Reiluus

Maxmin-reiluus

- Tärkeä näkökohta ‘best effort’ -tyyppisissä palveluissa
  - kenellekään ei anneta kvantitatiivisia QoS-takuita
  - kaikkien pitää saada palvelua tasapuolisesti
- Reiluuden *maxmin*-määritelmä

Reilu palvelu maksimoi heikointa palvelua saavan asiakkaan palvelun laadun

- Tämä ei yleensä määrittele yksikäsitteisesti resurssien jakoa
  - jos vapausasteita on vielä jäljellä, jatketaan maksimoimalla toseksi heikointa palvelua saavan asiakkaan palvelun laatu jne
- Reilussa jaossa jokainen asiakas *joko*
  - saa sen palvelun mitä on pyytänyt *tai*
  - lisäresurssin antaminen ko. asiakkaalle heikentäisi jonkun huonompaa tai samantasoista palvelua saavan asiakkaan palvelun laatua
  - ts. ei ole mahdollista lisätä minkään asiakkaan palvelun laatua vain parempaa laatua saavien asiakkaiden kustannuksella

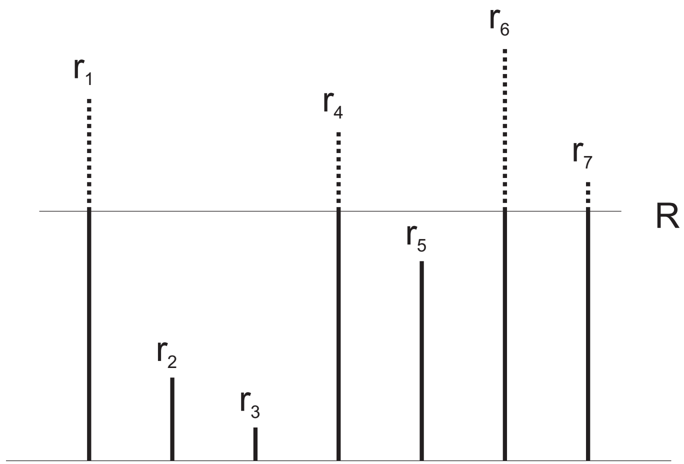
Maxmin-reiluus (jatkoa)

- Tarkastellaan niitä yhteyksiä, joille linkki ℓ on pullonkaulalinkki
  - reilun palvelun tapauksessa näiden yhteyksien nopeudet ovat yhtäsuuret
  - muuten pienimmän nopeuden yhteyttä voitaisiin nopeuttaa antamalla sille lisää kaistaa suuremman nopeuden yhteyksiltä
  - yhteinen 'katto' R_ℓ

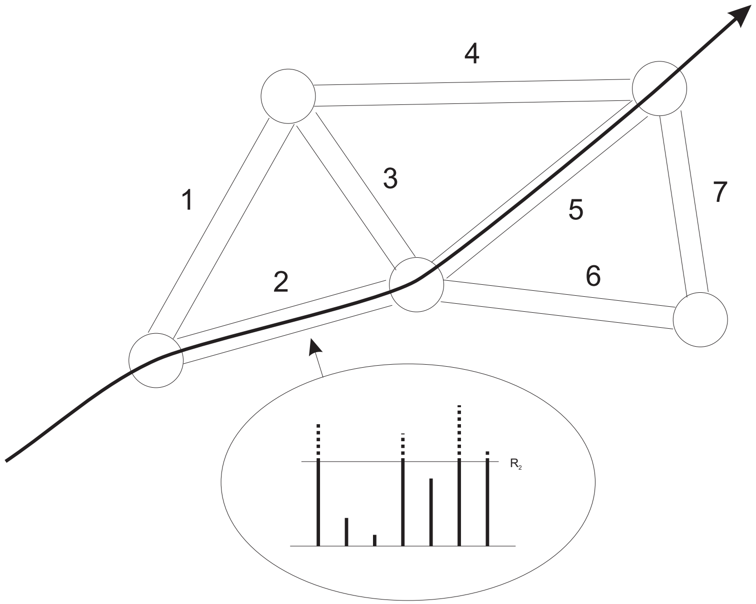
$$\sum_{i \in \mathcal{S}_\ell} \min(s_i, R_\ell) = C_\ell$$

$\mathcal{S}_\ell$ = niiden yhteyksien joukko, jotka kulkevat linkin ℓ kautta

s_i = lähteen i nopeus



Maxmin-reiluus monisolmuisessa verkossa



$$
\begin{cases}
\displaystyle\sum_{i \in \mathcal{S}_\ell} \min(r_i^{(\ell)}, R_\ell) = C_\ell, \ \forall \ell \\[2ex]
r_i^{(\ell)} = \displaystyle\min_{\ell' \in \mathcal{L}_i - \{\ell\}} (R_{\ell'}, s_i)
\end{cases}
$$

$$
\begin{cases}
\mathcal{S}_\ell = & \text{niiden yhteyksien joukko, jotka kulkevat linkin } \ell \text{ kautta} \\
\mathcal{L}_i = & \text{niiden linkkien joukko, joiden kautta yhteys } i \text{ kulkee} \\
r_i^{(\ell)} = & \text{nopeus, jolla yhteys } i \text{ `haluaisi' lähettää linkissä } \ell
\end{cases}
$$

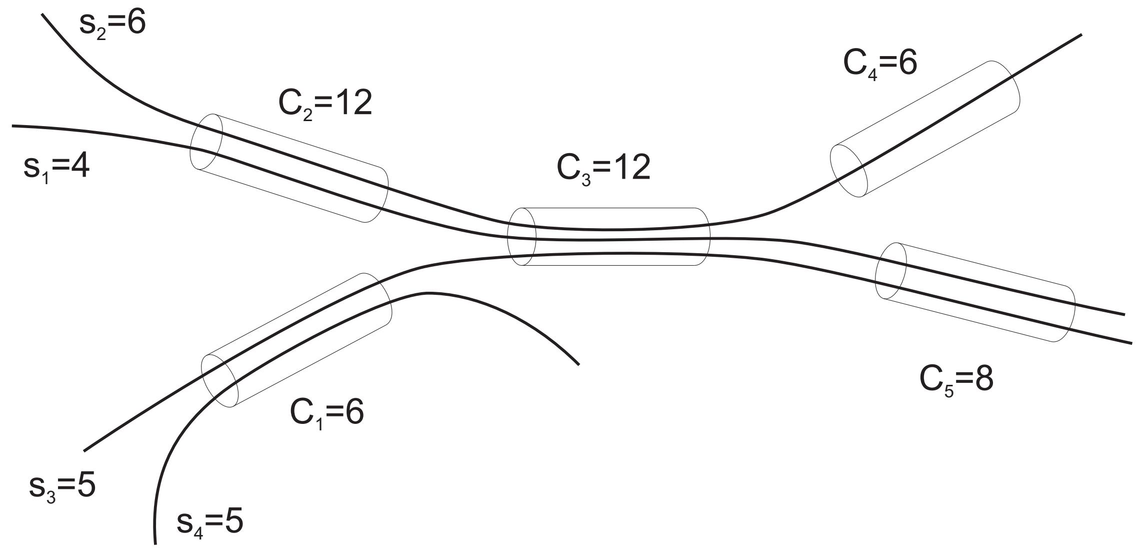
$$
r_i = \min_{\ell' \in \mathcal{L}_i} (R_{\ell'}, s_i) = \min_{\ell' \in \mathcal{L}_i} r_i^{(\ell)} \quad \text{lähteelle } i \text{ allokoitu nopeus}
$$

Maxmin-reilun jaon etsiminen ("täyttöalgoritmi")

1. Alussa asetetaan kaikkien yhteyksien nopeudet nollaksi, $r_i = 0, \ \forall i$
2. Kasvatetaan kaikkia nopeuksia (samansuuruisina), kunnes *joko*
   - jonkun yhteyden nopeus on saavuttanut lähdenopeuden *tai*
   - jonkin linkin kapasiteettiraja on saavutettu
3. 'Jäädytetään' tämän yhteyden / näiden yhteyksien nopeudet nykyiseen tasoonsa ja jatketaan muiden yhteyksien osalta nopeuksien kasvatusta kohdan 2 mukaisesti

- Kuvattu algoritmi edellyttää keskitettyä tietoa koko verkosta
- On kehitetty myös hajautettuja toteutuksia algoritmista (esim. ATM-vekon ABR-palvelun yhteydessä), jossa lähteet ja kytkimet vaihtavat keskenään tietoa
  - muutaman iteraation jälkeen nämä hajautetut algoritmit suppenevat kohti reilua jakoa

Esimerkki verkon kapasiteetin maxmin-reilusta jaosta



yhteys	1	2	3	4	Huom.
s_i	4	6	5	5	lähteen (pyydetty) nopeus
kierros					
0	0	0	0	0	
1	1	1	1	1	
2	2	2	2	2	
3	3	3	$\underline{3}$	$\underline{3}$	linkki 1 täynnä
4	$\underline{4}$	4			yhteyden 1 lähdenopeus
5		$\underline{5}$			linkki 3 täynnä

- Esimerkissä on nopeuden lisäykset tehty yksinkertaisuuden vuoksi ykkösen askelin
- Todellisuudessa kasvatus pitää tehdä liukuvasti
  - on helppo päätellä, mikä raja tulee seuraavaksi vastaan

Maxmin-reiluuden formaali määritelmä

Edelläolevat tarkastelut voidaan pukea vielä täsmällisempään matemaattiseen muotoon:

Määritelmä 1: Yhteyksien nopeusvektori $\mathbf{r} = \{r_i \mid i \in \mathcal{S}\}$ on mahdollinen, jos

$$0 \leq r_i \leq s_i \quad \forall i$$
$$\sum_{i \in \mathcal{S}_\ell} r_i \leq C_\ell \quad \forall \ell$$

Määritelmä 2: Nopeusvektori $\mathbf{r}$ on max-min-reilu, jos se on mahdollinen ja jos jokaisella yhteydellä i ja jokaisella mahdollisella nopeusvektorilla $\hat{\mathbf{r}}$, jolla $\hat{r}_i > r_i$, on olemassa toinen yhteys j, jolla $r_j \leq r_i$ ja $\hat{r}_j < r_j$

Määritelmä 3: Tietyn mahdollisen nopeusvektorin $\mathbf{r}$ suhteen linkki ℓ on pullonkaulalinkki yhteydelle $i \in \mathcal{S}_\ell$, jos $\sum_{k \in \mathcal{S}_\ell} r_k = C_\ell$ ja $r_j \leq r_i \; \forall j \in \mathcal{S}_\ell$

Voidaan osoittaa, että näistä määritelmistä seuraa

Lause 1: Mahdollinen nopeusvektori $\mathbf{r}$ on max-min-reilu, jos ja vain jos jokaiselle yhteydelle i jokin linkki on pullonkaula tai $r_i = s_i$

Lause 2: Max-min-reilu nopeusvektori $\mathbf{r}$ on yksikäsitteinen

Utiliteettipohjaiset reiluuden määritelmät

- Maxmin-reiluus "klassillinen" ja parhaiten tunnettu reiluuskäsite
  - yhden linkin tapauksessa kaistan tasanjako on ilmiselvästi reilu
  - verkon tapauksessa yleinen reiluuden määritelmä ei ole mitenkään ilmeinen
  - maxmin-reiluus on vain yksi mahdollinen määritelmä, joskin sitä voidaan hyvin perustella
- Myös muita määritelmiä on ehdotettu
- Ns. utiliteettipohjaiset reiluuskäsitteet sisältävät monia mahdollisia määritelmiä
  - maxmin-reiluus on erikoistapaus utiliteettipohjaisista reiluuskriteereistä
- Lähtökohtana on määritellä utiliteetti- l. hyötyfunktio $U(x_r)$, joka kuvaa käyttäjän r (reittiä r käyttävä vuo) verkosta saamaa hyötyä, kun sen saama kapasiteetti on x_r
- Tavoitteena on sitten maksimoida koko käyttäjäkunnan yhdessä saama hyöty

$$U = \sum_{r \in \mathcal{R}} n_r U(x_r)$$

missä $\mathcal{R}$ on kaikkien reittien joukko ja n_r reittiä r käyttävien voiden (käyttäjien) lukumäärä

Utiliteettipohjaiset reiluuden määritelmät (jatkoa)

- Utiliteettikriteerin mukainen reilu kapasiteetin jako voidaan nyt määritellä seuraavan optimointitehtävän ratkaisuna

$$\begin{cases} \text{max} & \sum_{r\in\mathcal{R}} n_r U(x_r) \\ \text{subject to} & Ax \le C \\ \text{over} & x \ge 0. \end{cases}$$

missä

$$\begin{array}{rcll} x & = & \{x_r, r \in \mathcal{R}\} & \text{voiden lukumääristä eri reiteillä muodostuva vektori} \\ C & = & \{C_j, j \in \mathcal{J}\} & \text{linkkikapasiteeteista muodostuva vektori} \\ A & = & \{a_{jr}, j \in \mathcal{J}, r \in \mathcal{R}\} & \text{linkki-reitti-matriisi;} \\ & & & a_{jr} \text{ on } n_r \text{ jos reitti } r \text{ kulkee linkin } j \text{ kautta ja 0 muutoin.} \end{array}$$

Utiliteettipohjaiset reiluuden määritelmät (jatkoa)

- Järkevä ja melko yleinen valinta utiliteettifunktioksi on

$$U(x) = \frac{x^{1-\alpha}}{1-\alpha}$$

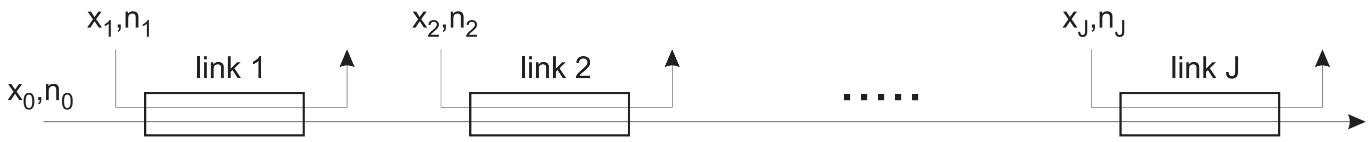
missä α on vapaa parametri

- Erityiset α:n arvon valinnat johtavat seuraaviin tärkeisiin erikoistapauksiin

α	konsepti	$\max\limits_x \sum\limits_{\mathcal{R}} n_r U(x_r)$
0	maksimoi kokonaisläpäisy	$\max\limits_x \sum\limits_{\mathcal{R}} n_r x_r$
1	suhteellinen reiluus	$\max\limits_x \sum\limits_{\mathcal{R}} n_r \log x_r$
2	minimoi potentiaalinen viive	$\min\limits_x \sum\limits_{\mathcal{R}} \dfrac{n_r}{x_r}$
∞	maxmin-reiluus	$\max\limits_x \min\limits_{r \in \mathcal{R}} x_r$

- Pienet α:n arvot suosivat yhteistä (verkon) hyötyä yksilöiden kustannuksella; suuret α:n arvot korostavat reiluutta heikointa kohtaan.

Esimerkki: Utiliteettipohjainen reiluus lineaarisessa verkossa



- Pitkää reittiä kulkevaa vuota indeksoidaan 0:lla; lyhyiden reittien voita indeksoidaan vastaavan linkin numerolla; kaikkien linkkien kapasiteetti on 1

α	konsepti	x_0
0	maksimoi kokonaisläpäisy	0
1	suhteellinen reiluus	$\dfrac{1}{n_0 + \sum_j n_j}$
2	minimoi potentiaalinen viive	$\dfrac{1}{n_0 + \sqrt{\sum_j n_j^2}}$
∞	maxmin-reiluus	$\dfrac{1}{n_0 + \max_{j \geq 1} n_j}$

- Parametrin α kasvaessa 0:sta kohti ääretöntä, eri allokaatiot antavat suhteellisesti yhä enemmän kaistaa pitkille reiteille