei olla äärettömän monta
- raapaistaan pintaa

# Palautusverkko

- palautusverkon solmu on pari  $(p, t)$
- solmu  $p$  palautetaan ajanhetkellä  $t$
- verkon särmä kuvaa palautuksen etenemistä järjestelmää kuvaavassa verkossa
- palautusverkko on suunnattu ja sykliiton
- jos palautusverkko on äärellinen, algoritmi on vakautuva kehälle vakiintuneisuuden suhteen

## Parametrin $\alpha$ valitseminen

- nähtiin, että itseisarvoltaan liian pieni  $\alpha$  johtaa ongelmiin
- puumaiset verkot nytkin ongelmattomia
- ratkaiseva asia on nyt verkon pisimmän reiän eli jänteettömän syklin pituus  $T_G$
- valitsemalla  $|\alpha| \geq T_G - 2$  algoritmi on vakautuva

# Lopuksi

- äärellinen askeltava järjestelmä
- $K$ :n täytyy olla suurempi kuin minimaalisen syklikannan pisimmän syklin pituus
- $\alpha$ :n itseisarvon täytyy vähintään yhtä suuri kuin verkon pisimmän reiän pituus  $-2$

