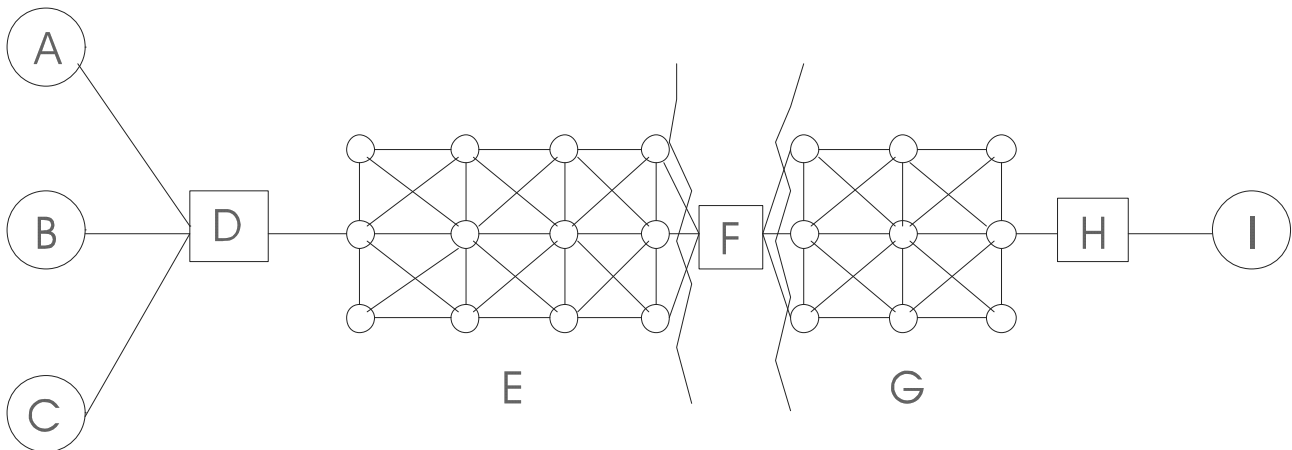


Dr. A. Murgu
MSc (mat) Juhani Takkinen

Seuraavassa siis lyhyt kuvaus omasta osastani (J.Takkinen) projektissa 3.3.
Projektissa on kolme erillistä osa-aluetta ja näistä olen siis itse sijoittunut vain yhteen ja siihenkin melko matemaattisessa mielessä. Toivottavasti seuraava vapaamuotoinen selostus selventää hieman omaa osuuttani.

Aluksi havainnollistava kuva:



A, B ja C (koneita) ovat yhteydessä verkkoon E palvelimen D kautta.
I (käyttäjän kone) on yhteydessä verkkoon G palvelimen H kautta.

Verkot E ja G eivät ole saman protokollan verkkoja, ja palvelin F toimii verkkojen välisenä tulkkina.

Oletetaan, että A lähettää I:lle sähköpostia.
B lähettää I:lle ääntä ja liikkuvaa kuvaa.
C:ssä toimii sovellus, jota I etäkäyttää.

Nyt siis A, B ja C lähettävät keskenään ”eriarvoista” tietoa.

Esimerkiksi ei haittaa mikäli sähköposti-datapaketit saapuvat perille vaihtelevalla viiveellä, mutta näissä paketeissa ei saa olla yhtään virheitä.

Toisaalta äänessä ja liikkuvassa kuvassa ei haittaa pienet virheet, sillä silmä tai korva ei niitä erota, mutta datapakettien on tultava perille tasaiseen tahtiin ja mahdollisimman pienellä väliviiveellä, jottei ääni tai kuva pätki.

Koska kuitenkin A, B ja C lähettää dataa samaan osoitteeseen, on palvelimen D järkevää kasata näiden lähettämiä datapaketteja isommiksi kokonaisuuksiksi, jolloin verkko E pystyy käsittelemään niitä joutuisammin. Ideana siis periaatteessa se, että on parempi kuljettaa mustikoita metsästä monta ämpärissä, kuin kädessä yksi kerrallaan. Tässä vaiheessa D:llä on ongelmana se, että missä suhteessa A:n, B:n ja C:n lähettämää dataa on järkevää kasata aina yhteen isompaan kokonaisuuteen (sanotaan tätä kokonaisuutta tästä eteenpäin ”kontiksi”).

Oletetaan nyt, että verkolla E on viisi luokkaa datalle. Edellä luokat tarkoittavat siis sitä, millä tavalla verkko E priorisoi välittämänsä tiedot (tyyliin: kiireellinen, ei kiireellinen, ei yhtään virheitä, alle 100 virhettä paketissa, tai näiden yhdistelmiä jne...). Kun D nyt antaa kontin verkon E vietäväksi, sijoittaa se kontin tätä ennen johonkin viidestä luokista ja verkko E kuljettaa sitä luokkansa mukaisella tavalla, kunnes kontti saapuu verkon rajalle, eli palvelimelle F.

Nyt on syytä huomata se, että verkolla E on tietty rajallinen kapasiteetti jokaiselle dataluokalle, eli jokaiseen konttiin ei voi laittaa ”ykkösluokan” leimaa.

Tässä kohdassa sijaitsee minun osuuteni (eli palvelimen F toiminnan matemaattinen mallintaminen/parantaminen):

Nyt palvelin F purkaa sille verkosta E saapuneet kontit.

Palvelin F käy läpi konteissa olleet paketit ja tarkistaa, onko jokin paketeista esimerkiksi myöhässä, jolloin se olisi hyvää lähettää eteenpäin kiireellisenä.

Koska nyt on niin, että G on eri protokollan verkko kuin E, on siinä todennäköisesti erilainen luokittelu datalle ja luokkia voi olla vähemmän, esimerkiksi vaikka vain neljä.

Palvelimen F tehtävänä on siis luokitella paketit uusiin kontteihin siten, että se ottaa huomioon verkossa G tarjolla olevat dataluokat ja pyrkii mahdollisimman hyvin korjaamaan verkon E aikana tulleita mahdollisia viiveitä tai muita ongelmia. Palvelimella F on nyt se pulma, että myös verkossa G on vain rajallinen kapasiteetti jokaiselle dataluokalle, eli F:n ei ole mahdollista korjata kaikkia viiveitä laittamalla ”ykkösluokkaan”, mutta sen on pyrittävä mahdollisimman hyvään tulokseen.

Tämän jälkeen kontit kulkevat verkon G läpi palvelimelle H, joka purkaa ne ja lähettää datapaketit oikeassa järjestyksessä määränpäähän I.

Edellä olevaan raamitettuun kohtaan pyrimme Murgun kanssa yhteistyössä luomaan matemaattista ja erityisesti Matlab:illa simuloitavissa olevaa ratkaisumallia. Ongelmassa on se hyvä puoli, että sen ratkaisemiseen tarvittavia matemaattisia työkaluja on runsaasti ja niitä on jo vuosikautia sovellettu arkipäivän ongelmissa, kuten rahdin- ja postinkuljetuksessa tai tuotantolaitosten optimoinnissa. Ensisijaisena pyrkimyksenämme on siis ymmärtää miten nämä työkalut ja mallit saadaan siirrettyä ”tietoliikennemaailmaan”, jonka jälkeen loppu pitäisi olla melko helppoa.

Edellisessä esimerkissä palvelin F oli sijoitettu kahden eri verkon rajalle, mutta todellisuudessa tilanne voi olla myös se, että F on esimerkiksi jokin verkon E tai G koneista (korjausta tehdään siis myös matkan aikana). Lisäksi voi olla niin, että E ja G ovat saman protokollan verkkoja, mutta E on sähköinen verkko ja G optinen tai *langaton*. Tällöin F siis jollain tavalla pyrkii kompensoimaan verkkojen välistä nopeuseroa tai ottaa huomioon verkkojen fyysiset erikoispiirteet (sähköisellä pienempi kapasiteetti kuin optisella, jne...).

Kaikkeen yllä olevaan liittyy myös iso kasa termejä kuten IP, QoS, MPLS... jne...jne, mutta koska en itsekään ole niiden asiantuntija, tiedän niistä vain matemaattisen osa-alueeni kannalta oleellisen, niin tulen lopulta siihen tulokseen että edellisen kaltainen selitys on selkein mitä pystyn tällä hetkellä asiasta antamaan, enkä sotkenut näitä termejä sen kummemmin mukaan. (IP liittyy pakettien kuljettamiseen, QoS dataluokkiin, MPLS ”konttien” kasaamiseen...)

Koska projektissa on niukat resurssit eli ainoastaan kaksi tutkijaa, niin muut projektin osa-alueista on jätetty tällä haavaa lähes huomiotta, eli etenemme eteenpäin osa-alue kerrallaan. Täten myös minulle Murgulta välittyvä tieto rajoittuu vain kulloisenkin osa-alueen kannalta oleelliseen (mikä on siis vain käytettävissä olevan ajankäytön optimointia).