

VELOITUSPERUSTEIDEN KÄYTTÖ LIIKENTEEN OHJAUKSESSA

Martti Kurne

SISÄLLYSLUETTELO

1. Johdanto

2. Internetin "Best Effort Service"

2.1 Eri Internet-palveluiden suorituskyky

3. Internet-palveluiden prioriteetti-hinnoittelu-tutkimus

3.1 Hyötyfunktiot

3.2 Käyttäjien reagoinnista

3.3 Simulaatio ja tuloksia

4. Yhteenveto ja keskustelua

Liite -1

Table 1 ja Table 2

1. Johdanto

Internet-tekniikkaa toteutettaessa ei ole alunperin huomioitu ollenkaan laskutusta ja veloitusta. Tämä selittyy Internetin syntyhistoriasta, joka sai alkusysäyksen Yhdysvaltain puolustushallinnon (ARPA, Advanced Research Projects Agency) hankkeesta. Se käynnistyi vuonna 1968. ARPA asetti tavoitteekseen Internet-teknologian, arkkitehtuurien ja protokollien kehittämisen 1972. ARPANETista tuli Internetin runkoverkko ja Internet laajeni, kun tutkimusverkkoihin kytkettyjä tietokoneita alettiin päivittää uusiin TCP/IP-protokolliin sopiviksi. Välitellen Internet laajeni tutkijoiden ja opiskelijoiden kansainväliseksi tietoverkoksi /7/.

Internet tarjoaa hyvin yksinkertaisen palvelumallin. Kaikki liikennepaketit saavat saman parhaan käytettävissä olevan palvelun (best effort service). Uusien reaaliaikaisuutta vaativien palvelujen tulo Internetiin on käynnistynyt ja verkon liikenne ja datamäärä kasvavat jatkuvasti. 90-luvulla Internetiä on alettu soveltaa myös liike-elämän tarpeisiin. Nyt myös erilaisten veloituserusteiden käyttö liikenteen ohjaamiseksi Internetissä voi tulla tavoiteltavaksi ominaisuudeksi.

2. Internetin “Best Effort Service”

“Best effort”-termi tarkoittaa sitä, että verkko yrittää lähettää paketteja eteenpäin niin pian kuin mahdollista, mutta se ei vastaa palvelun laadusta käyttäjille. Tämä palvelumalli-malli on toteutettu reitittimissä FIFO-jonotus-periaatteella (First In First Out). Ilman Admission Controlia jäävä, FIFO- periaatteella toimiva pakettien jakelu ei voi tarjota muuta kuin “best effort”-palvelun. Tällöin kaikkia palvelun verkon liikennettä (flows) kohdellaan samoin. Yhdenmukainen Quality of Service (QoS) näyttää hyvältä niin kauan kuin mitataan viiveitä ja hävinneitä paketteja. Ongelmia ei esiinny, jos verkko on vain hiukan kuormitettu, mutta kovalla kuormalla QoS on todella huono. Internetin peruspalvelut kuten FTP, SNMP-sähköposti ja Telnet-päätekäyttö eivät tosin ole kovin herkkiä verkon kuormituksen vaihteluille.

Mikäli Internetin ydinpalvelut tulevat aikariippuviksi, mallin pitää muuttua. Tämä tarkoittaa ei-FIFO perusteisten pakettialgoritmien käyttöönottoa reitittimiin tai ATM-

kytkimiin. Monia tämäntyyppisiä Internetin palveluita on tulossa, kuten FAX, Video on Demand, multimedia kokoukset, virtuaalitodellisuus jne.

Kun tarkastellaan palvelun laatua Quality of Service (QoS), todetaan, että palvelumalli on sopimus, joka kuvaa verkon vastetta käyttäjän palvelupyyntöihin. Ydinpalvelumalli sisältää QoS-sopimukset. Palvelusopimus (commitment) voidaan jakaa kahteen luokkaan joko lähtien määrällisestä tai laadullisesta näkökulmasta. Määrällinen lähestymistapa asettaa rajoja esimerkiksi maksimiviiveen. Laadullinen lähestymistapa määrittelee ja sääntelee verkossa liikkuvien pakettien välisiä suhteita. Palvelusopimukset voidaan jaotella myös sen mukaan mitä verkon osaa (entity) ne koskevat. Yleensä palvelusopimukset kohdistetaan liikennevuohon (flows). Liikenne voi toimia sille annetuissa rajoissa. Asiaa voidaan myös lähestyä jakamalla resursseja eri lailla verkon kollektiivisten osien kesken. Jokin yritys, instituutio, protokollaperhe tai sovellus saa tietyn palvelutason. Käytännössä kaksi yritystä voi jakaa esimerkiksi optisen linkin keskenään /8/.

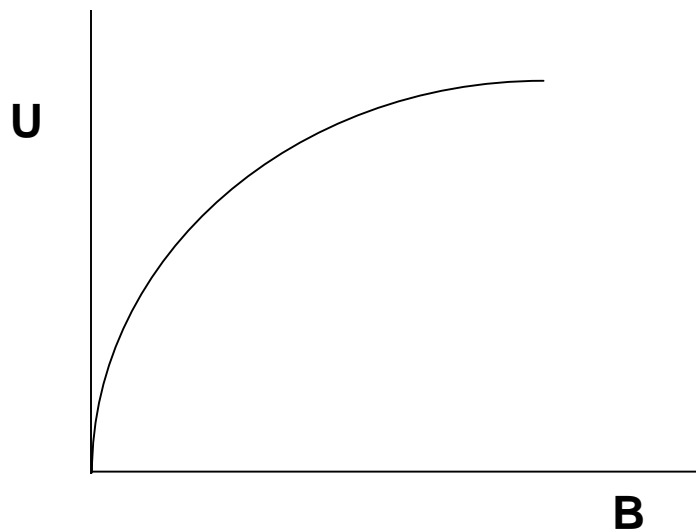
2.2 Eri Internet-palveluiden suhde verkkoon

Printteiset data-siirron sovellukset kuten FTP, sähköposti ja Telnet etäyhteydet sietävät hyvin viiveitä. Ne eivät välittömästi hyödy kaistan lisäämisestä. Näitä voidaan kutsua *elastisiksi* sovelluksiksi (kuva-1).

Toisena äärimmäisyytenä on viiveelle hyvin herkät sovellukset, joilla on tiukat reaaliaikavaatimukset. Nämä sovellukset tarvitsevat aina datan tietyn viiveen sisällä. Sovellukset eivät tietenkään häiriinny, vaikka data tulisi hieman aikaisemminkin, mutta häiriintyvät heti, mikäli viiveraja ylittyy. Tällaisia palveluja ovat puhelut ja kaikki piirikytkentäisyyttä kaipaavat sovellukset. Näitä sovelluksia voidaan kutsua *hard realtime*-sovelluksiksi.

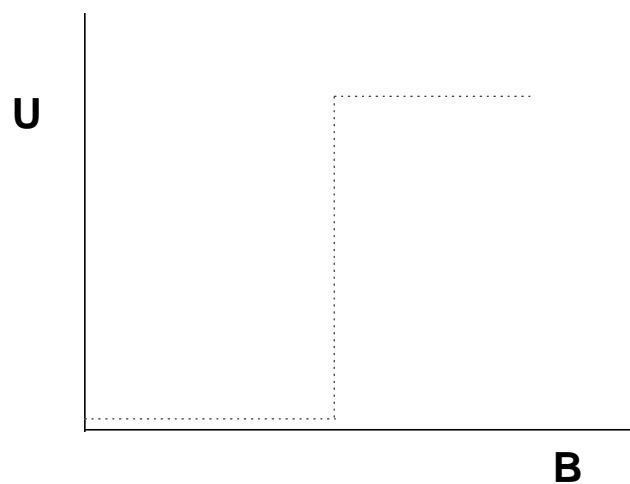
Mikäli viiverajavaatimukset tulevat täytetyksi, pysyy suorituskyky vakiona, mutta heti kun kastaleveys putoaa alle viiverajojen tarvitseman arvon, putoaa suorituskyky jyrkästi nolnaan. Tällainen järjestelmä ylikuormittuu heti kun kaistaleveyden osuus putoaa alle kriittisen tason. Sovellukset, joilla on kovat reaaliaikavaatimukset toimisivat paljon paremmin verkossa, jossa olisi toimiva *admission control*, joka takaisi että kaistanleveys ei koskaan putoaisi alle kriittisen tason. Perinteisesti video ja audio sovellukset on suunniteltu

kovien reaaliaikavaatimusten mukaisiksi. Kuitenkin nykyiset Internetiin rakennetut koejärjestelmät osoittavat, että useimmat audio- ja video-sovellukset voidaan toteuttaa sietämään tilapäisiä viiverajarikkomuksia ja hävinneitä paketteja.



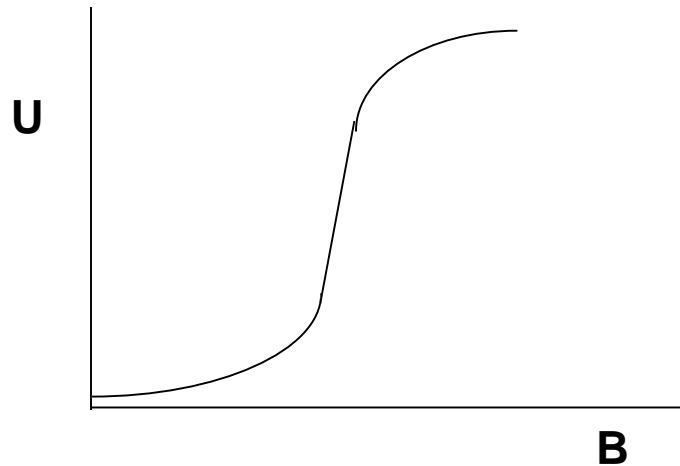
U= hyödyllisyys/ suorituskyky, B= kaista

Kuva-1 Elastinen sovellus



Kuva-2 Hard Realtime- sovellus

Kuitenkin video- ja audio- sovelluksilla on luontainen kaistanleveyden tarve, koska dataa generoituu riippumatta verkon tukkoisuudesta. Siten suorituskyky laskee heti kun kaistanleveys osuus tulee pienemmäksi kuin luontainen datan generoitumisnopeus. Viiveeseen sopeutuvan, *delay adaptive* audio ja video sovelluksen käytettävyyden käyrä voisi näyttää seuraavanlaiselta (kuva-3).



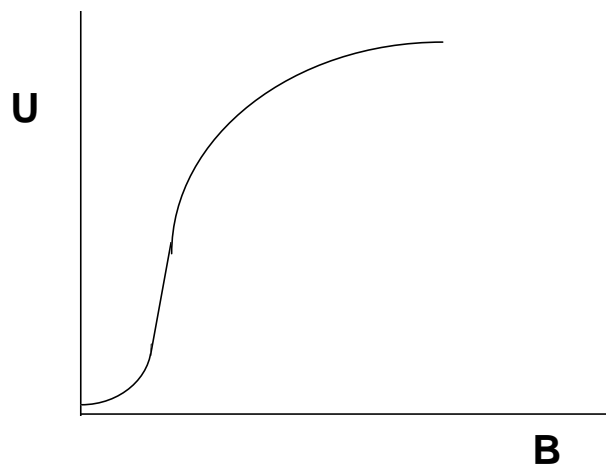
Kuva-3 Delay-Adaptive sovellus

On hyvä huomata, että suorituskyvyn pudotus ei ole läheskään yhtä suuri kuin hard realtime-sovelluksissa, vaikkakin käyrä muuten ominaisuuksiltaan on samantapainen. Erityisesti tämä käytettävyydsfunktio on kupera (konvekssi) eikä kovera nollan lähiympäristössä. Siten verkko voi ylikuormittua nollan lähellä toimivilla sovelluksilla. Tarkka ylikuormituksen alkamiskohta riippuu käyrän muodosta ja käännepisteen sijainnista.

Tämä analyysi osoittaa, miksi Internet-yhteisö ja puhelin-yhteisö on ollut umpikujassa kistelllessään "best-effort" ja reaaliaikapalvelun välillä monien vuosien ajan. Internet rakensi verkon, jonka palvelut tukevat data-siirtosovelluksia, jotka vuodaan luokitella elastisiksi. Lähtökohtana oli tarjota best-effort palveluita oli oikea ja luonnollinen. Nykyisellään kun video ja muut reaaliaikaiset palvelut ovat tulleet Internetiin, on myös vaatimus admission controllista lisääntynyt.

Samalla tavalla puhelin-yhteisö rakensi verkon, jolla on kovat reaaliaikavaatimukset puheen ympärille. Kun yhteisö on suunnitellut ATM:n palvelemaan uusia reaaliaikavaatimuksia, admission control oli jälleen luonnollinen ja oikea valinta.

On vielä toisen tyyppisiä reaaliaikasovelluksia. Nopeuteen sopeutuvia, *rate adaptive*, sovelluksia, jotka sopeuttavat siirtonopeuden suhteessa verkossa olevaan estoon. Tämä mukauttaminen pitää viiveet kohtuullisina eikä se ota kantaa kaistaleveyden osuuteen. Siten sovelluksen suorituskyky riippuu täysin signaalin laadusta. Korkeilla kaistaleveyksillä marginaalinen lisäkaistan hyöty on hyvin vähäinen, koska signaalin laatu on paljon nykyistä tarvetta parempi. Myös hyvin pienillä kaistanleveyksillä marginaalinen hyöty on hyvin pieni, koska signaalin laatu sietämättömän alhainen. Nopeuteen sopeutuvat käyrät saavat seuraavanlaisen muodon (kuva-4)/9/

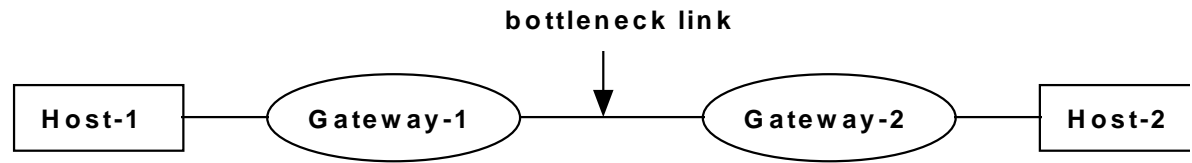


Kuva-4 Rate-Adaptive sovellus

3. Internet-palveluiden prioriteettihinnoittelu-tutkimus

Tekninen kehitys ei ole ainoa tärkeä tekijä, joka vaikuttaa verkon suorituskyvyn lisääntymiseen. Kun selvitetään verkon käytön kannustimia (incentives) ja erilaisen käytön syitä, löytyy monia eri näkökulmia, jotka usein johtavat rahataloudellisiin (monetary) syihin. Hinnoittelupolitiikka tuo käyttäjille rahataloudellisia houkuttimia, joka puolestaan johtaa suorituskyvyn tehostumiseen.

Prioriteettihinnoittelun soveltuvuutta Internetin eri palveluille on simuloitu koeympäristössä. (kuva-5)/10/.



Kuva-5 Simulointiympäristö

Jokaista käyttäjää edustaa sovellus, joka lähettää dataa host-1:ltä host-2:lle. Käyttäjät huolehtivat käyttökustannuksista (C). Käyttäjät huolehtivat myös sovelluksen suorituskyvystä. Sovelluksen suorituskyky on riippuvainen verkon suorituskyvystä. Sovelluksen suorituskyvyn heikentymistä käyttäjälle merkitään V:llä. Mitä suurempi V:n arvo on sitä huonompi on sovelluksen suorituskyky. Jokaiselle sovellukselle määritellään eri V, jotta saadaan mitattua aina tietyn sovelluksen herkkyys suorituskyvyn alenemiselle.

Kuinka sitten suorituskyky muuttui eri sovelluksilla ja kuinka sovelluksen suorituskyvyn heikkeneminen riippuu verkon käyttäytymisestä. Käytetyt sovellukset olivat sähköposti (E-mail), file transfer service (FTP), remote login (Telnet) sekä reaaliaikainen ääni (Voice).

Sähköpostia käytetään asynkronisesti monen käyttäjän kesken eikä välitöntä jakelua odoteta. Oletusarvona on sähköposti saapuu minuuttien viiveellä eikä käyttäjä koe muutamien minuuttien viivettä järjestelmän sähköpostin suorituskyvyn heikentymisenä.

Sähköpostissa käytettiin seuraavaa mallia V:lle.

$$V(\text{Email}) = 0.1(\text{average message delay, in seconds}) + (\% \text{ of messages not delivered in loose delay bound})$$

FTP on yhden käyttäjän interaktiivinen sovellus ja käyttäjät haluavat saada valmiiksi FTP-siirtonsa ennen kuin etenevät seuraavaan työvaiheeseensa. Kun oletetaan, että FTP-käyttäjät haluavat melkoisen pikaista palvelua. FTP:hen liittyvä malli on seuraava

$V(FTP) = (\text{average file transfer time, in seconds})$

Telnet edustaa todellista interaktiivista sovellusta. Käyttäjä johtaa Telnet-istuntoa ja odottaa välitöntä reaaliaikaista vastetta. Oletetaan, että Telnetin käyttäjät häiriintyvät yli muutaman sadasosasekunnin yli menevistä viiveistä. Oleellinen viive Telnetissä syntyy kulkuajasta, jonka paketti kulkee kohdekoneeseen ja tulee sietä takaisin päätteelle (round trip). Telnetin malli on seuraava.

$V(Telnet) = (\text{average packet round trip time in seconds})$

Äänipalvelu on reaaliaikapalvelu, joka on äärimmäisen herkkä viiveille. Äänipalvelut eivät siedä yli 200 millisekunnin viiveitä. Äänipalvelu eroaa muista edellä mainituista palveluista. Puhelu ei kaipaa 100%:n luotettavuutta siirrossa, koska ihmisäänessä on riittävästi ylimääräistä informaatiota (redundanssia), jotta oikea tulkinta saataisiin muutamien data-bittien katoamisesta huolimatta. Puhemallissa kaikki paketit, jotka eivät pysy viiverajojen sisällä poistetaan.

$V(Voice) = (\% \text{ of packets not obeying tight bound})$

3.1 Hyötyfunktiot

Käyttäjän lopullinen tyytyväisyystaso on riippuvainen sekä sovelluksen suorituskyvystä että käytöstä aiheutuvista kustannuksista. Suorituskyvyn ja kustannuksien yhdistettyä riippuvuutta kuvataan hyötyfunktioilla *Utility Functions* (U).

U riippuu sekä V:stä että C:stä (V kuvaa suorituskyvyn alentumista ja C kustannuksia). Mitä suurempi on U:n arvo, sitä suurempi on tyytyväisyyden aste. Oletuksena on että käyttäjät

pitävät alhaisista kustannuksista ja hyvästä suorituskyvystä. U voi olla mikä tahansa monotoonisesti laskeva V :n ja C :n funktio. Tässä on valittu

$$U = -V - C \quad (1)$$

Mikä sitten hyötyfunktio oikein on? Hyötyfunktiot ovat olleet keskeisessä osassa talousteorioissa 1900-luvun puolenvälin jälkeen. Hyötyfunktiot esittävät pelkästään yksilöiden suorittamaa valintaa, siis valitaa useista mahdollisuuksista etusijalle jostain joukosta. Kun vertaillaan eri yksilöiden välisiä hyötyfunktioita, antaa funktio tietoa siitä, mitä valintoja tietty yksilö suosii ja mitä ei, eron suuruudella ei ole merkitystä vain etumerki on tärkeä. Koska hyötyfunktio edustaa vain valintaa, ei ole järkevää vertailla absoluuttisia funktion arvoja eri yksilöiden välillä. Ei myöskään ole tarkoitus V :n avulla vertailla suhteellista suorituskyyä kahden eri sovelluksen välillä. Hyötyfunktiot vain ja pelkästään heijastavat valintojen paremmuusjärjestystä yksittäisen verkossa toimivan sovelluksen osalta muuttuvissa verkon olosuhteissa/12/.

3.2 Käyttäjien reagoinnista

Kuinka sitten käyttäjät reagoivat erilaisiin kannustimiin (incentives)? Malli ei yritä ottaa huomioon kysynnän joustoa. Otetetaan, että käyttäjän generoima liikenne ei muutu hinnoittelun myötä. Tällöin mallinnetaan vain loppukäyttäjien palveluluokan valintapäätöstä. Käyttäjät yrittävät maksimoida kokonaistyytyväisyyden U . Jos vain suorituserusteiset (V) kannusteet olisivat mukana, niin kaikki asettaisivat molemmat *prioriteetti-liput* ykköseksi. Jos vain rahataloudelliset kannusteet olisivat mukana, niin kaikki asettaisivat molemmat *prioriteettiliput* nollassi. Molempien kannustimien tapauksessa ei tilanne ole lainkaan selvä.

Kertamaksun tapauksessa rahataloudelliset kannusteet eivät ole merkitseviä, koska kustannus on aina sama (C). Siten kaikki käyttäjät asettavat molemmat *prioriteetti liput* päälle. Kun käytetään *prioriteettihinnoittelua*, on oletuksena, että käyttäjät asettavat *prioriteettilipun* päälle vain, *mikäli se kohentaa verkon suorituskyyä, niin että siitä on hyötyä sovelluksen kannalta katsottuna*.

Seuraavassa on valittu oletuksena käyttäjien tekemiä palveluluokkavalintoja kohtalaisesti kuormitetulle verkkolle:

V(Email) riippuu vain hiukan keskimääräisestä viiveestä. Niinpä simulaatiossa Emailin käyttäjät eivät käytä kumpaakaan prioriteettilippua. V(Telnet) on hyvin herkkä viiveelle ja niinpä simulaatiossa Telnet-käyttäjät asettavat molemmat prioriteettiliput ykköseksi. V(Voice) on hyvin herkkä viiveelle, mutta sietää muutamien bittien putoamisia. Niinpä simuloinnissa käyttäjä asettaa prioriteetin puhepalvelulle, mutta eivät bittien katoamisen (bit drop) estolippua. V(FPT) on riippuvainen keskimääräisestä tiedoston siirtoajasta. Koska tiedostot ovat yleensä hyvin suuria, niin siirtoaika on funktionaalisesti riippuvainen keskimääräisestä läpäisykyvystä (throughput) kuin yksittäisten pakettien viiveistä. FTP:n käyttäjät jättävätkin valitsematta palveluprioriteetin, mutta valitsevat bittien katoamislipun (bit drop) ykköseksi.

3.3 Simulaatio ja tuloksia

Käytettyjä yhteisprotokollia ovat UDP äänipalvelussa ja TCP muissa palveluissa. UDP valittiin äänipalvelulle, koska siihen sopivat ankarat viiverajat ja äänisovellukset eivät hyödy TCP: tarjoamasta pakettien uudelleen lähettämisestä.

Simulaatiomallissa käyttäjä tekee jatkuvasti palvelupyynnöitä omalta sovellukseltaan. Jokaista palvelupyynnöä voidaan luonnehtia koon *size* (s) ja aikaintervallin *time* (t) mukaan suhteessa edelliseen palvelukutsuun. Simulaattorissa tämä on käyttäjän toiminta on mallinnettu satunnaisprosessiksi, jossa sekä *size* (s) että *time* (t) ovat eksponentiaalisesti jakautuneita satunnaismuuttujia keskiarvoinaan \underline{s} ja \underline{t} . E-mailin ja FTP:n osalta *size* (s) viittaa lähetettävän sanoman tai tiedoston kokoon. Käytetään maksimikokoisia paketteja 500 bytes. Telnetin suhteen koko *size* (s) tarkoittaa ryöpyssä generoituvien merkkien määrää tilanteessa, kun jokainen merkki lähetetään ja kaiutetaan erikseen käyttäen 50 bytes-suuruisia paketteja.

Äänen eli Voicen *size* (s) tarkoittaa keskustelua ja sen aikakestoja. Keskustelun aikana 180 byten paketteja lähetetään 64Kbit/s.

Testi koostuu kahdesta erilaisesta simulaation *konfiguroinnista*. Ensimmäisessä tapauksessa on 2 Voice-sovellusta, 4 FTP sovellusta, 5 E-mail sovellusta ja 2 Telnet sovellusta. Toisessa konfiguraatiossa on 2 Voice-sovellusta, 3 FTP sovellusta, 5 E-mail

sovellusta ja 4 Telnet sovellusta (ks.liite-1). Pullonkaulayhteys (ks. kuva-5) on ensimmäisessä konfiguraatiossa 772 Kbits ja 600 Kbits toisessa konfiguraatiossa. Molemmissa simulaatioissa kerätyt tulot olivat \$ 100. Ensimmäisessä konfiguraatiossa $P(\text{flat}) = \$2.43 \cdot E^{-7}$ ja $P(\text{Priority}) = \$1.28 \cdot E^{-7}$ ja toisessa konfiguraatiossa $P(\text{flat}) = \$3.05 \cdot E^{-7}$ ja $P(\text{Priority}) = \$1.75 \cdot E^{-7}$. Molemmissa konfiguraatioissa halvin prioriteettiluokka $P(0,0)$ on karkeasti puolet $P(\text{Flat})$ tariffin suuruudesta. Keskivälin prioriteettiluokan tariffit $P(1,0)$ ja $P(0,1)$ ovat suunnilleen samalla tasolla kuin $P(\text{Flat})$ sensijaan korkein prioriteettiluokka $P(1,1)$ ylittää 50% korkeammalle kuin tasamaksu $P(\text{Flat})$.

Millaiset ovat ensimmäisessä ja toisessa konfiguraatiossa syntyneet eri sovellusten osuudet koko liiketeestä (Byte %, Taulukko).

	Telnet	Email	Voice	FTP
1. conf.	0.37 %	16%	20.13 %	63.5%
2. conf.	1%	27%	26%	46%

Taulukko: Eri sovellusten käyttöosuudet 1. ja 2. konfiguraatiossa

Suurin ero näiden kahden konfiguraatioiden välillä on se, että 2.konfiguraatiolla on noin 3-kertaa enemmän Telnet-liikennettä ja 2-kertaa enemmän E-mail-liikennettä. Havaitaan myös että 2. konfiguraatolla on hiukan enemmän Voice-liikennettä ja yksi neljännes vähemmän FTP-liikennettä kuin 1. konfiguraatiossa. Yksi simulaatio kestää aina 90 minuuttia. Molemissa tapauksissa (1. ja 2. konfiguraatio) pullonkaulana oleva yhteys toimi keskimäärin 80%:n kuormalla (bottleneck, kuva -5).

Molemmilla konfiguraatioilla verrataan tasamaksun (Flat rate) ja prioriteettiin pohjautuvaa (Priority pricing) hinnoittelumalleja keskenään. Pääasiallinen tulos antaa korkeamman tyytyväisyystason prioriteettipohjaisella hinnoittelulla kuin tasamaksuun perustuvalla hinnoittelulla. (U:n arvot). E-mail oli ainoa sovellus, jossa palvelu huonontui siirryttäessä prioriteettipohjaiseen hinnoitteluun (Liite-1, table 1 ja table 2).

Eri sovellusten välillä ei kannata V:n ja U:n arvojen vertailua tehdä. V ja U vertailee pelkästään simuloitavan sovelluksen saavuttamaa tyytyväisyystasoa kahdessa eri tilanteessa (konfiguraatiossa). V:n arvot ovat erilaisia eri FTP-käyttäjille. Tulos aiheutuu erilaisista liikenteen generoitumisparametreistä s ja t eikä suinkaan siitä, että jokin palvelun käyttäjä olisi tyytymättömämpi kuin jokin toinen. Tästä aiheutuu verkon suorituskyvyn alenemista. Telnetin viiveet liikkuvat 250ms:n ja 1 sekunnin välillä molemmissa konfiguraatioissa. Tämä ei johdu moniluokkaisen palvelumekanismen puuttumisesta, vaan pikemminkin käyttäjän kannusteiden puuttumisesta. Oletuksenahan oli, että kaikki käyttäjät vaativat korkealaatuista palvelua.

Kun jokin kiinteä hinta asetetaan, kaikki simuloitavat sovellukset käyttävät korkeinta käytettävissä olevaa palvelun laatuluokkaa. (molemmat prioriteettiliput ovat pystyssä). Prioriteettihinnoittelussa käyttäjät ovat motivoituneita valitsemaan tarkoituksenmukaisen palveluluokan. Moniluokkainen palvelujärjestys (multiclass service discipline) tekee mahdolliseksi verkon suorituskykyyn herkästi reagoivien sovellusten saada parempaa palvelua. Karkeasti 99% Voice-paketeista jaetaan viiverajojen sisällä. Telnet viiveet tulevat kertaluokkaa pienemmiksi ja FTP viiveet pienenevät merkittävästi. (2. konfiguraatiossa parannukset ovat vähäisempiä). Jokaiselle simuloitavalle sovellukselle kustannusten kasvaminen palvelun paranemisen johdosta oli hintansa väärä, koska sovelluksen suorituskyky parani. Siten prioriteetti-pohjainen hinnoittelun rakenne on tarjonnut kannustimia ja simuloidut sovellukset valitsevat aina tarkoituksenmukaisen palveluluokan.

Rahataloudellisia kannusteita voidaan käyttää houkutusena, jotta tarkoituksenmukaiset palveluluokat tulisivat valituiksi. On mahdollista, että verkon kaikki käyttäjät saavat hyötyä (utility) ja etua prioriteettihinnoittelun käytöstä. Johtopäätöksenä suoraviivainen ehdotus hinnoittelumalliksi. $P(0,0)=P(\text{Priority})$, $P(0,1)=P(1,0)=2P(\text{Priority})$ ja $P(1,1)=3P(\text{Priority})$. On hyvä huomata, että on myös monia muita vaihtoehtoja hinnoittelumalliksi, joita käyttäjät pitävät parempana kuin tasahinnoittelua/10/.

4. Yhteenveto ja keskustelua

Mikäli Internet verkko haluaa käyttää tehokkaasti koko kastanleveytensä, sen pitäisi pystyä tarjoamaan rikas palvelun tarjonata. Rikas palvelurajapinta on hyvä ominaisuus, jos sitä ei rakenneta pelkästään jälleenmyyjille, vaan että se kannustaa myös loppukäyttäjiä vaatimaan asianmukaista palvelua. Siten hinnoittelun rakenne täytyy saada käyttäjäpohjaiseksi, QoS-

riippuvaksi ja käytön määrään sidotuksi. Sellainen hinnoittelun rakenne vaatii tärkeitä muutoksia verkon laskutusrajapintaan. sekä tietokoneiden käyttöjärjestelmätasolla että loppukäyttäjäraja-rajapinnassa. Itseasiassa, mikäli QoS:iin liittyviä laskutusominaisuuksia ei tarvita tai käytetä paljoa ja jolloin hinnoittelu perustuu yhteyksien määrään, voi pehmeä siirtyminen QoS:n piirteiden käyttöön sekä käytön määrän mukaiseen hinnoitteluun tulla kyseeseen, jos kaistanleveydestä tulee pula. Olisi tärkeää, että de-facto standardit sisältäisivät täydellisen laskutusinfrastruktuurin ja QoS palvelurajapinnan. On muistettava, että standardointielimet eivät yritä löytää parasta verkkoarkkitehtuuria, vaan pikemminkin ne tekevät teknispainotteisia valintoja, jotka nojaavat hyvin kapea-alaiseen suunnittelufilosofiaan /11/. Simulaatiotutkimuksen mukaan /10/ rahataloudellisia kannusteita (incentives) voidaan käyttää houkutusena, jotta tarkoituksenmukaiset palveluluokat tulisivat valituiksi Internetin peruspalveluihin (Telnet, FTP, E-mail) sekä Voice-äänipalvelun osalta. On mahdollista, että Internetin käyttäjät hyötyvät ja saavat etua prioriteettihinnoittelun käytöstä. Verkon suorituskyky kasvaa ja verkko kuormittuu tasaisemmin, kun käytetään muuta kuin tasahinnoittelua.

Lähdeluettelo

/1/ If You can't bill forget it, forget it (ATM service billing), Telephony Journal, 16 sept 1996, Alexander P., Cisco Systems inc., San Jose CA USA.

/2/ Agent communication system for multimedia communication services, Kishimoto, R NTT Japan, IEEE INFOCOM '96, The Conference on Computer Proceedings.

-
- /3/ Broadband intelligent network project, Martikainen, O; Molin K., Telecom Finland, Helsinki, Finland, Proceedings of IFIP Workshop on Intelligent Networks, 1994.
- /4/ Morris Sloman, Network and Distributed Systems Management, ISBN 0-201-62745-0/94
- /6/ A long distance telephone operator billing system for major accounts, 1993
- /7/ WWW:n perusteet, Internetpalvelutuotanto, Telealan liiketoiminnan tutkimusseminaari, TKK, TiK, Veikko Pihalehto, 1997
- /8/ A Service Model for an Integrated Services Internet, Scott Shenker, David D. Clark, Lixia Zhang, October 1993
- /9/ Fundamental Design Issues for the Future Internet, IEEE Journal, Vol 13, NO 7, 1995
Scott Shenker, Member, IEEE
- /10/ A study of Priority Pricing in Multiple Service Class Networks, Ron Cocchi, Scott Shenker, Lixia Zang ACM 1991
- /11/ Service Models and Pricing Policies for Integrated Services Internet, Preliminary Draft, Scott Shenker, 1993