

Aalto-yliopisto
Perustieteiden korkeakoulu
Teknillisen fysiikan ja matematiikan koulutusohjelma

Henkilöstön kustannustehokas lomasuunnittelu

Mat-2.4108 Sovelletun matematiikan erikoistyö
29.12.2014

Teemu Kinnunen

Valvoja: Professori Harri Ehtamo
Ohjaaja: TkT Antti Punkka, VR-Yhtymä Oy

Työn saa tallentaa ja julkistaa Aalto-yliopiston avoimilla verkkosivuilla. Muilta osin kaikki oikeudet pidätetään

Sisällysluettelo

1. Johdanto.....	2
2. Kirjallisuuskatsaus.....	3
3. Optimointimalli	5
4. Lomien suunnitteluongelmia	9
4.1. 10 työntekijän tehtävä.....	10
4.2. 20 työntekijän tehtävä.....	11
4.3. 83 työntekijän tehtävä.....	13
5. Tulosten analysointi.....	15
Lähteet	16

1. Johdanto

Tässä erikoistyössä luodaan lineaarinen optimointimalli henkilöstön lomien suunnittelua varten. Mallissa lomat suunnitellaan viikkotasolla, mutta suunnittelu voidaan helposti muuttaa esimerkiksi päivä- tai kuukausitasolle tarpeen mukaan. Jokaiselle viikolle määritellään maksimi lomaa pitävien työntekijöiden määrälle. Lisäksi jokaisen työntekijän on pidettävä tietty määrä lomaa vuoden tai muun suunniteltavan ajanjakson aikana. Työntekijät jaetaan kahteen ryhmään heidän lomaviikkojensa määrän mukaan.

Mallissa työmäärä ja työntekijöiltä vaadittava työpanos vaihtelevat ajankohdasta toiseen. Esimerkiksi kesällä ja juhlapyhien aikaan tarvitaan monilla aloilla vähemmän työvoimaa, kuin tavallisesti. Vaihteleva työmäärä mallinnetaan asettamalla eri arvoja viikoittaisille lomaa pitävien työntekijöiden maksimimäärille. Epävarmuus huomioidaan kohdefunktion avulla, se maksimoi pienintä viikko kohtaista ylijäävien lomamahdollisuuksien määrää. Tämä tasoittaa lomien jakautumista ja jättää jokaiselle viikolle varmuusvaran äkillisille muutoksille.

Malli sisältää kolme kautta, joille lomien pitäminen jaetaan: kesää edeltävä aika, kesälomakausi ja kesän jälkeinen aika. Jokaiselle kaudelle on omat rajoitteensa, joiden avulla voidaan määrittää, että kuinka paljon lomaa tiettyyn ryhmään kuuluvan työntekijän on pidettävä kauden aikana. Lisäksi malliin on sisällytetty kaksi eri tapaa pakottaa ratkaisuja sellaisiksi, että lomat pidetään peräkkäin kauden sisällä.

Pohjatietona mallin käytölle eri yksiköissä voidaan käyttää yksikkökohtaisia työmääräennusteita. Niiden avulla voidaan määrittää sopivat arvot lomaa pitävien työntekijöiden sallituille määrille eri viikoilla. Lomasuunnittelun onnistuessa hyvin työntekijäresurssit saadaan käytettyä mahdollisimman kustannustehokkaasti, kun tarvittava ylitöiden määrä pienenee.

Erikoistyön toinen luku sisältää kirjallisuuskatsauksen, jossa käydään läpi lomien ja vapaapäivien optimointia eri aloilla. Kolmannessa luvussa formuloidaan ILP-tehtävä, jonka avulla voidaan ratkaista erilaisia henkilöstön lomien optimointitehtäviä. Neljännessä luvussa esitellään kolme esimerkkiongelmää, ja ratkaistaan ne kehitetyllä optimointimallilla. Viidennessä luvussa tehdään yhteenveto tuloksista ja mallin toiminnasta. Lisäksi pohdintaan miten mallia voisi parantaa.

2. Kirjallisuuskatsaus

Puhtaasti lomien optimointiin keskittyviä julkaisuja on hyvin vähän. Sen sijaan läheisesti aiheeseen liittyvä vapaapäivien optimointi on saanut kirjallisuudessa paljon enemmän huomiota. Baker ja Magazine (1977) esittelevät ratkaisutavan vapaapäivien suunnitteluongelmaan, jossa arkipäivinä on tietty työntekijöiden tarve N ja viikonloppupäivinä tietty työntekijöiden tarve n . Kirjoittajat johtavat suljetun muodon esityksen työntekijöiden tarpeen minimille työntekijöiden päivittäisten tarpeiden funktiona. Burns (1978) esittää myös laskentakaavan työntekijöiden tarpeen minimille. Hänen laskentatapansa on ensimmäinen, joka toimii ongelmissa, joissa työntekijöille halutaan vapaita viikonloppuja. Bartholdi ym. (1980) käyttävät syklisen vapaapäivien optimointiongelman ratkaisemiseen rajoitettuja verkkovirtaustehtäviä. He myös esittelevät pyöristysalgoritmin, jonka avulla relaxoidulle tehtävälle löydetään kokonaislukuratkaisu. Morris ja Showalter (1983) mallintavat vapaapäivien optimointiongelman joukonpeittämisiongelmana (*set covering problem*). Tehtävän ratkaisemisessa käytetään apuna leikkaavan tason menetelmää.

Chong ja Strevell (1985) esittelevät julkaisussaan huutokauppaan perustuvan menetelmän, jonka avulla armeijan lentolaivueen miehistöille jaetaan lomat. Tavoitteena on täyttää miehistöjen toiveet mahdollisimman hyvin. Lomien määräämisessä käytetään pistejärjestelmää, joka antaa jokaiselle miehistölle mahdollisuuden asettaa tarjouksia haluamilleen loma-ajoille. Rajoitteena lomien määräämisessä on, että kullakin ajanhetkellä on tietty työntekijöiden tarve, jota ei saa alittaa.

Ensin miehistöt merkitsevän avoimesti näkyvillä olevaan kalenteriin haluamiaan loma-aikoja. Muiden haluamia loma-aikoja nähdessään miehistö voi muuttaa valintojaan ja siirtää omia lomapyyntöään ajankohtiin, joissa on vähemmän päällekkäisyyksiä. Toisessa vaiheessa lomien ajankohdat lukitaan, ja kukin miehistö asettaa tarjouksia valitsemilleen ajankohdille (jokaisella miehistöllä on käytössään 20 pistettä). Tietyn lomajakson saa itselleen korkeimman tarjouksen tehnyt miehistö. Muut, jotka panostivat samaan lomajaksoon saavat pisteensä takaisin ja voivat tarjota niiden avulla vapaana olevista lomajaksoista. Menetelmä antaa jokaiselle miehistölle tasa-arvoiset mahdollisuudet ja jakaa lomapyynnot aikaisempaa tasaisemmin koko vuodelle. Menetelmän käyttöönoton myötä miehistöjen tyytyväisyys lomien määräämisprosessiin nousi huomattavasti.

Beaumont (1997) kuvaa vapaapäivien optimointiongelmaa, jossa käsitellään koko vuoden töitä. Ongelma formuloidaan MILP-tehtävänä. Mallin rakentamisessa keskitytään pitkän aikavälin ratkaisuun. Lyhytaikaiset (esimerkiksi sään tai sairauspoissaolojen aiheuttamat) muutokset työntekijöiden tarpeessa otetaan huomioon palkkaamalla töihin ulkopuolisia urakoitsijoita, tai kutsumalla töihin työntekijöitä, jotka ovat valmiita tekemään ylimääräisiä työpäiviä. Työntekijöiden työ- ja vapaapäiville tulee löytää syklejä, jotka täyttävät työntekijöiden ja yrityksen johdon tekemien sopimuksien ehdot:

- Peräkkäisiä työpäiviä pitää olla aina 4-7
- Kalenteriviikon aikana saa olla enintään 5 työpäivää
- Vapaapäiviä tulee olla peräkkäin 2-4
- 47 viikon syklin aikana työpäiviä pitää olla 223 tai 224

Ongelman ratkaisemisessa käytettiin hyväksi leikkaavan tason menetelmää, ja ratkaiseminen suoritettiin CPLEX – ohjelmistolla. Kehitetty ratkaisumenetelmä mahdollistaa erilaisten syklien luomisen paljon nopeammin ja aiempaa pienemmällä vaivalla.

Carter ja Lapierre (2001) käsittelevät artikkelissaan ensiavun lääkäreiden aika-auluttamista, joka on haastavimpia lääkäreiden aikataulutusesongelmia. Ensiapu on auki 24 tuntia viikon jokaisena päivänä. Kirjoittajat esittelevät kaksi esimerkkitapausta aikataulutussääntöjen kehittämistä Montrealin sairaaloissa.

Charles-Lemoyne Hospital käytti alun perin syklistä vuorotaulua, joka oli sama kaikille, ja sisälsi kokonaisia viikonloppuvapaita vain joka kolmantena viikonloppuna. Uudet vuorotaulut luotiin tietokoneohjelmistolla käyttäen apuna tabuhaku -metaheuristiikkaa. Tuloksena saatiin työvuorolista, jossa oli joka toinen viikonloppu vapaana. Vuorotaulu ei ole enää niin säännöllinen kuin ennen, mutta lääkäreiden tyytyväisyys siihen nousi. Jewish General Hospital käytti alun perin monessa vaiheessa osin manuaalisesti ja osin automaattisesti tehtyä työvuorolistaa, joka ei ollut syklinen. Kolmen kuukauden listan tekemiseen kului noin 40 tuntia työaikaa. Lisäksi tarvittavien tietojen kasaamiseen kului noin 6 viikkoa, sillä lääkärit kävivät töissä epäsäännöllisesti ja suunnittelun aloittamiseen tarvittiin kaikkien lääkäreiden itse kirjaamat vapaapäivä- ja lomatoiveet. Kirjoittajat muokkasivat erästä aiemmin kehitettyä algoritmia vastaamaan sairaalan tarpeita, ja saivat positiivista palautetta sairaalan työvuorolistat tekevältä lääkäriltä.

Knust ja Schumacher (2011) käsittelevät pienen öljy-yhtiön säiliöauton kuljettajien aikataulusongelmaa. Yhtiöllä on erilaisia säiliöautoja ja kuljettajia vaihtelevilla osaamisilla. Tavoitteena on laittaa jokaiseen työvuoroon kuljettaja, jolla on riittävä osaaminen, huomioiden samalla laki- ja turvallisuussäädökset. Kuljettajien työajat pyritään saamaan tietyllä välillä ja heidän loma-aikatoiveitaan pyritään kunnioittamaan. Tehtävän ratkaisemiseksi formuloidaan ensin MIP - tehtävä, joka sisältää 8 kovaa ja 4 pehmeää rajoitetta. Kohdefunktio on pehmeiden rajoitusten rikkomusten painotetun summan minimointi. Malli on niin monimutkainen, että sille ei löydetä optimiratkaisua ilman kalliita kaupallisia ohjelmistoja, joihin yhtiö ei halua panostaa. Täten kirjoittajat päätyivät ratkaisemaan ensin relaxoidun tehtävän, josta on poistettu kaksi vähintään tärkeää pehmeää rajoitetta. Lopuksi saatua ratkaisua yritetään parantaa kahden pois jätetyn rajoitteen suhteen. Kehitetyllä menetelmällä saadaan aikaan hyviä ratkaisuja jo 10 minuutin laskenta-ajoilla. Kaupallisella ohjelmistolla (CPLEX) voitiin ratkaista koko ongelma kerralla, ja saatiin parempia tuloksia, mutta laskenta-ajat olivat tuntien pituisia. Joka tapauksessa kehitetty menetelmä on suuri parannus aikaisempaan manuaaliseen suunnitteluun, sillä aiemmin käypää ratkaisua ei edes löydetty ja kuljettajien työn määrät vaihtelivat paljon.

3. Optimointimalli

Muodostetaan työntekijöiden lomien optimointia varten ILP-tehtävä, jonka perusajatuksena on, että työntekijä m ($m = 1, 2, \dots, M$) on tiettyinä viikkona n ($n = 1, 2, \dots, N$) joko lomalla tai töissä. Tämä ilmaistaan binaarisilla päätös-muuttujilla $x_{m,n}$: jos työntekijä m on lomalla viikolla n , $x_{m,n} = 1$ ja jos työntekijä on töissä, $x_{m,n} = 0$. Jokaiselle viikolle n asetetaan maksimimäärä lomalla oleville työntekijöille t_n .

Työntekijät ovat työsuhteensa pituuden perusteella joko pitkälomalaisia tai lyhytlomalaisia. Olkoon pitkälomalaisten määrä p ja lyhytlomalaisten määrä $l = M - p$. Indeksoidaan työntekijät siten, että työntekijät $1, 2, \dots, p$ ovat pitkälomalaisia, ja työntekijät $p + 1, p + 2, \dots, M$ ovat lyhytlomalaisia. Pitkälomalaisilla on v_p lomaviikkoa vuodessa (tai muussa suunniteltavassa ajanjaksossa) ja lyhytlomalaisilla on v_l lomaviikkoa. Nämä voidaan kirjoittaa rajoituksina:

$$\sum_{n=1}^N x_{m,n} = v_p, \quad m = 1, 2, \dots, p \quad (1)$$

$$\sum_{n=1}^N x_{m,n} = v_l, \quad m = p + 1, p + 2, \dots, M \quad (2)$$

Viikolla n lomalla voi olla korkeintaan t_n työntekijää. Tästä saadaan rajoitus-ehdot:

$$\sum_{m=1}^M x_{m,n} \leq t_n, \quad n = 1, 2, \dots, N \quad (3)$$

Kun merkitään, että viikolla n pitämättä jääneet mahdolliset lomat ovat s_n , saadaan rajoitukset yhtälömuotoon:

$$s_n = t_n - \sum_{m=1}^M x_{m,n}, \quad n = 1, 2, \dots, N \quad (4)$$

Vuosi jaetaan kolmeen lomanpitokauteen: kesälomakausi, kesää edeltävä aika ja kesän jälkeinen aika. Kausien määrittämistä varten asetetaan kesälomakauden ensimmäiseksi viikoksi k_e ja kesälomakauden viimeiseksi viikoksi k_v . Jokaisella työntekijällä tulee olla tasan q_k viikkoa lomaa kesälomakauden aikana.

$$\sum_{n=k_e}^{k_v} x_{m,n} = q_k, \quad m = 1, 2, \dots, M \quad (5)$$

Asetetaan minimeiksi ennen kesälomakautta pidettävien lomaviikkojen määrälle pitkälomalaisille q_{ep} ja lyhytlomalaisille q_{el} .

$$\sum_{n=1}^{k_e-1} x_{m,n} \geq q_{ep}, \quad m = 1, 2, \dots, p \quad (6)$$

$$\sum_{n=1}^{k_e-1} x_{m,n} \geq q_{el}, \quad m = p + 1, p + 2, \dots, M \quad (7)$$

Vastaavasti asetetaan minimeiksi kesälomakauden jälkeen pidettävien lomaviikkojen määrälle pitkälomalaisille q_{jp} ja lyhytlomalaisille q_{jl} .

$$\sum_{n=k_v+1}^N x_{m,n} \geq q_{jp}, \quad m = 1, 2, \dots, p \quad (8)$$

$$\sum_{n=k_v+1}^N x_{m,n} \geq q_{jl}, \quad m = p + 1, p + 2, \dots, M \quad (9)$$

Yhtenä mallin tavoitteena on mahdollistaa sellaisten ratkaisujen luominen, joissa lomat ovat mahdollisimman yhtäjaksoisia. Kunkin lomanpitokauden sisällä työntekijän kaikkien lomien tulee olla peräkkäin, eli työntekijä saa jäädä lomalle vain yhden kerran kauden aikana. Lomalle jäämistä indikoimaan luodaan jokaiselle päätösmuuttujalle $x_{m,n}$ binaarinen apumuuttuja $y_{m,n}$, joka ilmaisee, alkaako työntekijällä m loma viikolla n . Merkitään lomanpitokauden ensimmäistä viikkoa i :llä ja viimeistä viikkoa i_v :llä. Taulukkoon 1 on kerätty i :n ja i_v :n arvot kullekin lomanpitokaudelle.

Taulukko 1: Lomanpitokausien ensimmäiset ja viimeiset viikot.

Lomanpitokausi	i	i_v
Ennen kesää	1	$k_e - 1$
Kesä	k_e	k_v
Kesän jälkeen	$k_v + 1$	N

Kuhunkin lomanpitokauteen luodaan seuraavat rajoitusehdot jokaiselle työntekijälle:

$$x_{m,i} - y_{m,i} = 0, \quad m = 1, 2, \dots, M \quad (10)$$

$$x_{m,n-1} + y_{m,n} \leq 1, \quad m = 1, 2, \dots, M, \quad n = i + 1, i + 2, \dots, i_v \quad (11)$$

$$-x_{m,n} + y_{m,n} \leq 0, \quad m = 1, 2, \dots, M, \quad n = i + 1, i + 2, \dots, i_v \quad (12)$$

$$x_{m,n-1} - x_{m,n} + y_{m,n} \geq 0, \quad m = 1, 2, \dots, M, \quad n = i + 1, i + 2, \dots, i_v \quad (13)$$

Testataan rajoitusehtojen toiminta kaikissa neljässä eri $x_{m,n-1}$ ja $x_{m,n}$ arvot sisältävässä tapauksessa.

i) $x_{m,n-1} = 0, \quad x_{m,n} = 0$

Rajoitusehdosta (12) saadaan $0 + y_{m,n} \leq 0$, joten $y_{m,n} = 0$.

- ii) $x_{m,n-1} = 0, x_{m,n} = 1$
Rajoitusehdosta (13) saadaan $0 - 1 + y_{m,n} \geq 0$, joten $y_{m,n} = 1$.
- iii) $x_{m,n-1} = 1, x_{m,n} = 0$
Rajoitusehdosta (11) saadaan $1 + y_{m,n} \leq 1$, joten $y_{m,n} = 0$.
- iv) $x_{m,n-1} = 1, x_{m,n} = 1$
Rajoitusehdosta (11) saadaan $1 + y_{m,n} \leq 1$, joten $y_{m,n} = 0$.

Lisäksi jos työntekijällä m on lomanpitokauden ensimmäisellä viikolla loma ($x_{m,i} = 1$), niin vastaava apumuuttuja $y_{m,i} = x_{m,i} = 1$ yhtälön (10) perusteella. Yhtälöiden (10) – (13) avulla saadaan siis apumuuttujat $y_{m,n}$ toteuttamaan:

$$\begin{cases} y_{m,n} = 1, & x_{m,n}: \text{stä alkaa loma } (x_{m,i} = 1 \text{ tai } x_{m,n-1} = 0, x_{m,n} = 1) \\ y_{m,n} = 0, & \text{muulloin} \end{cases} \quad (14)$$

Tehdään apumuuttujien avulla rajoitusehdot, jotka mahdollistavat lomalle jäämisen vain kerran kunkin lomanpitokauden aikana (i ja i_v arvot eri lomanpitokausille on esitetty taulukossa 1).

$$\sum_{n=i}^{i_v} y_{m,n} = 1, \quad m = 1, 2, \dots, M \quad (15)$$

Lomallejäämiskertoihin perustuva laskentatapa osoittautui käytännön testeissä laskennallisesti liian raskaaksi joissakin tapauksissa. Tämän vuoksi esitellään vielä vaihtoehtoinen tapa, jolla voidaan pakottaa ratkaisuihin kullekin työntekijälle kesälomat yhtenäisiksi, kun kesälomakaudelle määrätty lomaviikkojen määrä on $q_k = 3$. Luodaan rajoitteet:

$$x_{m,n} - x_{m,n+1} \leq 0, \quad m = 1, 2, \dots, M, \quad n = k_e \quad (16)$$

$$\begin{aligned} -x_{m,n-1} + x_{m,n} - x_{m,n+1} &\leq 0, & m = 1, 2, \dots, M, \\ n &= k_e + 1, k_e + 2, \dots, k_v - 1 \end{aligned} \quad (17)$$

$$-x_{m,n-1} + x_{m,n} \leq 0, \quad m = 1, 2, \dots, M, \quad n = k_v \quad (18)$$

Rajoitteiden (16) – (18) avulla estetään yksittäiset lomaviikot. Rajoite (16) on kesälomakauden ensimmäistä viikkoa varten ja rajoite (18) kesälomakauden viimeistä viikkoa varten. Rajoite (17) estää yksittäiset lomaviikot kauden ensimmäisen ja viimeisen viikon välissä. Jos työntekijällä olisi yksittäinen lomaviikko kesälomakaudella, rajoite (17) saisi muodon:

$$0 + 1 - 0 \leq 0$$

Mistä saadaan ristiriita, eli rajoite (17) toimii yksittäisten lomaviikkojen estäjänä. Jos työntekijällä olisi jossain vaiheessa kesää kaksi lomaviikkoa peräkkäin, tämä tarkoittaisi, että jossain muussa kohdassa kesää olisi yksittäinen lomaviikko, mikä ei ole mahdollista aiemman nojalla. Vastaavat päätelmät voidaan tehdä myös rajoitteita (16) ja (18) käyttäen. Täten rajoitteet (16) – (18) pakottavat kaikki kesälomat peräkkäin, kun kesäloman määrä on kolme viikkoa.

Kohdefunktio

Muuttuja s_n kuvaa, kuinka monta työntekijää enemmän voisi olla lomalla viikolla n . Etsimällä ratkaisua, jolla pienin luvuista s_n on mahdollisimman suuri, saadaan aikaan yllätyksille, kuten sairaslomille, robusti ratkaisu.

Kohdefunktio voidaan kirjoittaa muodossa:

$$\max f, \quad \text{s. e.} \quad f \leq s_n = t_n - \sum_{m=1}^M x_{m,n}, \quad n = 1, 2, \dots, N \quad (19)$$

Lomien jakautumista voidaan tasoittaa vielä enemmän, jos käytetään s_n :lle ylärajaa s_{max} .

4. Lomien suunnitteluongelmia

Ongelmien ratkaisemiseen on käytössä kolme tapaa esiteltyä mallia käytettäessä. Ensimmäisessä tavassa lomien yhtenäisyyteen liittyviä rajoituksia ei käytetä. Toisessa tavassa otetaan käyttöön rajoitteet (16) – (18), jotta saadaan pakotettua kesälomat yhtenäisiksi. Kolmannessa tavassa otetaan käyttöön apumuuttujat $y_{m,n}$ ja rajoitteet (10) – (15). Näin saadaan jokaisen lomanpitokauden sisällä lomat yhtenäisiksi, riippumatta lomien pituuksista.

Ratkaisemiseen käytetään tietokonetta, jossa on Intel Core i5-4210U prosessori, 8 GB muistia ja 64 bittinen käyttöjärjestelmä. Isoja tehtäviä (kuten tässä luvussa

esitettävä 83 työntekijän tehtävä) ratkottaessa on hyvä käyttää 64 bittistä käyttöjärjestelmää, jotta muisti ei lopu kesken. Malli formuloitiin MATLAB -ohjelmistolla, ja ratkaisemisessa käytettiin `lp_solve` -lisäpakettia.

4.1. 10 työntekijän tehtävä

Testataan mallin toimintaa ensin kymmenen työntekijän tehtävässä. Käytetään seuraavia parametreja:

$$\begin{aligned} n &= 52 \\ m &= 10; \\ p &= 2 \\ s_{max} &= 2 \\ k_e &= 22 \\ k_v &= 36 \\ q_k &= 3 \\ q_{ep} &= q_{el} = q_{jp} = q_{jl} = 1 \\ \begin{cases} t_n = 4, & n = k_e, k_e + 1, \dots, k_v \\ t_n = 2, & \text{muulloin} \end{cases} \end{aligned}$$

Tehtävä ratkeaa kaikilla kolmella tavalla nopeasti. Tulokset on esitetty kuvissa 1 - 3. Kuvien taulukoissa rivit vastaavat työntekijöitä ja sarakkeet viikkoja. Jokaisen sarakkeen viimeisenä alkiona on kyseistä viikkoa vastaava s_n , eli ylimääneiden lomaviikkojen määrä kyseisellä viikolla. Kohdefunktion arvo on pienin näistä arvoista. Jotta lomaviikot jakautuisivat mahdollisimman tasaisesti, suurimman ja pienimmän s_n :n välisen erotuksen olisi hyvä olla mahdollisimman pieni. Tätä voidaan kontrolloida säätämällä s_{max} -parametria.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	2	2	2	1	1	2	2	1	1	2	2	1	1	1	1	2	2	2	2	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	1	1	1	2	1	1	1	1	1	1	1	1	1

Kuva 1: Tulokset, tapa 1.

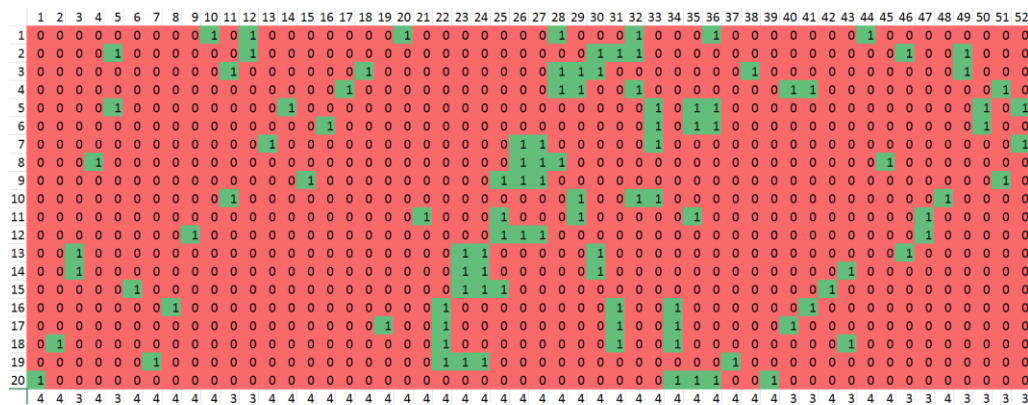
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0		
2	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0		
3	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	1	1	2	2	1	1	2	2	1	2	2	2	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	1	1	2	1	1	1	1	1	1	1	2	2		

Laskenta-ajat olivat 0,035 s, 0,57 s, ja 1,58 s, joten ensimmäisellä tavalla tehtävä ratkeaa noin 15 kertaa nopeammin kuin toisella tavalla ja noin 45 kertaa nopeammin, kuin kolmannella tavalla. Tulosten perusteella malli vaikuttaa toimivalta: kuvassa 2 kaikki kesälomat ovat yhtäjaksoisia ja kuvassa 3 kaikki lomat ovat yhtäjaksoisia lomanpitokausien sisällä. Kaikissa tapauksissa saadaan kohdefunktion arvoksi 1.

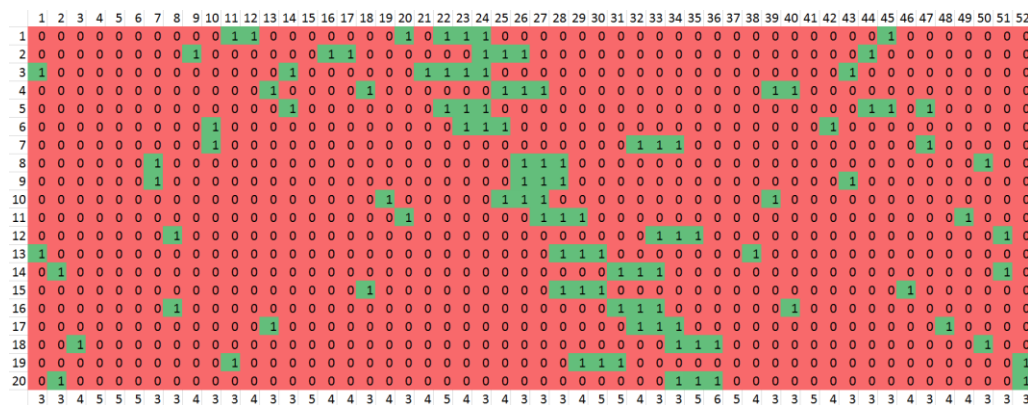
Luodaan seuraavaksi 20 työntekijän tehtävä. Käytetään seuraavia parametreja:

$$\{t_n = 5, \quad \text{muulloin}$$

Ratkaistaan tehtävä kaikilla kolmella tavalla. Tulokset on esitetty kuvissa 4 – 6. Kuvat vahvistavat, että malli toimii oikein käytettäessä kutakin ratkaisutapaa. Tapoja 2 ja 3 käytettäessä jouduttiin asettamaan $s_{max} = 6$, eli löysäämään ylimääräisten vapaapäivien ylärajan rajoitusta, jotta ratkaisut saatiin laskettua järkevässä ajassa (tehtävä ratkesi tavalla 2 myös arvolla $s_{max} = 5$, mutta tähän kului 4806 s, eli noin tunti ja 20 minuuttia). Laskenta-ajoiksi saatiin 0,0564 s, 0,1985 s ja 6,9284 s. Nyt ensimmäinen tapa oli vain noin 4 kertaa nopeampi kuin toinen tapa, mutta peräti 120 kertaa nopeampi kuin kolmas tapa.



Kuva 4: Ratkaisu, tapa 1, $S_{max} = 4$



Kuva 5: Ratkaisu, tapa 2, $S_{max} = 6$

4.3. 83 työntekijän tehtävä

$$\begin{aligned} n &= 52; \\ m &= 83; \\ p &= 21; \\ s_{max} &= 7; \\ k_e &= 22; \\ k_v &= 36; \\ q_k &= 3; \\ q_{ep} &= q_{el} = q_{jp} = q_{jl} = 1 \\ \begin{cases} t_n = 22, & n = kesae, kesae + 1, \dots, kesav \\ t_n = 12, & muulloin \end{cases} \end{aligned}$$

Tehtävä ratkeaa ensimmäisellä tavalla 0,87 sekunnissa. Tulokset on esitetty kuvassa 7. Toisella tavalla tehtävä ei ratkennut 8 tunnissa, vaikka parametrin s_{max} arvo nostettiin äärettömäksi. Myös kolmannella tavalla ratkaisua ei saatu 8 tunnissa.

[illegible]

Kuva 7: Tulokset, tapa 1

5. Tulosten analysointi

Erikoistyyssä kehitettiin ILP-malli työntekijöiden lomien suunnittelua varten. Malliin sisällytettiin kolme erilaista laskentatapaa. Ensimmäinen tapa ei ottanut ollenkaan huomioon lomaviikkojen peräkkäisyyttä. Sen sijaan toinen tapa pakotti kesälomaviikot peräkkäin, ja kolmas tapa pakotti lomat peräkkäin kaikkien lomanpitokausien sisällä. Luvussa 4 laskettujen esimerkkitehtävien perusteella malli on toimiva.

Tehtävän koon kasvaessa laskenta-aika kasvaa nopeasti. Kymmenen työntekijän tehtävässä saavutettiin optimiratkaisu alle kahdessa sekunnissa jokaisella laskentatavalla. Kun tehtävän kokoa kasvatettiin 20 henkilöön, jouduttiin löysäämään ylimääräisten vapaaviikkojen ylärajan määräävää parametria, jotta ratkaisut saatiin toisella ja kolmannella laskentatavalla järkevässä ajassa. Kun tehtävän koko kasvatettiin 83 henkilöön, ratkaisu saatiin edelleen alle sekunnissa ensimmäisellä tavalla, mutta muilla tavoilla ratkaisua ei saatu ollenkaan, kenties siksi että käypää ratkaisua oli vaikea löytää.

Yksi keino saada isompiinkin tehtäviin lomiam peräkkäin on pilkkoa tehtävä osiin. Esimerkiksi pilkkomalla tehtävän kahteen osaan, kesäaika ja muu vuosi, ratkaisuja voisi saada huomattavasti nopeammin kuin kokonaisessa tehtävässä. Tämä olisi erityisen tehokasta ratkaisutapaa 2 käytettäessä, sillä siinä lomien peräkkäisyys ehdot koskevat vain kesäaikaa. Näin kesäaikaa koskevat rajoitukset eivät olisi häiritsemässä muun ajan ratkaisun tekemistä, ja kesäajan ratkaiseminenkin olisi todennäköisesti nopeampaa irrallaan muusta vuodesta. Tehtävä voitaisiin pilkkoa myös esimerkiksi 10 työntekijän palasiin, sillä 10 työntekijän tehtävät vaikuttivat ratkeavan nopeasti kaikilla tavoilla. Tässä muodostuisi kuitenkin ongelmaksi se, että jokaiselle 10 työntekijän tehtävälle tulisi erikseen omat $s_n:t$, joten tällä tavalla ei todennäköisesti saavutettaisi optimaalista ratkaisua koko tehtävälle. Isojen tehtävien ratkaisemisesta voisi auttaa myös erilaisten heuristiikkojen käyttö. Niiden avulla voitaisiin saada lähes optimaalisia ratkaisuja käsiteltävikin isommille tehtäville.

Lähteet

- [1] Baker, K.R. & Magazine, M.J. "Workforce Scheduling with Cyclic Demands and Day-Off Constraints" 1977, *Management Science*, vol.24, no.2, pp. 161-167.
- [2] Bartholdi, J., Orlin, J. & Ratliff, H. 1980, "Cyclic Scheduling via Integer Programs with Circular Ones. *Operations Research*, Vol. 28, No. 5, pp. 1074-1085.
- [3] Beaumont, N. 1997, "Using Mixed Integer Programming to Design Employee rosters" *The Journal of the Operational Research Society*, vol. 48, no. 6, pp. 585–590.
- [4] Burns, R. 1978, "Manpower Scheduling with Variable Demand and Alternate Weekends off", *INFOR*, vol. 16. no. 2, pp. 101-111.
- [5] Carter, M.W. & Lapierre, S.D. 2001, "Scheduling emergency room physicians" *Health care management science*, vol. 4, no. 4, pp. 347-60.
- [6] Chong, P.S. & Strevell, M.W. 1985, "A vacation scheduling algorithm for military flight crews: Maximizing satisfaction while maintaining military preparedness", *Journal of Operations Management*, vol. 5, no. 2, pp. 205-211.
- [7] Knust, S. & Schumacher, E. 2011, "Shift scheduling for tank trucks", *Omega -international Journal Of Management Science*, vol.39, no. 5, pp. 513-521.
- [8] Morris, J. and Showalter, M. 1983. "Simple Approaches to Shift, Days-off and Tour Scheduling Problems." *Management Science*, vol.29, no.8, pp. 942–950.