

Puhelinverkon simulointia mikrotietokoneella

Laajojen hierarkisten ja vuorovaikutteisten systeemien simulointi on tietokonetekniikan ja ohjelmistojen kehityksen myötä saanut yhä keskeisemmän sijan tutkimustyössä. Vielä muutamia vuosia sitten vaadittiin näihin tutkimuksiin supermini- tai mainframe-koneita, mutta nyky-

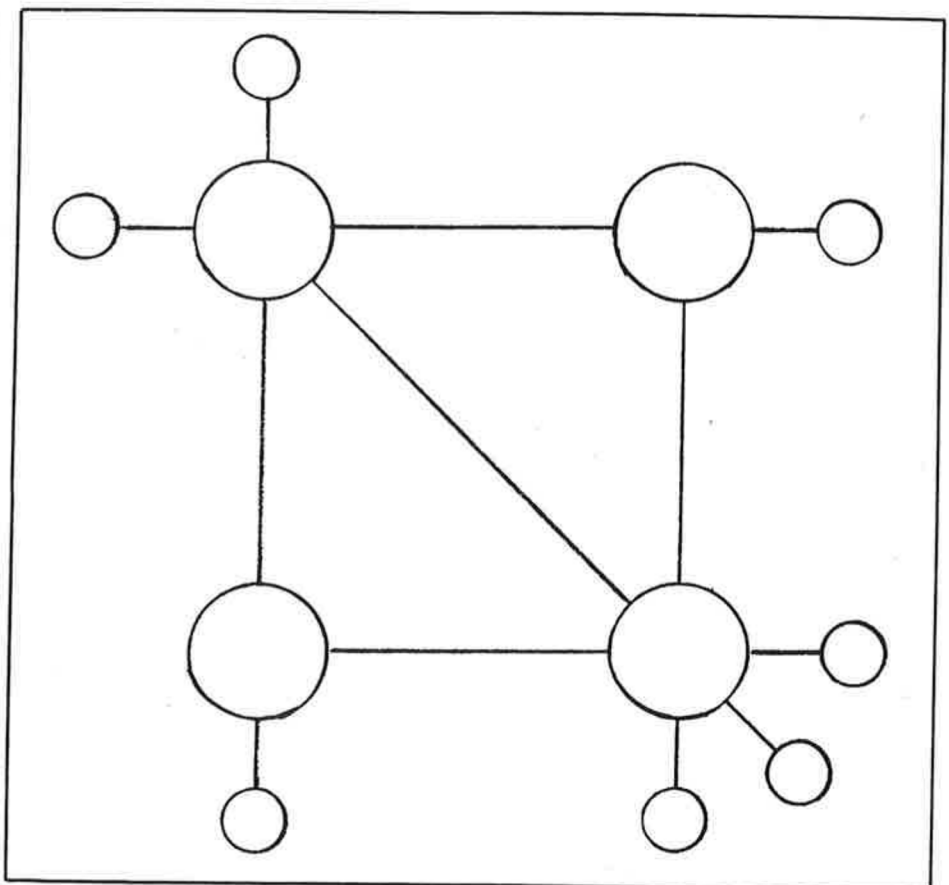
ään jo tehokkaalla mikrotietokoneellakin on toteutettavissa vaativia projekteja.

Seuraavassa eräästä tällaisesta, yhtymän puhelinverkon simulointiin soveltuvasta ohjelmistosta yleispiirteitä ja käyttökokemuksia.

Simulointi, jonka luontevin suomennos lienee jäljittely, voidaan täsmällisesti määritellä tekniikaksi, jossa keinotekoisien dynaamisten mallien käyttäytymisestä tehdään päätelmiä todellisen systeemin toiminnasta. Eräät tutkijat tosin rajoittavat simulointiin kuuluvaksi vain mallituksen ja ohjelmistotekniikan katsoen ajon ja päätelmien teon olevan ns. pelaamista. Simuloinnin yleistymiseen on keskeisesti vaikuttanut tarve saada tuntumaa järjestelmiin, jotka pystytään analysoimaan pitkälle, mutta jotka ovat ihmisen säätelykyvyn ulottumattomissa tai sitten pelaaminen on inhimillisistä, taloudellisista ym. syistä johtuen mahdotonta.

Simulointitutkimuksen rakenne

Käytännössä simulointitutkimuksen voidaan katsoa jakautuvan viiteen vaiheeseen: 1. Järjestelmän analyysi eli esiintyvien olioiden, niiden attribuuttien (ominaisuuksien) ja relaatioiden määrittäminen, 2. Järjestelmän mallittaminen eli oleellisten olioiden, attribuuttien ja relaatioiden valinta ja yhdistäminen systeemiksi, 3. Mallin verifiointi eli virheettömyyden tarkastelu, 4. Mallin validointi eli jäljittelevyyden testaus ja 5. Mallin analyysi, vastaako se toiminnaltaan tavoitteita. Malli voi olla luonteeltaan de-



Kuva 1. Eräs simulaattorissa käytettävä verkkomalli, neljä solmua siirtoteineen ja ti-laajineen.

terministinen (järjestelmän tunnettuja matemaattisia tai fysikaalisia deterministisiä riippuvuuksia laskeva) tai stokastinen (havaittuja prosesseja pseudosatunnaisesti jäljittelevä), diskreetti tai jatkuva päivytysperiaatteen mukaan tai dynaaminen tai staattinen mallin parametrien luonteesta riippuen. Mallin tyyppi ratkeaa yleensä varsin luonteesta systeemin ominaisuuksien asettamien erityisehtojen perusteella, mutta myös laskentaresurssit on syytä huomioida.

Puhelinverkkomalli

Simulantina tutkimuksessa on pidetty yhtymää palvelevaa verkkoa, joka ajatellaan olevan fyysisesti n. 100×100 km:n alueelle levittäytyneenä. Se sisältää vähintään neljä solmua, joihin on suoraan tai alasolmujen tai keskittimien avulla kytketty useita satoja puhelimia. Kunkin tilaajan tai ainakin häntä lähinnä palvelevan solmun oletetaan tietävän kutsuttavan puhelimen solmun eli jonkinlaiset puhelinluettelot ovat olemassa, eikä täysin umpimähkäisiä menetelmiä, esim. tulvahuakua sovelleta. Väylöityksen optimaalisuuteen pyritään vain impilisiittisesti, suositusluonteisin prioriteetein. Verkko on määritelty sangen riippumattomaksi, eikä sen yhteyksiä johonkin yliverkkoon tarkastella. Sen sijaan oleelliseksi piirteeksi on katsottu verkon liikkuvuus; mikä tahansa solmuista voi erota tai liittyä verkkoon itsenäisesti. Kuvaan 1 on hahmoteltuna eräs verkon mahdollinen konfiguraatio.

Koska jo nelisolmuisella verkolla on simuloitaessa oleellisesti kaikki tämän tyyppisen verkon ominaisuudet, on laskentatyön rajoittamiseksi pitäyditty siinä. Kuvassa 2 on esitetty simulantin perusversio, täysin kytketty verkko, jossa on numeroituna kaikki oliot ts. solmut ja väylät. Niistä voidaan valita tarkasteltavaksi mikä tahansa mielekäs kombinaatio eli sellainen, jossa väylät päätyvät solmuihin. Alasolmut ja keskittimet on abstrahoitu ja kutsuja käsitellään vasta kun ne saapuvat solmuun eli verkko käyttäytyy kuten puhelimet olisivat siihen suoraan kytketyt. Solmujen ajatellaan olevan käsivälitteisiä keskuksia ja ne hakevat muita solmuja välitysprioriteettien ohjaamina.

Järjestelmälle on luotu stokastinen, dynaaminen ja diskreetti mikrotietokonepohjainen simulointimalli. Stokastisuudella on vältetty jäykkyys ja piirreköyhyys, jota esim. odotuskuormitusten laskenta puhtaasti deterministisillä laskelmoilla aiheuttaa ja samalla mahdollistettu olioiden attribuuttien optimointi, joka on ollut eräs tavoitteista. Verkon liikkuvuuden ja optimoinnin välttämättömänä premissinä taas on ollut dynaamisuus parametriasetuksissa, jota on toteutettu mahdollisimman laajasti.

Laskentaympäristö

Laitteistona, jolle ohjelmisto on laadittu, on ollut Olivettin M 20 mikrotietokone 160 ktavun keskusmuistilla ja 2×320 ktavun minikalvoväyläasemalla. Suoritin,

Zilogin 16 bittinen 8001 on sangen nopea ja käyttöjärjestelmä, laitteistovalmistajan PCOS, ei yksinkertaisuudestaan johtuen rasita liikaa keskusyksikköä. Ohjelmointikielenä on ollut simulointiin sinänsä täysin erikoistumaton Microsoft BASIC. Tulkatuksi sen suoritus on kuitenkin yllättävän nopeaa ottaen huomioon simulointikoodin ehto-, hyppy- ja silmukkatheyden; yksinkertaiset verkko-konfiguraatiot lasketaan 3–4 kertaisella nopeudella reaaliaikaa nähden. Työtillä on simulointiohjelmaa ajatellen liian pieni, 33 ktavua, mutta overlay-tekniikalla ohjelmien kokorajoitusta voidaan kiertää. Levyviittausten hitaudesta johtuen tämä tietenkin pitkittää hieman laskentaa, samoin kuin työtilan pakkolaajennus, joka syö systeemiohjelmien työtilaa.

Ohjelmallinen toteutus

Laajimman simulaattoriversion, jota seuraavassa yksinomaan käsitellään, rakennettiin esittäjä makromoduleina kuvan 3 mukaisesti. Kukin moduli sisältää tehtäväkohtaisesti erikoistuneita pääohjelmia, aliohjelmia ja alirutiineja yhteensä 28 kappaletta. Nuolet kuvaavat kontrollin mahdollisia haarautumisteitä.

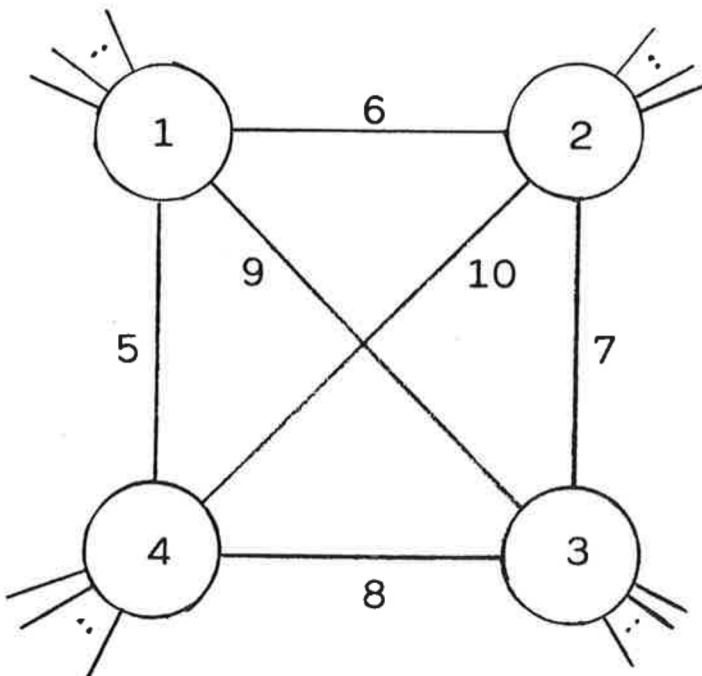
Aloitukset

Initialisointilohkolla ohjelmalle luetaan levyiltä tai päätteeltä sen tarvitsemat aloitusparametrit. Näitä ovat mm. verkon topologian määrittelevä kytkentätaulukko (4×4 matriisi kuvan 2 numeroinnein), priorisointitaulut (kullekin kutsujasolmulle muihin solmuihin ensimmäisen, toisen ja kolmannen tason puhelun välittäjäsolmut naapureista) ja puhelinten varausoptio. Samoin määritellään liikenneparametrit; puhelun keskipituus, estyneen kutsun uusintasoiton odotusaika, tilaajan kärsivällisyys ja kutsutiheydet solmupareittain suunta huomioiden. Näistä ohjelma laskee virtuaaliparametrisia ja mm. Christensenin estimaatit solmujen ja väylien kapasiteeteille. Nämä voidaan hyväksyä tai antaa paremmat, jotka tulostetaan levyille. Kun vielä mm. ohjelman kolme pseudosatunnaisrutiinia on siemenetty, voidaan haarautua kontrollilohkoon.

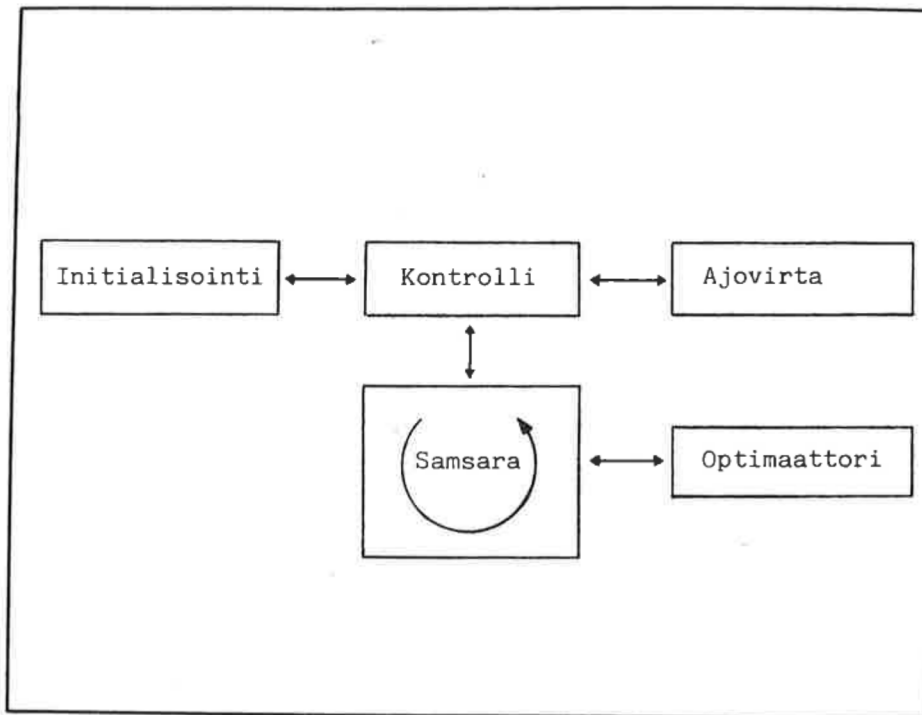
Aloitusmodulissa käsitellään suurta osaa ohjelman kaikkiaan yli 40:stä taulukosta. Näistä ensin osa on kokonaislukutyyppeistä, kuten skalaarimuuttujistakin, sillä keskusmuistitilaa on näin pyrittävä säästämään. Silti ohjelman välittömästi tarvitsemat parametrit vievät yli 4 ktavua muistia.

Ohjaus

Kontrollimodulissa, jota ohjataan päätteeltä tai ajovirrasta, päätetään ehtoluotien perusteella laskentatyyppi, joka voi olla: 1. Normaali päätekyttö, systeemillä pelaaminen, 2. Erikseen ohjelmoitavan ajovirran alaisuudessa suoritettava prosessointi tai 3. Edelliset yhdistettynä verkon eriaistaiseen optimointiin.



Kuva 2. Simulantin perusversio — täysin kytketty verkko. Oliot, solmut ja väylät on numeroitu 1–10.



Kuva 3. Laajimman simulaattoriversion ohjelman rakenne makromodulein esitettynä.

Samsara

Simulaattorin ytimenä voidaan pitää Samsaraa, jossa syklistä 3–10 sekunnin (reaaliaikaa) kvanteissa luodaan kutsuja, pyritään reitittämään ne puheliksi, päätetään puheluita ja päivitetään kuormituksia. Kutsuja luodaan Poisson-jakautuneesti tietty määrä kvanttia kohden koko verkolle. Ne jaetaan kutsujavastannottajapareille solmujen aktiivisuuden ja liikenteen jakautuman mukaisesti. Laskennassa käytetään tasajakautuneita satunnaislukuja, jotka kuvataan inversiomenetelmän mukaisesti jakautumien kertymäfunktioista laskettujen solmu- ja solmuparikohtaisten taulukoiden avulla.

Solmut ja väylät ovat simulaattorin olioiden ja attribuutteja ovat solmuilla niiden kytkineliemien ja puhelimien määrät ja väylillä johto- tai kanavamäärät. Määrillä tarkoitetaan sekä maksimimääriä että käytössä olevia määriä ja näiden suhdetta, erityisesti estotilannetta. Relatiivit ovat topologinen naapurisuus, solmuilla naapurisolmut ja väylät niihin ja väylillä päätesolmut. Jokainen systeemiin tullut kutsu aiheuttaa ns. aktiviteetin eli tutkimuksen, jossa reitityksen erikoistunut koodi vertailee olioiden attribuutteja ja pyrkii priorisointien ohjauksessa reitittämään kutsun vastaanottajalle. Jos rutiini onnistuu, kirjataan puhelun reitti taulukoon tai sen ylivuototiedostoon levyille, päivitetään sen käyttämien olioiden attribuutit ja arvotaan puhelulle eksponentiaalisesti jakautunut kesto.

Jos kutsu ei löydy tietä vastaanottajalle, generoidaan sille eksponentiaalisesti jakautunut odotusaika ja mikäli tämä ei ylitä käyttäjän asettamaa kärsivällisyysparametria, pinotaan kutsu odottamaan. Jokaisella kvantilla seulotaan ensin uusittavien kutsujen pino ja näiden reitityksen jälkeen käsitellään vastasyntyneiden

kutsujen pino.

Reitityksen jälkeen siirrytään Samsaraan purkaus-moduliin, jossa etsitään kvantin aikana päättyvät puhelut. Jokainen näistä synnyttää prosessin, jossa puhelun käyttämien olioiden attribuutit päivitetään ja taulukkovaraukset puretaan. Modulissa suoritetaan myös systeemin kuormituksen tilastointi jos tätä suorittavat rutiinit on optimaattorista aktivoitu. Purkauksen jälkeen tutkitaan onko keskeytyksiä ennen seuraavaa kvanttia ja mikäli ei, hypätään seuraavaan kvanttiin jatkamaan kiertoa.

Laajennukset

Verkon dynamiikan lisäämiseksi ja esim. solmujen vapaan liikkuvuuden simuloimiseksi on ohjelmallisesti toteutettu erittäin laajojen systeemimuutosten mahdollisuus. Näillä tarkoitetaan mielivaltaisia liikenteen vaihteluita, solmujen ja väylien kapasiteettien muutoksia tai kutsuprioriteettien uudelleenmäärittelyä. Uudet arvot luetaan ohjelmallisesti levytiedostosta ajanhetkellä, joka määritellään ohjelman eräessä modulissa ohjelmoitavissa ajovirtapinoissa. Tämä FIFO-rakenne määrää systeemin tulevaisuuden haluttuun aikaan saakka täydellisesti. Ajovirta voidaan suorittaa haluttu määrä peräkkäin. Kapasiteettien muutoksilla voidaan kuvata rakenneosien pettämistä, korjausta, laajennusta jne. Puhelujen keskeytetään ohjelmallisesti minimimäärä kapasiteettien alentuessa kuormitusta pienemmäksi. Liikenteen muutoksilla voidaan kuvata mitä moninaisinta ilmiökokoaisuutta; vuorokautisten vaihtelujen lisäksi voidaan simuloida mm. puheluiden rinnalla välitettävää sanomaliikennettä ja

väylille reititystä odottamaan jääviä kutsuja.

Verkko-liikenne-konfiguraation optimointiin on ohjelmistossa erillinen rutiini, selektiivinen, dynaaminen ja takaisin-kytketty tilasäättäjä, joka levyiltä tai päätteeltä lukemansa datan ohjaamana säätelee solmujen ja väylien kapasiteetteja portaittain optimiarvoihinsa. Se pyrkii primaaristi minimoimaan optimoitaviksi valittujen olioiden aikaeston alle annettun tason ja sekundaarisesti maksimoimaan samojen olioiden käyttöastetta. Optimaattoria voidaan ohjata ajovirrasta kuten systeemimuutoksia – näitä voidaan esim. vuorotella, jolloin sopivin optimaattorin parametrisetukset saadaan ajovirrasta lopputulokseksi tietyin systeemihistorian optimaalisesti annetuilla esto- ja käyttöastetasoilla sietävä verkko.

Yhteenvedon voidaan todeta, että ohjelmistototeutuksessa on pyritty modulaariseen selkeyteen ja koodissa mahdollisimman kompaktiin esitykseen. Systeemin määrittelyssä ja parametrisoinnissa on rajoitukset supistettu minimiin. Muistitilavaatimus on ilman grafiikan ja muiden täydennysten käyttöä saatu rajoitettua 45 ktavuun.

Simulaattori työkaluna

Simulaattoria voidaan interaktiivisessa pääteikäytössä soveltaa lähinnä mielenkiintoisten verkko-liikenne-konfiguraatioiden etsintään, optimaattorin parametrien testaukseen ja tulevaisuuksien sommitteluun ajovirtaa varten. Varsinaiset luotettavuuden ja välityskyvyn testaukset ja optimointi suoritetaan ajovirralla. Näin sen vuoksi, että merkitsevät näytemäärät saadaan vain riittävän laajalla otannalla, johon päästään joko vilkkaalla liikenteellä (100–200 erlangia, reaaliaikainen laskenta) tai pitkällä tarkkailujaksoilla. Testaukset useilla erilaisilla parametrijaksoilla voidaan suorittaa yhdessä ajovirrassa, joten laskenta on hyvin itsenäistä ja vaivatonta.

Simulaattori ei ole vielä lopullisesti validoitu, mutta ankarastikin arvioiden silloin on arvoa tutkimusvälineenä selvittäessä mm. vaihtoehtoväylöityksiä eri verkkotyypin luotettavuutta annettua historialla. Mallin laajentamisen pääasialliset esteet mikroympäristössä näyttäivät palautuvan suorittimen tehoon; mutkikkaat päivitysopeeraatiot aiheuttavat reaaliaikaisesta laskennasta jäämisen. Siten esim. alakeskusten yksilöllisten ominaisuuksien ja puhelimien sekä puheluiden erityispriorisointien laaja huomioonottaminen ei kannata.

Lähteitä

Naylor T H, Balintfy V L, Burdick D S, Chu K: Computer Simulation Techniques, Wiley -66

Rahko K, Viitaharju K, Harva S, Tams M, Kukko A: Kaukopuhelinverkon simulointi, VTT Telet. lab., raportti 10/74