VERKKOJEN MITOITUKSESTA

1. Piirikytkentäiset verkot

- Linkkien kapasiteettien (johtoja/linkki) määrääminen siten, että verkon kokonaiskustannukset minimoituvat, kun päästä-päähän estolle on annettu yläraja.
- Moen periaate

2. Pakettikytkentäiset verkot

- "Neliöjuurimenetelmä"
- Linkkien kapasiteettien määrääminen siten, että paketin keskiviive verkossa minimoituu, kun kustannuksille on asetettu yläraja.

Piirikytkentäisten verkkojen mitoituksesta

Piirikytkentäisen verkon mitoitustehtävä koskee yleensä seuraavaa asetelmaa. On annettu

- Verkon solmupisteet ja topologia (tieto siitä, mitkä solmupisteet on yhdistetty keskenään suoralla linkillä)
- Liikennematriisi, joka kertoo tarjotun liikenteintensiteetin $a_{i,j}$ mistä tahansa lähtösolmusta i mihin tahansa kohdesolmuun j. Tarjottu liikenne oletetaan poissoniseksi.
- Vaadittu palvelun laatutaso ilmaistuna sallitun eston B suuruutena. Periaatteessa sallittu esto $B_{i,j}$ voi olla spesifioitu erikseen jokaiselle yhteysvälille (i, j).
- Linkkikustannukset; riippuvuus linkin kapasiteetista ja pituudesta.

Tehtävänä on mitoittaa linkkien kapasiteetit C_j siten, että

- kaikkien yhteyksien päästä-päähän estot ovat annetun rajan alapuolella ja että tämän ehdon vallitessa
- verkon kustannukset minimoituvat

Mitoitustehtävä

Kysymyksessä on tyypillinen epälineaarisen ohjelmoinnin (optimoinnin) tehtävä.

- Rajoitushdot riippuvat epälineaarisesti kapasiteeteista.
- Kustannukset saattavat riippua epälineaarisesti kapasiteeteista.

On huomattava, että päästä-päähän estojen laskenta sinänsä on oma ongelmansa. Tarkka laskenta on yleensä mahdotonta. Kysymystä voidaan lähestyä kolmella eri tarkkuustasolla:

1. Karkein approksimaatio muodostuu siitä, että
   - oletetaan kuhunkin linkkiin tarjottu liikenne poissoniseksi
   - liikenteen intensiteetti summaksi kaikkien linkin kautta kulkevien liikennevirtojen intensiteeteistä
   - lasketaan linkin esto Erlangin kaavalla
   - arvioidaan päästä-päähän estoa linkkiestojen summalla
2. Tätä voidaan parantaa ns. ohennetun kuorman menetelmällä, jota käsitellään myöhemmin
3. Joissakin tapauksissa (mm. liityntäverkko) tarkka analyysi on mahdollinen. Tähän palataan kurssin lopussa.

Mitoitustehtävä

Seuraavassa tyydytään karkeimman approksimaation käyttöön päästä-päähän estojen laskennassa.

Perinteisen piirikytkentäisen verkon tapauksessa kysymyksessä on lisäksi kokonaislukuoptimointi, koska kapasiteetti (johtojen määrä) on luonteeltaan diskreetti.

Sinänsä epälineaarisen optimointitehtävän ratkaiseminen on standarditehtävä ja siihen on käytettävissä erilaisia valmiita ohjelmapaketteja. Erityisesti verkon mitoitusta varten on kehitetty omia erikoisohjelmiaan.

Seuraavassa tarkastellaan yksinkertaisen Moen periaatteen käyttöä optimointitehtävän ratkaisemisessa. [1]

[1] Esitys tältä osin on peräisin K. Kilkin luennoista

Moen periaate yhden linkin tapauksessa

Moen periaatteessa on kysymys inkrementaalisesta suurimman hyöty/kustannus-suhteen lähestymistavasta.

Tarkastellaan ensin yhtä linkkiä, jonka johtoluku on n ja tarjottu liikenneintensiteetti on a. Jos johtolukua lisätään yhdellä, niin välitetyn liikenteen intensiteetti kasvaa määrällä

$$\Delta a = a(E(n,a) - E(n+1,a)) \ , \quad \text{missä } E(n,a) \text{ on Erlangin estofunktio (B-kaava).}$$

Tämän avulla voidaan arvioida, vastaako lisäjohdosta saatava hyöty kustannuksia.

- Johtoja kannattaa lisätä niin kauan kuin hyöty on suurempi kuin kustannukset.
- Johtoja lisättäessä uudesta johdosta saatava lisähyöty pienenee koko ajan.

Moen periaate verkossa

Verkkotapauksessa Moen periaateen mukaisesti toimittaessa lähdetään liikkeelle verkosta, joka on alimitoitettu ja jossa mikään päästä-päähän esto ei täytä asetettua vaatimusta.

Tämän jälkeen lisätään kapasiteettia yksi johto kerrallaan sinne, missä se on kustannustehokkainta eli sinne, missä tietyllä hinnalla saadaan suurin päästä-päähän eston pieneneminen.

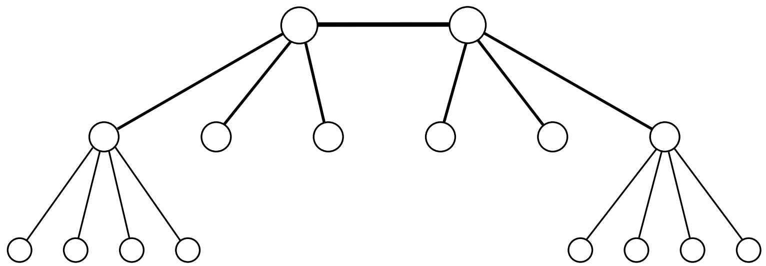
Tätä jatketaan niin kauan, kunnes kaikki rajoitusehdot on täytetty eli kaikki päästä-päähän estot saatu annetun rajan B alle.

Tilannetta mutkistaa se, että tietyn linkin kautta kulkee useita yhteyksiä ja kapasiteetin lisäyksen vaikutus näihin yhteyksiin on erilainen. Tällöin voidaan ottaa tarkasteltavaksi esim. vaikutus kyseisen linkin kautta kulkevista (palvelurajan alittavista) yhteyksistä heikointa palvelua saavan yhteyden laatuun.

Moen periaatteen mukainen inkrementaalinen lähestymistapa ei välttämättä johda globaaliin optimiin mutta antaa yleensä kelvollisen ratkaisun.

Moen periaate symmetrisessä hierarkkisessa verkossa

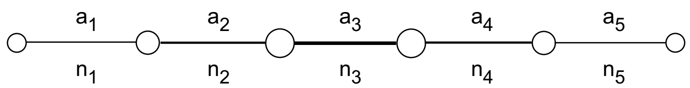
Tutkitaan esimerkkinä kuvan mukaista symmetristä hierarkkista verkkoa. Verkossa on viiden hierarkkiatason solmuja ja viiden eri hierarkkiatasojen linkkejä. Merkitään linkkien lukumäärää hierarkkiatasolla i m_i:llä.



Oletetaan, että kaikki tietyllä hierarkkiatasolla olevat linkit ovat samassa asemassa: linkkien pituudet ovat samat ja niiden kautta kulkee sama määrä (tarjottua) liikennettä: tasolla i intensiteetti on a_i.

Tällöin on selvää, että optimimitoituksessa saman tason linkeille kannattaa antaa sama kapasiteetti n_i.

Verkkoa voidaan kuvata yksinkertaisemmalla kaaviolla.



Moen periaate symmetrisessä hierarkkisessa verkossa (jatkoa)

Moen periaatteen mukaisesti johtoja lisätään sille tasolle, jolla saavutetaan suurin eston pienenminen kustannusyksikköä kohti. Tätä varten lasketaan suhde

$$h_i = \frac{E(n_i, a_i) - E(n_i + 1, a_i)}{m_i c_i}\ , \qquad \text{missä}$$

$$\begin{cases} n_i &= \text{johtojen lukumäärä yhdessä tason } i \text{ linkissä} \\ m_i &= \text{tason } i \text{ linkkien lukumäärä} \\ c_i &= \text{yhden johdon kustannus tasolla } i \\ a_i &= \text{tarjotun liikenteen intensiteetti yhdellä tason } i \text{ linkillä} \end{cases}$$

Etsitään se taso, jolla h_i on suurin ja lisätään yksi johto kaikille tämän tason linkeille (yhteensä m_i kappaletta).

Jatketaan kunnes päästä-päähän esto $\sum_i E(n_i, a_i)$ tulee pienemmäksi kuin B.

Symmetrisen hierarkkisen verkon mitoitus: esimerkki

Optimoidaan verkko, jonka parametrit ovat oheisen taulukon mukaiset.

Tavoitteena on päästä alle 3 %:n kokonaisestoon.

Lähdetään liikkeelle alla vasemmalla olevan taulukon alkutilanteesta.

Suurin h_i on tasolla 3. Johtoja kannattaa lisätä sinne, kunnes tämän tason h_i on tullut pienemmäksi kuin seuraavaksi suurin h_i tasolla 2.

Näin jatketaan, kunnes kokonaisesto on tullut pienemmäksi kuin 3 %. Lopputulos on oikeanpuoleisessa taulukossa.

i	m_i	c_i	a_i
1	50	100	3
2	5	200	25
3	1	300	100
4	10	200	12
5	100	100	2

i	n_i	$B(n_i)$%	$B(n_i+1)$%	kust. Meuro	lisäkust. Meuro	h_i %/Meuro
1	7	2.19	0.81	35	50	0.3
2	33	2.28	1.65	33	10	0.6
3	110	2.75	2.41	33	3	1.1
4	18	2.65	1.65	36	20	0.5
5	6	1.21	0.34	60	100	0.1
	yht.	11.08		197		

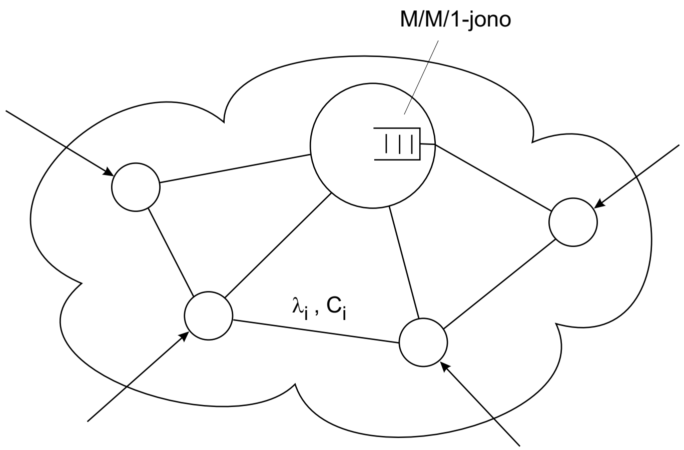
i	n_i	B_i %	kust. Meuro
1	8	0.81	40
2	39	0.23	39
3	126	0.16	38
4	21	0.56	42
5	6	1.21	60
	yht.	2.96	219

Lopullisessa ratkaisussa estoa on vähemmän suuriliikenteisillä linkeillä.
Jos kokonaisesto olisi jaettu tasan linkkien kesken, olisi kustannus ollut 230 Meuro.

Pakettiverkon mitoitus (“neliöjuurimentelmä”)

- Pakettiverkossa on M linkkiä, $i = 1, \ldots, M$.
- Välitettävien sanomien kokojakauma oletetaan eksponentiaaliseksi; keskipituus $1/\mu$ (bittiä)
- Mallinnetaan pakettiverkon solmujen (reititimien) lähtöpuskureita $M/M/1$-jonoina.

$\Lambda \Rightarrow$



$$\begin{cases} \lambda_i & = \text{pakettivirta (pkt/s) linkillä } i \\ C_i & = \text{linkin } i \text{ kapasiteetti} \\ d_i & = \text{linkin } i \text{ ominaiskustannus (mk/bps)} \\ 1/\mu & = \text{sanoman keskikoko (bit)} \\ \Lambda & = \text{verkon kaikkiin solmuihin yhteensä tuleva kokonaispakettivirta (pkt/s)} \\ & \text{huom. } \Lambda \neq \sum_i \lambda_i \end{cases}$$

Paketin keskimääräinen lähetysaika linkillä i $1/(\mu C_i)$ $\quad\Rightarrow\quad$ linkin välityskyky μC_i (pkt/s).

Paketin keskimääräinen viipymisaika linkillä i (jonotus + lähetys)

$$\boxed{T_i = \frac{1}{\mu C_i - \lambda_i} = \frac{1}{\mu C_i} \cdot \frac{1}{1 - \lambda_i/\mu C_i}} \qquad \rho_i = \lambda_i/\mu C_i \text{ linkin } i \text{ kuorma}$$

Optimoinnin tavoite ja rajoitusehto

Halutaan minimoida verkkoon saapuvan paketin keskimääräinen viipymäaika verkossa

$$\boxed{T = \frac{1}{\Lambda}\sum_i \lambda_i T_i}$$

- Tämä on painotettu keskiarvo viiveistä eri linkeillä.
- Voidaan tulkita myös Littlen tuloksen perusteella (koko verkko mustassa laatikossa):

$$\begin{cases} \lambda_i T_i & = \text{ keskimääräinen pakettien lkm (jonottavat + lähetyksessä oleva) puskurissa } i \\ \sum_i \lambda_i T_i & = \text{ pakettien keskimäärärinen lkm koko verkossa} \\ \frac{1}{\Lambda}\sum_i \lambda_i T_i & = \text{ paketin keskimääräinen viipymä verkossa} \end{cases}$$

Optimoinnin reunaehtona on, että verkon kokonaiskustannus ei saa ylittää annettua arvoa D,

$$\sum_i d_i C_i \le D$$

On selvää, että viive minimoituu, kun kaikki raha käytetään kapasiteettien lisäämiseen, joten reunaehto voidaan korvata yhtäsuuruudella

$$\boxed{\sum_i d_i C_i = D}$$

Tehtävänä on etsiä kapasiteetit C_i siten, että tämän ehdon vallitessa T minimoituu.

Minimointi rajoitusehdon vallitessa, Lagrangen kertoja

Minimointitehtävän reunaehto otetaan huomioon käyttäen Lagrangen kertojamenettelyä

- Minimoitavaan funktioon lisätään reunaehdon määrittelevä funktio kerrottuna toistaiseksi määräämättömällä kertoimella β. Minimoitava funktio on siten

$$G = \frac{1}{\Lambda} \sum_i \frac{\lambda_i}{\mu C_i - \lambda_i} + \beta \Big(\sum_i d_i C_i - D \Big)$$

- Jokaisella arvolla β funktiolle G voidaan etsiä (globaali, rajoittamaton) minimi.
- Minimin sijainti ja arvo riippuvat parametrista β.
- Määrätään nyt β siten, että minimipiste totetuttaa reunaehdon eli että $\sum_i d_i C_i - D = 0$.
- Tällöin lausekkeen G globaali minimi sijaitsee reunaehdon määräämällä hyperpinnalla.
- Funktion G rajoitus hyperpinnalle $\sum_i d_i C_i - D = 0$ on (β:sta riippumatta) sama kuin alkuperäinen kohdefunktio.
- G:n globaali minimi on varmasti myös hyperpinnalle rajoitetun funktion minimi eli alkuperäisen kohdefunktion minimi hyperpinnalla.
- Löydetty ratkaisu siis minimoi alkuperäisen kohdefunktion ja totetuttaa vaaditun reunaehdon eli on etsitty tehtävän ratkaisu.

Minimoinnin suorittaminen

$$G = \frac{1}{\Lambda}\sum_i \frac{\lambda_i}{\mu C_i - \lambda_i} + \beta\Big(\sum_i d_i C_i - D\Big)$$

$$\begin{aligned}
\frac{\partial G}{\partial C_i} &= \frac{1}{\Lambda}\frac{-\lambda_i \mu}{(\mu C_i - \lambda_i)^2} + \beta d_i = 0, \qquad i = 1,\ldots,M \\
&\Rightarrow\ (\mu C_i - \lambda_i)^2 = \frac{\lambda_i \mu}{\Lambda \beta d_i} \\
&\Rightarrow\ C_i = \frac{\lambda_i}{\mu} + \frac{1}{\sqrt{\Lambda\beta\mu}}\sqrt{\frac{\lambda_i}{d_i}} \qquad \text{(miinusmerkki ei tule kysymykseen)}
\end{aligned}$$

Ratkaistaan nyt β reunaehdosta

$$D = \sum_i d_i C_i = \sum_i d_i \frac{\lambda_i}{\mu} + \frac{1}{\sqrt{\Lambda\beta\mu}}\sum_i \sqrt{\lambda_i d_i} \quad \Rightarrow \quad \frac{1}{\sqrt{\Lambda\beta\mu}} = \frac{D - \sum\limits_i d_i \dfrac{\lambda_i}{\mu}}{\sum\limits_i \sqrt{\lambda_i d_i}}$$

- λ_i/μ on linkin i keskimääräinen bittivirta; vähintään tämä kapasiteetti tarvitaan linkillä i, jotta kuorma ylipäätään kyetään kuljettamaan.
- Vastaavasti $\sum\limits_i d_i \dfrac{\lambda_i}{\mu}$ on järjestelmän vähimmäiskustannus.

Minimointi (jatkoa)

Merkitään

$$D_e = D - \sum_i d_i \frac{\lambda_i}{\mu}$$

Tämä on ylijäämäraha, joka vähimmäiskustannusten kattamisen jälkeen on käytettävissä järjestelmän optimointiin.

D_e:n avulla reunaehto antaa

$$\frac{1}{\sqrt{\Lambda\beta\mu}} = \frac{D_e}{\sum_i \sqrt{\lambda_i d_i}}$$

Sijoitetaan tämä takaisin optimaalisen C_i:n lausekkeeseen:

$$\boxed{C_i = \frac{\lambda_i}{\mu} + D_e \frac{\sqrt{\lambda_i/d_i}}{\sum_{j=1}^{M} \sqrt{\lambda_j d_j}}}$$

Linkin kapasiteetti ylittää vähimmäisarvon λ_i/μ määrällä, joka on verrannollinen $\sqrt{\lambda_i/d_i}$:hin.

Keskimääräinen viipymä optimoidussa verkossa on

$$\boxed{T_{\min} = \frac{1}{\Lambda\mu D_e} \Big(\sum_{i=1}^{M} \sqrt{\lambda_i d_i} \Big)^2}$$

Erikoistapaus

Jos erityisesti kaikilla linkeillä on sama ominaiskustannus, voidaan asettaa

$$\begin{cases} d_i = 1 \\ D = C \quad \text{käytettävissä oleva kokonaiskapasiteetti} \end{cases}$$

Merkitään lisäksi

$$\rho = \frac{1}{C}\sum_{i=1}^{M}\frac{\lambda_i}{\mu} \qquad \text{linkkien keskimääräinen käyttöaste}$$

Tällöin kaavat saavat yksinkertaisemman muodon:

$$\boxed{\begin{cases} C_i = \dfrac{\lambda_i}{\mu} + (1-\rho)\,C\,\dfrac{\sqrt{\lambda_i}}{\sum_j \sqrt{\lambda_j}} \\[2ex] T_{\min} = \dfrac{\left(\sum_i \sqrt{\lambda_i}\right)^2}{(1-\rho)\Lambda C \mu} \end{cases}}$$

Ylimääräkapasiteetin jako tapahtuu $\sqrt{\lambda_i}$:den suhteessa.