TURUN YLIOPISTO

Kauppamatkustajan ongelman virtausmuotoilut

Aapo Pankkio

LuK-tutkielma
Marraskuu 2025

MATEMATIIKAN JA TILASTOTIETEEN LAITOS

Tarkastajat:
Leht. Yury Nikulin

TURUN YLIOPISTO, Matematiikan ja tilastotieteen laitos

LuK-tutkielma
Pääaine: Sovellettu matematiikka
Tekijä: Aapo Pankkio
Otsikko: Kauppamatkustajan ongelman virtausmuotoilut
Ohjaaja: Leht. Yury Nikulin
Sivumäärä: 17 sivua
Aika: Marraskuu 2025

Tässä tutkielmassa käsitellään lyhyesti graafi- ja verkostoteoriaa, minimikustannusvirtausongelmaa ja järjestelyongelmaa. Tutkielmassa esitetään kauppamatkustajan ongelman perinteinen malli ja Dantzig-Fulkerson-Johnson muotoilu. Tämä kaikki huipentuu tutkielman pääaiheeseen eli kauppamatkustajan ongelman virtausmuotoiluihin. Tutkielmassa esitetään kauppamatkustajan ongelman yhden, kahden ja monen resurssin virtausmuotoilut ja näiden ominaisuuksia verrataan lyhyesti toisiinsa ja Dantzig-Fulkerson-Johnson muotoiluun.

Asiasanat: kauppamatkustajan ongelma, graafiteoria, verkostoteoria, minimikustannusvirtausongelma, järjestelyongelma, kauppamatkustajan ongelman virtausmuotoilu.

Sisällys

1 Johdanto

Kauppamatkustajan ongelma (travelling salesman problem), jatkossa voidaan käyttää englanninkielistä lyhennettä TSP, on yksi kombinatorisen optimoinnin suosituimmista ongelmista. Tämä ongelma on helppo ymmärtää, mutta vaikea ratkaista optimaalisesti. Se alkoi nousta suosioon noin 1950-luvulla. Teoksen *Solution of a large-scale traveling-salesman problem* [Dantzif, Fulkerson & Johnson, 1954] esiintyminen lehdessä Journal of the Operations Research Society of America oli yksi kombinatorisen optimoinnin historian päätapahtumista [3]. On kuitenkin olemassa todisteita siitä, että termiä "kauppamatkustajan ongelma" käytettiin mahdollisesti ensimmäisen kerran matemaattisissa piireissä jo vuosina 1931-1932 [3]. Tietojen mukaan ensimmäisen virtausmuotoilun kauppamatkustajan ongelmalle julkaisivat Gavish ja Graves teoksessaan *The travelling salesman problem and related problems* vuonna 1978.

Kauppamatkustajan ongelma on optimointiongelma, jonka ymmärtäminen ei vaadi edistynyttä matematiikan taitoa. Oletetaan, että tiedossa on joukko kaupunkeja ja kaikkien kaupunkiparien väliset etäisyydet (kustannukset). Kaupungeista yksi valitaan aloituskaupungiksi. Nyt kauppamatkustajan ongelma on etsiä optimaalinen (toisin sanoen lyhin tai edullisin) reitti, jossa jokaisessa kaupungissa vieraillaan tasan kerran, jonka jälkeen palataan takaisin lähtökaupunkiin.

Kauppamatkustajan ongelma on hyödyllinen käytännön ongelmien ratkaisemiseen. Monet merkittävät tosielämän ongelmat voidaan muotoilla kauppamatkustajan ongelmaksi. Muutamia tällaisia ongelmia ovat esimerkiksi piirilevyjen poraaminen, tilausten keräily varastoissa ja tietokoneen johdotus.

Kauppamatkustajan ongelmalla on kolme huomattavaa variaatiota. Yleisesti ottaen kauppamatkustajan ongelma voidaan luokitella symmetriseen kauppamatkustajan ongelmaan (sTSP), asymmetriseen kauppamatkustajan ongelmaan (aTSP), ja monen kauppamatkustajan ongelmaan (mTSP) [4]. Jos etäisyys (kustannus) kaupungista i kaupunkiin j on sama kuin kaupungista j kaupunkiin i kaikilla kaupunkipareilla niin on kyseessä sTSP. Taas aTSP tapauksessa nämä kaupunkien väliset etäisyydet (kustannukset) voivat olla erisuuret tai jotkin näistä etäisyyksistä (kustannuksista) voivat puuttua. Monen kauppamatkustajan ongelmassa kauppamatkustajia on useampia kuin yksi. Monen kauppamatkustajan ongelmasta kiinnostuneet voivat perehtyä Lawler, E. L. teoksen *The traveling salesman problem: a guided tour of combinatorial optimization* lukuun 2 sivuihin 23-25.

Tässä tutkielmassa esitetään yhden kauppamatkustajan ongelmalle yhden, kahden ja monen resurssin virtausmuotoilut ja näiden muodostamiseen tarvittava graafija verkostoteoria sekä muutama oleellinen optimointiongelma. Kauppamatkustajan ongelman virtausmuotoiluissa on suhteellisen vähän rajoituksia, mikä tekee niistä käyttökelpoisia käytännön ongelmien ratkaisemiseen. Virtausmuotoilut voi myös yksinkertaisesti mallintaa erilaisilla optimointiohjelmilla lineaarisena sekalukuoptimointitehtävänä, mikä tekee niistä myös käyttäjäystävällisiä. Näiden lisäksi virtausmuotoilujen relaksaatiot ovat rajoituksien määrään verrattuna suhteellisen vahvoja.

Tutkielman luvussa 2 käsitellään lyhyesti graafi- ja verkostoteoriaa, ja luvussa 3 optimointiongelmia. Luvussa 4 käsitellään kauppamatkustajan ongelmaa sekä tutkielman pääaihetta eli kauppamatkustajan ongelman virtausmuotoiluja. Päälähteinä toimivat Marko M. Mäkelä (2019): *Matemaattinen optimointi II* [1] ja Kontoghiorghes, E. J., & Gatu, C. (2007). *Optimisation, Econometric and Financial Analysis* [2]. Virtausmuotoilut mukailevat jälkimmäisen päälähteen esityksiä, jotka on pienillä muutoksilla tehty vastaamaan Turun yliopiston merkintätapoja.

2 Graafi- ja verkostoteoriaa

Tässä luvussa käsitellään lyhyesti graafi- ja verkostoteoriaa. Graafi- ja verkostoteorian käsitteet ovat tarpeellisia esitellessä kauppamatkustajan ongelmaan liittyviä tuloksia ja kauppamatkustajan ongelman muotoiluja. Kaupunkiverkostoa ja monia muitakin asioita voidaan esittää *graafina* (Q, E), missä $Q = q_1, q_2, q_3, ...$ on *solmujen* ja $E = e_1, e_2, e_3, ...$ on niitä yhdistävien *kaarien* joukko. Kaari $e_k = (q_i, q_j)$ voi olla *suunnattu* tai *suuntaamaton*. Suunnatussa tapauksessa kaarta pitkin voi kulkea ainoastaan suuntaan $q_i \to q_j$ ja usein tällaista kaarta kutsutaan *nuoleksi*. Tällöin q_i on alkusolmu, josta kuljetaan loppusolmuun q_j. Suuntaamattomassa tapauksessa kaarta pitkin voi kulkea molempiin suuntiin. Kaaren merkintä ei ole yksikäsitteinen, joten voidaan sopia, että järjestysnumeroltaan pienempi solmu on ensimmäisenä. Jos kaarilla (nuolilla) on reaalilukupainot c_{ij} niin voidaan formuloida ja ratkaista erilaisia laskennallisia optimointiongelmia, jolloin graafia kutsutaan verkoksi. Kaarien (nuolien) paino voi kuvata etäisyyttä, hintaa, aikaa tai muuta suuretta.

Esitellään seuraavaksi verkkoihin liittyviä tämän tutkielman kannalta oleellisia käsitteitä :

- Verkko on *suunnattu*, jos sen kaikki kaaret ovat nuolia.

- Verkko on *suuntaamaton*, jos sen kaikki kaaret ovat suuntaamattomia.

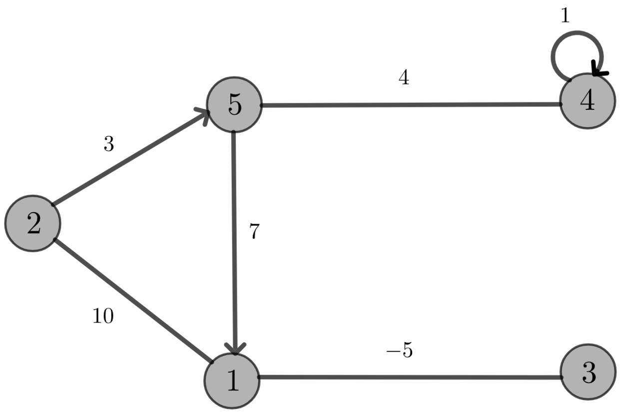
- *Polku* tai *tie* solmujen i ja j välillä on katkeamaton kaari- tai nuolijono (i, k_1), (i, k_2), ..., $(k_{p-1}, k_p), (k_p, j)$, joka yhdistää verkon solmut i ja j. Nuolet on kuljettava oikeassa järjestyksessä eikä missään solmussa käydä kahdesti.

- *Polun pituus* on polkuun kuuluvien kaarien (nuolien) painojen summa.

- *Silmukka* on nuoli (i, i) solmusta i itseensä.

- *Sykli* on polku, jonka alku- ja loppupiste ovat samoja ja $p \geq 1$. Sykli on positiivinen, jos sen pituus on positiivinen ja sykli on negatiivinen, jos sen pituus on negatiivinen.

Verkoilla voidaan havainnollisesti esittää monenlaisia ongelmia. Useat ongelmat voidaan myös formuloida verkko-ongelmiksi, vaikka niiden verkkorakenne ei olisikaan ilmeinen. Kuvassa 1 esitetään esimerkki verkosta. Suuntaamattoman verkon voi formuloida suunnatuksi verkoksi asettamalla jokaisen suuntaamattoman verkon kaaren (i, j) tilalle kaksi nuolta (i, j) ja (j, i). Tällä tavalla symmetrisen suuntaamattoman verkon voi kuvata asymmetrisenä suunnattuna verkkona.



Kuva 1: Esimerkki mahdollisesta viiden solmun muodostamasta verkosta.

3 Verkostovirtaukset

Virtausongelmat ovat lineaarisen optimoinnin erikoistapauksia. Virtausongelmat voidaan yleisesti esittää verkkojen avulla. Virtausongelmissa verkon kaaria pitkin virtaa yhtä tai useampaa resurssia solmusta toiseen. Nämä resurssit voivat olla konkreettisia, kuten ihmisiä, autoja, vettä tai resurssi voi olla jotakin käsitteellistä, kuten kauppamatkustajan ongelman tapauksessa myöhemmin esitetään. Kun puhutaan virtauksesta on syytä olettaa, että verkko on suunnattu ja että virtaus kulkee aina nuolen suuntaan. Seuraavissa alaluvuissa esitetään tutkielman pääaiheelle oleellisia virtausongelmia.

3.1 Minimikustannusvirtausongelma

Tarkastellaan *minimikustannusvirtausongelmaa*

$$\min \quad \sum_{(i,j)\in E} c_{ij}x_{ij} \tag{1}$$

$$\text{s.t.} \quad \sum_{j\in Q,(k,j)\in E} x_{kj} - \sum_{i\in Q,(i,k)\in E} x_{ik} = b_k, \quad \forall\ k \in Q \tag{2}$$

$$0 \leq x_{ij} \leq u_{ij}, \quad \forall\ (i,j) \in E, \tag{3}$$

jossa päätösmuuttujat ovat

x_{ij} : Nuolta (i, j) pitkin tapahtuvan virtauksen määrä.

Ja parametrit ovat

c_{ij} : Nuolella (i, j) virtaavan resurssin yksikkökustannus

b_k : Virtauksen kysyntä tai tarjonta, eli nettotarjonta, solmussa k

u_{ij} : Virtauksen yläraja (kapasiteetti) nuolella (i, j).

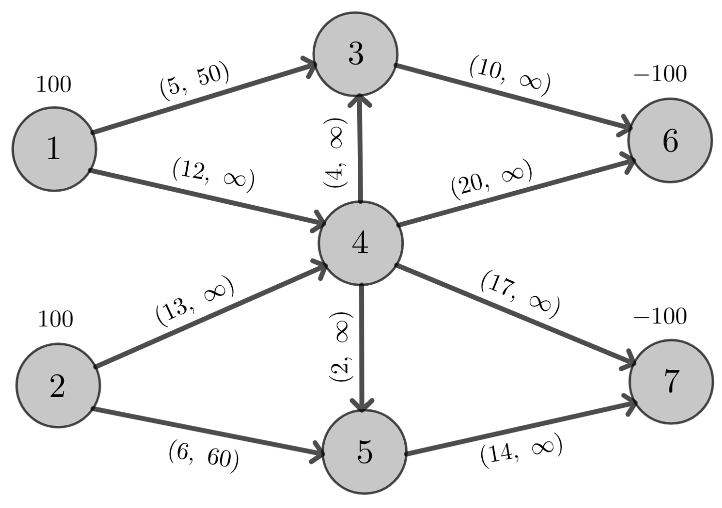
Jos nettotarjonta $b_k > 0$, kutsutaan solmua k *lähteeksi* ja se sisältää resurssia b_k yksikköä. Tämä määrä resurssia virtaa yhteensä lähteestä muihin solmuihin. Jos nettotarjonta $b_k < 0$, kutsutaan solmua k *nieluksi* ja tällaisella solmulla on kysyntää b_k yksikköä. Nieluun virtaa yhteensä b_k yksikköä resurssia muista solmuista. Jos $b_k = 0$, kutsutaan solmua *kauttakulkusolmuksi* ja se ei sisällä tai tarvitse resurssia. Kysynnän ja tarjonnan summa on oltava $\sum_{k \in Q} b_k = 0$. Tarvittaessa voidaan muodostaa ylimääräinen *virtuaalinielu* tai *virtuaalilähde*, jotta kysyntä ja tarjonta saadaan tasapainoon.

Minimikustannusvirtausongelmassa voi esiintyä myös useampaa erillistä resurssia. Tällaisessa tapauksessa täytyy jokaiselle resurssille määritellä omat muuttujansa. Näillä resursseilla voi olla toisistaan eroavat nuolet, kustannukset ja nuolien ylärajat. Edellä mainitussa minimikustannusvirtausongelman mallissa resursseja on vain yhtä tyyppi.

Minimikustannusvirtausongelman kohdefunktiossa (1) minimoidaan jokaisella nuolella kulkevien virtauksien kustannuksien summaa eli kokonaiskustannusta. Rajoitus (2) varmistaa, että kaikki lähteen sisältämät resurssit virtaavat muihin solmuihin, nielut vastaanottavat riittävästi resursseja ja kauttakulkusolmut eivät muuta virtaavan resurssin määrää. Näissä rajoituksissa solmusta lähtevät virtaukset ovat ei-negatiivisia ja solmuun saapuvat virtaukset ei-positiivisia. Nettotarjonta b_k on solmusta lähtevän ja solmuun tulevan resurssimäärän erotus. Lähteistä siis virtaa sen nettotarjonnan verran enemmän resurssia kuin sinne tulee ja nielujen tapauksessa tämä on päinvastoin. Näin lähteet kasvattavat virtaavan resurssin määrää, nielut vähentävät virtaavan resurssin määrää ja kauttakulkusolmut pitävät virtaavan resurssin määrän ennallaan. Rajoitus (3) varmistaa, että kaikilla nuolilla tapahtuva virtaus on ei-negatiivista ja alle nuolen ylärajan. Luonnollisesti nuolia pitkin ei voi virrata resurssia negatiivista määrää.

Parametrien b_k ja u_{ij} ollessa kokonaislukuja löytyy minimikustannusvirtausongelmalle varmasti ainakin yksi kokonaislukukomponenttinen optimiratkaisu. Esimerkissä 1 ratkaistaan yksinkertainen minimikustannusvirtausongelma.

Esimerkki 1. Yritys valmistaa tynnyreitä kahdessa eri tehtaassa. Eräs tilaus tarvitsee yhteensä 200 tynnyriä. Molemmat yrityksen tehtaat ovat valmistaneet tilausta varten 100 tynnyriä. Tynnyrit on ensiksi kuljetettava yrityksen kolmeen varastoon, joista ne voidaan vasta kuljettaa tilaajan kahdelle eri toimipisteelle jotka molemmat tarvitsevat 100 tynnyriä. Tehtaat ovat lähteitä, varastot kauttakulkusolmuja ja toimipisteet nieluja. Tässä tehtävässä ei tarvitse muodostaa virtuaalinielua tai virtuaalilähdettä. Etsitään reitit tehtaista varastoiden kautta toimipisteille, joissa kuljetuskustannukset minimoidaan. Tämän voi formuloida verkko-ongelmana. Yleensä verkon nuolien viereen merkitään pari (c_{ij}, u_{ij}), jossa c_{ij} on tässä tapauksessa yhdestä tynnyristä aiheutuva kuljetuskustannus nuolella (i, j) euroina, ja u_{ij} on rajoitus sille kuinka monta tynnyriä nuolta (i, j) pitkin voi korkeintaan kuljettaa. Verkkoon merkitään myös lähteiden ja nielujen nettotarjonnat. Kuvassa 2 näkyy tehtävän verkko ja tämän alla optimointitehtävä.



Kuva 2: Tynnyriesimerkin verkko

$$
\begin{aligned}
\text{min} \quad & 5x_{13} + 12x_{14} + 13x_{24} + 6x_{25} + 10x_{36} + 20x_{46} + 17x_{47} + \\
& 14x_{57} + 4x_{43} + 2x_{45} \\
\text{s.t.} \quad & x_{13} + x_{14} = 100 && \text{(solmu 1)} \\
& x_{24} + x_{25} = 100 && \text{(solmu 2)} \\
& x_{36} - x_{13} - x_{43} = 0 && \text{(solmu 3)} \\
& x_{46} + x_{47} - x_{14} - x_{24} = 0 && \text{(solmu 4)} \\
& x_{57} - x_{25} - x_{45} = 0 && \text{(solmu 5)} \\
& -x_{36} - x_{46} = -100 && \text{(solmu 6)} \\
& -x_{47} - x_{57} = -100 && \text{(solmu 7)} \\
& x_{13} \leq 50 && \text{(nuolen } (1,3) \text{ kapasiteetti)} \\
& x_{25} \leq 60 && \text{(nuolen } (2,5) \text{ kapasiteetti)} \\
& x_{ij} \geq 0, \quad \forall\, (i,j) \in E. && \text{(nuolet ei-negatiivisia)}
\end{aligned}
$$

Solmu 1 ja solmu 2 kuvaavat tehtaita eli niissä on molemmissa 100 tynnyriä. Näistä solmuista on molemmista siis virrattava 100 tynnyriä muihin solmuihin. Solmu 3, solmu 4 ja solmu 5 kuvaavat varastoja joiden kautta tynnyrit kulkevat toimipisteisiin. Nämä solmut toimivat kauttakulkusolmuina. Solmu 6 ja solmu 7 kuvaavat tilaajan toimipisteitä ja niihin on siis molempiin virrattava 100 tynnyriä muista solmuista. Kahdella nuolella on myös kapasiteettirajoitukset. Kapasiteettirajoitukset voivat johtua esimerkiksi käytetyn kuljetusvälineen tilarajoituksesta. Tynnyreitä ei voi kuljettaa negatiivista määrää ja siksi nuolilla virtaavien tynnyrien määrä on oltava ei-negatiivinen.

Tämän tehtävän voi ratkaista esimerkiksi optimointiohjelmalla CPLEX. Tehtävän optimiratkaisu on

$$x_{13} = 50, x_{14} = 50, x_{24} = 40, x_{25} = 60, x_{36} = 50, x_{46} = 50, x_{47} = 40, x_{57} = 60, x_{43} = 0, x_{45} = 0.$$

Tällä ratkaisulla kohdefunktion arvo on 4750€.

3.2 Järjestelyongelma

Järjestelyongelmassa muodostetaan kahden eri joukon alkioista optimaaliset parit. Olkoon ensimmäinen joukko A ja toinen joukko B. Määritellään järjestelyongelmaa varten päätösmuuttuja

$$x_{ij} = \begin{cases} 1, & \text{jos alkio } i \in A \text{ yhdistetään alkioon } j \in B \\ 0, & \text{muulloin.} \end{cases}$$

Parametreina toimivat c_{ij}, jotka kuvaavat parin (i, j) yhteensopivuutta. Nyt voidaan etsiä ratkaisua, jossa muodostettujen parien yhteensopivuuksien summa on suurin tai voidaan etsiä ratkaisua, jossa muodostettujen parien yhteensopivuuksien summa on pienin. Järjestelyongelman malli on

$$\text{opt} \quad \sum_{i \in A} \sum_{j \in B} c_{ij} x_{ij} \tag{4}$$

$$\text{s.t.} \quad \sum_{j \in B} x_{ij} = 1, \quad \forall\, i \in A \tag{5}$$

$$\sum_{i \in A} x_{ij} = 1, \quad \forall\, j \in B \tag{6}$$

$$x_{ij} = 0, 1, \quad \forall\, i \in A,\ \forall\, j \in B, \tag{7}$$

jolloin ratkaistavana on tavallinen lineaarinen järjestelyongelma. Rajoitus (5) varmistaa, että jokainen alkio yhdistetään tasan yhteen toiseen alkioon. Toisin sanoen jokaisesta solmusta $i \in A$ voi valita vain yhden lähtevän nuolen. Rajoitus (6) varmistaa, että jokaiseen alkioon yhdistetään tasan yksi toinen alkio. Toisin sanoen jokaiselle solmulle $j \in B$ voi valita vain yhden solmuun saapuvan nuolen. Rajoitus (7) varmistaa, että muuttuja x_{ij} on binäärinen. Pari siis joko muodostetaan tai jätetään muodostamatta. Pareja ei voi muodostaa osittain. Rajoitus (7) voidaan myös korvata rajoituksella (3), jossa kaikkien nuolien yläraja $u_{ij} = 1$. Tämä on sallittua, koska rajoituksien (5) ja (6) oikeat puolet ovat kokonaislukuja, jolloin myös ratkaisun komponenttien on oltava kokonaislukuja ja ykköstä suurempaa lukua ei voi esiintyä. Täten muodostuneen lineaarisen optimointitehtävän ratkaisu on myös järjestelyongelman ratkaisu.

4 Kauppamatkustajan ongelma

Olkoon kaikissa muotoiluissa kaupunkien joukko $Q = \{1, 2, ..., n\}$, $n \geq 2$ ja aloituskaupunki on kaupunki 1. Kun kaupunkeja on yksi tai kaksi, eli $n = 1, 2$, on kauppamatkustajan ongelman ratkaisu triviaali. Silmukat eivät ole kauppamatkustajan ongelmassa oleellisia, joten ne estetään jokaisessa muotoilussa erikseen. Luvussa tutkitaan kauppamatkustajan ongelman asymmetrisiä muotoiluja sillä nämä ovat yleisempiä. Kauppamatkustajan ongelman symmetrisiä tapauksia voi ratkaista asymmetrisillä muotoiluilla asettamalla $x_{ij} = x_{ji},\ \forall\ i, j \in Q$ asymmetrisessä muotoilussa.

Kauppamatkustajan ongelman perustavanlaatuisena piirteenä on, että jokaisessa kaupungissa voi vierailla tasan kerran. Järjestelyongelman kohdalla määriteltiin jo rajoitukset (5) ja (6) joiden tarkoituksena oli varmistaa, että jokainen alkio yhdistetään tasan yhteen toiseen alkioon ja jokaiseen alkioon yhdistetään tasan yksi toinen alkio. Nämä rajoitukset ovat siis usein myös oleellinen osa kauppamatkustajan ongelman muotoiluja. Nämä rajoitukset eivät kuitenkaan ole kauppamatkustajan ongelman tapauksessa riittävät. Ne eivät huomio mahdollisten alisyklien muodostumista. Siksi järjestelyongelman rajoituksien lisäksi on muodostettava rajoituksia, jotka estävät alisyklien muodostumisen.

Määritellään binäärimuuttuja

$$x_{ij} = \begin{cases} 1, & \text{jos nuoli } (i,j) \text{ on osa reittiä} \\ 0, & \text{muulloin.} \end{cases}$$

Toisin sanoen muuttuja $x_{ij} = 1$, jos kaupungista i kuljetaan seuraavaksi kaupunkiin j. Tämä muuttuja on käytössä myös kauppamatkustajan ongelman virtausmuotoiluissa. Parametri c_{ij} kuvaa etäisyyttä tai kustannusta nuolella kaupungista i kaupunkiin j. Nyt kauppamatkustajan ongelman kohdefunktiona voidaan käyttää järjestelyongelman kohdefunktiota (4), jossa minimoidaan edellä määritellyllä muuttujalla reitissä käytettyjen nuolien kustannuksien summaa, eli reitin kokonaiskustannusta. Nyt voidaan muodostaa kauppamatkustajan ongelman perinteinen malli

$$\begin{aligned} \min \quad & \sum_{i,j \in Q, i \neq j} c_{ij} x_{ij} \\ \text{s.t.} \quad & \sum_{j \in Q, j \neq i} x_{ij} = 1, \quad \forall\, i \in Q \\ & \sum_{i \in Q, i \neq j} x_{ij} = 1, \quad \forall\, j \in Q \\ & + \text{Alisyklien estäminen} \qquad (8) \\ & x_{ij} = 0,1, \quad \forall\, i,j \in Q. \end{aligned}$$

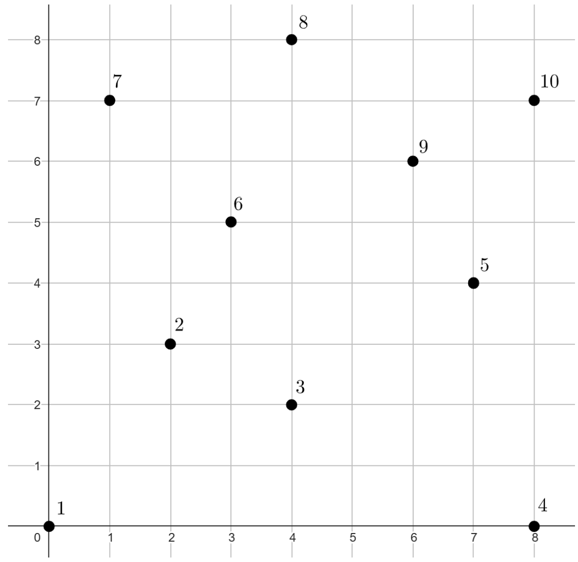
Tämä malli sisältää rajoitukset (5) ja (6) ja näiden lisäksi rajoituksien (8) (subtour elimination constraints (SEC)) tarkoituksena on estää alisyklien muodostuminen. Monet kauppamatkustajan ongelman eri muotoilut eroavat siinä, miten alisyklit estetään. Tähän tarkoitukseen on kehitetty useita erilaisia keinoja. Perinteisin näistä on ehto

$$\sum_{i,j \in S, i \neq j} x_{ij} \leq |S| - 1, \quad S \subseteq \{2, ..., n\},\ 2 \leq |S| \leq n-1, \qquad (9)$$

joka on toimiva, mutta rajoiterikas tapa estää alisyklien muodostuminen. Tällä ehdolla rajoitetaan kuinka monta nuolta osajoukon S solmuista lähtevistä nuolista voi valita reittiin. Sillä jokaisessa osajoukossa S voi valita korkeintaan $|S| - 1$ nuolta, ei näissä osajoukoissa voi syntyä syklejä, koska tämä vaatisi $|S|$ nuolta. Tämän vuoksi alisyklejä ei voi esiintyä reitissä. Kutsutaan kauppamatkustajan ongelman muotoilua, jossa alisyklit estetään ehdon (9) avulla, sen kehittäjien mukaan Dantzig–Fulkerson–Johnson muotoiluksi. Dantzig–Fulkerson-Johnson muotoilussa on $2^{n-1} + n$ rajoitusta ja $n(n-1)$ binäärimuuttujaa. Tämä muotoilu on työläs implementoida optimointiohjelmalla CPLEX, sillä muotoilussa täytyy muodostaa eksponentiaalinen määrä joukon Q osajoukkoja S.

Kauppamatkustajan ongelman virtausmuotoilut noudattavat perinteistä mallia ja alisyklien muodostuminen estetään niissä virtausten avulla, jolloin rajoituksia on edellistä keinoa vähemmän ja muotoilu on yksinkertainen muodostaa CPLEX optimointiohjelmalla. Esimerkki 2 havainnollistaa miksi alisyklien estäminen on kauppamatkustajan ongelman tapauksessa tarpeellista.

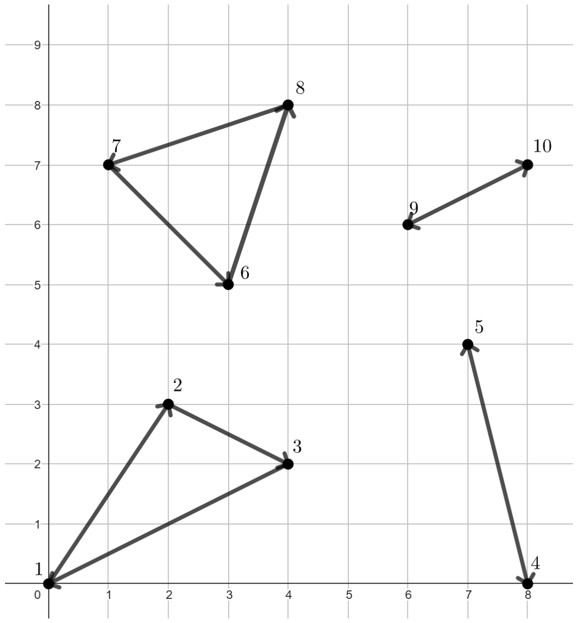
Esimerkki 2. Tarkastellaan piirilevyyn tarvittavien reikien optimaalista porausta. Ratkaistaan tämä minimoivana järjestelyongelmana. Kuvassa 3 näkyy reikien sijainnit ja taulukossa 1 on näkyvillä reikien väliset etäisyydet kahden desimaalin tarkkuudella. Näistä tiedoista muodostetun optimointitehtävän ratkaisun reitti näkyy kuvassa 4. Tämä reitti koostuu neljästä erillisestä alisyklistä. Tämä ratkaisu ei siis ole kauppamatkustajan ongelman ratkaisu ja siksi alisyklien estäminen kauppamatkustajan ongelmassa on tärkeää.



Kuva 3: Piirilevyn reiät

i/j	1	2	3	4	5	6	7	8	9	10
1	-	3,61	4,47	8,00	8,06	5,83	7,07	8,94	8,49	10,63
2	3,61	-	2,24	6,71	5,10	2,24	4,12	5,39	5,00	7,21
3	4,47	2,24	-	4,47	3,61	3,16	5,83	6,00	4,47	6,40
4	8,00	6,71	4,47	-	4,12	7,07	9,90	8,94	6,32	7,00
5	8,06	5,10	3,61	4,12	-	4,12	6,71	5,00	2,24	3,16
6	5,83	2,24	3,16	7,07	4,12	-	2,83	3,16	3,16	5,39
7	7,07	4,12	5,83	9,90	6,21	2,83	-	3,16	5,10	7,00
8	8,94	5,39	6,00	8,94	5,00	3,16	3,16	-	2,83	4,12
9	8,49	5,00	4,47	6,32	2,24	3,16	5,10	2,83	-	2,24
10	10,63	7,21	6,40	7,00	3,16	5,39	7,00	4,12	2,24	-

Taulukko 1: Reikien väliset etäisyydet



Kuva 4: Tehtävän optimireitti

4.1 Virtausmuotoilut

Kauppamatkustajan ongelman virtausmuotoilut yhdistävät piirteitä minimikustannusvirtausongelmasta ja järjestelyongelmasta. Alisyklien muodostuminen voidaan, riippuen resurssien lukumäärästä, estää määrittelemällä yksi tai useampi jatkuva muuttuja ja määrittelemällä näille virtausrajoituksia (2). Kaupunkien nettotarjonnat valitaan niin, että jokaisessa kaupungissa on vierailtava ja aloituskaupunkiin palattava. Tämä yhdessä rajoituksien (5) ja (6) kanssa estävät alisyklien muodostumisen. Eri kauppamatkustajan ongelman virtausmuotoilut estävät alisyklit hieman eri tavoilla.

4.1.1 Yhden resurssin virtausmuotoilu

Ensimmäiseksi esitellään kauppamatkustajan ongelman yhden resurssin virtausmuotoilu. Tätä muotoilua varten täytyy määritellä jatkuva muuttuja

$$y_{ij} = \text{Virtauksen määrä nuolella } (i, j).$$

Nyt perinteisen mallin rajoitukset (4) ja (5) säilyvät ja (8) korvataan rajoituksilla

$$0 \leq y_{ij} \leq (n-1)x_{ij}, \quad \forall\ i, j \in Q,\ i \neq j \tag{10}$$

$$-\sum_{i \in Q, i \neq 1} y_{i1} = -(n-1) \tag{11}$$

$$\sum_{j \in Q, j \neq k} y_{kj} - \sum_{i \in Q, i \neq k} y_{ik} = 1, \quad \forall\ k \in Q \backslash \{1\}. \tag{12}$$

Rajoitus (11) varmistaa, että $n-1$ yksikköä resurssia virtaa kaupunkiin 1 ja rajoitus (12) varmistaa, että yksi yksikkö resurssia virtaa ulos kaikista muista kaupungeista.

Aloituskaupunki toimii siis nieluna ja muut kaupungit lähteinä. Aloituskaupungin nettotarjonta on $-(n-1)$ ja muiden kaupunkien nettotarjonnat ovat 1. Rajoitus (10) rajoittaa virtaukset nuolille, joita reitissä käytetään. Tämä takaa, että virtaukset riippuvat valitusta reitistä. Muuten ne voisi valita vapaasti kunhan virtausrajoitukset silti pätevät, mikä ei estä alisyklien muodostumista. Jokaisessa kaupungissa on siis vierailtava, jotta $n-1$ resurssia voi virrata kaupunkiin 1. Lisäksi rajoitukset (5) ja (6) varmistavat, että aloituskaupungista voi lähteä ja sinne voi palata vain kerran. Täten kaupungeista ei voi muodostua alisyklejä. Lopullinen kauppamatkustajan ongelman yhden resurssin virtausmuotoilu on

$$
\begin{aligned}
\min \quad & \sum_{i,j\in Q, i\neq j} c_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{j\in Q, j\neq i} x_{ij} = 1, \quad \forall\, i \in Q \\
& \sum_{i\in Q, i\neq j} x_{ij} = 1, \quad \forall\, j \in Q \\
& 0 \leq y_{ij} \leq (n-1)x_{ij}, \quad \forall\, i,j \in Q,\ i \neq j \\
& -\sum_{i\in Q, i\neq 1} y_{i1} = -(n-1) \\
& \sum_{j\in Q, j\neq k} y_{kj} - \sum_{i\in Q, i\neq k} y_{ik} = 1, \quad \forall\, k \in Q\backslash\{1\} \\
& x_{ij} = 0, 1, \quad \forall\, i,j \in Q,\ i \neq j.
\end{aligned}
$$

Tätä muotoilua voi parantaa tiukentamalla rajoitusta (10) kaupungeille $i \neq 1$ asettamalla

$$y_{ij} \leq (n-2)x_{ij}, \quad \forall\, i,j \in Q\backslash\{1\},\ i \neq j.$$

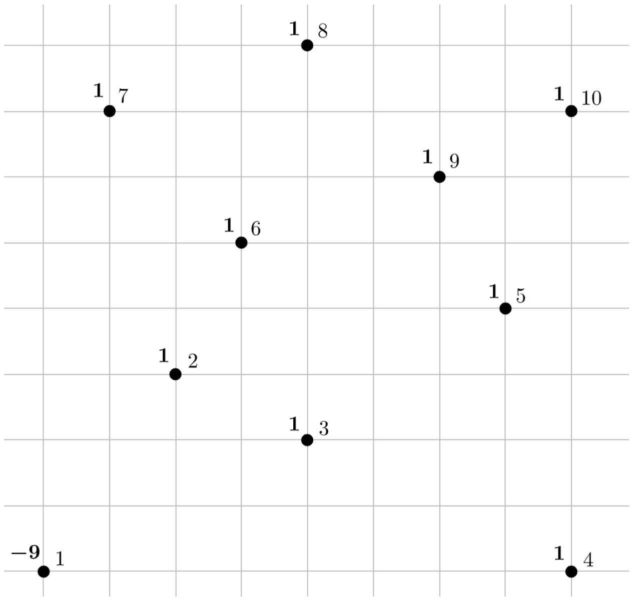
Tämä perustuu havaintoon, että korkeintaan $n-2$ yksikköä voi virrata pitkin nuolia, jotka eivät lähde kaupungista 1 [4].

Tässä muotoilussa on $n(n+2)$ rajoitusta, $n(n-1)$ binäärimuuttujaa, ja $n(n-1)$ jatkuvaa muuttujaa [4]. Esimerkissä 3 ratkaistaan kauppamatkustajan ongelman yhden resurssin virtausmuotoilu.

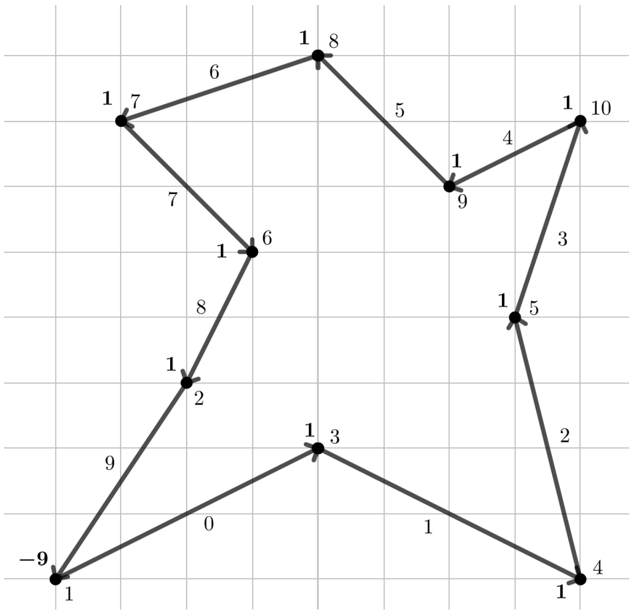
Esimerkki 3. Ratkaistaan esimerkin 2 tapaus käyttämällä kauppamatkustajan ongelman yhden resurssin virtausmuotoilua. Nyt aloituskaupunki voidaan merkitä nieluksi ja muut kaupungit lähteiksi. Muodostettu verkko näkyy kuvassa 5. Verkkoon ei ole merkitty nuolia selkeyden vuoksi. Nyt käyttämällä yhden resurssin virtausmuotoilulle CPLEX optimointiohjelmaa saadaan optimiksi kuvassa 6 näkyvä reitti, jolloin kohdefunktion arvo on 33,13. Jatkuvien muuttujien y_{ij} virtaukset on optimireitissä merkitty nuolien viereen.

4.1.2 Kahden resurssin virtausmuotoilu

Kauppamatkustajan ongelman kahden resurssin virtausmuotoilu säilyttää perinteisen mallin rajoitukset (5) ja (6). Tätä muotoilua varten täytyy määritellä jatkuvat



Kuva 5: Yhden resurssin virtausmallin reiät



Kuva 6: Tehtävän optimaalinen reitti

muuttujat

$$
\begin{aligned}
y_{ij} &= \text{Resurssin 1 virtauksen määrä nuolella } (i,j) \\
z_{ij} &= \text{Resurssin 2 virtauksen määrä nuolella } (i,j)
\end{aligned}
$$

Perinteisen mallin rajoitukset (8) korvataan rajoituksilla

$$
\sum_{j \in Q, j \neq 1} y_{1j} - \sum_{i \in Q, i \neq 1} y_{i1} = -(n-1) \tag{13}
$$

$$
\sum_{j \in Q, j \neq k} y_{kj} - \sum_{i \in Q, i \neq k} y_{ik} = 1, \quad \forall\, k \in Q \backslash \{1\} \tag{14}
$$

$$
\sum_{j \in Q, j \neq 1} z_{1j} - \sum_{i \in Q, i \neq 1} z_{i1} = n-1 \tag{15}
$$

$$
\sum_{j \in Q, j \neq k} z_{kj} - \sum_{i \in Q, i \neq k} z_{ik} = -1, \quad \forall\, k \in Q \backslash \{1\} \tag{16}
$$

$$
\sum_{j \in Q, j \neq i} y_{ij} + \sum_{j \in Q, j \neq i} z_{ij} = n-1, \quad \forall\, i \in Q \tag{17}
$$

$$
y_{ij} + z_{ij} = (n-1)x_{ij}, \quad \forall\, i,j \in Q,\ i \neq j \tag{18}
$$

$$
y_{ij} \geq 0,\ z_{ij} \geq 0, \quad \forall\, i,j \in Q,\ i \neq j.
$$

Rajoitukset (13) ja (14) varmistavat, että resurssia 1 virtaa $n-1$ yksikköä kaupunkiin 1 ja kaikista muista kaupungeista virtaa resurssia 1 yhden yksikön verran muihin kaupunkeihin. Nämä rajoitukset toimivat resurssin 1 suhteen samalla tavalla kuin yhden resurssin virtausmuotoilun vastaavat rajoitukset. Rajoitukset (15) ja (16) varmistavat, että resurssia 2 virtaa kaupungista 1 $n-1$ yksikköä ja kaikkiin muihin kaupunkeihin virtaa resurssia 2 yhden yksikön verran. Resurssilla 2 aloituskaupunki on siis lähde ja kaikki muut kaupungit ovat nieluja. Resurssilla 2 aloituskaupungin nettotarjonta on $n-1$ ja muiden kaupunkien nettotarjonnat ovat -1. Rajoitukset (17) ja (18) varmistavat, että jokaista nuolta pitkin virtaa yhteensä korkeintaan $n-1$ resurssia ja sallivat virtausta ainoastaan nuolissa joita reitissä käytetään. Tämä rajoitus on lähes sama kuin yhden resurssin virtausmuotoilun rajoitus (10), mutta tässä nuolella virtaavan resurssin määrä on aina $n-1$, sillä lähtötilanteessa kaupungista 1 virtaa $n-1$ yksikköä resurssia 2 ja jokaisessa aloituskaupungista eroavassa kaupungissa resurssin 1 määrä virtauksessa kasvaa yhdellä ja resurssin 2 määrä pienenee yhdellä. Kaikissa kaupungeissa on vierailtava, jotta $n-1$ yksikköä resurssia 1 virtaa kaupunkiin 1 ja jokaiseen muuhun kaupunkiin virtaa yksi yksikkö resurssia 2. Alisyklejä ei muodostu sillä kaupungista 1 voi rajoituksien (5) ja (6) mukaan lähteä ja palata vain kerran.

Lopullinen kauppamatkustajan ongelman kahden resurssin virtausmuotoilu on

$$\begin{aligned}
\min \quad & \sum_{i,j\in Q, i\neq j} c_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{j\in Q, j\neq i} x_{ij} = 1, \quad \forall\, i \in Q \\
& \sum_{i\in Q, i\neq j} x_{ij} = 1, \quad \forall\, j \in Q \\
& \sum_{j\in Q, j\neq 1} y_{1j} - \sum_{i\in Q, i\neq 1} y_{i1} = -(n-1) \\
& \sum_{j\in Q, j\neq k} y_{kj} - \sum_{i\in Q, i\neq k} y_{ik} = 1, \quad \forall\, k \in Q\backslash\{1\} \\
& \sum_{j\in Q, j\neq 1} z_{1j} - \sum_{i\in Q, i\neq 1} z_{i1} = n-1 \\
& \sum_{j\in Q, j\neq k} z_{kj} - \sum_{i\in Q, i\neq k} z_{ik} = -1, \quad \forall\, k \in Q\backslash\{1\} \\
& \sum_{j\in Q, j\neq i} y_{ij} + \sum_{j\in Q, j\neq i} z_{ij} = n-1, \quad \forall\, i \in Q \\
& y_{ij} + z_{ij} = (n-1)x_{ij}, \quad \forall\, i,j \in Q,\ i \neq j \\
& x_{ij} = 0,1,\ y_{ij} \geq 0,\ z_{ij} \geq 0, \quad \forall\, i,j \in Q,\ i \neq j.
\end{aligned}$$

Tämä muotoilu sisältää $n(n+4)$ rajoitusta, $n(n-1)$ binäärimuuttujaa ja $2n(n-1)$ jatkuvaa muuttujaa [4].

4.1.3 Monen resurssin virtausmuotoilu

Viimeinen esiteltävä muotoilu kauppamatkustajan ongelmalle on monen resurssin virtausmuotoilu, jossa eri resurssien määrä $k = n-1$. Tätä muotoilua varten täytyy määritellä jatkuvat muuttujat

$$y_{ij}^k = \text{Resurssin } k \text{ virtauksen määrä nuolella } (i,j).$$

Tämä muotoilu säilyttää perinteisen mallin rajoitukset (5) ja (6) ja rajoitukset (8) korvataan rajoituksilla

$$y_{ij}^k \leq x_{ij}, \quad \forall\ i, j, k \in Q,\ k \neq 1,\ i \neq j \tag{19}$$

$$-\sum_{i \in Q, i \neq 1} y_{i1}^k = -1, \quad \forall\ k \in Q \backslash \{1\} \tag{20}$$

$$\sum_{j \in Q, j \neq 1} y_{1j}^k = 0, \quad \forall\ k \in Q \backslash \{1\} \tag{21}$$

$$\sum_{j \in Q, j \neq k} y_{kj}^k = 1, \quad \forall\ k \in Q \backslash \{1\} \tag{22}$$

$$\sum_{i \in Q, i \neq k} y_{ik}^k = 0, \quad \forall\ k \in Q \backslash \{1\} \tag{23}$$

$$\sum_{j \in Q, j \neq l} y_{lj}^k - \sum_{i \in Q, i \neq l} y_{il}^k = 0, \quad \forall\ l, k \in Q \backslash \{1\},\ l \neq k \tag{24}$$

$$y_{ij}^k \geq 0, \quad \forall\ i, j, k \in Q,\ k \neq 1,\ i \neq j. \tag{25}$$

Rajoitukset (20) ja (21) varmistavat, että kaupunkiin 1 virtaa yksi yksikkö jokaista resurssia ja sieltä ei virtaa resursseja muihin kaupunkeihin. Rajoitukset (22) ja (23) varmistavat, että yksi yksikkö resurssia k virtaa ulos kaupungista k ja resurssia k ei virtaa kaupunkiin k. Rajoitus (24) takaa, että jokainen kaupunki kaupunkia k ja 1 lukuunottamatta on resurssille k kauttakulkusolmuja. Jokaisella resurssilla on nieluna aloituskaupunki, jonka nettotarjonta on -1 kaikilla resursseilla. Resurssin k ainoa lähde on solmu k, jonka nettotarjonta on 1, ja kaikki muut aloituskaupungista ja solmusta k eroavat kaupungit ovat tälle resurssille kauttakulkusolmuja. Rajoitus (19) varmistaa, että resurssia virtaa ainoastaan sellaisia nuolia pitkin, jotka ovat osa reittiä. Tämä rajoitus on jokaisella resurssilla sama kuin yhden resurssin virtausmuotoilussa. Jokaisella kaupungilla on oma resurssinsa ja jokaisen niistä on virrattava aloituskaupunkiin. Kaupungista 1 voi rajoituksien (5) ja (6) mukaan lähteä ja palata vain kerran. Täten jokaisessa kaupungissa on vierailtava ja alisyklejä ei voi muodostua.

Lopullinen kauppamatkustajan ongelman monen resurssin virtausmuotoilu on

$$
\begin{aligned}
\min \quad & \sum_{i,j\in Q, i\neq j} c_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{j\in Q, j\neq i} x_{ij} = 1, \quad \forall\, i \in Q \\
& \sum_{i\in Q, i\neq j} x_{ij} = 1, \quad \forall\, j \in Q \\
& y_{ij}^k \leq x_{ij}, \quad \forall\, i,j,k \in Q,\ k \neq 1,\ i \neq j \\
& -\sum_{i\in Q, i\neq 1} y_{i1}^k = -1, \quad \forall\, k \in Q\backslash\{1\} \\
& \sum_{j\in Q, j\neq 1} y_{1j}^k = 0, \quad \forall\, k \in Q\backslash\{1\} \\
& \sum_{j\in Q, j\neq k} y_{kj}^k = 1, \quad \forall\, k \in Q\backslash\{1\} \\
& \sum_{i\in Q, i\neq k} y_{ik}^k = 0, \quad \forall\, k \in Q\backslash\{1\} \\
& \sum_{j\in Q, j\neq l} y_{lj}^k - \sum_{i\in Q, i\neq l} y_{il}^k = 0, \quad \forall\, l,k \in Q\backslash\{1\},\ l \neq k \\
& x_{ij} = 0,1,\ y_{ij}^k \geq 0, \quad \forall\, i,j,k \in Q,\ k \neq 1,\ i \neq j.
\end{aligned}
$$

Tämä virtausmuotoilu sisältää n^3+n^2+6n-3 rajoitusta, $n(n-1)$ binäärimuuttujaa ja $n(n-1)^2$ jatkuvaa muuttujaa [4].

4.2 Muotoilujen vertailu

Tässä aliluvussa verrataan lyhyesti tutkielmassa esitettyjen kauppamatkustajan ongelman muotoilujen ominaisuuksia.

Virtausmuotoiluissa on hyvä huomata, että muuttujien x_{ij} arvojen ollessa kiinnitettyjä on kaikissa muotoiluissa jatkuvien muuttujien ylärajat kokonaislukuja. Jatkuvat muuttujat muodostavat virtausongelman ja jokaisen kaupungin kohdalla nettotarjonta on kokonaisluku. Jatkuvien muuttujien arvot ovat optimiratkaisussa siis kokonaislukuja.

Malli	Rajoitteet	Binäärimuuttujat	Jatkuvat muuttujat
Dantzig-Fulkerson-Johnson	$2^{n-1}+n$	$n(n-1)$	0
Yhden resurssin virtaus	$n(n+2)$	$n(n-1)$	$n(n-1)$
Kahden resurssin virtaus	$n(n+4)$	$n(n-1)$	$2n(n-1)$
Usean resurssin virtaus	n^3+n^2+6n-3	$n(n-1)$	$n(n-1)^2$

Taulukko 2: Eri mallien rajoitteiden ja muuttujien lukumäärät.

Taulukkoon 2 on merkitty eri muotoilujen rajoitteiden, binäärimuuttujien ja jatkuvien muuttujien lukumäärät. Taulukkoon 3 on vielä tiivistetty eri muotoilijen rajoitteiden ja muuttujien aikavaatimukset. Taulukon 3 tiedot ovat peräisin lähteestä [5].

Malli	Rajoitteet	Muuttujat
Dantzig-Fulkerson-Johnson	$\mathcal{O}(2^n)$	$\mathcal{O}(n^2)$
Yhden resurssin virtaus	$\mathcal{O}(n^2)$	$\mathcal{O}(n^2)$
Kahden resurssin virtaus	$\mathcal{O}(n)$	$\mathcal{O}(n^2)$
Monen resurssin virtaus	$\mathcal{O}(n^3)$	$\mathcal{O}(n^3)$

Taulukko 3: Eri mallien rajoitteiden ja muuttujien aikavaatimukset.

Aikaisemmin mainitut muotoilut voidaan relaksoida lineaarisiksi optimointiongelmiksi muuntamalla binäärimuuttuja $x_{ij} = 0, 1$ jatkuvaksi muuttujaksi $0 \leq x_{ij} \leq 1$. Relaksoidun tehtävän optimi on vähintään yhtä hyvä kuin alkuperäisen ja täten se on alkuperäisen tehtävän optimille alaraja. Tätä alarajaa voi käyttää tehostamaan algoritmeja, kuten Branch-and-Bound. Mitä lähempänä relaksaation optimi on alkuperäisen tehtävän optimia niin sitä vahvempi ja parempi relaksaatio on. Tässä tukielmassa mainittujen muotoilujen relaksaatioiden vahvuutta tutkitaan Öncan et al. teoksessa *A comparative analysis of several asymmetric traveling salesman problem formulations* ja seuraavat tiedot ovat peräisin tästä lähteestä. Monen resurssin virtausmuotoilulla ja Dantzig-Fulkerson-Johnson muotoilulla on vahvimmat relaksaatiot ja näiden relaksaatiot ovat ekvivalentit. Yhden ja kahden resurssin virtausmuotoilujen relaksaatiot ovat ekvivalentit ja näiden relaksaatiot ovat heikompia, kuin monen resurssin virtausmuotoilun ja Dantzig-Fulkerson-Johnson muotoilun. Monen resussin virtausmuotoilu ja Dantzif-Fulkerson-Johnson muotoilujen relaksaatioiden optimiarvot eroavat keskimäärin 1,073 prosenttia alkuperäisen tehtävän optimiarvosta. Yhden resurssin ja kahden resurssin virtausmuotoilujen relaksaatioiden optimiarvot eroavat keskimäärin 6,825 prosenttia alkuperäisen tehtävän optimiarvosta.

Dantzig-Fulkerson-Johnson muotoilulla ja monen resurssin virtausmuotoilulla relaksaatiot ovat yhtä vahvat, mutta monen resurssin virtausmuotoilussa on vähemmän rajoitteita. Tosin tässä muotoilussa on myös enemmän muuttujia. Yhden resurssin ja kahden resurssin virtausmuotoilujen relaksaatiot ovat yhtä vahvat, mutta yhden resurssin virtausmuotoilussa on vähemmän rajoitteita ja muuttujia.

Viitteet

[1] Marko M. Mäkelä (2019): *Matemaattinen optimointi II.* Luentomoniste, Turun yliopisto.

[2] Kontoghiorghes, E. J., & Gatu, C. (2007). *Optimisation, Econometric and Financial Analysis* (E. J. Kontoghiorghes & C. Gatu, Eds.; 1. Aufl., Vol. 9). Springer-Verlag. https://doi.org/10.1007/3-540-36626-1

[3] Lawler, E. L. (1985). *The traveling salesman problem: a guided tour of combinatorial optimization.* Wiley.

[4] Davendra, D. (2010). *Traveling salesman problem, theory and applications* (D. Davendra, Ed.). IntechOpen.

[5] Temel Öncan, İ. Kuban Altınel, Gilbert Laporte, *A comparative analysis of several asymmetric traveling salesman problem formulations*, Computers & Operations Research, Volume 36, Issue 3, 2009, Sivut 637-654, ISSN 0305-0548, https://doi.org/10.1016/j.cor.2007.11.008.