

Tosiaikainen ajoneuvoliikenteen reitinsuunnittelu

Petri Blomqvist, Matrex Oy

Frank Cameron, Tampereen teknillinen yliopisto, Porin yksikkö

Terhi Kaarakka, Joensuun yliopisto

Tuomo Kauranne, Lappeenrannan teknillinen yliopisto

Anne Setämaa, Jyväskylän yliopisto

Risto Silvennoinen, Tampereen teknillinen yliopisto

Liisa Torikka, Lappeenrannan teknillinen yliopisto

Esko Turunen, Tampereen teknillinen yliopisto

1. Johdanto

Henkilöauton kuljettajan reitinvalinta on ilmiö, johon vaikuttaa monia tekijöitä. Valintaan vaikuttavat niin henkilökohtaiset mieltymykset kuin tiedot mahdollisista reittivaihtoehdoista. Monet matkustajat haluavat päästä mahdollisimman nopeasti perille, joten reitin matka-aika on yksi keskeinen reitin houkuttelevuutta kuvaava tekijä.

Useimmiten kuljettajan tiedot eri reittivaihtoehdoista ovat puutteellisia. Tilanteet tie- ja katuverkolla muuttuvat koko ajan ja paras reitti kello 8.00 ei välttämättä ole paras enää kello 9.00. Tosiaikaisen ajoneuvoliikenteen reitinsuunnittelujärjestelmän tarkoituksena on tuottaa ajantasaisia reittiohjeita ajoneuvojen kuljettajille. Lähtökohtana on pidetty sitä, että ajoneuvot ohjataan sellaisille reiteille, että kohdeliikenteen matka-ajan summa minimoituu. Kohdeliikenteellä tarkoitetaan tässä reitinsuunnittelujärjestelmällä ohjattavia ajoneuvoja. Reitinsuunnittelujärjestelmä optimoi ajoneuvojen reittejä koko systeemin kannalta. Tämä tarkoittaa sitä, että kaikkia yksittäisiä autoja ei välttämättä ohjata niiden kannalta nopeimmalle reitille, jos tämä hidastaa muuta liikennettä. Systeemioptimi antaa siis kaikkien kohdeliikenteen ajoneuvojen kokonaisuuden kannalta parhaimmat reitit. Tämä on liikenneviranomaisten näkökulma asiaan, sillä sen tavoitteena on liikenteen yleinen sujuvuus.

Tässä työssä tutkittuun reitinsuunnittelujärjestelmään on kaavailtu kahdenlaista käyttäjärajapintaa. Ensinnäkin järjestelmä tuottaa ajantasaisia reittiohjeita yhdelle käyttäjälle. Tämä tarkoittaa sitä, että autoilija lähettää reitinsuunnittelujärjestelmälle kyselyn, jossa hän kertoo minne hän haluaa päästä. Järjestelmä palauttaa käyttäjälle parhaan reitin, joka ottaa huomioon tilanteen tieverkolla sillä hetkellä. Järjestelmän kertoma reitti perustuu matka-aikaennusteisiin. Jos auto on kaukana määränpästä, tilanne liikenneverkolla ehtii muuttua moneen kertaan ja paras reitti muuttuu sen myötä. Matka-aikaennusteiden avulla pystytään arvioimaan näitä muutoksia ja laskemaan paras reitti ottaen huomioon nämä ennusteet.

Matka-aikaennusteet perustuvat senhetkisen liikenteen ja eri taustamuuttujien (esim. sää) tarkkaan hahmottamiseen ja tulevaisuustietojen huomioimiseen (tiedetään esimerkiksi, että jäähallissa on alkamassa puolen tunnin päästä jääkiekko-ottelu). Nykytilanteen ja tulevaisuustietojen perusteella voidaan historiatietoihin perustuvasta tietokannasta etsiä mahdollisimman samankaltainen tilanne. Senhetkisten matka-aikojen voidaan arvioida kehittyvän samalla tavalla kuin vastaavan kaltaisessa tilanteessa menneisyydessä.

Toinen käyttäjärajapinta on teiden varsilla olevat reittiopasteet, jotka on tarkoitettu useampien ajoneuvojen ohjaamiseen. Reittiopasteet soveltuvat tilanteisiin, jossa reittivaihtoehtoja on vain muutama. Reittiopastejärjestelmä voi ennakoida ruuhkien syntymistä ja pyrkii ohjaamaan liikennettä vaihtoehtoisille reiteille siten että mikään reiteistä ei ruuhkautuisi. On tärkeää että reittiohjetta päivitetään riittävän usein. Esimerkiksi kahden rinnakkaisen väylän tapauksessa ei saisi tapahtua niin, että liikennettä ohjataan toiselle reitille niin kauan kunnes se on tukkeutunut, jonka jälkeen ohjetta muutetaan.

2. Menetelmät

Tosiaikaisen ajoneuvoliikenteen reitinsuunnittelujärjestelmän systeemikaavio on esitetty kuvassa 2. Järjestelmä koostuu kolmesta pääkomponentista.

1. Ohjelma, joka sijoittelee kohdeliikenteen ajasta riippuvalle liikenneverkolle. Sijoitteluohjelmalla voidaan käyttää esimerkiksi Emme/2:sta. Ohjelma tarvitsee lähtötietoina kohdeliikenteen kysyntämatriisin (lähtö/määräpaikkamatriisi) tarkasteltavana ajanhetkenä ja aikalaajennetun liikenneverkkokuvauksen.
2. Linkkikustannusten ennustemalli ennustaa tieverkon senhetkisten liikennemäärien, säätilan, ajanhetken ja muiden muuttujien avulla tulevaisuuden linkkikustannukset (=matka-ajat). Ennustemalli etsii tietokannan historiatiiedoista mahdollisimman paljon sen hetkistä tilannetta muistuttavan tapauksen ja ennustaa matka-aikojen kehittyvän samalla tavalla kuin silloin tapahtui. Linkkikustannusten ennustemalli voi ottaa huomioon myös tietoja ennakoituista tulevaisuuden tapahtumista.
3. Iterointiprosessi sijoitteluohjelman ja ennustemallin välillä. Sijoitteluohjelma tuottaa alkuarvauksen kohdeliikenteen reittivalinnoista. Tämän jälkeen ennustemallilla tuotetaan ennuste linkkikustannuksista, kun kohdeliikenne on lisätty liikenneverkolle muun liikenteen sekaan. Ennustetut linkkikustannukset siirretään sijoitteluohjelmaan, joka laskee uudet reittivalinnat, jotka jälleen palautetaan ennustemallille, joka laskee uudet linkkikustannukset. Näin edetään kunnes kahden peräkkäisen iterointikierroksen reittivalinnat eivät enää muutu huomattavasti, eli iterointi on supennut kohti tasapainotilannetta. Iterointiprosessia voidaan täydentää tarvittaessa geneettisen algoritmin tuottamalla tai tietokannasta poimituilla vaihtoehtoisilla reittivalinnoilla, mikäli normaali iteraatioprosessi ei suppene.

Seuraavassa tutkitaan tarkemmin kutakin yllä mainittua menetelmäosaa.

3. Liikenteen sijoittelu

Alkuratkaisun hakemiseen voidaan käyttää lyhimmän polun algoritmeja verkoille, joissa solmujen välisten linkkien kustannukset ovat ajasta riippuvia. Lähteessä [12] on tähän tarkoitukseen esitetty versio Dijkstran algoritmista. Usean ajoneuvon tapausta käsitellään kirjoituksessa [8].

Toinen lähestymistapa on käyttää aikalaajennettuja liikenneverkkoja. Kuvassa 1 on esitetty aikalaajennetun verkon koodausperiaate. Kuvan oikealla puolella on kaksi

staattista verkkoa ajanhetkillä $t=0$ ja $t=1$. Vasemmalla puolella staattiset verkot on yhdistetty yhdeksi dynaamiseksi verkkoksi. Dynaamiseen verkkoon on otettu koordinaattien (x,y) lisäksi kolmanneksi ulottuvuudeksi aika. Linkit on piirretty aina eri aikatasojen välille.

Esimerkiksi ajanhetkellä 0 solmusta 1 solmuun 2 kuluu aikaa 1 minuutti. Tämä näkyy dynaamisessa verkossa siten, että linkin 10 ja 21 välinen matka-aika on 1 minuutti. Ajanhetkellä 0 solmusta 1 solmuun 4 kuluu aikaa 2 minuuttia. Dynaamisessa verkossa on siis linkki solmujen 10 ja 42 välillä ja sen matka-aika on 2 minuuttia. Ajanhetkellä 1 solmujen 1 ja 4 välinen matka-aika on lyhentynyt yhteen minuuttiin. Dynaamisessa verkossa tämä näkyy siten, että solmujen 11 ja 42 välisen linkin matka-aika on 1 minuutti.

Aikalaajennettu verkkokuvaus johtaa siihen, että solmujen ja linkkien määrä kertautuu aika-askelten lukumäärällä verrattuna staattiseen verkkoon. Se mahdollistaa kuitenkin sen, että kysynnän ja matkavastusten muuttumista ajan funktiona voidaan mallintaa perinteisillä sijoitteluohjelmilla kuten Emme/2:lla.

Lyhimmän polun hakemiseen aikalaajennetussa verkossa voidaan käyttää ahnetta Dijkstran algoritmia. On kuitenkin huomattava, että kyse ei ole aivan suoraan aikalaajennetun verkon kahden solmun välisen lyhimmän reitin hausta, koska reitin loppusolmu ei ole tiedossa, sehän liikkuu ajan mukana.

Aikalaajennettu verkko näyttää melko epähavainnolliselta suhteessa staattiseen verkkoon. On kuitenkin mahdollista rakentaa tietokoneohjelmia, jotka muuttavat normaalin staattisen verkkokuvauksen dynaamiseksi. Eri aika-askleet voidaan laittaa tällöin päällekkäin (samat x - ja y -koordinaatit). Lisäksi voidaan tehdä ohjelmia jotka muuttavat dynaamisen sijoittelun tulokset staattiseen muotoon (summataan liikennevirrat dynaamisilla linkeillä staattisten linkkien virroiksi).

Edellä olevaa voidaan soveltaa yhden tai useamman ajoneuvon tilanteeseen. Jos käyttäjiä on kuitenkin paljon, niin niiden välinen kytkentä on otettava huomioon, muuten järjestelmä voi jopa edesauttaa ruuhkan syntymistä. Siis probleema siirtyy käyttäjien lukumäärän mukana dynaamisesta lyhimmän reitin hausta (Dynamic Shortest Path) dynaamiseen liikenteen sijoitteluun (Dynamic Traffic Assignment). Jälkimmäisestä on laaja katsausartikkeli viitteessä [10].

4. Ennustemalli

Perusoletuksena linkkikustannusten (matka-aikojen) ennustamisessa on, että tilanne on suurin piirtein säännöllinen, ei kaoottinen. Myös poikkeustilanteissa (kolarit jne) voi olla säännöllisyyttä, ks. näiden käsittelystä esim. [11].

Näin voidaan kerätä tietokanta tyypillisistä liikennetilanteista. Esimerkiksi: "Alkava aamuruuhka alueella ..., kello 7.00-8.00, kesä, arkipäivä, hyvä sää, ..., liikennemäärä on ... ". Tietojen luonteen vuoksi sumeiden joukkojen käyttö voi olla eduksi.

Syöttötietoina ennusteelle voivat olla liikennemäärät kaikissa linkeissä, viikonpäivä, kellonaika, sää, ruuhkatiedot, onnettomuudet, tietyöt jne.

Tietojen keräämiseksi tyypillisistä liikennetilanteista voidaan käyttää myös asiantuntijoiden (taksinkuljettajat, bussinkuljettajat, liikenneviranomaiset, jne.) haastattelua. Tai suuresta tietoaaineistosta voidaan louhia data-mining-tekniikoilla oleellista tietoa. Eräs mahdollinen näistä on GUHA (general unary hypothesis analysis, <http://lispminer.vse.cz/index.html>), joka hakee oleellisia ja tilastollisesti merkitseviä riippuvuuksia numeerisesta matriisista.

Tietojen kokoamisen työmäärästä esimerkkinä mainittakoon, että Lahti-Heinola välin matka-ajan ennustemallin tietojen koostamiseen meni 1-2 henkilötyöviikkoa (Esko Turunen).

Ennustemalli voitaisiin rakentaa seuraavasti. Tietokantapankissa on staattisesti kuvattuina kukin tyypillinen liikennetilanne erikseen. Esimerkiksi:

JOS $A = x_1$ ja $B = y_1$ ja $C = z_1$, NIIN matka-ajat ovat T_1 (täsmällisiä lukuja)

⋮

JOS $A = x_k$ ja $B = y_k$ ja $C = z_k$, NIIN matka-ajat ovat T_k (täsmällisiä lukuja)

Todellista liikennetilannetta verrataan erikseen jokaisen säännön JOS-osaan eli etsitään se sääntö, joka muistuttaa todellista input-arvoa eniten. (Tämä ei ole välttämättä yksikäsitteinen, asiantuntijaa tarvitaan antamaan erillisiä sääntöjä 50-50 tilanteessa.) Silloin output-arvona on kyseiseen sääntöön liittyvä output T_j . (Ks. yksityiskohtat esim. kirjasta [13].)

5. Iterointi

Ehdotetun sijoittelua ja ennustemallia vuorottelevan iteroinnin suppenemisestä ei ole varmuutta. Laskennalliselta vaativuudeltaan ajasta ja liikennemääristä riippuva liikennesijoittelu on ilmeisesti NP-kova. Tarkoilla menetelmillä ei siis realistisen isoja aineistoja pystytä ratkaisemaan optimaalisuuteen asti. Heuristisista algoritmeista edellisten tueksi voidaan ajatella ainakin geneettistä algoritmia. Lähteessä [4] on dynaamiseen lyhimmän polun ongelmaan kehitetty geneettinen algoritmi.

6. Johtopäätökset

Tässä työssä on hahmoteltu tosiaikaisen ajoneuvoliikenteen reitinsuunnittelujärjestelmän keskeiset toimintaperiaatteet puuttumatta vielä yksityiskohtiin. Kuvassa 2 esitetty systeemikaavio voidaan toteuttaa asteittain. Systeemin ytimenä toimii liikenteen sijoitteluohjelma, joka etsii parhaat reitit dynaamisella verkolla. Dynaamisen verkon rakentamista ja toimivuutta on syytä testata todellisella liikenneaineistolla. On selvítettävä kuinka paljon parempia (= paremmin todellisuutta vastaavia) tuloksia sillä saavutetaan suhteessa staattiseen verkkokuvaukseen. Aikalaajennettu verkkokuvaus ja sen vertailu staattiseen verkkokuvaukseen on hyvä aihe lisätutkimuksille (esim. diplomityölle).

Tärkeysjärjestyksessä seuraavana tulee linkkikustannusten ennustemalli. Staattisella verkolla linkkikustannusten oletetaan kasvavan liikennemäärän funktiona (ks. kuva 3). Staattisen verkon linkkikustannusfunktioita voidaan käyttää myös lähtökohtana dynaamisessa verkossa. Jotta dynaamisesta verkkokuvauksesta saataisiin

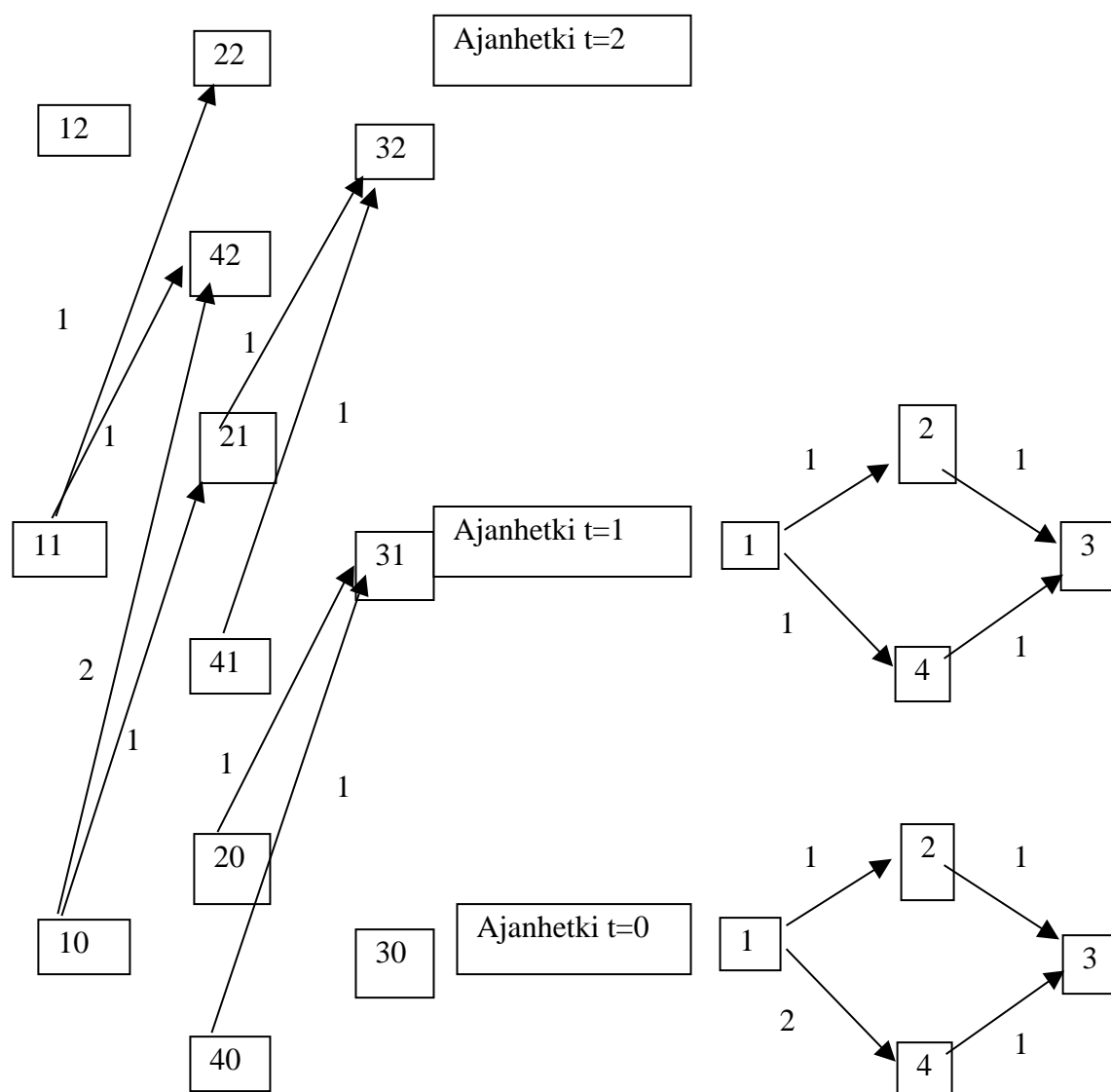
mahdollisimman paljon hyötyä ja jotta reitinsuunnittelujärjestelmä tuottaisi hyviä reittiohjeita, tulisi staattisista kustannusfunktioista kuitenkin siirtyä kohti linkkikustannusten ennustemallia.

Linkkikustannusten ennustemallin toteuttaminen vaatii paljon resursseja, sillä jokainen tieverkon linkki pitää periaatteessa mallintaa erikseen. Tämä vaatii hyvin paljon historiatietoja liikennemääristä ja eri taustamuuttujista. Valtavasta tilastoaineistosta oleelliset relaatiot voidaan louhia esiin esim. data-mining tekniikoilla. Käytännössä linkit voidaan todennäköisesti luokitella erilaisiin ryhmiin siten, että ryhmien sisällä voidaan käyttää samanlaisia kustannusten ennustemalleja.

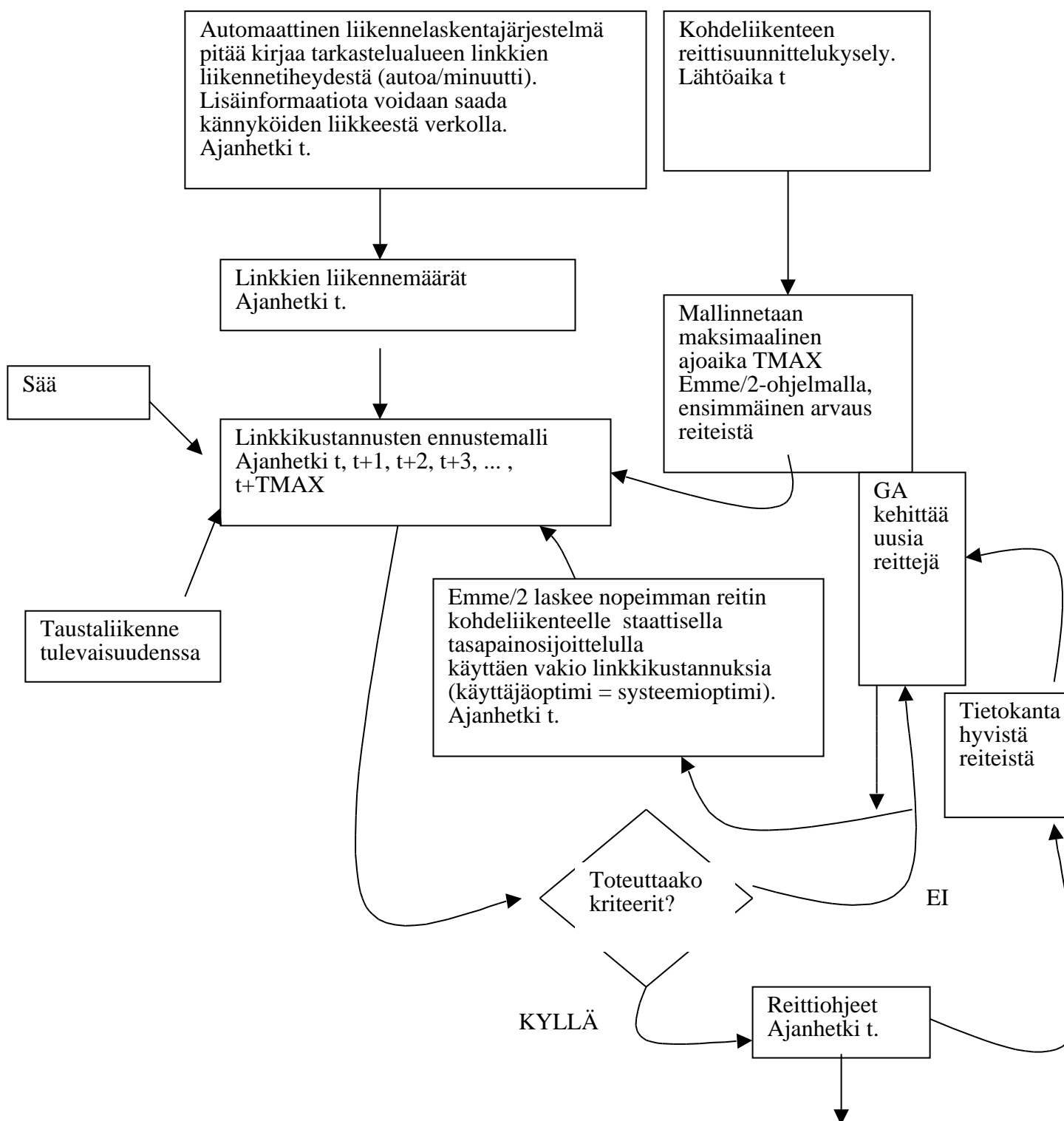
Kirjallisuutta

1. Akamatsu: A dynamic traffic equilibrium assignment paradox, Transportation Research Part B, 1999.
2. Azaron, A. and Kianfar, F.: Dynamic shortest path in stochastic dynamic networks: Ship routing problem, Eur. J. Op. Res. 144 (2003), 138-156.
3. Cai, Sha and Wong: Time-varying minimum cost flow problems, European Journal of operational research, 1999.
4. Davies, C. and Lingras, P.: Genetic Algorithms for rerouting shortest paths in dynamic and stochastic networks. Eur. J. Op. Res. 144 (2003), 27-38.
5. Eppstein 1998: Finding the k shortest paths, Siam J. Comput Vol 28, No. 2, pp. 652-673.
6. Fleischer, L. and Tardos, E.: Efficient continuous-time dynamic network flow algorithms, Operations Research Letters 23 (1998) 71-80.
7. Fu, L.: An adaptive routing algorithm for in-vehicle route guidance systems with real-time information, Transportation Research Part B, 35 (2001), 749-765.
8. Ichoua, J.S., Gendreau, M. and Potvin, J-Y: Vehicle dispatching with time-dependent travel times, Eur. J. Op. Res. 144 (2003), 32-39.
9. Kaufman, Nonis, Smith 1998: A mixed integer linear programming model for dynamic route guidance, Transpn Res.-B, Vol. 32, No. 6, pp. 431-440.
10. Peeta, P. and Ziliaskopoulos: Foundations of Dynamic Traffic Assignment: The Past, the Present and the Future, Networks and Spatial Economics, 1 (2001), 233-265

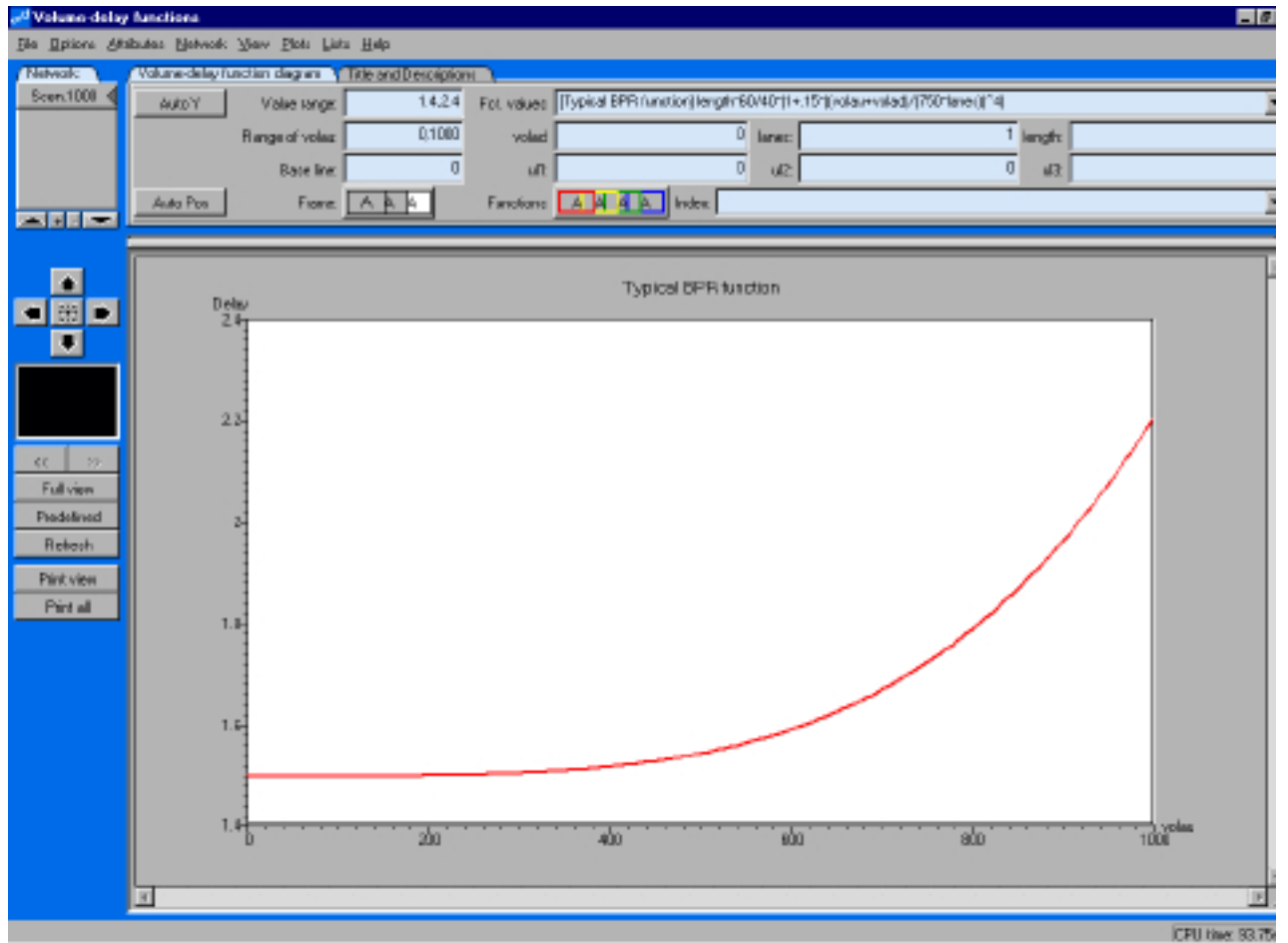
11. Sadek, A.W., Smith, B.L. and Demetsky, M. J.: A prototype case-based reasoning system for real-time freeway traffic routing, *Transportation Research Part C*, 9 (2001) 353-380.
12. Sung, K. Bell, M.G. H., Seong, M. and Park, S.: Shortest paths in a network with time-dependent flow speeds, *Eur. J. Op. Res.* 121 (2000), 32-39.
13. Turunen, Esko: *Mathematics behind Fuzzy Logic*, Springer Verlag, Heidelberg, 1999.
14. Waller, S. T. and Ziliaskopoulos, A.: On the Online Shortest Path Problem, December 2000, Submitted to *Networks*.



Kuva 1: Aikalaajennettu verkko



Kuva 2: Systemikaavio



Kuva 3: Tyypillinen linkin kustannusfunktio (vaaka-akselilla liikennemäärä, pystyakselilla matka-aika)