Paikallisuus hajautetuissa verkkoalgoritmeissa

Juhana Laurinharju

Tieteellinen kirjoittaminen
Helsingin Yliopisto
Tietojenkäsittelytieteen laitos

Helsinki, 27. helmikuuta 2013

Johdanto

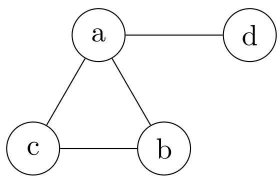
1 Määritelmiä

1.1 Verkko

Suuntaamaton verkko on pari $G = (V, E)$, missä V on *solmujoukko* ja E on *kaarijoukko*. *Kaari* solmusta $v \in V$ solmuun $u \in V$ on kaksikko $\{v, u\} \in E$. Kaarta voidaan myös merkitä lyhyemmin vu. Esimerkiksi verkko $G = (V, E)$, missä

$$\begin{aligned} V &= \{a, b, c, d\} \text{ ja} \\ E &= \{\{a, b\}, \{b, c\}, \{c, a\}, \{a, d\}\} \\ &= \{ab, bc, ca, ad\} \end{aligned}$$

näyttää seuraavalta



1.2 Laskennan malli

Olkoon $G = (V, E)$ suuntaamaton verkko. Verkon jokaisessa solmussa $v \in V$ on tietokone. Laskenta koostuu *kommunikointikierroksista*. Yhden kommunikaatiokierroksen aikana jokainen solmu voi:

1. suorittaa mielivaltaista laskentaa
2. lähettää viestin jokaiselle naapurilleen
3. vastaanottaa naapureiden lähettämät viestit

Lisäksi jokaiselle solmulle $v \in V$ on annettu yksikäsitteinen tunniste $\mathrm{ID}(v) \in \{1, \ldots, |V|\}$. Laskennan päätyttyä jokaisen solmun tulee tietää oma tulosteensa.

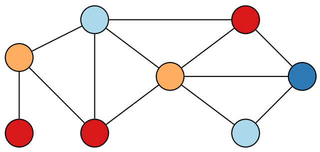
TODO: motivointia sille, että tarkastellaan vain kommunikaatiokierrosten lukumäärää aikavaativuutena.

1.3 Verkon väritys

Verkko on *väritetty*, jos jokaiseen solmuun $v \in V$ on liitetty jokin *väri* ja kahdella vierekkäisellä solmulla ei koskaan ole samaa väriä. Tarkemmin, verkon

$G = (V, E)$ *solmuväritys* on kuvaus $c : V \to \{1, \dots, k\}$ jollain luonnollisella luvulla $k \in \mathbb{N}$. Lisäksi vaaditaan, että jos verkossa on kaari solmusta v solmuun u, eli $vu \in E$, niin $c(v) \neq c(u)$.

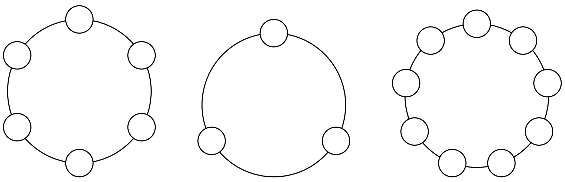
Verkon voi värittää k:lla värillä jos löytyy yllä olevan ehdon täyttävä kuvaus $c : V \to \{1, \dots, k\}$. Tällaista väritystä kutsutaan *k-väritykseksi*.



Jos verkkoa väritetään hajautetulla algoritmilla, niin jokaisen solmun tulee tietää oma värinsä laskennan päätyttyä.

1.4 Sykli

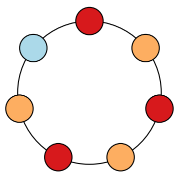
Verkko on *sykli*, jos se on yhtenäinen ja sen jokaisella solmulla on tasan kaksi naapuria.



Tarkemmin sanoen, n-sykli, missä $n \geq 3$, on verkko $C_n = (V, E)$ jolla

$$\begin{aligned} V &= \{v_1, v_2, \dots, v_n\} \\ E &= \{v_i v_{i+1} \mid 1 \leq i < n\} \cup \{v_n v_1\} \end{aligned}$$

Syklin voi aina värittää kolmella värillä.



1.5 Iteroitu logaritmi $\log^*$

Iteroitu logaritmi $\log^*$ kertoo kuinka monta kertaa luvusta täytyy ottaa logaritmi, kunnes lopputulos on korkeintaan yksi. Tarkemmin,

$$\log^* x = \begin{cases} 0, & \text{jos } x \leq 1, \\ 1 + \log^*(\log x), & \text{muutoin.} \end{cases}$$

Esimerkiksi

$$\begin{aligned}\log^* 16 &= \log^* 2^{2^2} = 1 + \log^* 2^2 \\ &= 2 + \log^* 2 = 3 + \log^* 1 = 3\end{aligned}$$

ja

$$\begin{aligned}\log^* 65536 &= \log^* 2^{2^{2^2}} = 1 + \log^* 16 \\ &= 4,\end{aligned}$$

joten $\log^* n$ on arvoltaan pienempi kuin 5 kun $n < 2^{65536}$. Iteroitu logaritmi on siis äärimmäisen hitaasti kasvava funktio.

1.6 Näkymä

Hajautetussa algoritmissa solmu $v \in V$ saa k:ssa kierroksessa selville oman k-ympäristönsä. Toisaalta solmu ei pysty tässä ajassa saamaan mitään selville solmuista, joiden etäisyys v:stä on yli k. Hajautettu algoritmi, jonka ajoaika on k kierrosta on siis funktio, jonka lähtöjoukkona on solmujen mahdolliset k-ympäristöt.

TODO: määrittele etäisyys

TODO: tää kaipaa varmaan vähän selvennystä

TODO: kuvasarja havainnollistamaan tätä

Erityisesti syklissä algoritmi, jonka ajoaika on k kierrosta, tekee päätöksensä k:n edeltäjän, k:n seuraajan ja oman solmun tunnisteiden perusteella. Toisin sanoen solmun $v_l \in V(C_n)$ tuloste on funktio arvoilta

$$\left(\mathrm{ID}(v_{l-k}), \mathrm{ID}(v_{l-k+1}), \ldots, \mathrm{ID}(v_{l-1}), \mathrm{ID}(v_l), \mathrm{ID}(v_{l+1}), \ldots, \mathrm{ID}(v_{l+k})\right),$$

Missä yhteen- ja vähennyslaskut suoritetaan modulo n.

TODO: Kuvasarja solmun näkymästä syklissä.

Erityisesti jos algoritmi tuottaa 3-värityksen syklissä k:ssa kierroksessa, niin täytyy olla olemassa sellainen funktio $f : [n]^{2k+1} \to [3]$, joka tuottaa laillisen 3-värityksen riippumatta siitä miten solmuille on annettu tunnisteet.

1.7 Naapurustoverkot

TODO: liitä tää syklien näkymiin

Naapurustoverkko $B_{t,n} = (V, E)$, missä V on kaikkien vektoreiden $(x_1, \ldots, x_{2t+1})$ joukko joilla x_i:t ovat keskenään erisuuria kokonaislukuja joukosta $[n]$. Verkossa $B_{t,n}$ solmut muotoa

$$(x_1, \ldots, x_{2t+1}) \text{ ja } (y, x_2, \ldots, x_{2t})$$

ovat naapureita, kun $y \neq x_{2t+1}$.

TODO: esittele $[n]$ merkintä

Verkossa $B_{t,n}$ on $n(n-1)(n-2)\cdots(n-2t)$ solmua ja sen kaikkien kaarten asteluku on $2(n-2t-1)$.

TODO: termi asteluku esittelemättä

TODO: kierrosmäärä on välillä k ja välillä t

Hajautettu algoritmi, joka 3-värittää syklin t kierroksessa on funktio $c : V(B_{t,n}) \to [3]$.

TODO: tätä vois perustella

Nyt c on myös laillinen 3-väritys verkolle $B_{t,n}$, sillä jos c antaa solmuille

$$(x_1, \ldots, x_{2t+1}) \text{ ja } (y, x_2, \ldots, x_{2t})$$

saman värin, niin se antaa myös syklissa kahdelle vierekkäiselle solmulle saman värin kun syklissä esiintyy pätkä

$$y, x_1, x_2, \ldots, x_{2t+1}.$$

Siis jos näytetään, että verkkoa $B_{t,n}$ ei voi 3-värittää, niin ei voi myöskään olla hajautettua algoritmia joka värittäisi n-syklin kolmella värillä t kierroksessa.