

Kotihoidon optimointi

Joni Pakarinen

28.1.2014

Itä-Suomen yliopisto
Tietojenkäsittelytiede
Pro gradu -tutkielma

Tiivistelmä

Tässä tutkielmassa tutustutaan työvuorojen aikataulutukseen ja kotihoidon optimointiongelmaan kirjallisuuden sekä esimerkkijärjestelmän avulla. Lisäksi pohditaan kauppatkustajan ongelman soveltuvuutta kotihoidon optimointiin. Tutkielmassa käsitellään myös reitityspalveluita ja algoritmeja. Näiden ohella tutustutaan reitityspalvelun rakentamiseen sekä siihen tarvittaviin materiaaleihin. Lopussa vertaillaan linnuntie-etäisyyden eroa tieverkko-etäisyyteen käyttäen rakennettua reitityspalvelua.

ACM-luokat (ACM Computing Classification System, 1998 version): G.1.6, G.2.2, H.2.8, H.4.1, J.3

Avainsanat: Digiroad, Graafi, Verkko, TSP, Kotihoito, Aikataulutus, Reititys, Optimointi, Hilkka®

Esipuhe

Tämä tutkielma sai alkunsa FastROI Oy:n teknologiajohtaja Ilpo Kettusen kanssa käydyn keskustelun jälkeen ensin kandidaatintutkielman muodossa, josta jatkumo eteni tähän Pro gradu -tutkielmaan. Työn ohjaaja vaihtui alkumetreillä, mutta tahdon silti kiittää myös alkuperäistä ohjaajaani lehtori Simo Juvastetta hänen tuomistaan ideoista. Aihion luominen näin laajalle aihepiirille vaati oman aikansa, mutta lopulta pääsimme uuden ohjaajani professori Pasi Fräntin kanssa yhteisymmärrykseen. Tavoitteiden selkiytyessä varsinainen työosuus kesti kolme kuukautta. Aihio koottiin FastROI Oy:n aihepiirin käytännön kokemuksesta sekä tutkija Mikko Malisen tekemästä tutkimustyöstä. Edellämainittujen henkilöiden lisäksi haluan kiittää arjen tuesta puolisoani Sonja Nykästä ja Salsa Del Este ry:tä.

Sisältö

1	Johdanto	1
2	Työvuorojen aikataulutus ja optimointi	4
2.1	Yleinen aikataulutuksen ongelma	4
2.2	Kotihoidon aikataulutuksen ongelma	6
2.2.1	Ominaisuusmäärittelyt	7
2.2.2	Työvuorojen optimointi	10
2.2.3	Työtehtävien optimointi	11
2.2.4	Työtehtävien välinen etäisyys työtehtävien jaossa	15
3	Reititys algoritmit	19
3.1	Kahden pisteen välinen etäisyys	19
3.2	Verkon määritelmä	22
3.3	Lyhimpien polkujen selvittäminen	24
3.3.1	Dijkstran algoritmi	24
4	Työvuorojen ja reitityksen yhteisoptimointi	27
4.1	Kauppamatkustajan ongelma	27
4.2	Kauppamatkustajan ongelman sovitus	28
5	Reitityspalvelut	30
5.1	Kriteerejä reitityspalvelulle	30
5.2	Web-api -palveluntarjoajat	32
5.3	Oman reitityspalvelun rakentaminen	37
5.3.1	Materiaalille asetettavat vaatimukset	37
5.3.2	Maanmittauslaitoksen aineisto	38
5.3.3	Digiroad	39
5.4	Reitityspalvelun rakentaminen Digiroad K-aineistosta	41
6	Kokeelliset tulokset	46
6.1	Monen pisteen välinen etäisyys	46
6.2	Kahden pisteen välinen etäisyys	47
7	Yhteenveto	49
	Viitteet	50

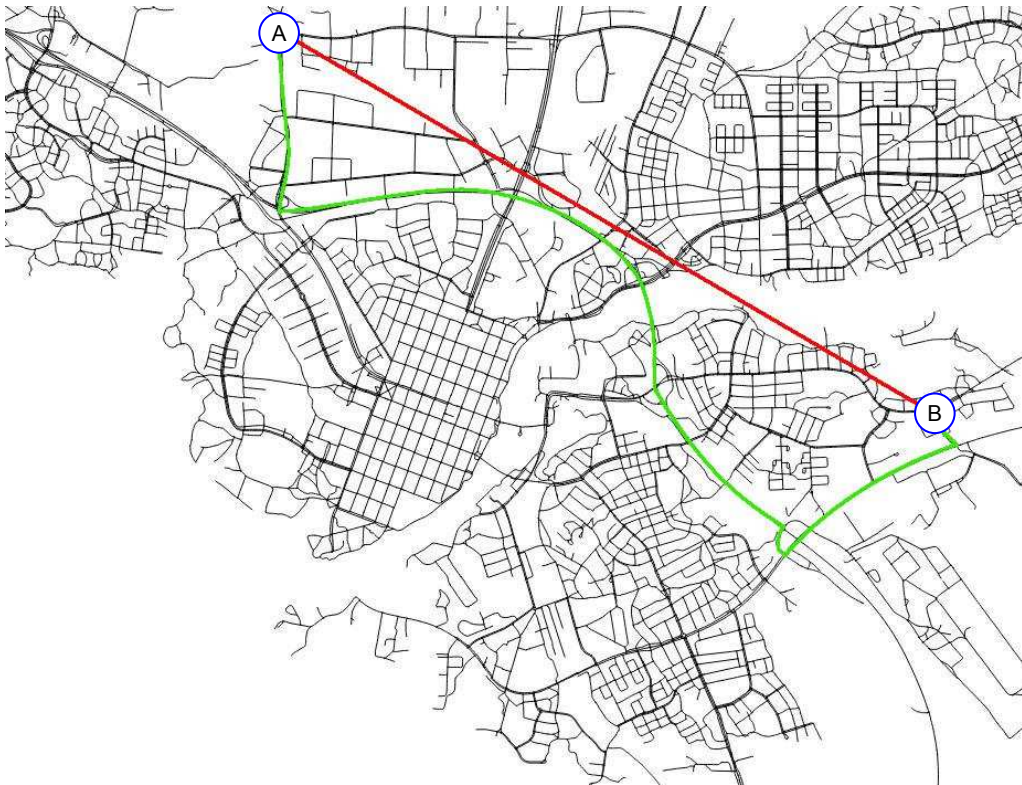
1 Johdanto

Tässä tutkielmassa käsitellään kotihoidon työvuoro- ja työtehtäväsuunnittelun automatisointia. Kotihoito käsittää sairaanhoidon, avustavan sairaanhoidon ja yleisen avustamisen tuomista potilaiden kotiin. Työvuorosuunnittelussa sovitetaan työvuoroja työntekijöiden kanssa tehtyjen sopimusten ja niihin liittyvien työehtosopimusten rajoissa siten, että töissä olisi aina tarvittava määrä henkilökuntaa. Työvuorosuunnittelussa jaetaan samalla päivän työtehtävät, joka myös osaltaan varmistaa, että tarvittava määrä henkilökuntaa on paikalla. Kaikki työtehtävät eivät kuitenkaan sovellu kaikille työntekijöille. Kotihoidon työvuorosuunnittelussa tulee myös ottaa huomioon työtehtävien vaatimustaso ja työntekijöiden osaamistaso. Koska kotihoidossa työtehtävät suoritetaan asiakkaiden kotona, täytyy työtehtävien jakamisessa ottaa huomioon työtehtävien välillä siirtymiseen kuluva aika.

Hilkka on FastROI Oy:n toiminnanohjausjärjestelmä hoitokoteihin ja kotihoitoon. Hilkassa automaattista työvuorosuunnittelua sekä työtehtävien jakoa kutsutaan optimoinniksi. Hilkassa pidetään kirjaa työntekijöistä, työvuoroista, potilaista ja potilaiden hoitotiedoista ja tehtävistä monien muiden asioiden ohella. Optimoinnin tarkoituksena on automatisoida työvuorojen suunnittelu sekä työtehtävien jako. Hilkassa määritellään millaisia työvuoroja hoitajat voivat tehdä työsopimuksensa nojalla sekä kirjataan asiakkaiden tarvitsemat hoitokäynnit, eli työtehtävät. Työvuorot suunnitellaan kolmen viikon jaksoissa, jolloin samalla jaetaan työtehtävät. Työtehtäviin on kirjattu ominaisuustiedot, jotka kertovat vaatimustason ja mahdolliset esteet. Näitä vastaavat tiedot merkitään työntekijöiden tietoihin, joka mahdollistaa työtehtävien jakamisen oikeille henkilöille.

Työtehtävien jakamisen lisäksi Hilkan optimointi tekee työntekijöille päivän työtehtävien suoritusjärjestyksen. Järjestelmän tarkoituksena on helpottaa töiden suunnittelua ja ohjausta, kuten myös varsinaista työntekoa. Suunnittelutyön siirtäminen pois ihmisiltä vähentää työtuntien tarvetta ja keventää työnjohtajien työkuormaa. Tutkielman alussa käsitellään aikataulutusta yleisesti, jonka jälkeen keskitytään kotihoidon työvuoro- ja työtehtäväsuunnittelun aikataulutuksen osa-ongelmaan. Tutkielmassa käytetään esimerkkinä Hilkka -järjestelmää.

Työtehtävien välillä siirtymiseen kuluvan ajan arvioiminen on toinen ongelma-alue, jota käsitellään tässä tutkielmassa. Ajan arvioimista varten tulee tietää reitin pituus työ-



Kuva 1: Linnuntie ja todellinen reitti Joensuun halki. (Digiroad 2010)

tehtävien välillä. Työntekijät voivat liikkua työtehtävien välimatkat autolla, pyörällä tai kävellen. Matka-aika riippuu käytettävästä liikennevälineestä, liikenteestä ja valitusta reitistä. Tarkoituksena on etsiä sellainen tehtäväjako, jossa matkustamiseen käytetään mahdollisimman vähän aikaa. Matka-aikaa voidaan arvioida esimerkiksi laskeamalla etäisyydet työtehtävien välillä joko linnuntienä tai tieverkostoa pitkin kulke-malla. Kuvassa 1 havainnollistetaan linnuntie-etäisyys kahden pisteen välillä Joensuun halki käyttämällä punaista viivaa, sekä nopein autolla kuljettava reitti tieverkkoa käyttämällä vihreää viivaa. Linnuntie antaa mahdollisuuden arvioida työtehtävien välillä siirtymiseen kuluva aikaa, mutta se ei kuitenkaan vastaa todellista kuljettavaa reittiä.

Työtehtäville on määritelty osoitetieto, jonka avulla reitti kahden työtehtävän välillä etsitään. Reitin etsimiseen kahden osoitetiedon väliltä tarvitaan tieverkkoa, joka koostuu risteyksien ja teiden sijainneista, niiden pituuksista, teiden nimistä ja talonumeroista, sekä vallitsevista liikennemääräyksistä. Näiden tietojen avulla tiestöstä voidaan muodostaa verkkoesitys, josta verkkoalgoritmeja käyttämällä voidaan etsiä lyhyimpiä reittejä matkallisesti. Mikäli halutaan arvioida matka-aikaa, tarvitaan myös tieosuuksien nopeusrajoitukset. Luvussa 3 käsitellään verkkojen teoriaa ja reittien etsimis-

tä verkossa. FastROI käyttää karttapohjana Suomen tiestötiedon sisältävää Digiroad -materiaalia, joka sisältää kaikki tarvittavat tiedot kohteiden välisten etäisyyksien ja siirtymään kuluvan ajan arvioimiseen tieverkkoa pitkin.

Jotta Digiroadia tai muuta vastaavaa materiaalia voitaisiin käyttää reittien hakemiseen, täytyy sen sisältämä tieto muokata verkkomuotoon. Tieverkon risteykset muunnetaan verkon solmuiksi ja niitä yhdistävät tieosuudet kaariksi. Luvussa 3.2 käydään läpi verkkojen teoriaa ja haetaan tarkoitukseen parhaiten sopivaa algoritmia. Verkkojen teorian ohella tutustutaan algoritmeihin, joilla etsitään lyhimpiä polkuja. Esimerkkinä tällaisesta algoritmista on Dijkstran algoritmi (Cormen & al 2005). Verkkototeutuksen tarkastelussa käsitellään myös kuinka liikennejärjestelmän ominaisuudet otetaan huomioon verkkototeutuksessa. Verkkojen käsittelyyn ja tallentamiseen tarvitaan myös omat tietokantarakenteet.

Kauppamatkustajan ongelmassa käydään kotihoidon optimoinnin tapaan kaikissa kohteissa mahdollisimman tehokkaasti. Kauppamatkustajan ongelmasta on olemassa useita variantteja joiden soveltamismahdollisuutta kotihoidon optimointiin tarkastellaan lyhyesti luvussa 4.

Luvussa 5 käsitellään Digiroadin ominaisuuksia ja soveltuvuutta optimointiin. Kuva 1 on luotu Digiroad materiaalin avulla. Tässä tutkielmassa käsitellään myös vaihtoehtoja, kuten web-api -palveluita, jotka tarjoavat valmiin rajapinnan nopeimpien ja lyhimpien reittien hakuun eri liikennevälineille. Luvussa 5 vertaillaan myös reitinhakupalvelun rakentamisen ja valmiin web-api -palvelun käyttämisen välisiä eroja ja kustannuksia. Lisäksi vertaillaan kolmannen osapuolen reitityspalveluita, materiaaliveaihtoehtoja reititykseen, sekä reitityspalvelun rakentamista Digiroad -materiaalista.

Luvussa 6 vertaillaan linnuntie-etäisyyttä tieverkkoa pitkin kuljettuun matkaan. Vertailua tehtiin Googlen palveluiden avulla lasketuilla etäisyyksillä, sekä Digiroad -materiaalista rakennetun reitityspalvelun avulla lasketuilla etäisyyksillä. Vertailussa käytettiin kotihoidon käyntitietojen lisäksi Mopsi -palvelun valokuvien sijainteja.

Lopuksi luvussa 7 esitetään yhteenveto, jossa käydään läpi tutkielmaa tehdessä havaitut asiat.

2 Työvuorojen aikataulutus ja optimointi

Tässä luvussa tutustutaan aikataulutuksen ongelmaan työvuorojen ja työtehtävien kannalta. Lisäksi esitellään kotihoidon aikataulutuksen ongelma esimerkkijärjestelmän avulla.

2.1 Yleinen aikataulutuksen ongelma

Aikataulutusta tehdään monilla toimialoilla kuten esimerkiksi hoiva- ja kuljetusalalla. Eri toimialueiden aikataulutussmalleissa on yhteneviä piirteitä. Aikataulutuksen perusidea on sovittaa työtehtävät ja työntekijät yhteen. Yleensä työtehtävien määrä, vaatimustaso ja suoritusajankohta ovat vaihtelevia. Aikataulutuksen avulla varmistetaan, että tarvittava määrä työntekijöitä on suorittamassa työtehtäviä ja kaikki työtehtävät saadaan suoritettua. Tavoitteena on saavuttaa tarvittava palvelutaso mahdollisimman pienillä työvoimaresursseilla. Työtehtävät pyritään jakamaan tehokkaasti työntekijöiden kesken, jotta henkilöstökulut pysyisivät mahdollisimman pieninä.

Aikataulutus voidaan jakaa työvuorojen ja työtehtävien aikataulutukseen. Aikataulutuksessa tulee ottaa huomioon lakisääteiset asiat, kuten oikeus saada tietää työvuorot etukäteen seuraavaksi kolmeksi viikoksi, ylityörajoitukset ja -korvaukset, sekä loma- ja lepoajat. Lakisääteisten asioiden lisäksi täytyy tarkastella vuorojen työtehtäviä. Kaikki työntekijät eivät osaa tai voi tehdä kaikkia työtehtäviä ja kokeneemmat työntekijät eivät välttämättä ehdi tekemään kaikkia tarjolla olevia vaativampia työtehtäviä.

Mikäli kaikki työntekijät eivät voi suorittaa kaikkia työtehtäviä, täytyy aikataulutuksessa työtehtävien määrän lisäksi ottaa huomioon laatu. Työvuorojakoa suunnitellessa täytyy määritellä työtehtävien vaatima osaamistaso sekä työntekijöiden osaamiset. Osaamismäärittelyn avulla varmistetaan työntekijän ja työtehtävän yhteensopivuus. Hoiva-alalla tällaisia määrittelyitä ovat esimerkiksi insuliinipistoksen antaminen, mikä vaatii sairaanhoitajan pätevyyden. Poikkeuksia aikataulutuksen luomiseen tuovat myös työntekijöiden poissaolot sekä työvuorojen pituusrajoitteet. Toimialakohtaisesti voi olla tarpeellista lukita tietyt työtehtävät tietyille henkilöille. Esimerkiksi hoiva-alalla potilaan tehtävien lukitseminen samalle työntekijälle (omahoitaja) parantaa potilasviihtyvyyttä. Aikataulutuksen lopputuotteena on työvuorolista ja mikäli työtehtävät jaetaan, niin samalla saadaan myös työntekijäkohtainen työtehtävälista, jossa on päivän työtehtävät

samaan tapaan kuin lukujärjestyksessä.

Osaamismäärittelyt työtehtävissä voidaan jakaa pakollisiin ja suositeltaviin. Pakollinen määrittely on aina sellainen joka työntekijällä tulee olla, jotta työ voidaan kohdistaa hänelle. Suositeltavat osaamiset ovat sellaisia jotka työntekijän olisi hyvä osata, mutta eivät kuitenkaan estä työtehtävän kohdistamista. Suositeltavat ominaisuudet tuovat tai vähentävät kustannuksia työtehtävien jaossa. Suositeltavan ominaisuuden rikkominen tuo lisäkustannuksen, joka voi olla esimerkiksi ylimääräinen aika, joka puutteellisen osaamisen omaavalla työntekijällä kuluu tehtävän suorittamiseen. Työtehtävien jaossa pyritään päättämään mikä kustannus ominaisuusmäärittelyn rikkomisesta tulee, jotta voidaan arvioida onko järkevämpää rikkoa määrittelyä vai aikatauluttaa taidon omaava työntekijä muualta, mikäli mahdollista. (Ernst & al 2004)

Castillo-Salazar & al (2012) kokosivat tutkimuksessaan aikataulutuksessa yleisesti huomioon otettavia asioita seuraavasti:

- *Aikaikkunat:* Työtehtäville määritellään aikaikkunat, joiden puitteissa työtehtävä tulee aloittaa. Aikaikkunat voivat olla luonteeltaan joustavia tai tiukkoja riippuen työtehtävän laadusta ja sovellettavasta toimialasta.
- *Kulkuneuvon käyttäminen:* Työntekijät voivat toimialasta ja työhön vaadittavista varusteista riippuen käyttää työtehtävän luo siirtymiseen eri kulkuneuvoja kuten autoa, polkupyörää, julkista liikennettä tai liikkuu jalkaisin.
- *Työpäivän aloitus ja lopetuspaikat:* Päivän aloitus ja lopetus voi tapahtua työntekijän kotiosoitteesta tai yhteisestä lähtöpisteestä kuten toimistolta. Jotkin toimialat voivat vaatia esimerkiksi tarvikkeiden tai työvälineiden noutoa ennen töiden aloittamista.
- *Osaamis- ja vaatimusmäärittelyt:* Mikäli toimialan työt ovat sellaisia, joita kaikki työntekijät eivät ole päteviä suorittamaan, laaditaan työntekijöille ja työtehtäville toisiaan vastaavat osaamis- ja vaatimusmäärittelyt. Määrittelyiden avulla varmistetaan työtehtävien kohdistuminen osaavalle henkilökunnalle.
- *Työn kesto:* Arvioitu aika joka kuluu työtehtävän suorittamiseen. Arviointi on hankalaa, sillä työtahti riippuu työntekijästä. Yleisesti työn kesto pyritään arvioimaan siten, että kaikki työntekijät ehtivät suorittaa työtehtävän helposti.

- *Liitetyt työtehtävät:* Työtehtävät voivat liittyä toisiinsa suoritusjärjestyksen tai yhtäaikaisuuden muodossa. Esimerkiksi rakennustöissä voidaan aikatauluttaa niin, että tarvikkeiden noutotehtävä, kuljetus ja töiden aloittaminen tulee samalle työntekijälle.
- *Ryhmätyö:* Työtehtävät voivat vaatia yhden tai useamman työntekijän suorittamaan määriteltyä tehtävää. Tällöin useita työntekijöitä voi aikatauluttaa yhtenä ryhmänä. Ryhmien osaamismäärittelyt täytyy tehdä huolella, jotta tarvittavan pätevyyden omaavat työntekijät ovat oikeissa ryhmissä.
- *Klusterointi:* Joillakin toimialoilla työntekijät jaetaan alueisiin, jonka sisällä olevia työtehtäviä työntekijät suorittavat. Esimerkiksi tienhoidossa toimitaan alueittain. Tällöin samassa kunnassa asuvat työntekijät hoitavat lähialueidensa tiestön kunnossapidon.

Työvuorojen aikataulutusongelma on helppoissakin tapauksissa NP-kova. Ongelmaa on tutkittu paljon ja erilaisia lähestymistapoja on useita. Ratkaisutapoihin kuuluu esimerkiksi *parviälykyys* (swarm intelligence), *sirontahaku* (scatter search), *hyperheuristiikat* (hyperheuristics), *kokonaislukuohjelmointi* (integer programming), *metaheuristiikat* (metaheuristics), *simuloitu jäähdytys* (simulated annealing), *tabuhaku* (tabu search) ja *naapurustohaku* (variable neighborhood search). (Kyngäs 2011)

2.2 Kotihoidon aikataulutuksen ongelma

Kotihoidossa työntekijät ovat *kotiavustajia*, *sairaanhoitajia*, *lääkäreitä* tai *lähihoitajia*. Työtehtävät ovat erilaisia yleisiä avustus- tai terveydenhuollon tehtäviä, joiden tarkoituksena on tukea vanhusten ja vammaisten kotiasumista. Työntekijät suorittavat näitä tehtäviä lukuisten potilaiden luona, jotka asuvat eri puolilla kaupunkia. Ongelmaan on useita ratkaisutapoja, mutta kaikissa sovellusalue otetaan huomioon samaan tapaan (Misir & al 2010). Kotihoidon aikataulutuksessa työtehtävät on määritelty ennalta ja siinä pyritään hakemaan päteviä työntekijöitä suorittamaan näitä työtehtäviä. Työntekijöiden sopivuus työtehtäviin varmistetaan määrittelemällä työtehtäville niiden suorittamiseen vaaditut välttämättömät taidot, sekä määrittelemällä työntekijöille samaan tapaan heidän osaamansa taidot.

Seuraavaksi käsitellään kotihoidon työvuorojen aikataulutusta käyttäen esimerkkinä

FastROI Oy:n Hilkka-järjestelmää, sekä vertaamalla tätä kirjallisuudessa esille tulleisiin järjestelmiin. Hilkka pyrkii tehostamaan kotihoiton työnohjausta ja parantamaan kotihoitopalvelun laatua. Se sisältää optimointialgoritmin, jonka avulla työnohjausta tehdään kahdessa vaiheessa. Ensin suunnitellaan työntekijöiden työvuorolistat kolmeksi viikoksi eteenpäin, minkä jälkeen optimoidaan työntekijöiden työtehtävät päiväkohtaisesti. Näin ratkaistaan kaksi tehtävää, jotka työnjohtajat ovat aiemmin suorittaneet manuaalisesti itse.

Hoitajat kirjaavat työtehtävänsä Hilkka-mobiilisovelluksella, josta he tarkistavat myös työtehtävien suoritusjärjestyksen. Hoitajat voivat käyttää puhelimen navigaattoria tai erillistä navigaattoria seuraavan kohteen löytämiseen. Mobiilisovelluksen avulla on tulevaisuudessa mahdollista avata puhelimen oma navigointisovellus, johon seuraavan työn kohde asetetaan automaattisesti.

2.2.1 Ominaisuusmäärittelyt

Hilkka-optimoinnissa on tehty osaamismäärittely hoitajille samalla periaatteella kuin Misir & al (2010) tutkimuksessa, mutta työtehtävien ominaisuusmäärittely on tätä laajempi. Työntekijöille, asiakkaille ja työtehtäville laaditun määrittelyn avulla työtehtävät kohdistetaan työntekijöille. Asiakkaan ominaisuusmäärittelyihin kuuluu osaamismäärittelyt ja vaatimukset, jotka ovat voimassa kaikissa asiakkaan työtehtävissä. Yksittäisiin töihin voidaan myös lisätä osaamisvaatimuksia työtehtäväkohtaisesti.

Ominaisuusmäärittelyt hoitajille:

- *Kulkuneuvo*: Kotihoitajat liikkuvat autolla, polkupyörällä tai kävellen. Kullekin hoitajalle määritellään kulkuneuvo, jonka mukaan matka-aikaa arvioidaan.
- *Aloit- ja lopetuspaikka*: Kotihoitajat voivat aloittaa samasta keskuksesta tai vaihtoehtoisesti kukin omasta kodistaan. Töiden lopetus voi tapauskohtaisesti päättyä keskukseen tai hoitajan omaan kotiin. Yleisesti töihin kuitenkin lähdetään toimistolta.
- *Työaika*: Hoitajan päivittäinen ja viikoittainen työsopimuksessa määriteltä työaika. Lisäksi määritellään prosenttimäärä mikä työajasta on käytettävissä tehtävien hoitamiseen.

- *Taitotaso:* Hoitajan omat työskentelytaidot. Taitotasomäärittelyt ovat identtisiä työtehtävillä ja asiakkaiden vaatiman taitotason kanssa. Hilkassa oletuksena olevat taitotasomäärittelyt esitellään tekstissä myöhemmin.
- *Rajoitteet:* Hoitajan työskentelyn estävät tekijät, kuten allergisuus asiakkaan lemmikille. Rajoitemäärittelyt ovat identtisiä työtehtäville ja asiakkaille määriteltyjen rajoitteiden kanssa.
- *Estot:* Hoitajalle voidaan asettaa estoja asiakkaille, joiden työtehtäviä ei voida kohdistaa kyseiselle hoitajalle. Esto voidaan asettaa joko asiakkaan tai työntekijän toimesta.

Ominaisuusmäärittelyt asiakkaille:

- *Omahoitaja:* Asiakkaalle voidaan asettaa yksi tai useampi omahoitaja, joka valitaan suorittamaan asiakkaan tehtäviä, mikäli mahdollista.
- *Rajoitteet:* Vastaavasti kuten hoitajilla. Asiakkaalla voi esimerkiksi olla koira tai asiakkaan tehtävät vaativat nostamista.
- *Estot:* Vastaavasti kuten hoitajilla.

Ominaisuusmäärittelyt työtehtäville:

- *Lukitut tehtävät / Omahoitajuus:* Työtehtäviä voidaan lukita halutuille hoitajille. Tällöin työtehtävä kohdistuu automaattisesti tälle hoitajalle, mikäli hänellä on työvuoro tehtävän aikaan.
- *Vaadittava taitotaso:* Kullekin tehtävälle määritellään vaadittava taitotaso, jotka vastaavat hoitajille määriteltyjä osaamisia ja joita voidaan tarvittaessa lisätä tai poistaa. Osaamismäärittelyt ovat identtisiä työtehtäville ja asiakkaille määriteltyjen vaadittavien taitotasojen kanssa.
- *Aikaikkuna:* Tehtävän suoritus täytyy aloittaa aikavälin sisällä. Aikaikkuna voi esimerkiksi alkaa kello 9:00 ja sen pituus olla kaksi tuntia. Aikaikkuna voi olla myös tiukka, jolloin työ on aloitettava tasan kello 9:00.
- *Arvioitu kesto:* Arvioitu kesto tehtävän suorittamiseen. Tehtävän kesto voi olla esimerkiksi 15 minuuttia.

- *Sukupuoli:* Tehtävän suorittajan täytyy olla valittua sukupuolta.
- *Sijainti:* Osoite missä suoritettava työtehtävä sijaitsee.
- *Rajoitteet:* Hoitajan työskentelyn estävät tekijät kuten allergiat.

Hilkan ominaisuusmäärittelyssä oletuksena käytössä olevat taitotasomäärittelyt:

Kielitaidot:

- Eestin kieli
- Englannin kieli
- Ruotsin kieli
- Venäjän kieli

Yleiset tehtävät:

- Kylvetys (KY)
- Lääkkeiden jako (LÄ)

Sairaanhoidolliset työt:

- Avanteen hoito
- Erikoisosaamista vaativa pistos (Pendysin, Trexan)
- Haavahoito
- Intramuskulaarinen pisto
- Insuliinin pisto
- Katetrointi
- Laskimoverinäytteiden otto
- MMSE-testi
- N-Lääkkeet
- Pika INR:n ottaminen
- Rokotukset (RO)
- Subkutaaninen pisto (Klexane)
- Sairaanhoidolliset työt
- Verenpaine
- Verensokerin mittaus

Ominaisuusmäärittelyssä hoitajien edellämainittu taitomäärittely kattaa erilaisia lääke- ja sairaanhoitoon liittyviä asioita, kuten rokotukset ja verenpaineen mittaamisen, sekä hoitajan kielitaidon. Optimointi ei tarjoa hoitajille tehtäviä, joiden suorittamiseen heillä ei ole pätevyyttä. Uusia taitoja voidaan tehdä tarvittaessa lisää ja vanhoja poistaa käytöstä, mikäli ne eivät ole enää tarpeellisia. Puutteet pätevyydessä eivät kuitenkaan estä tehtävän kohdistamista manuaalisesti. Työnjohtaja voi siis antaa jonkun tehtävän hoitajalle, jota optimointi ei ole voinut taitopuutteiden takia kohdistaa. Ominaisuusmäärittelyn käytöstä kerrotaan lisää luvussa 2.2.3.

2.2.2 Työvuorojen optimointi

Työvuorojaossa työntekijöille päätetään päiväkohtaiset työvuorot ja niiden kesto (aamu, iltä ja viikonloppu) kolmeksi seuraavaksi viikoksi. Työvuorojen jako suoritetaan työvoimatarpeen mukaan, jolloin optimoidaan työntekijöistä syntyviä kustannuksia. Työvuorot jaetaan ensisijaisesti niille työntekijöille, joille sopimuksen mukaan on luvattu töitä. Toissijaisena on työntekijät, joiden tuntimäärää voidaan säädellä. Sopimukset määrittävät lepo- ja työaikojen suuruudet, joiden rajoissa työvuorojen kesto voidaan muokata. Osa sopimuksista on tuntimäärien suhteen joustavampia kuin toiset ja näitä työntekijöitä käytetään vain tarvittaessa. Toisinaan voi olla kustannustehokkempaa teettää työntekijöillä ylitöitä uuden työntekijän palkkaamisen sijaan. (Ernst & al 2004)

Jakso		24.06.2013 - 14.07.2013													
Viikot	Päivät	26							27						
		24.06	25.06	26.06	27.06	28.06	29.06	30.06	01.07	02.07	03.07	04.07	05.07	06.07	07.07
		MA	TI	KE	TO	PE	LA	SU	MA	TI	KE	TO	PE	LA	SU
1	... Kaisa	07:30 15:30	07:30 15:30	07:30 15:30	07:30 15:30	07:30 15:30	VP	VP	07:30 15:30	07:30 15:30	07:30 15:30	15:00 21:30	15:00 21:30	VP	VP
	Tavoite: 114:45														
	Yhteensä: 117:00	08,00	08,00	08,00	08,00	08,00			08,00	08,00	08,00	06,30	06,30		
2	... Krista	07:30 15:30	07:30 15:30	07:30 15:30	07:30 15:30	07:30 15:30	VP	VP	07:30 15:30	07:30 15:30	07:30 15:30	15:00 21:30	15:00 21:30	VP	VP
	Tavoite: 114:45														
	Yhteensä: 112:30	08,00	08,00	08,00	08,00	08,00			08,00	08,00	08,00	06,30	06,30		
3	... Maija	07:30 15:30	07:30 15:30	07:30 15:30	07:30 15:30	07:30 15:30	VP	VP	15:00 21:30	15:00 21:30	15:00 21:30	07:30 15:30	07:30 15:30	VP	VP
	Tavoite: 90:00														
	Yhteensä: 115:30	08,00	08,00	08,00	08,00	08,00			06,30	06,30	06,30	08,00	08,00		

Kuva 2: Kahden viikon työvuorolistasta

Kuvassa 2 on kahden viikon otos Hilikka -järjestelmän avulla luodusta työvuorolistasta,

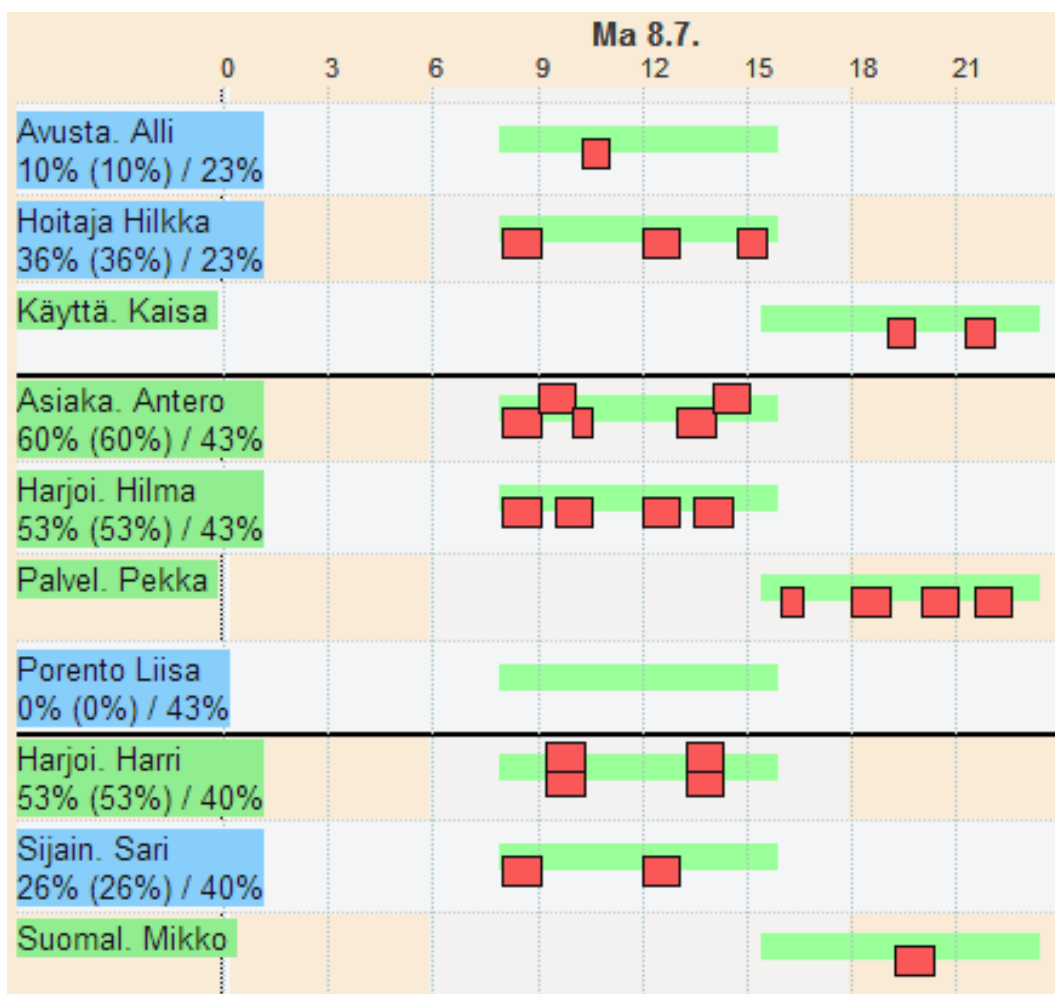
jossa näkyvät kolmen työntekijän työvuorot. Listassa näkyy päiväkohtaiset työvuoron aloitus- ja lopetusajankohta sekä vapaapäivät. Värikoodit määräytyvät työvuoron kes-
ton mukaan ja ovat muokattavissa halutuiksi. Kuvassa vihreällä olevat työajat ovat päiv-
vävuoroja ja oranssilla olevat kellonajat ovat iltavuoroja. Sinisellä merkitty VP tarkoittaa
vapaapäivää. Henkilön rivillä näkyy alhaalla harmaassa laatikossa kunkin työpäi-
vän kokonaistyöaika. Listassa pyhäpäivät on korostettu punaisella (myös arkipyhät) ja
lauantai korostettu persikan värisenä. Työntekijän nimen alla on jaksokohtainen tavoit-
tetuntimäärä, joka määräytyy sopimuksesta. Tavoitetuntimäärän alapuolella on ilmoit-
tettu todellinen jaksolle aikataulutettu tuntimäärä. Kaisalle ja Maijalle on aikataulutettu
normaalia enemmän tunteja ja Kristalle normaalia vähemmän.

Työtehtävien määrä ja laatu määrittävät työvoiman tarpeen, joten työvuoroja jaettaessa
täytyy tarkastella myös työtehtäviä. Vuorojaossa työtehtävien, asiakkaiden ja hoitajien
yhteensopivuus määritellään luvussa 2.2 esiteltyjen ominaisuuksien avulla. Työtehtä-
vät ovat todellisuudessa tiedossa kahdeksi viikoksi kerrallaan, mutta työntekijöillä on
lakisääteinen oikeus kolmen viikon työvuorolistaan. Lain vuoksi työntekijöille arvioi-
daan kahden ensimmäisen viikon tehtävien perusteella myös kolmannen viikon työ-
vuorot. Työtehtävien määrän ja työehtosopimuksen sääntöjen pohjalta haetaan hyvää
ratkaisua siten, että kaikille työtehtäville saadaan määritettyä työntekijä. Työvuorojen
optimointi on sidoksissa työtehtävien optimointiin, josta kerrotaan tarkemmin seuraa-
vassa luvussa.

2.2.3 Työtehtävien optimointi

Työtehtäviä optimoidaan päiväkohtaisesti, jolloin jaetaan päivän työtehtävät ja mää-
ritellään niitä suorittavat työntekijät. Työtehtäväjaon jälkeen yhden työntekijän tehtä-
vien suoritusjärjestys nähdään hänelle määrättyjen tehtävien suoritusajoista. Luvun 2.2
ominaisuusmäärittelyä hyödynnetään yhdistäessä sopivia tehtäviä työntekijöille. Työ-
tehtävien jaossa voidaan optimoida eri asioita, kuten työntekijöiden kulkemaa matkaa
tai siihen kuluvaan aikaan, sekä työntekijöiden määrää. Työvuorolista voi kuitenkin olla
myös ennalta määritelty, jonka perusteella työtehtäviä jaetaan mahdollisimman tehok-
kaasti käytössä olevalle henkilökunnalle.

Kuvassa 3 on otos päivän jaetuista työtehtävistä Hilkka -järjestelmästä. Vihreä suora-
kaide työntekijän nimen oikealla puolella näyttää päivän kokonaistyöajan ja punaiset
suorakaiteet ovat työntekijälle kohdistettuja ulkoisia tehtäviä. Punaisten suorakaitei-



Kuva 3: Työtehtävälista

den pituus kertoo samalla työtehtävän keston. Nimien taustalla oleva pohjaväri kertoo kuinka paljon työntekijälle on kohdistettu työtehtäviä.

Kohdistettujen työtehtävien suoritusaikaa suhteessa kokonaistyöaikaan kutsutaan *kuormitukseksi*. Kuormituksen määrä näkyy nimen alla prosenttilukuna sulkujen sisällä. Väri on sininen mikäli kuormitus on alle 50% työajasta, vihreä mikäli on kuormitus 50-60% ja punainen mikäli kuormitus on yli 60%. Sininen tarkoittaa kevyttä kuormitusta, vihreä normaalia ja punainen liiallista kuormitusta. Työntekijälle määritetään myös oma henkilökohtainen maksimikuormitusprosentti, joka otetaan optimoinnissa huomioon. Ensimmäisenä oleva prosenttiluku kertoo ulkoisten asiakaskäyntien tuoman kuormituksen. Mikäli työntekijälle olisi kohdistettu sisäisiä työtehtäviä, kuten toimistotöitä, näkyisivät ne keltaisena samaan tapaan kuin ulkoiset työtehtävät kokonaistyöaika-suorakaiteen päällä. Sisäiset työtehtävät myös kasvattavat kokonais-

kuormituksen määrää. Työntekijät on jaettu tiimeihin ja tiimejä kuvassa erottaa musta vaakaviiva. Oikeanpuoleisin prosenttiluku kertoo tiimin kokonaiskuormituksen.

Kaikkia tehtäviä ei kuitenkaan pystytä aina jakamaan. Työntekijöitä voi olla liian vähän tai työntekijälle määritellyt ominaisuudet estävät tehtävän antamisen työntekijälle. Osaa määrittelyistä voidaan tällöin painottaa niiden tärkeyden perusteella. Tällöin ominaisuudet jaetaan *koviin* ja *pehmeisiin*. Kovat ominaisuudet ovat pakollisia ja niitä ei voida rikkoa missään tapauksessa. Pehmeät ominaisuudet ovat rikottavissa, mutta vain mikäli parempaa ratkaisua ei löydetä. Mikäli tästäkään huolimatta kaikkia tehtäviä ei saada kohdistettua, työnjohtaja valitsee käytössä olevista työntekijöistä sopivimman.

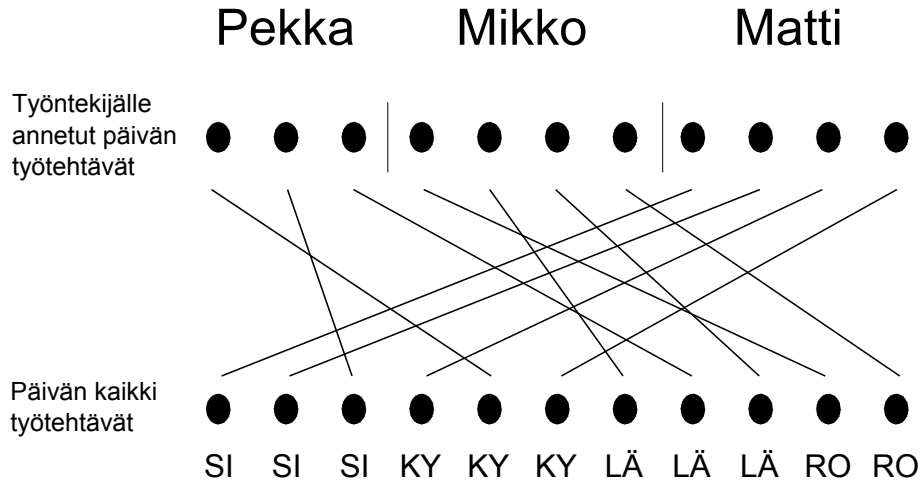
Optimoinnissa painotetaan eri taitoja niiden kriittisyyden perusteella. Lääketieteelliset taidot ovat kovia rajoitteita, mutta esimerkiksi yhteisen kielen osaaminen ei välttämättä ole poissulkeva tekijä aamupalan valmistamiselle. Hilikka -optimoinnissa pehmeänä rajoitteena on omahoitajuus. Työnjohtaja voi tällöin tapauskohtaisesti kohdistaa ominaisuusmäärittelyissä täsmäämättömän työtehtävän jollekin vapaalle työntekijälle, joka hänen mielestään pystyy suorittamaan tehtävän ristiriidasta huolimatta.

Optimoinnissa työntekijälle määritellään lisäksi hänen käyttämä kulkuneuvo, joka voi olla auto, polkupyörä tai kävely. Kulkuneuvomäärittelyn avulla arvioidaan, kuinka kauan työntekijällä kestää siirtyä kohteesta toiseen. Eri kulkuneuvoille määritellään keskimääräinen nopeus, jota ne kulkevat siirtyessään paikasta toiseen. Autolla tämä nopeus on 40 km/h, kävellen 4 km/h ja polkupyörällä 10 km/h. Näin arvioidaan kahden työtehtävän välillä siirtymiseen kuluva aika.

Hoitajia ei pakoteta menemään laskennassa käytettyä reittiä, vaan heille annetaan ainoastaan tehtävien suoritusjärjestys. Hoitajat päättävät loppukädessä itse mitä reittiä he haluavat käyttää kohteesta toiseen siirtymiseen. Kun hoitajat voivat päättää reitin itse, voivat he hyödyntää paikallistuntemustaan sopivimman reitin valitsemiseksi.

Järjestelmässä on työtehtäviä, jotka jaetaan työntekijöille tasaisesti edellämainitut määrittelyt huomioiden. Kuvassa 4 on havainnollistettuna 11 työtehtävän jakaminen tasan kolmen työntekijän kesken. Kuvan esimerkissä jokaiselle työntekijälle tulisi kohdistua maksimissaan neljä työtehtävää, jotta jako olisi tasainen. Yhtäläisen tehtävämäärän lisäksi on katsottava myös jaettujen tehtävien tuomaa kuormitusta.

Määrittelyt huomioidaan työtehtävien jaossa laskemalla työtehtävän suorittamises-



Kuva 4: Työtehtävien ja työntekijöiden paritus

ta tuleva *kustannus* jokaiselle työntekijälle. Seuraavassa tarkastellaan Malinen & al (2013) ratkaisua pienen esimerkkitapauksen kohdalla. Kunkin määritellyn ominaisuuden rikkomiselle määritellään kustannus, jota käytetään laskennassa. Kuvassa 5 on työtehtävien kustannuksista muodostuva matriisi. Mikäli mitään ominaisuusmäärittelyä ei rikota, kustannus on nolla. Jos työntekijän ja työtehtävän välillä on kova ominaisuus joka ei täyty, tulee työtehtävän kustannuksesta tällöin ääretön. Pekka ja Matti eivät osaa rokottaa, joten heidän kohdallaan rokotustehtävän kustannukseksi on merkitty ∞ . Matti ei myöskään saa jakaa lääkkeitä, joten lääkejakotehtävien kustannus hänelle on ∞ .

Pehmeän ominaisuuden kuten omahoitajuuden rikkominen tuo tehtävälle kustannuksen, joka on viisi yksikköä. Matti on kahden ensimmäisen ja Pekka kolmannen siivoustehtävän asiakkaan omahoitaja, joten näiden tehtävien kohdalla kustannus on nolla. Mikolla kylvetys ja siivoustehtävät yhtä lukuunottamatta rikkovat omahoitajuutta, jonka vuoksi niiden kustannukset hänelle ovat viisi. Lisäksi hänellä on koira-allergia, joka nostaa toisen siivoustehtävän kustannuksen kymmeneen. Kun jokaisen työtehtävän ja työntekijän välille on laskettu kustannus, pyritään löytämään työnjako, jonka kokonaiskustannus on mahdollisimman pieni. Tässä esimerkissä rokotustehtävä menee Mikolle, kylvetys Matille ja siivous Pekalle, jolloin kokonaiskustannukseksi jää 0.

Kustannusfunktio ominaisuusmäärittelyn rikkomisesta on määritelty seuraavasti,

$$R = \sum_{i=1}^n X_i \quad (1)$$

missä R on ominaisuusmäärittelyn rikkomisista tulevien kustannusten summa, n on tehtävien määrä ja X_i on työntekijälle kohdistetun tehtävän tuoma kustannus.

	Siivous		Kylvetys		Lääkejako		Rokotus			
	Siivous	Siivous	Kylvetys	Kylvetys	Lääkejako	Lääkejako	Rokotus	Rokotus		
Pekka	5	5	0	5	0	5	0	5	∞	∞
Pekka	5	5	0	5	0	0	5	0	5	∞
Pekka	5	5	0	5	0	0	5	0	5	∞
Mikko	5	10	5	5	5	0	0	5	0	0
Mikko	5	10	5	5	5	0	0	5	0	0
Mikko	5	10	5	5	5	0	0	5	0	0
Mikko	5	10	5	5	5	0	0	5	0	0
Matti	0	0	5	0	5	0	∞	∞	∞	∞
Matti	0	0	5	0	5	0	∞	∞	∞	∞
Matti	0	0	5	0	5	0	∞	∞	∞	∞
Matti	0	0	5	0	5	0	∞	∞	∞	∞

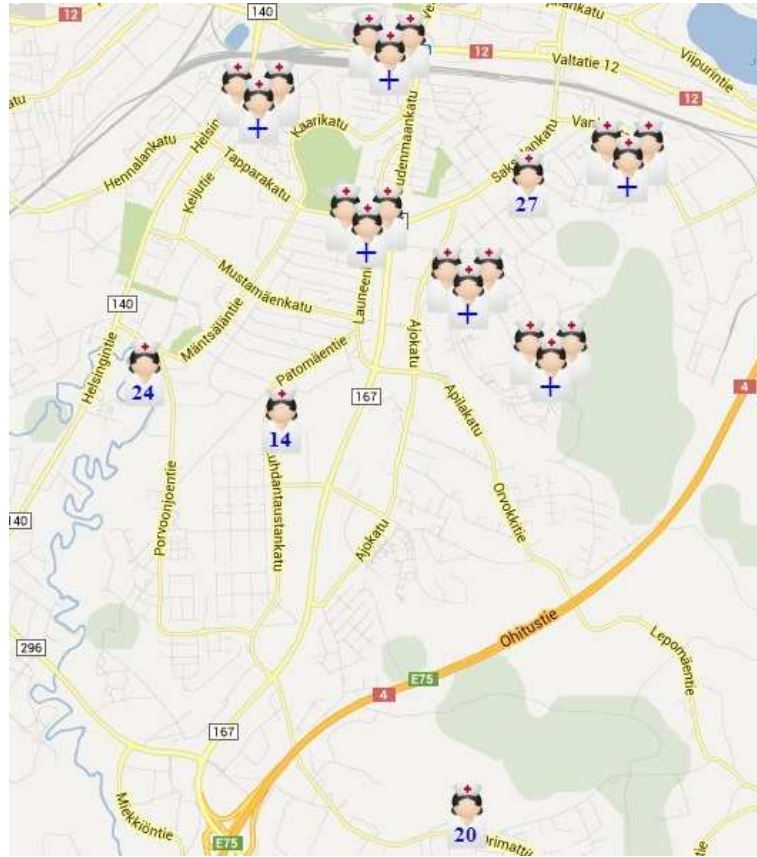
Kuva 5: Matriisiesitys työtehtävien jakamisesta muodostuvista kustannuksista

Työtehtävien jaossa on kyse *paritusongelmasta* (assignment problem). Työtehtävien paritus voidaan tehdä optimaalisesti *unkarilaisen algoritmin* avulla (Kuhn 1955). Tällöin työvuorot on päätetty ennakkoon ja työtehtäviä paritetaan työvuorossa oleville työntekijöille. Kustannuksena toimivat työtehtäville ja työntekijöille määritellyt ominaisuudet. Työvuororajoja on mahdollista rikkoa, mutta se tuo ennalta määrättyyn kustannuksen paritukseen.

2.2.4 Työtehtävien välinen etäisyys työtehtävien jaossa

Kotihoidon työtehtävien jaossa tulee ottaa huomioon työtehtävien välinen etäisyys ja sen mukana työtehtävien välillä siirtymiseen kuluva aika. Mikäli etäisyydet ovat pieniä ja niiden välillä siirtymiseen kuluva aika on muutamia minuutteja, niillä ei ole merkitystä. Työntekijän päivän aikana kulkema matka voi kuitenkin olla jopa kymmenen kilometriä, mikä jalkaisin kuljettuna kuluttaa työaikaakin noin kaksi tuntia. Kuvassa 6 on Lahden kotihoidon kaikkien hoitajien keskimääräiset sijainnit yhdelle päivälle. Kauimmaisten tehtävien välinen etäisyys linnuntietä pitkin on noin viisi kilometriä. Hoitajankuva kertoo yhden työntekijän työtehtävien keskimääräisen sijainnin. Symbo-

li, jossa on monta hoitajaa ja plus-merkki, tarkoittaa että kyseisellä alueella on usean työntekijän työtehtäviä. Tässä esimerkissä kullekin hoitajalle on kohdistettu keskimäärin seitsemän tehtävää ja kunkin työntekijän matka-aika on minimoitu.



Kuva 6: Työntekijöiden keskimääräinen sijoittuminen kartalle

Malinen (2013) lähestyy ongelmaa minimoimalla työntekijälle jaettujen tehtävien parittaisien matka-aikojen neliötä. Parittaiselle matka-ajan neliölle Malinen (2013) käyttää termiä *polkukustannus*. Työntekijälle jaetut tehtävät muodostavat *klusterin*, eli joukon tehtäviä. Parittaiset neliöetäisyydet ovat suoraan verrannollisia *kokonaisneliövirheeseen* (total squared error), kun klusterit ovat tasapainoitettuja eli kussakin on yhtä monta pistettä. Tämä perustuu Huygensin lauseeseen, jonka Späht (1980) todisti. Kun parittaisia neliöetäisyyksiä minimoidaan, niin kokonaisneliövirhe minimoituu ja tällöin voidaan arvioida, että myös kauppamatkustajan ongelman polku lyhenee.

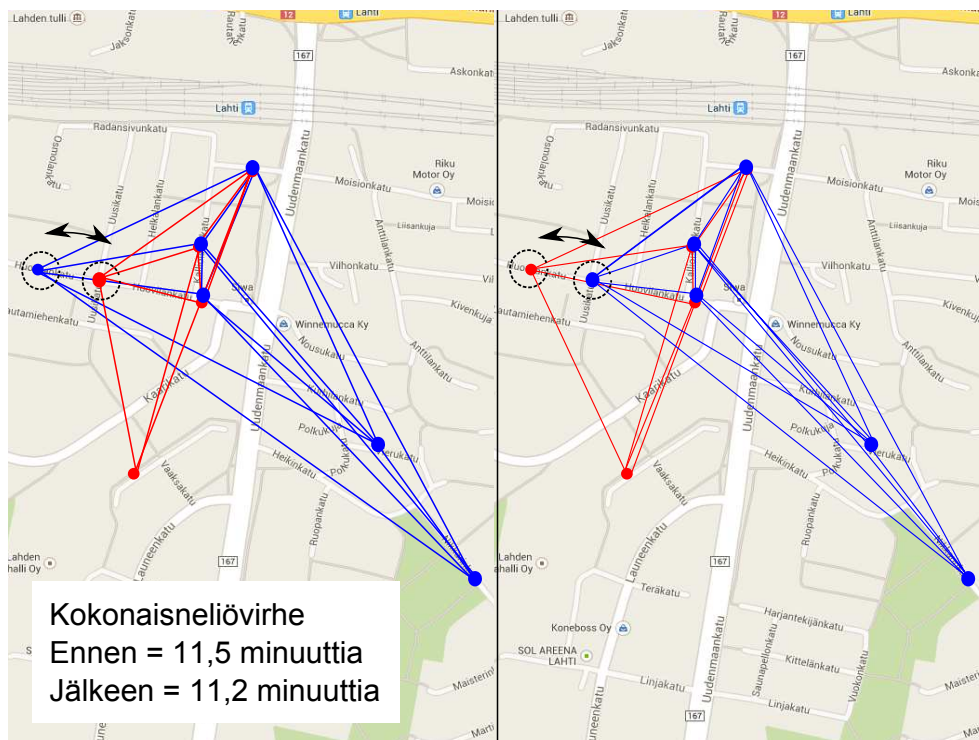
Kustannusfunktio työntekijän tehtäville määritellään seuraavasti,

$$\text{Total cost} = R + a \cdot \text{Total squared error} \quad (2)$$

$$= R + a \cdot \sum_{i=1}^{n-1} \text{TravelTime}(X_i, X_{i+1})^2 \quad (3)$$

missä R on ominaisuusmäärittelyn rikkomisista tulevan kustannusfunktion tulos ja vakio a on käyttäjän määriteltävissä oleva painokerroin kokonaisneliövirheelle.

Malisen (2013) esittämässä algoritmossa työntekijän kulkemaa kokonaismatka-aikaa minimoidaan vaihtamalla tehtäviä kahden työntekijän kesken, huomioiden tehtävien ja työntekijöiden ominaisuusmäärittelyt. Kahden työntekijän välisestä tehtävävaihdosta tarkastellaan kokonaishyötyä. Arvioidessa tehtävien vaihdosta saatavaa hyötyä verrataan kustannusfunktion arvoa ennen ja jälkeen muutoksen. Mikäli kokonaiskustannus on pienempi, työtehtävät vaihdetaan. Kustannusfunktiossa huomioidaan tehtävien ominaisuusmäärittelyt ja tehtävien välillä kulkemiseen kuluva matka-aika. Tehtävien vaihtoja jatketaan kunnes ennalta määritelty määrä vaihtoyrityksiä on suoritettu. Menetelmässä ei huomioida ensimmäiseen kohteeseen siirtymiseen kuluva matka-aikaa.



Kuva 7: Kahden työntekijän tehtävien välisten matka-aikojen neliöiden summa

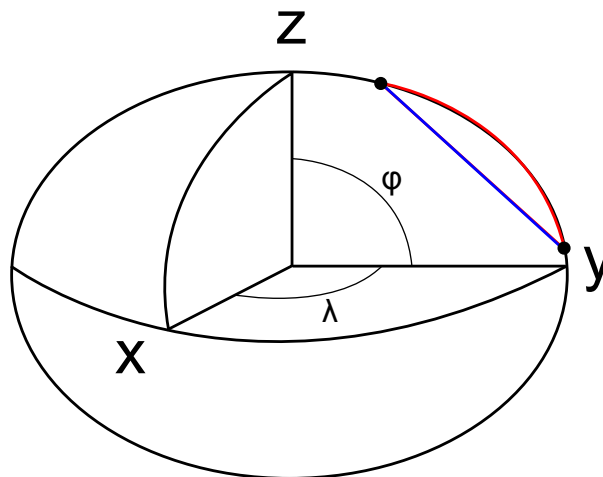
Kuvassa 7 havainnollistetaan tehtävien vaihtoa kahden Lahden kaupungin työntekijän työtehtävien avulla. Samanväriset työtehtävät kuuluvat samalle työntekijälle. Kuvassa vasemmalla on havainnollistettu lähtötilanne ja oikealla lopputilanne, jossa kaksi tehtävää vaihdetaan keskenään. Lähtötilanteessa punaisen työntekijän tehtävien kokonaisneliövirhe kävellen (6 km/h) on noin 5 minuuttia ja sinisen työntekijän tehtävien kokonaisneliövirhe on kävellen (6 km/h) noin 10,3 minuuttia. Yhteenlaskettu kokonaisneliövirhe kävellen on noin 15,4 minuuttia. Lopputilanteessa punaisen työntekijän kokonaisneliövirhe on noin 5,5 minuuttia ja sinisen työntekijän kokonaisneliövirhe on noin 9,8 minuuttia. Yhteenlaskettu kokonaisneliövirhe mahdollisen vaihdon jälkeen on noin 15,3 minuuttia. Koska kokonaisneliövirhe on kokonaisuudessaan pienempi, vaihto suoritetaan. Mikäli sininen työntekijä käyttäisi autoa (40 km/h) töiden suorittamiseen, olisi lähtötilanteen yhteenlaskettu kokonaisneliövirhe noin 5,3 minuuttia ja lopputilanteen noin 5,7 minuuttia, eli tällöin vaihto ei olisi kannattava.

3 Reititysalgoritmit

Tässä luvussa käydään läpi etäisyyden määritelmä sekä verkkojen teoriaa nopeimpien ja lyhimpien polkujen etsimisestä. Verkon kaksi yleisintä muotoa on suunnattu ja suuntaamaton verkko. Tässä luvussa käsitellään suunnattuja verkkoja (Cormen & al 2005).

3.1 Kahden pisteen välinen etäisyys

Etäisyys voidaan laskea kahden pisteen väliltä suoraa viivaa pitkin tasossa *euklidisena etäisyytenä* (linnutie) tai ellipsoidin pintaa pitkin *geodeesisenä etäisyytenä*. Laskentatapojen eroa havainnollistetaan kuvassa 8, jossa euklidinen etäisyys on merkitty sinisellä ja geodeesinen etäisyys on merkitty punaisella.



Kuva 8: Euklidinen etäisyys ja geodeesinen etäisyys

Euklidinen etäisyys voidaan laskea xyz-koordinaatistossa Pythagoraan lauseen avulla seuraavasti.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

Pisteen koordinaatit ilmoitetaan kuitenkin yleensä geodeettisesti, eli kulmamittoina suhteessa päiväntasaajaan ja nollameridiaaniin. Leveysasteen symbolina käytetään kreikkalaista symbolia φ (Fii) ja pituusasteelle symbolia λ (Lambda), kuten kuvassa 8 havainnollistetaan. Geodeesinen etäisyys voidaan tällöin laskea *haversin* -kaavan

avulla seuraavasti,

$$d = 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\varphi_2 - \varphi_1}{2} \right) + \cos(\varphi_1) \cos(\varphi_2) \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right)$$

missä:

r = ellipsoidin säde

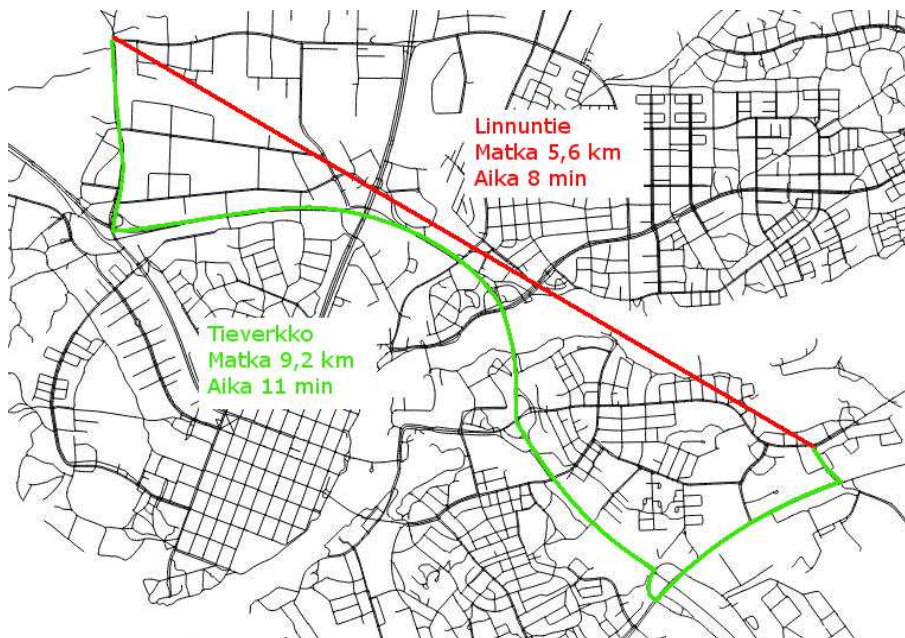
φ_1, φ_2 = leveysaste pisteille 1 ja 2

λ_1, λ_2 = pituusaste pisteille 1 ja 2

Mikäli koordinaatit ilmoitetaan *kartesisisesti* tasoon projisoituna kuten EUREF-FIN koordinaatistossa, voidaan etäisyys tällöin laskea Pythagoraan lauseella seuraavasti.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Eri laskentatavoilla saadaan toisistaan eroavia tuloksia, mutta lyhyillä välimatkoilla erot eivät ole merkittäviä. Yleisesti käytettyjen geodeesisten koordinaatien kanssa on kuitenkin sopivaa käyttää haversin -kaavaa, koska sillä voidaan laskea myös pitkien välimatkojen päässä toisistaan olevien pisteiden välisiä etäisyyksiä tarkasti.

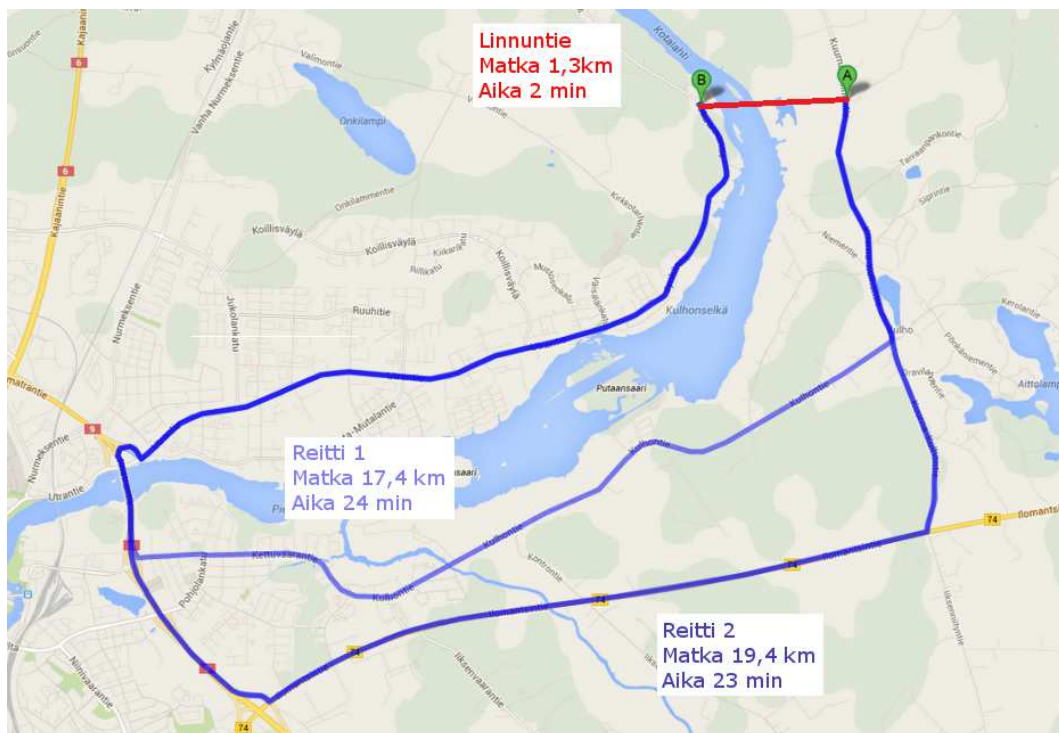


Kuva 9: Linnuntie ja todellinen reitti Joensuun halki. (Digiroad 2010)

Emme kuitenkaan liiku lentäen vaan kuljemme tietä pitkin. Kuvassa 9 nopein reitti tieverkkoa pitkin autolla nopeusrajoitukset huomioiden on 9,2 km (vihreä) ja matka-ajan

arvio 11 min. Vastaava linnuntie-etäisyys on 5,6 km (punainen) ja matka-ajan arvio (40 km/h) 8 min. Tieverkkoa pitkin kuljettaessa lyhin reitti ei ole aina nopein, sillä eri teillä voidaan liikkua eri nopeuksilla. Käytettävä liikenneväline vaikuttaa nopeusarvion lisäksi myös käytettävään reittiin, koska kaikilla teillä ei saa ajaa moottoriajoneuvolla. Kun kahden pisteen välisen lyhimmän reitin matka-aikaa arvioidaan tieverkkoa pitkin kulkevana ja käytettävän kulkuneuvon, sekä tiestön nopeusrajoitukset huomioiden, muuttuu matka-ajan arvioiminen totuudenmukaisemmaksi.

Edellisessä esimerkissä matka-ajan arvio poikkesi todellisuudesta 3 minuuttia, millä käytännössä ei ole juurikaan merkitystä. Kuvassa 10 on linnuntie arvion vääristävä tilanne, jossa arvio eroaa todellisesta matka-ajasta 21 minuuttia.



Kuva 10: Virhemahdollisuus linnuntien ja todellisen reitin välillä (Google 2013).

Mikäli käytössä on tiestön nopeusrajoitustieto tai tiettyyppi, on kuljettavalle reitille mahdollista laskea liikennesääntöjen mukainen maksiminopeus. Todellinen nopeus on yleensä tätä pienempi, sillä muu liikenne hidastaa liikkumista. Myös käytetty liikenneväline vaikuttaa reitin nopeuteen, sillä esimerkiksi kävellen kuljettu reitti voi olla lyhyempi kuin autolla kuljettu.

Kuvassa 11 on havainnollistettu liikennevälineen tuomaa eroa kahden pisteen väliseen etäisyyteen tieverkkoa pitkin. Kuvassa vasemmalla on Googlen reittiehdotus au-



Kuva 11: Kahden pisteen välinen etäisyys autolla ja kävelen (Google 2013).

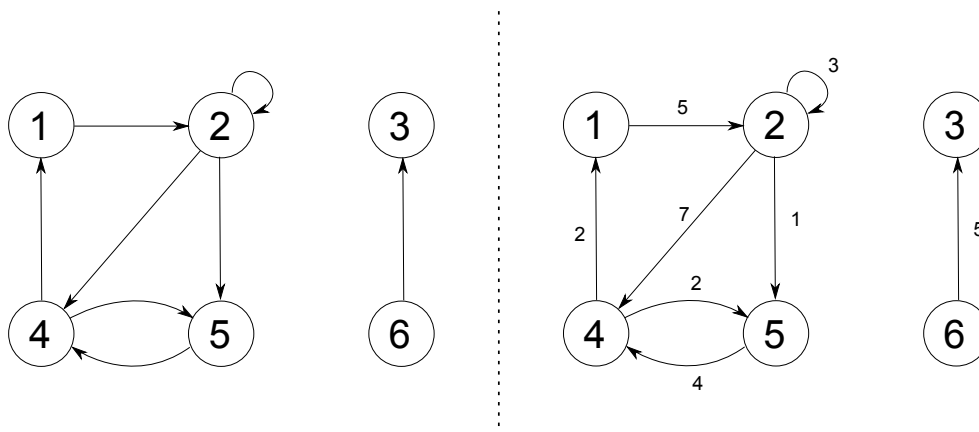
toa käyttäen 1,7 km ja aika-arvio 4 minuuttia. Kuvassa oikealla on reittiehdotus jalkakulkijalle 220 metriä ja aika-arvio 3 minuuttia.

3.2 Verkon määritelmä

Määritelmän mukaan *suunnattu verkko* (*directed graph*) muodostuu parista $G = (V, E)$. Näistä V on äärellinen joukko, joka muodostuu verkon solmuista ja E on kaarista muodostuva relaatiojoukko V :stä. Kaaret ovat suunnattuja ja koostuvat järjestetyistä pareista (u, v) . Tällöin u on solmu, josta kaari alkaa (lähtösolmu) ja v on solmu, johon kaari päättyy (maalisolmu). Kuvassa 12 on havainnollistettuna suunnattu verkko, jossa on solmujoukko $V = \{1, 2, 3, 4, 5, 6\}$ ja kaarijoukko $E = \{(1, 2), (2, 2), (2, 4), (2, 5), (4, 1), (4, 5), (5, 4), (6, 3)\}$

Kaaria voidaan ajatella yksisuuntaisina teinä, sillä polun kaaria voi kulkea vain yhteen suuntaan. Solmusta lähtevien kaarien maalisolmuja kutsutaan lähtösolmun *naapureiksi* (*adjacent*). Yhden kaaren luoma naapuruussuhde on yksisuuntainen. Kuvassa 12 solmulla 2 on naapuruussuhde itsensä, sekä solmujen 4 ja 5 kanssa. Solmujen välillä voi kuitenkin olla kaari kumpaankin suuntaan, jolloin naapuruus muuttuu kahdensuuntaiseksi.

Verkon solmuja yhdistävien kaarien muodostamia reittejä kutsutaan *poluiksi*. Esimerkiksi kuvasta 12 löytyvät polut $(6, 3)$ ja $(1, 2, 2, 5)$. Polku $(1, 2, 2, 5)$ muodostuu solmu-



Kuva 12: Suunnattu verkko ja suunnattu painotettu verkko (Cormen & al 2005)

joukosta $(1, 2, 2, 5)$ ja kaarijoukosta $\{(1, 2), (2, 2), (2, 5)\}$. Polun pituus on polun kaarijoukon kaarten lukumäärä. Polun kaarijoukossa voi olla sama kaari useampaan kertaan ja tällöin ne lasketaan kukin omaksi kaarekseen. Edellämainitun polun pituus on kolme, koska polku muodostuu neljästä solmusta, joita yhdistää kolme kaarta. Polkujen solmu ja kaarijoukot ovat järjestettyjä. Esimerkiksi kuvassa 12 polku $(4, 5, 2)$ ei ole mahdollinen, sillä solmusta 5 ei ole kaarta solmuun 2. Mikäli verkko olisi suuntaamaton, edellämainittu polku olisi mahdollinen. Polku $(2, 5, 4)$ on kuitenkin mahdollinen.

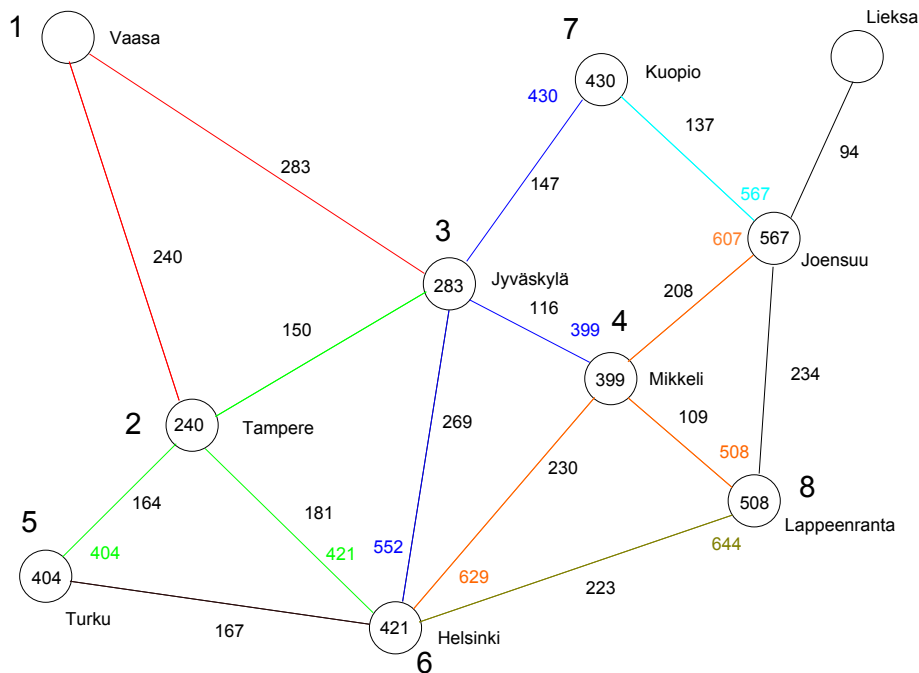
Suuntaamattomissa verkoissa solmuille määritellään *aste* (*degree*), joka on solmuun yhteydessä olevien kaarien lukumäärä. Suunnatussa verkossa solmuille voidaan erikseen määrittää *tulo-* ja *lähtöaste* (*in-degree* ja *out-degree*). Solmun aste on tällöin tulo- ja lähtöasteen summa. Mikäli solmun aste on 0, solmun sanotaan olevan *eristyksissä* (*isolated*). Etsittäessä polkua kahden solmun välillä on tärkeää, että kumpikaan solmu ei ole eristyksissä ja verkko on *yhtenäinen*. Vaikka solmu ei olisi eristyksissä on mahdollista että solmu- ja kaarijoukko on irrallaan muusta verkosta. Tällöin verkko ei ole yhtenäinen. Verkko on yhtenäinen, mikäli kaikkia verkon solmuja yhdistää polku. Tieverkko voi olla epäyhtenäinen mikäli esimerkiksi saarissa on teitä, joita silta- tai lauttayhteys ei yhdistä muuhun verkkoon.

Painotetussa verkossa kaarille asetetaan lisäksi painot, jotka voivat kuvata esimerkiksi kaaren (tai tieverkon tapauksessa tien) pituutta tai aikaa, joka kuluu kaarta vastaavan tien kulkemiseen. Kaarille voidaan myös antaa muitakin ominaisuustietoja kuten tien päällysteen laatu, tien luokitus ja nopeusrajoitukset (Koivisto 2001). Painotetussa verkossa polun pituus voidaan laskea kaarien painojen summana.

3.3 Lyhimpien polkujen selvittäminen

Lyhimpien polkujen selvittämiseen on useita algoritmeja. Tässä luvussa esittelen yhden Dijkstran algoritmin (Cormen & al 2005).

3.3.1 Dijkstran algoritmi



Kuva 13: Dijkstran algoritmin eteneminen ja vaiheet

Dijkstran algoritmi on tunnetuin lyhyimpien polkujen etsimiseen käytettävistä algoritmeista. Se on dynaamiseen ohjelmointiin perustuva ahneesti etenevä algoritmi lyhimpien polkujen etsimiseen. Algoritmi toimii sekä suunnatuissa että suuntaamattomissa painotetuissa verkoissa, mutta verkon kaarien painot eivät saa olla negatiivisia. Kuvassa 13 on suuntaamaton verkko, jota käytetään apuna algoritmin toiminnan havainnollistamisessa. Valintatilanteessa algoritmi valitsee parhaimman tarjolla olevan vaihtoehdon edetessään. Dijkstran tapauksessa liikkueensa solmujen välillä algoritmi valitsee tarjolla olevista solmuista sen, johon saapuvan kaaren paino on pienin yhteenlaskettuna jo kuljettujen kaarten kanssa. Tällöin aina kun solmu valitaan, tiedetään lyhin mahdollinen polku kyseiseen solmuun. Algoritmi löytää edetessään lyhyimpiä polkuja myös muihin solmuihin kuin maalisolmuun. Algoritmin syöte on lähtö- ja maalisolmu, joiden välistä lyhyintä polkua etsitään. Algoritmin edetessä ylläpidetään lyhintä löydettyä

polkua kaikkiin jo käsiteltyihin solmuihin.

Kuvissa 13 ja 14 havainnollistetaan algoritmin etenemistä. Esimerkissä etsitään lyhintä polkua Vaasasta Lappeenrantaan. Ensin katsotaan solmut jotka ovat Vaasan naapureita. Valinta on Tampere, koska kaaren paino on kevyin. Kun Tampere on valittu, tiedämme että ei ole olemassa lyhyempää polkua Vaasasta Tampereelle.

	1	2	3	4	5	6	7	8
1	Vaasa	0	0	0	0	0	0	0
	Lieksa	-	-	-	-	-	-	-
7	Kuopio	-	-	430	430	430	430	430
	Joensuu	-	-	-	607	607	607	607/567
3	Jyväskylä	283	390/283	283	283	283	283	283
4	Mikkeli	-	-	399	399	399	399	399
2	Tampere	243	243	243	243	243	243	243
8	Lappeenranta	-	-	-	508	508	508	508
5	Turku	-	404	404	404	404	404	404
6	Helsinki	-	421	421/552	421/552/629	421	421	421

Kuva 14: Dijkstran eteneminen

Toisessa vaiheessa valinnan kohteina on solmut Jyväskylä, Helsinki ja Turku, eli jo valittujen solmujen naapurit. Polkujen painot ovat Vaasa-Jyväskylä 283, Tampere-Jyväskylä $240+150=390$ (karsiutuu pois, koska tiedetään lyhempi polku Jyväskylään), Tampere-Helsinki $240+181=421$ ja Tampere-Turku $240+164=404$. Valinta on Vaasa-Jyväskylä, sillä polun yhteenlaskettu pituus on lyhin.

Kolmannessa vaiheessa valinnan kohteina ovat Jyväskylä-Kuopio $283+147=430$, Jyväskylä-Mikkeli $283+116=399$ ja toisessa vaiheessa mainitut reitit Tampere-Turku ja Tampere-Helsinkiin. Jyväskylä-Tampere ei enää tarkastella, koska lyhin polku Tampereelle on jo löydetty. Valinta on siis Mikkeli.

Kahdeksan vastaavan vaiheen jälkeen löydetään lyhyin polku Lappeenrantaan, joka on 508 km. Kuvassa 14 kunkin vaiheen valinta on ympyröity punaisella ja kuvassa 13

vaiheet on numeroituna. Samalla löydettiin lyhimmät polut Vaasasta kaikkiin muihin läpikäytyihin kaupunkeihin.



Kuva 15: Dijkstran eteneminen tieverkossa

Dijkstran algoritmi etenee samalla tavalla tieverkossa. Tieverkossa solmuja ovat risteykset ja kaaria niitä yhdistävät tieosuudet. Kuvassa 15 lähtöpistettä lähinnä olevat tieverkon solmut on merkitty punaisella. Jotta vihreällä merkitty polku löytyisi, etenee algoritmi lähtösolmusta edellisen esimerkin tapaan kaikkiin suuntiin kunnes maalisolmu löytyy.

4 Työvuorojen ja reitityksen yhteisoptimointi

4.1 Kauppamatkustajan ongelma

Kokonaisneliövirheen avulla arvioidaan työntekijän keskimääräistä matka-aikaa ja sitä minimoimalla pyritään pienentämään keskimääräistä matka-aikaa. Työntekijän kokonaismatka-aikaa on vielä mahdollista minimoida suunnittelemalla suoritusjärjestys kauppamatkustajan ongelman avulla siten, että valitaan nopein reitti tehtävien suorittamiseen.

Työntekijän tehtävien suorittamiseen käyttämä matka-aika voidaan mallintaa kauppamatkustajan ongelmana. Varsinaisessa kauppamatkustajan ongelmassa kierretään annettujen kaupunkien joukko lyhintä mahdollista reittiä pitkin siten, että kauppamatkustaja vierailee kussakin kaupungissa vain kerran.

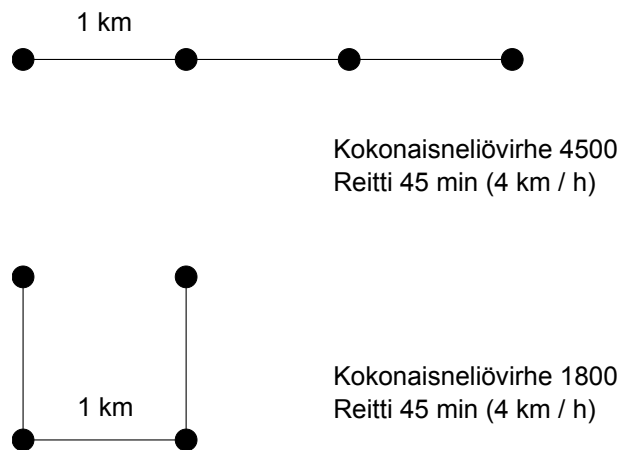
Kauppamatkustajan ongelmasta itsestään on kaksi alakategoriaa, jotka ovat *symmetrinen* ja *epäsymmetrinen* (symmetric / asymmetric travelling salesman problem). Symmetristä kauppamatkustajan ongelmaa (STSP) käytetään, kun verkko on suuntaamaton ja epäsymmetristä (ATSP) kun verkko on suunnattu. Symmetrinen tapaus on epäsymmetrisen erikoistapaus, sillä suuntaamattomasta verkosta voidaan aina johtaa suunnattu verkko korvaamalla kaksisuuntaiset kaaret kahdella suunnatulla kaarella (Gutin ja Punnen 2002). Kauppamatkustajan ongelma linnuntie-etäisyyksillä on symmetrinen ja voidaan ratkaista esimerkiksi lähimmän naapurin algoritmilla (nearest neighbor) joka käy työtehtävät läpi valiten aina edellistä työtehtävää lähimmän seuraavan työtehtävän. Lähimmän naapurin valitseva algoritmi ei kuitenkaan tuota lyhintä ratkaisua kaikissa tapauksissa, joten sen etsimää reittiä voidaan parantaa tabuhaualla (tabu search) tai parviälykkyydellä (swarm intelligence).

Taulukossa 1 kolme ensimmäistä työtehtäväjoukkoa on Lahden kotihoitotehtävistä, joille on ratkaistu nopein reitti kauppamatkustajan ongelman avulla. Kaksi viimeistä ovat esimerkkitapauksia, joissa matka-aika ja pisteiden määrä ovat samat, mutta kokonaisneliövirhe eri. Vaikka reitti 2 on nopeampi kuin reitti 1, on kokonaisneliövirhe lyhyemmässä nopeammassa reitissä silti suurempi. Taulukon matkat on ilmaistu metreinä ja matka-ajat minuutteina. Matkanopeus esimerkeissä on 4 km / h.

Kuvassa 16 havainnollistetaan pisteiden sijoittumisen vaikutus kokonaisneliövirheeseen. Samanpituisten reittien kokonaisneliövirheet voivat poiketa toisistaan.

Taulukko 1: Erilaisia neliövirheitä ja matka-aikoja

	Työtehtävien määrä	Kokonaisneliövirhe	Matka	Matka-aika
Reitti 1	3	88	585 m	8,8 min
Reitti 2	4	106	572 m	8,6 min
Reitti 3	5	177	645 m	9,8 min
Reitti 4	4	4500	3000 m	45 min
Reitti 5	4	1800	3000 m	45 min



Kuva 16: Hajonnan eroavaisuus samanpituisilla reiteillä

Koska tieverkko ei ole täysin symmetrinen, on meidän ongelmamme epäsymmetrinen kauppamatkustajan ongelma. Yksisuuntaiset kadut ja kääntymismääräykset lisäävät tieverkkosovitukseen epäsymmetrisyyttä. Erilaisia verkkoja käsiteltiin luvussa 3.2. Epäsymmetrinen kauppamatkustajan ongelma voidaan ratkaista *geneettisellä paikallisella* (genetic local search) (Freisleben & al 1996). Tämän työn tutkimuksessa käytettiin symmetristä kauppamatkustajan ongelmaa.

4.2 Kauppamatkustajan ongelman sovitus

Kauppamatkustajan alkuperäinen ongelmanmäärittely ei sovi sellaisenaan kotihoidon optimointiin, sillä läpikäytävät työtehtävät ovat sidottuja kellonaikaan. Alkuperäinen

ongelmanmäärittely etsii nopeimman suoritusjärjestyksen, mutta kotihoidon optimoinnissa nopein suoritusjärjestys täytyy etsiä siten, että työtehtävien aikaikkunat otetaan huomioon.

Kauppamatkustajan ongelmasta on olemassa alatyyppejä, jota kutsutaan *ajoneuvon reititysongelmaksi aikaikkunoilla ja rajoitteilla* (Vehicle routing problem with time windows and constraints), josta käytetään tästä eteenpäin lyhettä *VRP*. Siinä otetaan huomioon tehtävien aika-ikkunat ja ominaisuustiedot ja sen avulla saadaan ratkaistua kaikkien pisteiden kautta kulkeva lyhin reitti niin, että aikaikkunat huomioidaan (Desrochers & al 1988). Sekä kauppamatkustajan ongelma että VRP ovat molemmat NP-kovia. Jälkimmäinen voidaan ratkaista eksaktisti aina 15 pisteen instanssiin asti (Desrochers & al 1988). Tämä ratkaisu perustuu branch-and-bound -tekniikkaan, jossa alarajat solmuille lasketaan dynaamisen ohjelmoinnin avulla.

Tämän työn kokeellisessa osuudessa rajoitetaan kokonaisneliövirhettä ja kauppamatkustajan ongelmaa käyttäviin vaihtoehtoihin.

5 Reitityspalvelut

Mikäli kotihoidon työnohjauksessa ja suunnittelussa halutaan käyttää tieverkkoon perustuvaa matka-ajan arviota, tarvitaan keino selvittää reittejä työtehtävien väliltä. Reittejä voidaan selvittää *karttapalveluiden* avulla, joista esimerkkeinä Google Maps, MapQuest Maps, CloudMade ja Here. Edellämainitut palvelut ovat kuitenkin maksullisia. Vaihtoehtona karttapalveluille on hankkia karttadata ja rakentaa sen pohjalta oma palvelu reittien hakemista varten. Maanmittauslaitos tarjoaa ilmaista karttadataa Suomessa ja OpenStreetMap -palvelu tarjoaa ilmaista karttadataa ympäri maailman. Näiden lisäksi on kaupallisia karttadatan tarjoajia kuten Digiroad ja NavTeq. Karttadata ja reititykseen soveltuva tiestödata ovat kuitenkin kaksi eri asiaa, joita tarkastellaan tässä luvussa.

5.1 Kriteerejä reitityspalvelulle

Reititykseen voidaan käyttää valmista karttapalvelua tai rakentaa oma reitinhakupalvelu. Karttapalvelut tarjoavat *web-api* -kirjaston, jonka avulla reittejä haluttujen osoitteiden välillä voidaan hakea verkkokyselyiden avulla. Karttapalvelut ovat maailmanlaajuisia, joten yhden palvelun hankkimalla saa käyttöönsä lähes koko maailman kattavan reitinhakupalvelun.

Oman reitinhakupalvelun rakentaminen vaatii vähintään yhden ihmisen työpanoksen, sekä aineiston. Ylläpitovaiheessa tarvitaan ainakin yhtä palvelinta joka tuo omat kustannuksensa. Aineiston ylläpito ja mahdolliset korjaukset työllistävät päivitystarpeesta riippuen myös yhtä henkilöä satunnaisesti. Esimerkiksi Digiroad-materiaali päivittyy neljä kertaa vuodessa.

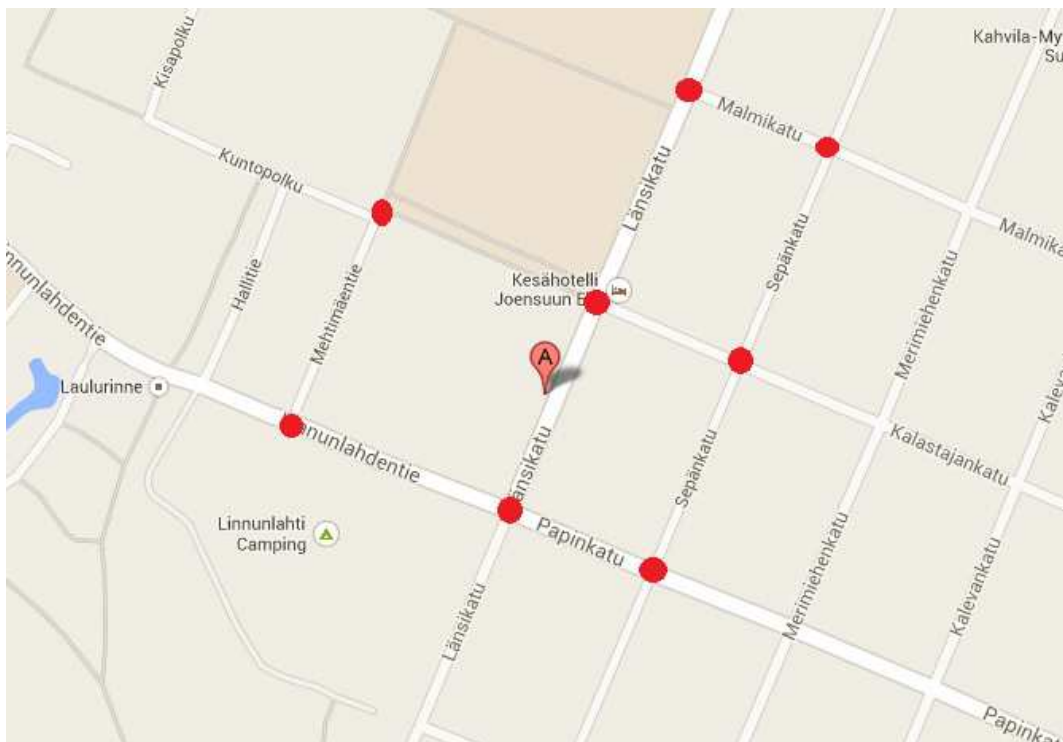
Mikäli käytetään maksullista aineistoa, vaikuttaa reitityspalvelun rakentamisen aineistokustannuksiin reitinhakualueen koko. Aineistojen hinta, saatavuus ja laatu vaihtelee eri valtioissa, joten niiden hankkimiseen, lisensointiin ja yhteensopivaksi muokkamiseen kuluu aikaa. Palvelun rakentamisen jälkeen reititysaineistoa tarvitsee päivittää aika-ajoin tiestön muuttuessa, mikä lisää lisensointikustannuksia.

Valinnassa on otettava huomioon seuraavat kriteerit:

- Työmäärä
- Kokonaishinta
- Luotettavuus
- Laajennettavuus
- Päivitettävyys

Kotihoidon kannalta tärkeät toiminnallisuudet reitityspalvelussa ovat:

- Geokoodaus
- Nopeimmat reitit eri liikennevälineille
- Aineiston ajantasaisuus



Kuva 17: Geokoodaus Länssikatu 15

Kotihoidon optimoinnin kannalta on tärkeää, että palvelu tukee *geokoodausta*, koska järjestelmissä osoitetiedot ilmoitetaan yleisesti kadunnimillä ja talonnumeroilla. Geokoodauksella tarkoitetaan koordinaattien antamista kohteelle. Tässä yhteydessä geokoodauksen avulla selvitetään osoitetiedon sijainti (koordinaatit) ja sen avulla sijainti

tieverkossa. Kuvassa 17 on havainnollistettu osoite Länsikatu 15 ja sen lähellä sijaitsevat verkon solmut. Reitinhakua varten tieverkkoon lisätään tilapäinen solmu ja kaaret sille tieosuudelle, joka toimii haun alku- tai päätepisteenä.

5.2 Web-api -palveluntarjoajat

Oman palvelun rakentamisen vaihtoehtona on käyttää karttapalveluiden web-api-rajapintoja. Valmiiden karttapalveluiden käyttö reittien haussa on helppoa ja niiden vaatima työmäärä on pieni. Tämän lisäksi karttapalvelut kattavat reitinhaun ympäri maapalloa. Karttapalveluiden yrityskäyttö on kuitenkin maksullista. Googlen api:n kaupallisen käyttämisen lisensiointihinnat alkavat 11000 \$ / vuosi ja hinta nousee tehtyjen api-kyselyiden määrän perusteella (Google 2013). Taulukossa 2 on listattuna neljä web-api:a tarjoavaa karttapalvelua. Google tarjoaa useita reittivaihtoehtoja eri liikennevälineille sisältäen matkan pituuden ja matka-ajan arvion. Palvelu sisältää siten kaikki kotihoidon reitinoptimoinnin tarvitsemat ominaisuudet. Reittikyselyt ovat tällöin riippuvaisia verkkoyhteyksien ja palveluntarjoajan toimivuudesta. Karttapalveluiden heikkoutena on se, että niiden käyttämää karttamateriaalia ei pääse muokkaamaan reaaliajassa. Palveluntarjoajille on kuitenkin mahdollista lähettää muutosehdotuksia ja virheilmoituksia. Kaikki vaihtoehdot tukevat geokoodausta, reititystä, taustakarttaa, sekä etäisyyden ja matka-ajan mittaamista eri liikennevälineille. CloudMade käyttää osittain OpenStreetMap-palvelun karttadataa, jota kuka tahansa voi muokata.

Taulukko 2: Web-api vaihtoehdot ja hinnat

Palvelu	Max. kyselyitä	Data	Hinta
Google Maps	100 000 / päivä	suljettu	11000\$ / vuosi
MapQuest	rajoittamaton	suljettu	2500\$ / vuosi
Here	100 000 / päivä	suljettu / navteq	18000\$ / vuosi
CloudMade	100 000 / kk	suljettu / OSM	15\$ / +100 000 kyselyä

Googlen web-api kyselyt tehdään http-rajapinnan kautta alla olevan esimerkin tapaan. Allaolevassa mallikyselyssä haetaan reitti Joensuusta Kuopioon. Lähtöpiste ilmoitetaan *origin*-parametrin avulla ja päätepiste *destination*-parametrin avulla. Kyselyssä tulee lisäksi olla mukana *sensor*-parametri, jonka avulla välitetään tieto onko kyselyä tekevässä laitteessa paikannuslaite. Yrityskäyttäjän täytyy lisäksi syöttää parametrit

client ja *signature* tunnistautumista varten.

```
http://maps.googleapis.com/maps/api/directions/json?  
origin=Joensuu&destination=Kuopio&sensor=false
```

Oletuksena haku palauttaa autoilijalle sopivan reitin. Mikäli halutaan reitti jollekin muulle kulkuneuvolle, syötetään kulkuneuvotieto *mode*-parametrin avulla ja lisätään se kyselyn perään. Lisäparametrina voidaan tällöin antaa esimerkiksi *avoid* ja tässä tapauksessa arvolla *highways*, jolloin pyritään välttämään valtateitä. Kulkuneuvovaihtoehdot ovat auto, kävely, pyöräily ja julkinen liikenne. Julkinen liikenne ei tosin toimi vielä kaikkialla. Kyselyssä voidaan välittää edellämainittujen lisäksi myös muita parametreja.

```
&avoid=highways&mode=bicycling
```

Edellämainituksessa kyselyssä vastaus pyydetään json-muodossa, mutta tulokset voi vaihtoehtoisesti pyytää xml-muodossa. Kysely palauttaa etäisyyden ja matka-ajan lisäksi muun muassa vaihteittaiset reittiohjeet html muodossa etäisyyksineen ja matka-aikoineen, sekä reitin koordinaatteina.

MapQuest-palvelussa kyselyt tehdään http-, json- tai xml-muodossa, mutta kaikki kyselyt vaativat api-avaimen toimiakseen. MapQuestin kyselyssä lähtöpiste välitetään *from* -parametrin avulla ja päätepiste välitetään *to*-parametrin avulla. Haluttu vastausmuoto ilmoitetaan *outFormat*-parametrin avulla (json tai xml). Googleta poiketen vastaus ei sisällä kokonaismatkaa ja matka-aikaa, vaan ainoastaan kunkin reitin vaiheen matkan pituuden ja matka-ajan.

```
http://www.mapquestapi.com/directions/v2/route?key=KEY_HERE  
&from=Joensuu&to=Kuopio&outFormat=json
```

MapQuest-palvelu tukee erilaisia kulkuneuvotyyppejä. Googleta poiketen siinä on mahdollista pyytää autolla kuljettava reitti joko nopeimpana tai lyhyimpänä ja vaihtoehtoina on pyöräily ja julkisen liikenteen käyttö (missä mahdollista). MapQuest tukee edellämainittujen lisäksi useita muita hienosäätöparametreja, kuten esimerkiksi *roadGradeStrategy*, jolla voidaan määritellä halutaanko reitillä vältellä tai suosia ylä- tai alamäkiä (MapQuest 2013).

Here ei tarjoa omassa http-rajapinnassaan mahdollisuutta tehdä kyselyä suoraan osoite-nimillä, joten osoitteet täytyy ensin muuttaa kahden allaolevan kyselyn tapaan koordinaateiksi. Palvelu palauttaa koordinaatit, joko json- tai xml -muodossa käyttäjän valinnan mukaan. Googlen tapaan Heren varmennus tapahtuu kahden lisäparametrin avulla, jotka ovat *app_id* ja *app_code*.

```
http://geocoder.cit.api.here.com/6.2/geocode.json?
app_id=DemoApp&app_code=code&gen=3&city=Joensuu
```

```
http://geocoder.cit.api.here.com/6.2/geocode.json?
app_id=DemoApp&app_code=code&gen=3&city=Kuopio
```

Muiden palveluiden tapaan Here tukee json ja xml -formaattia. Kulkuneuvotyypeistä Here tukee autoa, jalankulkijaa ja kuorma-autoa. Kuorma-auto tyyppi ottaa reitin valinnassa huomioon kuorma-auton tuomat rajoitukset liikenteessä kuten matalat sillat ja painorajoitukset, sekä käyttää pienempää matkanopeuden arviota (Nokia 2013). Reitin tyypit tekevät poikkeuksen muista. Tyyppeinä ovat nopein, lyhin, maisema, sekä 'nopein nyt' ja 'suora-ajo'. 'Nopein nyt' ottaa huomioon vallitsevat liikenneolosuhteet ja 'suora-ajo' vastaavasti ei. Kahteen viimeiseen on erikseen mahdollista asettaa ajon alkamisaika, oletuksena vallitseva kellonaika.

```
http://route.st.nlp.nokia.com/routing/6.2/calculateroute.json?
app_id=DemoApp&app_code=code&waypoint0=62.60423,29.75927
&waypoint1=62.89313,27.67613&mode=fastest;pedestrian
```

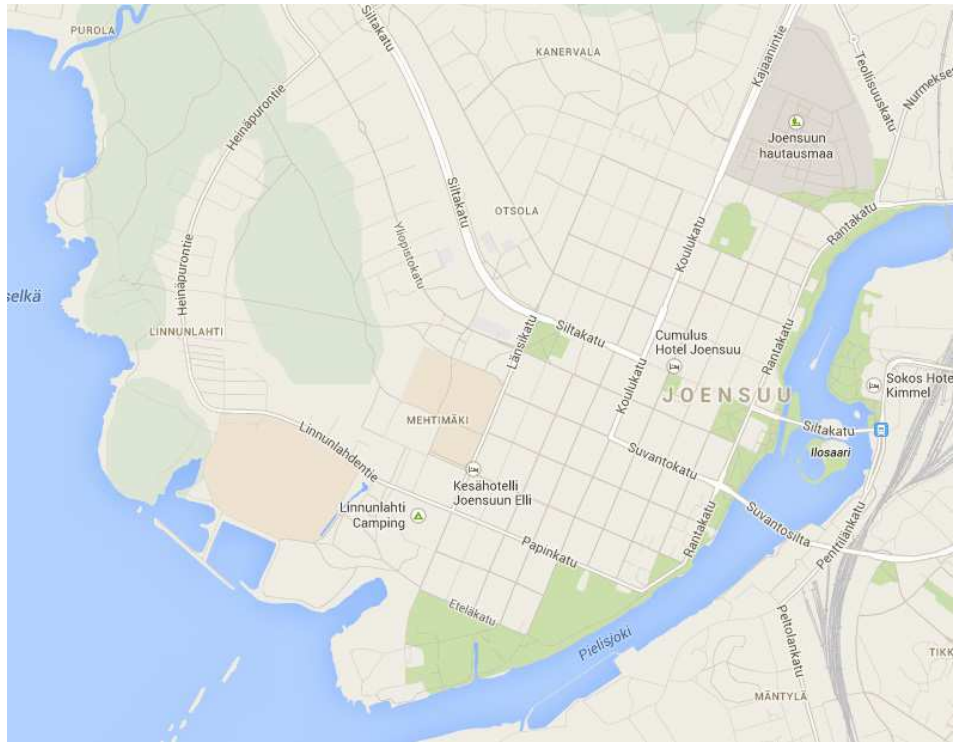
CloudMade-palvelussa kyselyt tehdään http- tai https-rajapinnan kautta ja se vaatii Heren tapaan alku- ja loppupisteet koordinaatteina. Koordinaatit ja reittikysely toimivat allaolevaan tapaan. CloudMade palauttaa tiedot muista poiketen gpx- ja json-formaatissa (CloudMade 2013).

```
http://beta.geocoding.cloudmade.com/v3/<APIKEY>/api/
geo.location.search.2?format=xml&source=OSM
&enc=UTF-8&limit=10&locale=fi&q=Joensuu
http://beta.geocoding.cloudmade.com/v3/<APIKEY>/api/
geo.location.search.2?format=xml&source=OSM
&enc=UTF-8&limit=10&locale=fi&q=Kuopio

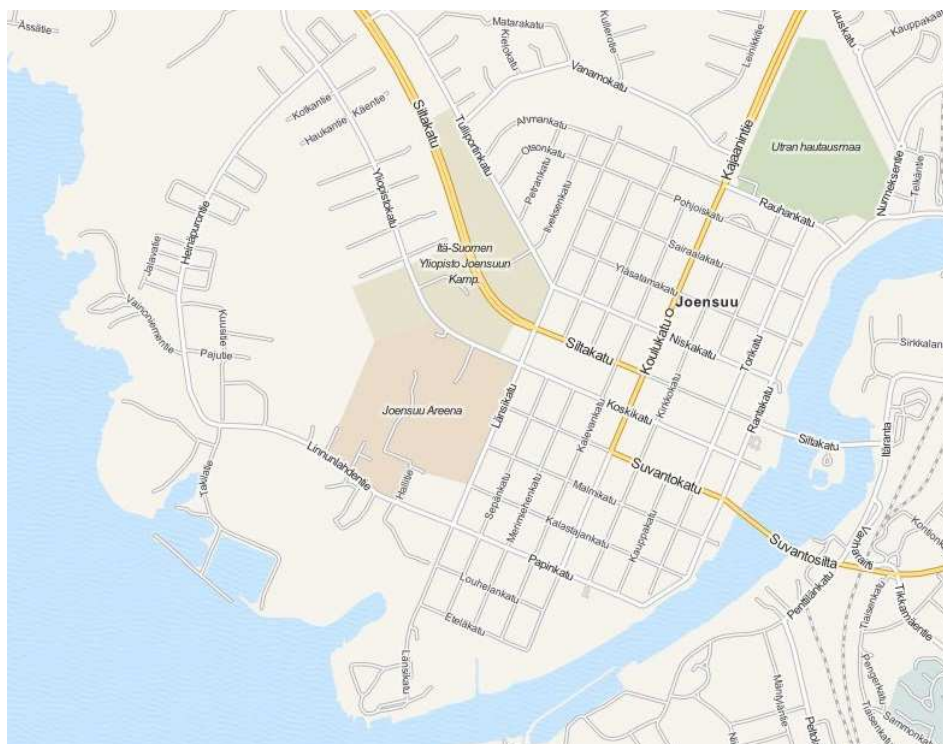
http://routes.cloudmade.com/<APIKEY>/api/0.3/
start_point,end_point/fastest.js
```

Kuvissa 18, 19, 20, 21 on esimerkkinä kartta Joensuusta kaikista edellämainituista palveluista, joista jokaisella on hiukan toisistaan poikkeava värimaailma ja kartalla esiintyvän informaation määrä vaihtelee. CloudMade tarjoaa kartallaan eniten sisältöä ja Here vähiten. Karttakuvista ilmenee että MapQuestin ja Heren kartoista puuttuvat pyörätiet ja lenkkeilypolut. Esimerkki puuttuvasta lenkkeilypolusta näkyy kuvassa 18 vasemmalla kolmiomaisena Linnunlahdentien, Yliopistokadun ja Heinäpurontien sisällä. Tämä juoksijoiden keskuudessa pikkukolmiona ja hullunkolmiona tunnettu 1,7 km pituinen lenkkeilypolku puuttuu kuvista 19 ja 20. Samoin puuttuu suosittu keskustasta

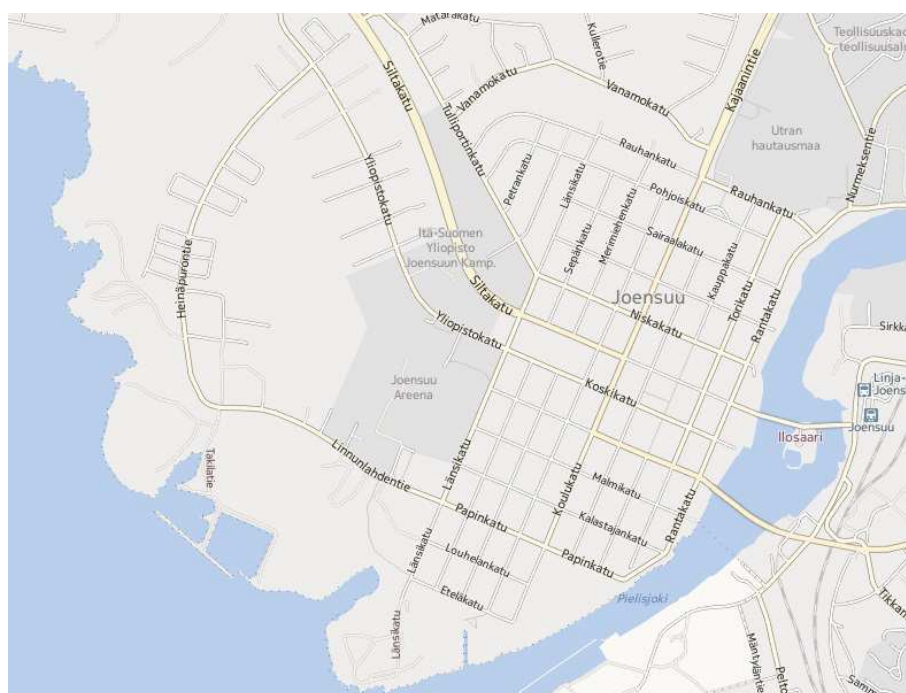
Noljakkaan kulkeva pyörätie. Samat lenkkipolut löytyvät kuitenkin kuvasta 21. Cloud-Made (kuva 21) ei esitä teiden nimiä allaolevassa tarkennustasossa. Teiden nimet löytyvät myös CloudMaden kartasta, mutta yhtä tarkennustasoa lähempää.



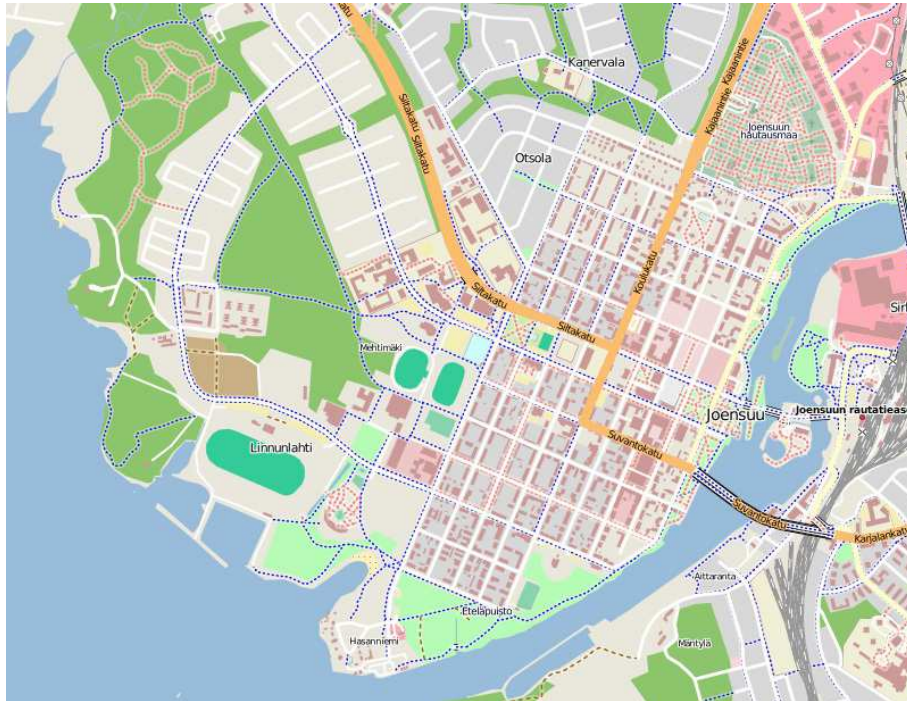
Kuva 18: Esimerkkikuva Joensuusta Googlen karttapalvelusta (Google 2013).



Kuva 19: Esimerkkikuva Joensuusta MapQuest-karttapalvelusta (MapQuest 2013).



Kuva 20: Esimerkkikuva Joensuusta Here-karttapalvelusta (Here 2013).



Kuva 21: Esimerkkikuva Joensuusta CloudMade-karttapalvelusta (CloudMade 2013, MapData ODbI 2013).

5.3 Oman reitityspalvelun rakentaminen

Valittaessa sopivaa materiaalia karttapalvelun rakentamiseen on otettava huomioon yrityskäytön tuomat vaatimukset ja tekniset ominaisuudet. Lyhyimpiä reittejä saadaan selvitettyä suunnatun painotetun verkon avulla, joka sopii kuvaamaan tieverkon etenemissuuntia ja risteyskoja. Materiaalin täytyy olla muokattavissa tähän muotoon.

OpenStreetMap (OSM) tarjoaa verkkoyhteisön keräämää karttadataa ja reittikyselyitä ilmaiseksi. OpenStreetMapin tarjoama karttadata on kuitenkin laadultaan yrityskäyttöön sopimaton, sillä data ei kata kaikkia alueita ja liikenneverkon ominaisuuksia, kuten kiellettyjä ajosuuntia ja kaistamääriä.

5.3.1 Materiaalille asetettavat vaatimukset

Jotta reitityspalvelu voitaisiin toteuttaa, on materiaalissa oltava kuvattuna seuraavat tieverkon ominaisuudet:

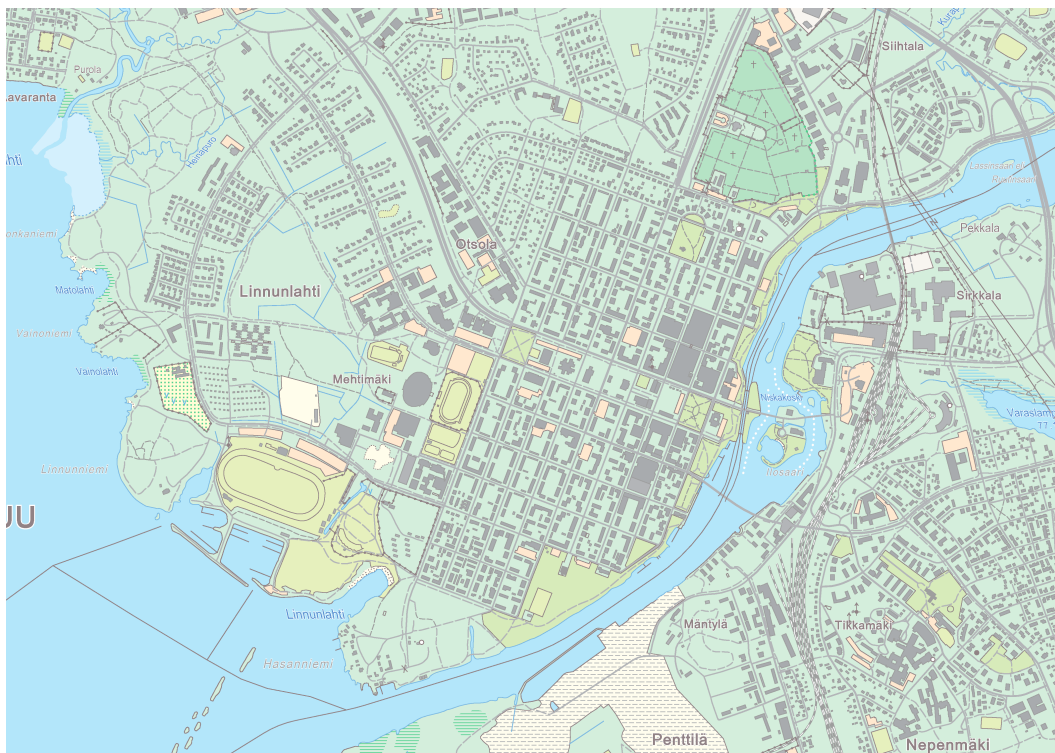
- *Yksisuuntaiset tiet*
- *Kääntymis- ja muut liikennemääräykset*
- *Tietyyppi*
- *Tienosuuden pituus*
- *Nopeusrajoitukset*
- *Teiden nimet*

Yksisuuntaiset tiet, kääntymismääräykset ja kielletyt ajosuunnat ovat tarpeellisia, jotta voidaan hakea reitti, joka noudattaa liikennesääntöjä. Eri liikennevälineitä varten on tiedettävä tiettyypit kuten kävelytie, pyörätie ja moottoritie, joiden avulla voidaan hakea reittejä, jotka soveltuvat kullekin liikennevälineelle parhaiten. Tämän lisäksi reitin matka-ajan arvioimisen kannalta on olennaista, että materiaali sisältää nopeusrajoitustiedot, sekä tiedot tieosuuksien pituudesta. Reitin alkua ja loppupisteen selvittämiseksi osoitetiedosta materiaalin täytyy sisältää kaikkien teiden nimet. Osoitetiedon talonnumero voidaan laskea tieosuuden pituudesta.

5.3.2 Maanmittauslaitoksen aineisto

Toukokuussa 2012 Suomen Maanmittauslaitos (MML) avasi kaiken digitaalisen aineistonsa yritysten ja yksityisten toimijoiden vapaaseen käyttöön ilmaiseksi. Avaaminen koski muiden aineistojen lisäksi myös ilmakuvia, maastotietokantaa, sekä siitä luotua vektori- ja rasteridataa. Aineistoista luodaan Suomen kartat eri mittakaavoissa.

Reitityskäyttöön sopivin aineisto tästä materiaalista on "Tiestö osoitteilla". Aineisto on vektorimuotoinen ja sisältää muun muassa tiestön keskilinjageometrian, sekä tieluokan. Aineisto on jaettu risteysvälien mittaisiin osiin, jotka muodostavat verkon kaaret. Kaarien päätepisteet muodostavat verkon solmut. Vektoritiedosta voidaan laskea tieosuuden pituus ja tieluokkien avulla on mahdollista päätellä nopeusrajoitukset ja erottaa kävelytiet autoteistä. Varsinainen nopeusrajoitus voi kuitenkin olla muu kuin tieluokan nopeusrajoitus. Tievektorit sisältävät tiedon yksisuuntaisuudesta, osoitenumeroinnin ja tienimen, mutta ei kääntymismääräyksiä (MML 2013). Aineiston avulla on mahdollista hakea tieverkon lyhimpiä reittejä, mutta edellämaitusta rajoituksista johtuen ei ole



Kuva 22: Joensuu taustakarttasarjasta 1:10000 (MML 2013)

varmuutta onko kyseistä reittiä mahdollista ajaa laillisesti. Aineisto ei siten täytä yrityskäytön laatuvaatimuksia. Maanmittauslaitoksen avaamat aineistot sisältävät kuitenkin erittäin käyttökelpoisen taustakarttamateriaalin, jonka päällä haettu reitti voidaan esittää. Kuvassa 22 on Joensuun kaupunkikeskusta Maanmittauslaitoksen taustakarttamateriaalista 1:10000 mittakaavaan muokattuna. Maanmittauslaitos on toimittanut tämän saman tiestötiedon Liikennevirastolle Digiroad-aineistoa varten, joka vuorostaan on toimittanut tiestön ominaisuustiedot Maanmittauslaitokselle (MML 2013). Digiroad aineisto on jatkojalostettu tästä materiaalista.

5.3.3 Digiroad

Digiroad on Suomen liikenneviraston tuottama kaupallinen materiaali Suomen tiestöstä. Sen hinnoittelu muodostuu materiaalin irroitukseen käytettävästä tuntityöstä, tiedostojen hinnasta ja toimituskuluista. Yhteen toimitukseen lasketaan kolme työtuntia, joiden lisäksi tulevat tiedostojen hinta ja toimituskulut. Tuntityöveloitus on 70 € / h, tiedostojen hinta 20 € ja toimituskulut 10 €. Digiroad maksaa yhteensä 240 € toimituskerralta. Digiroadin eri julkaisut ovat samanhintaisia, jolloin voi valita haluttuun

käyttötarkoitukseen sopivimman toimitusmuodon hinnasta riippumatta.

Toimituksen yhteydessä Digiroad-materiaaliin tehdään viiden vuoden käyttösopimus. Päivitystoimituksen tilaamalla käyttöoikeus jatkuu viisi vuotta tilaamispäivästä eteenpäin. Aineiston hyödyntämiseen ei ole yksiselitteisiä sääntöjä, vaan ne sovitaan tapauskohtaisesti toimituksen yhteydessä (Helander 2012). Sopimuksen päättyessä myös materiaalin käyttöoikeus päättyy.

Digiroad-aineistoa hyödyntävää sovellusta myytäessä sopimuksessa on mainittava, että sovellus käyttää Digiroad-aineistoa ja että sen tekijänoikeudet kuuluvat Liikennevirastolle. Lisäksi itse sovelluksessa, jossa aineistoa julkaistaan tai toimitetaan, tulee olla tekijänoikeusmerkintä. Digiroad-aineistoa voi jakaa eteenpäin asiakkaalle osana omaa sovellusta. Tämänkaltaista sovellusta myytäessä täytyy siitä olla sovittu etukäteen Liikenneviraston kanssa. Digiroad-materiaalista tehtävä sopimus asettaa myös hyödyntäjälle vastuun ilmoittaa materiaalissa havaitsemansa virheet Liikennevirastolle.

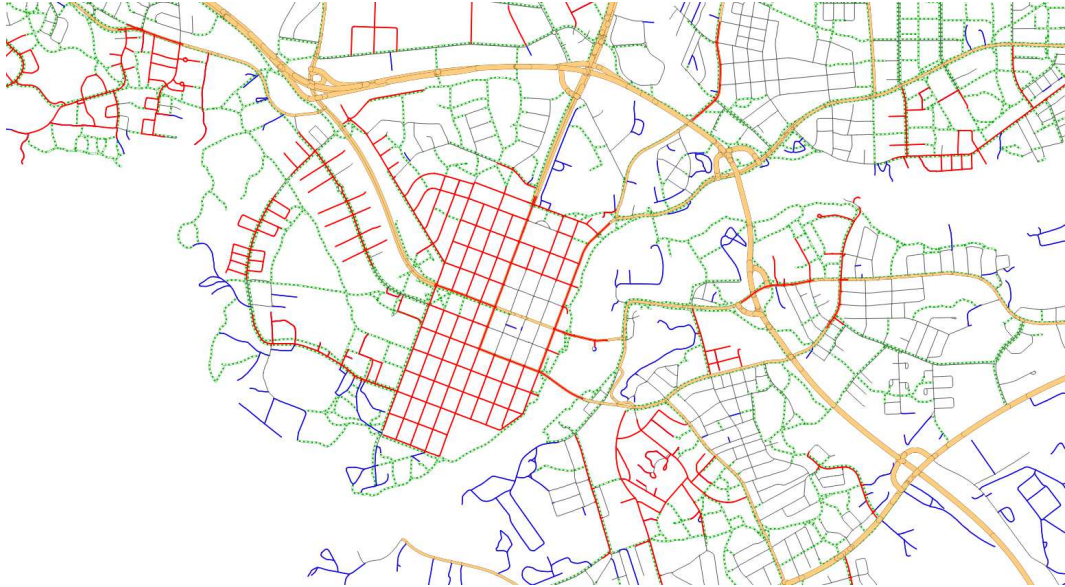
Digiroad-tiestötiedon sisältämät tiedot ovat luonteeltaan pistemäisiä tai viivamaisia. Esimerkiksi tiet ovat viivamaisia ja tiestön palvelut pistemäisiä. Viivamaiset tiedot Digiroadissa ovat *väyliä*. Mahdollisia väylätyyppejä ovat *maantie*, *katu*, *yksityistie*, *kevyen liikenteen väylä*, *rautatie* ja *lautta*. Väylistä ilmoitetaan myös väylän leveys ja onko se pinnoitettu vai ei. Tarkempaa pinnoitetyyppejä ei ole määritelty.

Digiroad-tiestötieto sisältää teiden nimet, jotka on ilmoitettu joko suomeksi ja/tai ruotsiksi riippuen siitä onko tielle annettu nimi molemmilla kielillä. Mikäli tieosuudella on useampi kaista, sisältää Digiroad tiedon niiden lukumäärästä. Tien liikennevirran suunta on myös ilmaistu, kuten myös ohituskieltoalueet, sekä tiellä vallitsevat nopeusrajoitukset. Myös rajoituksien vaikutussuunta on määritelty materiaaliin.

Teiden nimien lisäksi Digiroad sisältää osoitenumeroinnin. Digiroadin tie-elementissä on ilmaistuna ensimmäiset ja viimeiset osoitenumerot oikealla ja vasemmalla puolella tieosuutta (Digiroad 2011). Osoitenumeroitten tarkkoja sijainteja Digiroadissa ei ole, mutta ne voidaan arvioida tieosuuden pituuden perusteella. Osoitenumeroinnin tyyliä kuvataan Digiroadissa kolmella eri tavalla: parittomat ja parilliset talonumerot eri puolilla tietä, parittomat ja parilliset talonumerot samalla puolella tietä, ei säännönmukaisuutta. Yleensä osoitenumerot ovat kuitenkin säännöllisiä siten, että parittomat ja parilliset talonumerot ovat eri puolilla tietä.

Digiroad-materiaali on talletettu vektoreina ja niihin liitettyinä ominaisuustietoina.

Käyttääkseen Digiroad-materiaalia reittien hakemiseen täytyy Digiroad-materiaali ensin muuntaa verkkomuotoon, jolloin täytyy valita verkkototeutus joka sopii parhaiten kyseiseen käyttötarkoitukseen.



Kuva 23: Digiroadin sisältämiä ominaisuustietoja havainnollistettuna

Kuvassa 23 on Digiroad-materiaalista muodostettu kuva, jossa isoimmat tieluokat on keltaisella värillä, yksityistiet sinisellä, kevyenliikenteenväylät vihreällä, kadut mustalla ja 40 km / h nopeusrajoitusalueet punaisella.

5.4 Reitityspalvelun rakentaminen Digiroad K-aineistosta

Oma reitityspalvelu voidaan rakentaa käyttämällä *Digiroad K* -julkaisua ja avoimia ohjelmistotyökaluja. Tässä luvussa esimerkkityökaluina käytetään *Postgres*-tietokantaa ja sen *Postgis*- ja *Pgrouting*-lisäosia.

Digiroad-materiaali siirretään ensin tietokantaan käyttämällä *shp2pgsql*-ohjelmaa tai *pgAdmin3*-ohjelman graafista tuontityökalua, joka kuuluu *Postgis*-julkaisuun. Muunnoksen jälkeen geometriatiedoista muokataan verkkotietokanta *Pgrouting*-lisäosaa varten. Tiestössä solmuja ovat teiden risteykset ja tiet niitä yhdistäviä kaaria. Koska kaikki tiet eivät ole luonteeltaan ja ajosuunnaltaan samanlaisia, verkon täytyy olla suunnattu.

Digiroad K -julkaisusta verkon luomiseen sopii *liikenne-elementti*-tietue, jossa tiestötieto on jaoteltu vähintään risteysvälin mittaisiin osiin. Tällöin on helppoa, joskin ai-

kaavievää, etsiä risteykset eli verkon solmukohdat, jolloin liikenne-elementit muodostavat verkon kaaret. Verkon solmukohdat muodostuvat kaarien alku- ja loppupisteistä, joten solmuja etsiessä tulee käydä läpi kaikki Digiroad:in sisältämät liikenne-elementit. Seuraavilla sql-lauseilla kaaritauluun lisätään lähtö- ja maalisolmutietoa varten kentät source ja target:

```
ALTER TABLE digiroad_liikenne_elementti
ADD COLUMN "source" integer;
ALTER TABLE digiroad_liikenne_elementti
ADD COLUMN "target" integer;
```

Kutakin solmua varten tulee olla yksiselitteinen kokonaisluku, joka merkitään näihin kenttiin. *Pgrouting* tarjoaa työkalun solmujen luomista varten, mutta toimiakseen kysely vaatii muutoksia tiegeometrioiden korkeusmäärittelyihin (z-index), sillä kysely ei toimi mikäli se havaitsee päällekkäisiä solmuja eri korkeusmäärittelyillä. Kysely luo solmuja varten taulun ja merkitsee solmujen id-tietueet kaariin. Allaoleva kysely luo verkon tarvitsemat solmutiedot. Parametreinä kyselylle annetaan kaaritaulun nimi, virhetoleranssi solmujen etsintään, kaaritaulun geometriatiedon sisältävän sarakkeen nimi, kaaritaulun rivin yksilöivän sarakkeen nimi. Virhetoleranssia tarvitaan, koska teiden päätepisteet risteyksissä eivät ole aina koordinaateiltaan identtiset.

```
select pgr_createTopology
('digiroad_liikenne_elementti',0.00001, 'geom', 'gid');
```

Kyselyn tuottama solmutaulu:

```
CREATE TABLE digiroad_liikenne_elementti_vertices_pgr
(
  id bigserial NOT NULL,
  cnt integer,
  chk integer,
  ein integer,
  eout integer,
  the_geom geometry(Point),
  CONSTRAINT digiroad_liikenne_elementti_vertices_pgr_pkey
  PRIMARY KEY (id)
)
```

Tämän jälkeen kaaritaulun solmutiedot indeksoidaan hakujen nopeuttamiseksi:


```
CREATE INDEX source_idx ON digiroad_liikenne_elementti("source");
CREATE INDEX target_idx ON digiroad_liikenne_elementti("target");
```

Digiroadin tiedoissa on kentät *mitattu pituus* ja *matka-aika*, mutta niiden arvoja ei ole julkaistu. Pituus kullekin tieosuudelle voidaan laskea ja tallettaa valmiiseen *mitattu_pi* sarakkeeseen seuraavasti.

```
update digiroad_liikenne_elementti set mitattu_pi =
ST_Length_Spheroid
(geom, 'SPHEROID["GRS_1980",6378137,298.257222101]');
```

Näiden valmisteluiden jälkeen *Pgrouting*:in avulla voidaan etsiä lyhimmän reitin pituus seuraavasti. Luku 1069 on lähtösolmun tunniste ja luku 4906 maalisolmun tunniste.

```
SELECT sum(cost) FROM pgr_dijkstra('
SELECT gid AS id,
source::integer,
target::integer,
mitattu_pi::double precision AS cost
FROM digiroad_liikenne_elementti',
1069, 4906, false, false);
```

Edellinen kysely käyttää kuitenkin kaikkia kaarityyppejä, joten reitti on sopiva vain jalankulkijalle ja silloinkin vain jos reitin varrella ei ole maanteitä. Reitinhaussa käytettävät kaaret saadaan määriteltä *where*-ehtojen avulla esimerkiksi lisäämällä kyselyyn *where vaylatyypp != 4*, joka poistaa kevyenliikenteenväylät hakuavaruudesta.

```
SELECT gid, source, target, mitattu_pi, r_cost from
digiroad_liikenne_elementti where vaylatyypp != 4
```

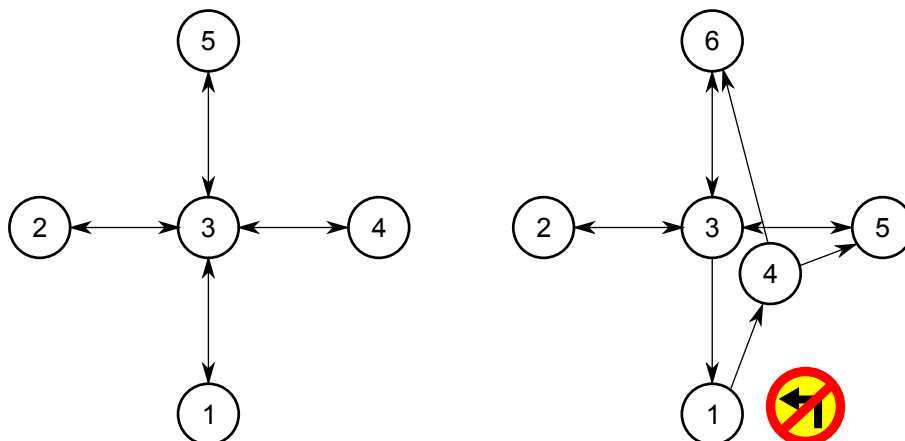
Edellämainittu kysely ei kuitenkaan ota kantaa yksisuuntaisiin katuihin ja tieosuuksiin. Yksisuuntaisten huomioimista varten kaaritauluun pitää määritellä sarake *r_cost*. Mikäli kaarta saa kulkea molempiin suuntiin on sen arvo sama kuin laskemamme *mitattu_pi*, muutoin jokin suuri arvo kuten 10000. Jos yksisuuntaisuus on kaaren loppupäästä alkupäähän, tällöin *mitattu_pi* asetetaan suureksi ja laskettu todellinen pituus asetetaan *r_cost*-sarakkeeseen. Yksisuuntaisuus on ilmoitettu samaan tapaan Digiroad-julkaisussa. Digiroadissa kullakin tien osalla on alku- ja loppupää, ja tieto liikennevirran suunnasta, jonka avulla *r_cost*-arvot saadaan määritettyä. Niiden määrittämisen jälkeen voidaan hakea reittejä liikennesuunnat huomioiden seuraavasti.

```

SELECT sum(cost) FROM pgr_dijkstra('
    SELECT gid AS id,
        source::integer,
        target::integer,
        mitattu_pi::double precision AS cost
    FROM digiroad_liikenne_elementti',
    1069, 4906, true, true);

```

Mikäli halutaan päästä vieläkin realistisempaan matka-ajan arviointiin, täytyy verkossa ottaa huomioon myös kääntymismääräykset. Ne voidaan huomioida luomalla päällekkäisiä solmuja risteyksiin, joissa kääntymismääräyksiä esiintyy. Kuvassa 24 havainnollistetaan kuinka kääntymiskielto toteutetaan verkkototeutuksessa. Kääntymismääräyksissä on kuitenkin huomioitava, että ne voivat rajoittaa esimerkiksi ainoastaan rekkojen kääntymistä. Tästä syystä kuvassa vasemmalla oleva rakenne on myös säilytettävä, mikäli vasemmalle voi kääntyä jalan tai pyöräillen. Mikäli kääntymismääräykset on toteutettu verkkoon edellämainitulla tavalla, niiden toteuttaminen ei vaadi muutoksia reitinhakualgoritmiin.



Kuva 24: Vasemmalle kääntyminen kielletty esitettynä verkkototeutuksena

Vaihtoehtoisesti voidaan luoda oma taulu kääntymismääräyksiä varten. Taulussa listataan kaikkien solmujen saapuvien ja poistuvien kaarien yhdistelmät jotka aiheuttavat lisäkustannuksen reittiin, eli kääntymiskiellot. Tällöin algoritmin täytyy huomioida tämän taulun ehdot. *Pgrouting* tarjoaa funktion, jonka avulla voidaan etsiä reittejä kääntymismääräykset huomioiden. Kääntymismääräykset täytyy ensin syöttää seuraavaanlaiseen tauluun.

```

CREATE TABLE restrictions (

```



```

rid serial,
to_cost double precision,
to_edge integer,
from_edge integer,
via text
);

```

Kun kääntymismääräykset on talletettu omaan tauluunsa, voidaan hakea reittejä kääntymismääräykset ja yksisuuntaiset huomioiden seuraavasti.

```

SELECT seq, id1 AS node, id2 AS edge, cost
FROM pgr_trsp(
  'SELECT gid AS id,
  source,
  target,
  cost
FROM digiroad_liikenne_elementti',
  1069, 4906, true, true,
  'SELECT to_cost, to_edge AS target_id,
    from_edge || coalesce('',' || via, '') AS via_path
  FROM restrictions'
);

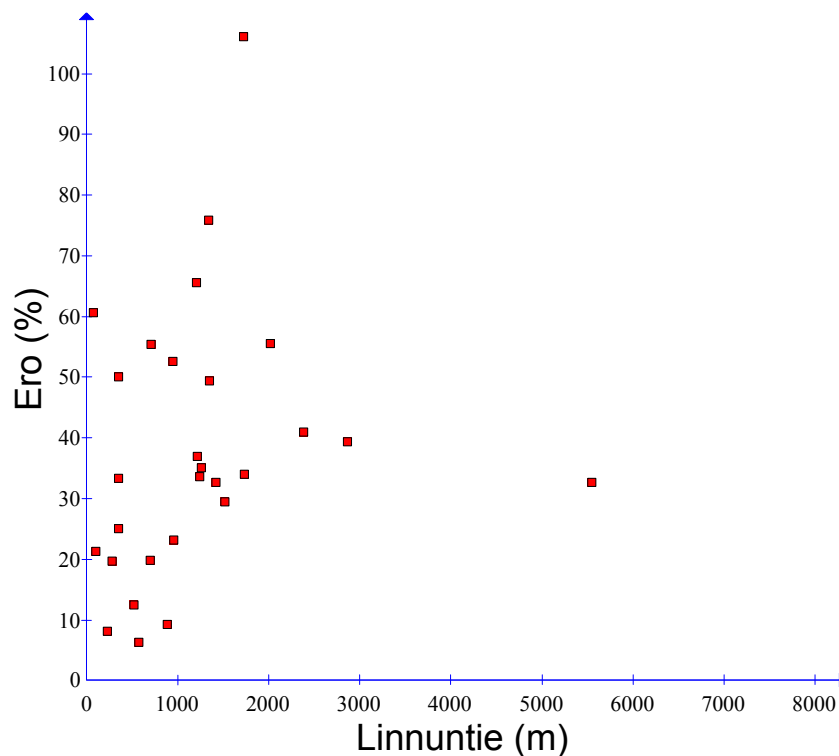
```

6 Kokeelliset tulokset

Kokeellisessa osuudessa selvitettiin linnuntie- ja tieverkko-etäisyyksien eroa kahden pisteen välillä, sekä tämän eron huomioimisen vaikutusta hoitajien päivän työtehtäviä optimoitaessa. Tällä pyritään selvittämään onko optimoinnissa tarpeellista käyttää tieverkkoetäisyyttä.

6.1 Monen pisteen välinen etäisyys

Materiaalina käytettiin Lahden kotihoidon työntekijöiden yhden päivän työtehtäviä. Tehtävät jaettiin Unkarilaisella algoritmilla ja jaon jälkeen kokonaisneliövirhe minimoitiin paikallishaun avulla. Tehtävien suoritusjärjestys ratkaistiin kauppamatkustajan ongelmana tabuhaku-algoritmin avulla siten, että tehtävien aikaikkunoiden rikkomiseen ei kiinnitetty huomiota. Tavoitteena ei ollut selvittää todellista reittiä työntekijöille, vaan selvittää kuinka paljon linnuntie-etäisyytenä lasketun työtehtävien välisen etäisyyden arvio poikkeaa tieverkko-etäisyydestä.



Kuva 25: Reitin pituusero jos käytetään tieverkkoa suhteessa linnuntie-.

Kyseisenä työpäivänä oli 32 työntekijää, mutta neljällä työntekijällä kaikki työtehtä-

vät sijaitsivat samassa paikassa, joten heidät jätettiin pois vertailusta. Vertailuun jäi 28 työntekijää. Tieverkkoa pitkin kuljettu matka oli keskimäärin 38% pidempi ja mediaani 34%. Minimissään ero oli 6% ja maksimissaan 106%. Etäisyyden vaikutusta muutokseen havainnollistetaan kuvassa 25, jossa yksi piste kertoo yhden työntekijän reitin pituuden linnuntie-etäisyytenä ja kuinka monta prosenttia pidempi vastaava reitti oli tieverkkoa pitkin.

6.2 Kahden pisteen välinen etäisyys

Reittien ohella tutkittiin laajemmin satunnaisesti valitun pisteparin välistä etäisyyttä linnuntietä ja tieverkkoa pitkin. Pisteinä käytettiin Mopsi-palveluun (2013) talletettujen kuvien koordinaatteja Joensuun alueelta. Etäisyyden mittaamista varten etsittiin pisteitä lähinnä olleet tieverkon solmut ja poistettiin duplikaatit. Mukaan otettiin pisteet, joiden etäisyys tieverkon solmuun oli 50 metriä tai vähemmän. Pisteet joihin ei johtanut tietä karsittiin. Pisteitä oli 9669, joista karsinnan jälkeen jäi jäljelle 1107.



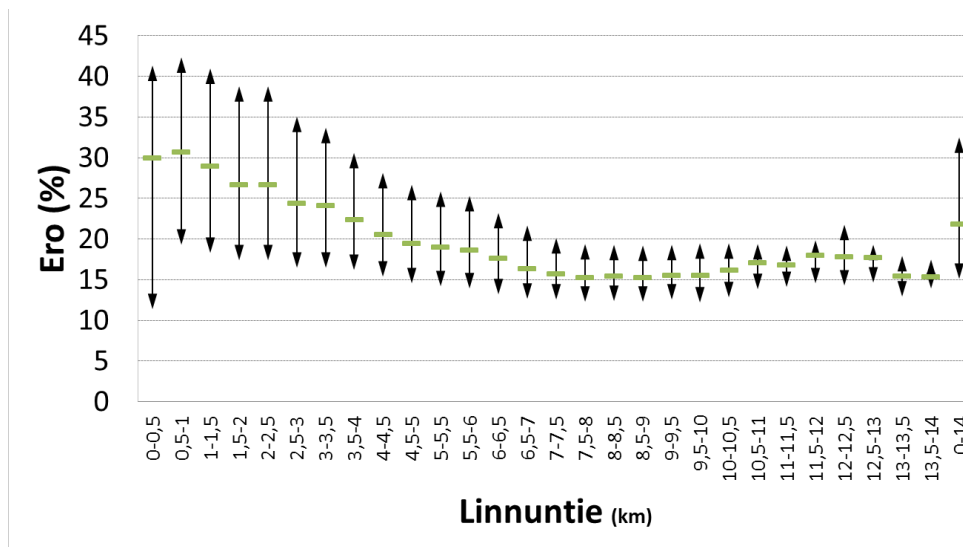
Kuva 26: Tutkimuksessa käytetyt tieverkonverkon solmut kartalla (Digiroad 2013).

Kuvassa 26 pisteitä lähimmät tieverkon solmut on esitetty kartalla (punainen). Reitin hakuun tuli näin yhteensä 612171 pisteparia, siten että saman pisteen välistä etäisyyttä ei lasketa ja pistepareista tarkastellaan vain yhtä suuntaa. Tiereitit laskettiin *Pgroutingin* dijkstran algoritmitoteutuksella käyttäen Digiroadista luotua tieverkkoa. Reitin haku koski kaikkia tietyyppisiä, mutta ei huomioinut tien kulkusuuntaa eikä kääntymis-

määräyksiä.

Solmuparien välisen etäisyyden ero oli pienimmillään 0% ja suurimmillaan 3156%. Tieverkon etäisyys oli keskimäärin 28% suurempi kuin linnuntie-etäisyys ja mediaani 22%. Pienin ero muodostui suoralla tieosuudella, jossa etäisyys on identtinen. Kuvassa 27 on havainnollistettu eri pituusalueiden reittien, sekä koko aineiston ensimmäinen, toinen ja kolmas kvartiili.

Suurimmat erot tulevat joen eri puolilla sillasta kaukana olevista pisteistä. Lisäksi reititys tuo virheitä siten, että osa lähellä toisiaan olevista tieverkon solmuista vaatii pitkähkön kiertotien vaikka todellisuudessa näin ei välttämättä olisi. Esimerkiksi Joensuuun Tiedepuiston ohitse kulkeva Länsikatu ja samansuuntaisesti sen vierellä kulkeva erillinen kevyenliikenteenväylä, joista puuttuvat risteyksien kohdalla olevat yhdistävät suojatiet. Tämän takia tien ylityksen sijaan reitti kulkee kaksi korttelia etelään, siirtyy kadulle ja palaa kaksi korttelia takaisin pohjoiseen.



Kuva 27: Reitit pituuseroa suhteessa linnuntie-etäisyyteen.

Tuloksista voidaan päätellä, että tieverkon etäisyyden ja linnuntie-etäisyyden ero ja vaihteluväli pienenee etäisyyden kasvaessa. Vaikka todellinen matka-aika on arviota monta kertaa pidempi, autolla liikkuessa nämä erot eivät ole merkittäviä, sillä aikataulussa tulee olla tarpeeksi joustoa ruuhkaista liikennettä varten. Kävellessä työnsä hoitavan kohdalla matkan tuplaantuessa ero on merkittävä. Esimerkiksi mikäli hoitajalla olisi viisi 10 minuuttia kestävää siirtymää päivän aikana, joiden todellinen kesto tulisikin olemaan 15 minuuttia ja päivä on aikataulutettu liian tiukaksi, hän voisi pahimmillaan olla 25 minuuttia myöhässä viimeiseltä työtehtävältä.

7 Yhteenveto

Tässä tutkielmassa keskityttiin kotihoidon optimointiin ja sen eri osa-alueisiin. Ongelmanmäärittelyn ohella tutustuttiin yhteen ratkaisumalliin, jota käsiteltiin laajemmin. Kokonaisongelman ratkaisemisen sijaan työ keskittyi reitityspalvelun rakentamiseen ja selvittämään sen avulla laskettujen reittien eroa linnuntie-arvioon. Eri ratkaisutapojen oikeellisuutta, tarkkuutta ja tehokkuutta olisi mielenkiintoista vertailla laajemmin, mikäli näiden käytännön toteutukset saataisiin käyttöön.

Reitityspalvelun avulla päästään totuudenmukaisempaan matka-ajan arvioon, mutta mikäli matkustamiseen varataan linnuntie-arvioon nähden kaksinkertainen määrä aikaa, on sitä myös mahdollista käyttää osana optimointia. Arviot erovat eniten silloin, kun reitin alku- ja päätepiste sijaitsevat eri puolilla jokea tai jotain muuta estettä, jota joudutaan kiertämään tieverkkoa pitkin kuljettaessa. Näissä tapauksissa matkan arvioiminen linnuntie-etäisyyttä käyttäen aiheuttaa suurimmat virheet.

Tutkielmassa ei huomioitu työtehtävien järjestystä, minkä ratkaiseminen VRP:n avulla olisi tutkimuskohteena mielenkiintoinen, mutta valitettavasti sen laajuus oli liian suuri tähän tutkielmaan. Myös TSP-ratkaisujen eroavaisuuden selvittäminen olisi ollut mielenkiintoinen tutkimuskohde.

Viitteet

- Späth H. (1980) *Cluster analysis algorithms for data reduction and classification of objects.* , Wiley, New York, 1980, page 52.
- Kuhn H. W. (1955) *The Hungarian method for the assignment problem.* Naval Research Logistics, 2: 8397. doi: 10.1002/nav.3800020109.
- Malinen M., Pakarinen J., Fränti P. (2013) *Optimizing Home Care Services.* Manuscript.
- Cormen T., Leiserson C., Rivest R., Stein C. (2005) *Introduction to algorithms.* The MIT Press, Cambridge, Massachusetts.
- Koivisto P., Niemistö R. (2001) *Graafiteoriaa.* Tampereen yliopisto.
- Freisleben B., Merz P. (1996) *A genetic local search algorithm for solving symmetric and asymmetric traveling salesman problems.* Proceedings of IEEE International Conference on Evolutionary Computation 616 - 621.
- Desrochers M., Lenstra J. K., Savelsbergh M. V. P., Soumis F. (1988) *Vehicle Routing With Time Windows: Optimization and Approximation.* Vehicle Routing: Methods and Studies, Elsevier Science Publishers B.V, North-Holland.
- Misir M., Verbeeck K., Causmaecker P., Berghe G. (2010) *Hyper-heuristics with a Dynamic Heuristic Set for the Home Care Scheduling Problem.* IEEE Congress on Evolutionary Computation, Belgium.
- Ernst A. T., Jiang H., Krishnamoorthy M., Sier D. (2004) *Staff scheduling and rostering: A review of applications, methods and models.* European Journal of Operational Research 153, 3-27.
- Castillo-Salazar J. A., Landa-Silva D., Qu R. (2012) *A Survey on Workforce Scheduling and Routing Problems.* Practice and Theory of Automated Timetabling (PATAT 2012), 29-31 August 2012, Son, Norway University of Nottingham Jubilee Campus.
- Gutin G., Punnen P. (2002) *Travelling Salesman Problem and its variations.* Kluwer Academic Publishers.
- Digiroad (2011) *Digiroad, Liikennevirasto.* WWW-sivusto, <http://www.digiroad.fi>

digiroad.fi/yleista/fi_FI/yleista/ (22.1.2011).

Helander T. (2012) Sähköpostiviesti 12.4.2012.

Google (2013) *The Google Directions API*. WWW-sivusto, <https://developers.google.com/maps/documentation/directions/> (16.10.2013).

Nokia (2013) *REST APIs*. WWW-sivusto, <http://developer.here.com/rest-apis/> (16.10.2013).

MapQuest (2013) *Mapquest developers*. WWW-sivusto, <http://www.mapquestapi.com/directions/> (16.10.2013).

CloudMade (2013) *Routing HTTP API Docs*. WWW-sivusto, <http://cloudmade.com/documentation/routing> (16.10.2013).

MML (2013) *Maastotietokohteet*. WWW-sivusto, <http://www.maanmittauslaitos.fi/sites/default/files/Maastotietokohteet.pdf> (21.10.2013).

Mopsi (2013) *Track your friends*. WWW-sivusto, <http://cs.uef.fi/mopsi/> (1.12.2013).