Tampereen yliopisto

Vesa Rantanen

SÄHKÖISTEN MATEMATIIKAN TENTTIEN KEHITTÄMINEN TAMPEREEN KORKEAKOULUYHTEISÖSSÄ

Opiskelijoiden kokemukset sähköisistä matematiikan tenteistä ja piirtoalustoista sekä satunnaisten matriisien luominen

Diplomityö
Tekniikan ja luonnontieteiden tiedekunta
Tarkastajat: Simo Ali-Löytty
Ulla Miekkala
Syyskuu 2023

TIIVISTELMÄ

Vesa Rantanen: Sähköisten matematiikan tenttien kehittäminen Tampereen korkeakouluyhteisössä
Diplomityö
Tampereen yliopisto
Teknis-luonnontieteellinen DI-ohjelma
Syyskuu 2023

Tutkielman tarkoituksena on matematiikan sähköisten tenttien kehittäminen. Tätä varten suoritettiin tutkimus, jonka tavoitteena oli selvittää Tampereen ammattikorkeakoulun opiskelijoiden mielipiteitä sähköisestä matematiikan tentistä sekä piirtoalustoista tentin apuvälineinä. Lisäksi luotiin MATLAB-algoritmi, jonka avulla on mahdollista luoda satunnaisia neliömatriiseja esimerkiksi matematiikan sähköisissä tenteissä käytettäväksi.

Digitalisaation sekä sähköisten ylioppilaskirjoitusten myötä sähköisten tenttien yleisyys on lisääntynyt korkeakouluissa. Sähköisten tenttien hyötynä on niiden joustavuus sekä aikataulullisesti että tehtävätyyppien osalta. Tampereen korkeakouluissa sähköiset tentit järjestetään TUNI EXAM -palvelussa. Tampereen ammattikorkeakoulussa matematiikan opetukseen ei liity matematiikan kirjoittaminen ohjelmistoilla, joten sähköisten matematiikan tenttien järjestäminen koulussa ei ole ollut mahdollista. Tampereen korkeakouluyhteisö pilotoi piirtonäyttöjä TUNI EXAM -luokissa. Piirtonäyttöjen avulla tenttiin vastaaminen on samanlaista kuin paperitentissä ja matematiikan kirjoittaminen kaavaeditorilla ei ole välttämätöntä.

Tutkimuksessa Tampereen ammattikorkeakoulun opiskelijat suorittivat sähköisen tentin, jossa piirtoalustaa pystyi hyödyntämään. Tutkimusta varten sähköiseen tenttiin luotiin ohjeistus opiskelijoita varten, jotta tentin tekeminen olisi mahdollisimman helppoa. Tutkimuksen aineisto kerättiin tentin jälkeen täytettävällä kyselyllä, jossa tiedusteltiin opiskelijoiden koulutustaustaa, kokemuksia sähköisestä tentistä ja heitä pyydettiin vertailemaan sähköistä tenttiä paperitenttiin. Tutkimukseen osallistui 16 oppilasta. Saatuja tutkimusaineistoja analysoitiin laadullisesti. Tutkimuksesta selvisi, että opiskelijat suhtautuivat suurelta osin sähköiseen tenttiin sekä piirtoalustan käyttämiseen myönteisesti, vaikka he kokivat joitakin ongelmia tentin aikana. Opiskelijat antoivat myös paljon palautetta ja erilaisia parannusehdotuksia sähköisen tentin suorittamiseen.

Tutkielmassa esiteltiin sähköisten tenttien kehittämistä varten luotu MATLAB-algoritmi. Algoritmin avulla on mahdollista tuottaa satunnaisia neliömatriiseja halutuin ehdoin, joita ovat esimerkiksi arvojen suuruus ja tiettyjen arvojen lukumäärä matriisissa. Algoritmi perustuu kokonaislukuoptimointiin ja sitä voidaan pitää erityisen hyödyllisenä työkaluna matematiikan sähköisten tenttikysymysten ja harjoitustehtävien luomisessa.

Avainsanat: sähköinen tentti, kokonaislukuoptimointi, piirtoalusta, satunnaiset matriisit

ABSTRACT

Vesa Rantanen: The development of electronic mathematics exams in Tampere Universities
Master of Science Thesis
Tampere University
Master's Programme in Science and Engineering
September 2023

The purpose of this thesis is the development of electronic mathematics exams. For this, a study was conducted, whose aim was to examine the opinions of the students of Tampere University of Applied Sciences about electronic mathematics exams and drawing tablets as tools in these exams. Additionally, a MATLAB-algorithm that can be used to produce random square matrices for electronic exams and exercises, was created.

Because of digitization and digital matriculation examinations, the amount of electronic exams has increased in universities. The benefit of electronic exams is their flexibility, both in terms of schedule and the types of questions that can be asked. Electronic exams are organized in TUNI EXAM service. In Tampere University of Applied Sciences writing mathematical notations on software is not a part of mathematics courses hence organizing electronic exams hasn't been possible. Tampere Universities are piloting drawing tablets in TUNI EXAM classes. With the help of drawing tables, answering in electronic exams is similar to paper-and-pencil exams and writing mathematical notations on software is not necessary.

In the study, students of Tampere University of Applied Sciences took an electronic exam, where it was possible to utilize a drawing tablet. Electronic exam instructions were made for the study to make the exam performance as easy as possible for the students. Research material of the study was gathered after the exam with an inquiry, in which students were asked about their educational background, experience of the electronic exam and they were asked to compare electronic and paper-and-pencil exams. 16 students participated in the study. The research material was analyzed qualitatively. The study revealed that the students were mostly positive on electronic exam and using a drawing tablet, even though they faced some problems during the exam. The students gave a lot of feedback and improvement suggestions about the exam performance.

A MATLAB algorithm was created for the development of electronic exams. With the algorithm, it is possible to manufacture random matrices with some desired conditions which are, for instance, the size of the values and the number of certain values in the matrix. The algorithm is based on integer programming, and it can be particularly beneficial tool for mathematics exams and exercises.

Keywords: electronic exam, integer programming, drawing tablet, random matrices

ALKUSANAT

Diplomityöni oli yli vuoden kestänyt projekti, joka eteni vaihtelevaa tahtia. Työ oli mielenkiintoinen ja opettavainen, joskin osaltaan myös turhauttava ja stressaava. Olen tyytyväinen lopputulokseen ja itseeni koko tutkinnon suorittamisesta.

Työn ohjaamisesta ja yhteistyöstä haluan kiittää ohjaajiani Simo Ali-Löyttyä, Ulla Miekkalaa sekä kaikkia työssä esiteltävään tutkimukseen osallistuneita opiskelijoita. Erityiset kiitokset kuuluvat perheelleni ja ystävilleni, jotka ovat tukeneet minua koko opintojeni ajan. Lisäksi annan ison kiitoksen kaikille minua ajan saatossa opettaneille, jotka antoivat avaimia tämän tutkinnon suorittamiseen.

Tampereella, 27. syyskuuta 2023

Vesa Rantanen

SISÄLLYSLUETTELO

LYHENTEET JA MERKINNÄT

0_n	$n \times n$ -nollamatriisi
$0_{m\times n}$	$m \times n$ -nollamatriisi
$\lfloor a \rfloor$	Luvun a lattiafunktio
A_{eq}	Kerroinmatriisi rajoitteen ollessa yhtälöryhmä (Luku 4)
B	Lineaarisen optimointiongelman kanta
b_{eq}	Vakiovektori rajoitteen ollessa yhtälöryhmä (Luku 4)
$\mathbf{f}$	Kohdevektori
I_n	$n \times n$ -identiteettimatriisi
$J_{m,n}$	$m \times n$ -ykkösmatriisi
$\mathbf{lb}$	Muuttujien alarajavektori
$\min_+$	Positiivinen minimi
$\mathbb{R}$	Reaalilukujen joukko
TAMK	Tampereen ammattikorkeakoulu
TUNI	Tampereen korkeakouluyhteisö (engl. Tampere Universities)
$\mathbf{ub}$	Muuttujien ylärajavektori
$\mathbf{x}_B$	Lineaarisen optimointiongelman kantavektori
$\mathbf{x}_{LP}$	Relaksoidun optimointiongelman optimaalinen ratkaisu
$\mathbb{Z}$	Kokonaislukujen joukko
Ω	Käypä joukko

1. JOHDANTO

Tutkielman tavoitteena on sähköisten tenttien kehittäminen. Tutkielmassa esitellään tutkimus, jonka tarkoituksena oli selvittää, millaiseksi Tampereen ammattikorkeakoulun (TAMK) tekniikan opiskelijat kokevat matematiikan tenttien suorittamisen sähköisesti. Sähköisissä tenteissä on pilotoitavana piirtoalustat, joiden hyödyntäminen on tärkeää TAMK:n matematiikan tenttien kannalta, sillä näissä opinnoissa ei harjoitella matemaattisten notaatioiden kirjoittamista kaavaeditorien avulla.

Matematiikan opetuksessa proseduurien osaamista harjoitellaan ja testataan eri lukuarvojen avulla. Näin ollen on tärkeää pystyä luomaan lukuarvoiltaan satunnaistettuja tehtäviä. Satunnaistetut tehtävät ovat hyvä keino myös plagioinnin ehkäisemiseen. Tehtävien satunnaistamisessa on otettava huomioon, mitä osaamisaluetta tehtävän halutaan mittaavan. Toisaalta on varmistettava, että satunnaistetut tehtävät ovat vaikeustasoltaan toisiinsa verrattavissa.

Tutkielman luvussa 2 esitellään lineaarisen optimoinnin ja kokonaislukuoptimoinnin ratkaisumenetelmiä. Luvussa esitellään lineaarisen optimoinnin peruskäsitteitä, erilaisia simplexmenetelmiä lineaaristen optimointiongelmien ratkaisemiseen sekä menetelmiä, joiden avulla on mahdollista ratkaista kokonaislukuongelmia. Menetelmien esittely työssä ei ole syvällistä, eikä kaikkien menetelmien vaiheiden syitä tai toimivuutta erikseen todisteta.

Luku 3 käsittelee sähköisiä tenttejä ja matematiikan opettamista pedagogisesta näkökulmasta. Sähköisten tenttien hyötyjä esitellään niin yleisesti kuin Tampereen korkeakouluyhteisön (TUNI) näkökulmasta. Luvussa kerrotaan automaattitarkisteisista STACK-tehtävistä ja satunnaistettujen tehtävien hyödyistä. Lisäksi luvussa vertaillaan suppeasti formatiivista ja summatiivista arviointia, sekä proseduraalista ja konseptuaalista tietoa matematiikassa.

Luvussa 4 tarkastelun aiheena on MATLAB-algoritmi, jonka avulla on mahdollista luoda satunnaisia $n \times n$ -matriiseja. Algoritmin toiminta perustuu lineaariseen kokonaislukuoptimointiongelmaan. Luvussa esitellään, miten algoritmi on rakennettu ja miten erilaisia ehtoja on mahdollista luoda. Luku sisältää myös teoriaa MATLAB:n sisäisestä kokonaislukuoptimointikomennosta. Luvun lopussa myös havainnollistetaan algoritmin toimintaa ja sen riippuvuutta eri parametreista esimerkkiajojen avulla.

Luvussa 5 esitellään tutkielman varsinainen tutkimus. Luvussa kerrotaan aineiston keräämisestä, piirtoalustan käyttöön laaditusta ohjeistuksesta ja tutkimuksessa tehdyn kyselyn tulokset esitellään. Luvussa 6 puolestaan pohditaan tutkimuksen tuloksia sekä luotettavuutta.

2. LINEAARISEN KOKONAISLUKUOPTIMOINNIN MENETELMIÄ

Tässä luvussa esitellään erilaisia menetelmiä lineaariseen optimointiin ja kokonaislukuoptimointiin. Luvussa esitellään tärkeimpiä lauseita ja todistuksia, mutta menetelmien kaikkien eri vaiheiden merkityksiä ei avata. Luvussa menetelmät ja notaatiot on esitetty suureksi osaksi Bernard Kolmanin teoksen "Elementary linear programming with applications" perusteella.

Optimointi on matematiikan osa-alue, jossa halutaan minimoida tai maksimoida jonkin funktion arvo. Optimointiongelman yksinkertaisin yleinen muoto on

$$\min_{\mathbf{x}\in\Omega} f(\mathbf{x}),$$

jossa $f : \mathbb{R}^n \to \mathbb{R}$ ja $\Omega \subset \mathbb{R}^n$. Tavoitteena on siis minimoida funktio f, jota kutsutaan kohdefunktioksi muuttujavektorin $\mathbf{x}$ kuuluessa käypään joukkoon Ω. Käypä joukko koostuu kaikista ongelman mahdollisista eli käyvistä ratkaisuista. Minimointi tapahtuu löytämällä optimaalinen ratkaisu $\mathbf{x}^*$, jolla kohdefunktio saa minimaalisen eli optimaalisen arvon. Optimointiongelmissa esiintyy tiettyjä ehtoja muuttujien arvoille ja niiden välisille suuruuksille. Optimointiongelma on usein muotoa

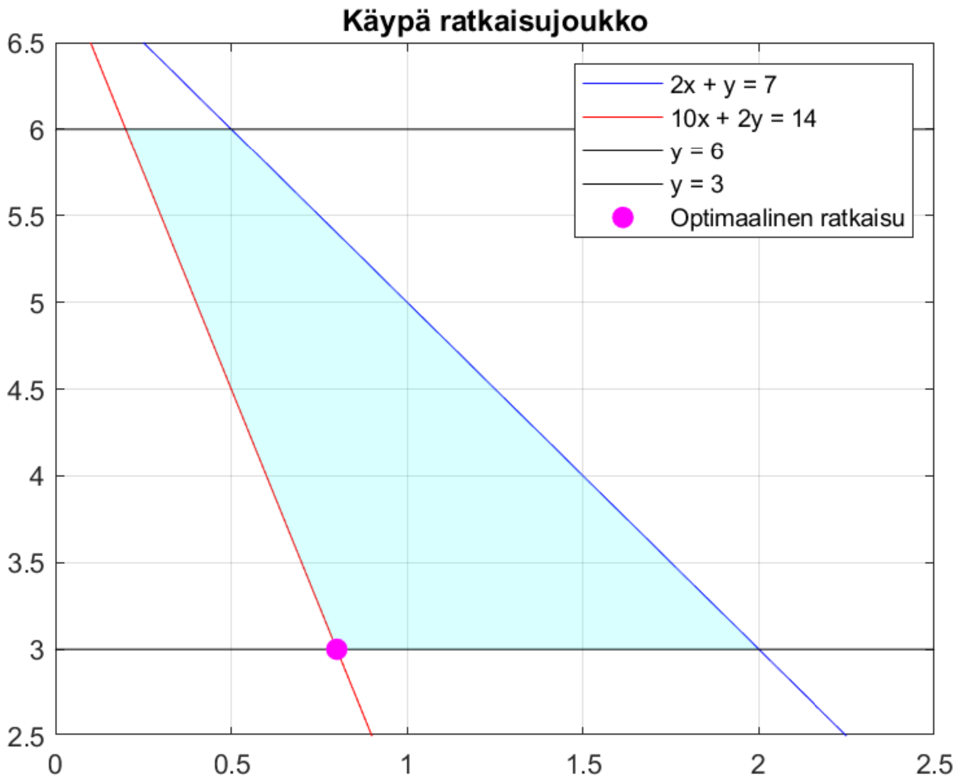
$$\begin{aligned}
&\min z = f(\mathbf{x})\\
&g_1(\mathbf{x}) = 0\\
&g_2(\mathbf{x}) \geq 0\\
&\qquad\vdots
\end{aligned}$$

jossa $g_i(\mathbf{x})$ on muuttujasta $\mathbf{x}$ riippuva polynomi. Yleensä optimointiongelmissa esiintyy useampi kuin yksi muuttuja. Tällöin rajoitteet g_i voivat riippua kaikista näistä muuttujista.

Esimerkki 2.1. Olkoon lineaarinen optimointiongelma seuraava:

$$\begin{aligned} \min z &= 5x + 3y \\ 2x + y &\leq 7 \\ 10x + 2y &\geq 14 \\ 3 \geq y &\geq 6 \end{aligned}$$

Tavoitteena on löytää muuttujien x ja y käyvät arvot, joilla kohdefunktio saavuttaa miniminsä. Kuvaan 2.1 on piirretty optimointiongelman rajoitteiden mukainen käypä ratkaisujoukko. Käyvästä joukosta löytyy kohdefunktion optimaalinen arvo $z = 13$, joka saavutetaan pisteessä $(0{,}8; 3)$.



Kuva 2.1. *Esimerkin 2.1 käypä ratkaisujoukko ja optimaalinen ratkaisu visualisoituna*

2.1 Lineaarisen optimoinnin peruskäsitteitä ja menetelmiä

Lineaarisessa optimoinnissa sekä kohdefunktio että ongelman rajoitteina toimivat yhtälöt ovat lineaarisia. Vastaavasti, jos jokin näistä ei olisi lineaarinen, olisi kyse epälineaarisesta optimoinnista. Yleisesti lineaarisessa optimointiongelmassa etsitään sellaisia muuttujien

$x_1, x_2, x_3, \ldots, x_n$ arvoja, jotka minimoivat yhtälön $z = c_1x_1 + c_2x_2 + \cdots + c_nx_n$ ehdoin

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1m}x_n &\le b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2m}x_n &\le b_2 \\ &\vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nm}x_n &\le b_m, \end{aligned}$$

joissa voi esiintyä myös symbolit $=$ tai $\ge$. Matriisien avulla optimointiongelma saadaan muokattua sen kanoniseen muotoon

$$\min z = \mathbf{c}^T\mathbf{x} \tag{2.1a}$$

$$A\mathbf{x} = \mathbf{b} \tag{2.1b}$$

$$\mathbf{x} \ge \mathbf{0}, \tag{2.1c}$$

jossa $\mathbf{x}, \mathbf{c} \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$ ja $\mathbf{b} \in \mathbb{R}^m$. Lineaariset optimointiongelmat eivät kuitenkaan aina ole tätä muotoa, mutta kaikki ongelmat voidaan muuttaa tähän muotoon.

Lause 2.1. *Lineaarista optimointiongelmaa voidaan muokata seuraavin tavoin:*

1. *Jokainen maksimointiongelma voidaan muuttaa minimointiongelmaksi ja päin vastoin, koska*
$$\max \mathbf{c}^T\mathbf{x} = -\min(-\mathbf{c}^T\mathbf{x}).$$
[18, s. 53]
2. *Epäyhtälö $g(x) \le b$ voidaan muokata pelivaramuuttujan $y \ge 0$ avulla muotoon $g(x) + y = b$. Pelivaramuuttuja lisätään myös kohdefunktioon. [14, s. 14]*
3. *Epäyhtälö $g(x) \ge b$ voidaan muokata ylijäämämuuttujan $y \ge 0$ avulla muotoon $g(x) - y = b$. Ylijäämämuuttuja lisätään myös kohdefunktioon. [14, s. 14]*
4. *Epäyhtälö $x_k \le 0$ voidaan muokata yhtälöksi $y_k \ge 0$ merkitsemällä $y_k = -x_k$. [18, s. 54]*
5. *Epäyhtälö $x_k \ge l$ voidaan muokata yhtälöksi $y_k \ge 0$ merkitsemällä $y_k = x_k - l$. [8, s. 7]*

Tutkitaan kanonisessa muodossa olevaa lineaarista optimointiongelmaa. Oletetaan, että $m \le n$ ja rank$(A) = m$ eli matriisilla A on m lineaarisesti riippumatonta saraketta. Nämä m saraketta muodostavat kääntyvän matriisin $B \in \mathbb{R}^{m\times m}$, jonka avulla voidaan kirjoittaa

$$A = \begin{bmatrix} B & N \end{bmatrix}. \tag{2.2}$$

Matriisi $N \in \mathbb{R}^{m\times(n-m)}$ koostuu puolestaan matriisin B kanssa lineaarisesti riippuvista sarakkeista [18]. Kun tutkitaan yhtälöä $\begin{bmatrix} B & N \end{bmatrix} \mathbf{x} = \mathbf{b}$, saadaan

$$B\mathbf{x}_B + N\mathbf{x}_N = \mathbf{b}. \tag{2.3}$$

Määritelmä 2.1. Lineaarisen optimointiongelman yksikäsitteinen ratkaisu

$$x = \begin{bmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{bmatrix} = \begin{bmatrix} \mathbf{x}_B \\ 0 \end{bmatrix}$$

on *kantaratkaisu*. Matriisia B kutsutaan *kannaksi* ja vektorin $\mathbf{x_B}$ komponentteja x_{Bi} kutsutaan *kantamuuttujiksi*.

Toisin sanoen kantaratkaisussa vain kantamuuttujien arvot voivat olla nollasta poikkeavia. Kantaratkaisu on aina eräs ratkaisu yhtälöön $A\mathbf{x} = \mathbf{b}$, mutta kaikki kantaratkaisut eivät välttämättä toteuta kaikkia ongelman rajoitteita, jolloin ne eivät ole optimointiongelman käypiä ratkaisuja.

Määritelmä 2.2. Lineaarisen optimointiongelman *käyväksi kantaratkaisuksi* kutsutaan kantaratkaisua, joka kuuluu käypään joukkoon Ω.

Määritelmä 2.3. Kahden käyvän kantaratkaisun sanotaan olevan *viereiset* (engl. adjacent), jos niiden kaikki kantamuuttujat yhtä lukuunottamatta ovat samat.

Kantaratkaisuun liittyvä käsite on kärkipiste.

Määritelmä 2.4. Konveksin joukon S pistettä $\mathbf{u}$ kutsutaan *kärkipisteeksi*, jos se ei ole yhdenkään tähän konveksiin joukkoon kuuluvan janan sisäpiste. Piste $\mathbf{u}$ on siis joukon S kärkipiste mikäli ei ole olemassa pisteitä $\mathbf{x}_1, \mathbf{x}_2 \in S$

$$\mathbf{u} = \lambda\mathbf{x}_1 + (1-\lambda)\mathbf{x}_2,$$

kun $0 < \lambda < 1$.

Esimerkiksi kuvassa 2.1 kärkipisteet ovat $(0{,}2; 6)$, $(0{,}5; 6)$, $(0{,}8; 3)$ ja $(2; 3)$.

Lause 2.2. [13] *Jos $\mathbf{x}$ on käyvän joukon Ω kantaratkaisu, se on myös joukon kärkipiste.*

Todistus. [13] Oletetaan, että $\mathbf{x}$ ei ole käyvän joukon Ω kärkipiste. Tällöin on olemassa kaksi erillistä käypää ratkaisua $\mathbf{y}$ ja $\mathbf{z}$ yhtälöihin (2.1a) ja (2.1b), jotka toteuttavat yhtälön

$$\mathbf{x} = \lambda\mathbf{y} + (1-\lambda)\mathbf{z},\ 0 < \lambda < 1.$$

Mikäli $\mathbf{x} = \mathbf{0}$, on oltava $\mathbf{y} = \mathbf{z} = \mathbf{0}$, koska $\mathbf{y}, \mathbf{z} \geq \mathbf{0}$. Näin ollen ratkaisut $\mathbf{y}$ ja $\mathbf{z}$ kuuluvat samaan kantaan kuin $\mathbf{x}$, mutta koska käypä kantaratkaisu on yksikäsitteinen kannassaan, saadaan ristiriita. Piste $\mathbf{x}$ ei siis ole tällöin käypä kantaratkaisu. $\square$

2.1.1 Simplex-menetelmä

Lineaarisen optimoinnin ongelmien ratkaisemiseen on olemassa useita erilaisia menetelmiä, joiden avulla saadaan optimaalinen ratkaisu. Yksi yleisimmistä meneltemistä on George B. Danzigin vuonna 1947 kehittämä simplex-menetelmä. Menetelmä perustuu käypien kantaratkaisujen etsimiseen ja niiden optimaalisuuden testaamiseen.

Olkoon lineaarinen optimointiongelma muotoa

$$
\begin{aligned}
\min z &= -\mathbf{c}^T\mathbf{x} \\
A\mathbf{x} &\leq \mathbf{b} \\
x_i &\geq 0,
\end{aligned}
$$

jossa $\mathbf{x}, \mathbf{c} \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$, $\mathbf{b} \in \mathbb{R}^m$ ja $i = 1, 2, \ldots, n$. Oletetaan vielä, että $\mathbf{b} \geq \mathbf{0}$. Muokataan ongelma Lauseen 2.1 avulla muotoon

$$
\begin{aligned}
\max z &= \mathbf{c}^T\mathbf{x} \\
A\mathbf{x} &= \mathbf{b} \\
x_i &\geq 0,
\end{aligned}
$$

jossa $i = 1, 2, \ldots, n, n+1, \ldots, n+m$. Näistä muuttujista $x_{n+1}, x_{n+2}, \ldots, x_{n+m}$ ovat pelivaramuuttujia. Muokkausten myötä

$$
A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} & 1 & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & a_{2n} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ a_{m1} & a_{12} & \cdots & a_{mn} & 0 & 0 & \cdots & 1 \end{bmatrix} \text{ ja } \mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.
$$

Simplex-menetelmän ensimmäinen vaihe on erään käyvän kantaratkaisun löytäminen. Koska $\mathbf{b} \geq \mathbf{0}$, voidaan asettaa muuttujat $x_1, x_2, \ldots, x_n$ nolliksi, jolloin yhtälöstä $A\mathbf{x} = \mathbf{b}$ saadaan

$$
x_{n+1} = b_1, x_{n+2} = b_2, \ldots, x_{n+m} = b_m,
$$

joka on Määritelmän 2.1 mukaan kantaratkaisu. Ratkaisu on myös käypä, sillä $\mathbf{b} \geq \mathbf{0}$, joten

$$\mathbf{x}^T = \begin{bmatrix} 0 & 0 & \cdots & 0 & b_1 & b_2 & \cdots & b_m \end{bmatrix}$$

on eräs käypä kantaratkaisu. Simplex-menetelmää käsitellään usein taulukkomuodossa. Kun ongelma muutetaan taulukkomuotoon, kohdefunktio kirjoitetaan muotoon

$$-\mathbf{c}^T\mathbf{x} + z = 0.$$

Taulukossa sarakkeet vastaavat muuttujia $x_1, x_2, \ldots, x_n, x_{n+1}, \ldots, x_{n+m}$ ja z. Kirjoitetaan optimointiongelman rajoitteiden ja kohdefunktion mukaiset kertoimet riveittäin taulukkoon:

$$\begin{array}{c|cccccccc|c}
 & x_1 & x_2 & \cdots & x_n & x_{n+1} & x_{n+2} & \cdots & x_{n+m} & z & \\
\hline
x_{n+1} & a_{11} & a_{12} & \cdots & a_{1n} & 1 & 0 & \cdots & 0 & 0 & b_1 \\
x_{n+2} & a_{21} & a_{22} & \cdots & a_{2n} & 0 & 1 & \cdots & 0 & 0 & b_2 \\
\vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\
x_{n+m} & a_{m1} & a_{m2} & \cdots & a_{mn} & 0 & 0 & \cdots & 1 & 0 & b_m \\
\hline
 & -c_1 & -c_2 & \cdots & -c_1 & 0 & 0 & \cdots & 0 & 1 & 0
\end{array} \tag{2.4}$$

Taulukon vasemmanpuoleisin sarake kertoo rivillä esiintyvän yhtälön kantamuuttujan, mistä voidaan päätellä, että jokaisella yhtälöllä on vain yksi kantamuuttuja. Alimmainen rivi puolestaan kuvaa kohdefunktiota ja sitä kutsutaan *kohderiviksi*.

Simplex-menetelmän seuraavassa vaiheessa tutkitaan, onko taulukon mukainen käypä kantaratkaisu optimaalinen. Ratkaisu on optimaalinen, jos kohderivillä kantamuuttujien kertoimet ovat nollia ja muiden muuttujien kertoimet ovat ei-negatiivisia. Mikäli tämä ehto toteutuu, on löydetty optimaalinen ratkaisu ja menetelmä voidaan keskeyttää. Muissa tapauksissa siirrytään menetelmän seuraavaan vaiheeseen.

Menetelmän seuraava vaihe perustuu viereisen käyvän kantaratkaisun löytämiseen, sillä optimaalinen ratkaisu on aina käypä kantaratkaisu. Tarkemmin ajateltuna, halutaan löytää kärkipiste, jolla muuttujan z arvo kasvaa. Tehokkain tapa muuttujan z kasvattamiseen on kasvattaa eniten negatiivista arvoa kohderivillä. Valitaan sarake, jolla on pienin arvo kohderivillä niin kutsutuksi *pivot-sarakkeeksi*. Jos useassa sarakkeessa kohderivillä on sama arvo, voi sarakkeen valita näistä mielivaltaisesti. Pivot-saraketta vastaavaa muuttujaa kutsutaan *tulevaksi muuttujaksi*.

Neljännessä vaiheessa valitaan *lähtevä muuttuja*. Ratkaistaan aluksi taulukon mukaiset yhtälöt kantamuuttujien suhteen kasvattamalla tulevaa muuttujaa x_t ja pitämällä muiden muuttujien arvot nollina, jolloin saadaan

$$x_{Bi} = b_i - a_{it}x_t,$$

jossa $i = 1, 2, \ldots, m$ ja indeksi t kuvaa pivot-saraketta. Muuttuja x_{Bi} kuvaa taulukon vasemman reunan kantavektoreita. Toisaalta tiedetään, että $\mathbf{x} \geq 0$, joten tulevaa muuttujaa voidaan kasvattaa, sillä rajoituksella, että lähtevä muuttuja ei saa negatiivista arvoa. Saadaan

$$\begin{cases} b_1 - a_{1t}x_t \geq 0 \\ b_2 - a_{2t}x_t \geq 0 \\ \vdots \\ b_m - a_{mt}x_t \geq 0 \end{cases} \Leftrightarrow \begin{cases} x_t \leq \frac{b_1}{a_{1t}} \\ x_t \leq \frac{b_2}{a_{2t}} \\ \vdots \\ x_t \leq \frac{b_m}{a_{mt}}. \end{cases}$$

Yksi näistä epäyhtälöistä antaa tiukimman rajoitteen tulevan muuttujan x_t kasvattamiselle. Muuttujan x_t arvoksi asetetaan siis pienin positiivinen suhde eli $\min_{+}\left\{\frac{b_1}{a_{1t}}, \frac{b_2}{a_{2t}}, \ldots, \frac{b_m}{a_{mt}}\right\}$. Mikäli $a_{it} = 0$, kyseistä suhdetta ei ole määritelty, joten sitä ei tarvitse ottaa huomioon suhteita vertailtaessa. Se kantamuuttuja, jonka määräämältä riviltä tulevan muuttujan x_t pienin arvo saadaan, saa uudeksi arvokseen 0 ja sitä kutsutaan *lähteväksi muuttujaksi*. Lähtevää muuttujaa vastaavaa riviä kutsutaan puolestaan *pivot-riviksi*. Mikäli $\frac{b_i}{a_{1i}} \leq 0$ kaikilla $i = 1, 2, \ldots, m$, ei tulevien muuttujien kasvattamisella ole rajoitteita, mistä voidaan päätellä, ettei ongelmalla ei ole äärellistä optimaalista ratkaisua. Kun tulevan kantamuuttujan arvo on määritetty ja tiedetään lähtevän muuttujan arvon olevan 0, tulee muiden kantamuuttujien arvot laskea uudelleen. Näin saadaan uusi käypä kantaratkaisu. Ratkaisun avulla saadaan uusi arvo kohdefunktiolle.

Menetelmän viimeinen vaihe koostuu *pivotoinnista* eli uuden taulukon muodostamisesta. Alkio, joka esiintyy sekä pivot-sarakkeessa että -rivillä, on *pivot-alkio*. Pivotoinnin ensimmäisessä vaiheessa jaetaan koko pivot-rivi pivot-alkion arvolla, jotta pivot alkion arvoksi saadaan 1. Seuraavaksi lisätään muihin riveihin pivot-rivin moninkertoja, jotta pivot-sarakkeen kaikkien muiden alkioiden arvo on 0. Tässä vaiheessa vaihdetaan myös tuleva muuttuja vasemman puoleiseen sarakkeeseen lähtevän muuttujan tilalle. Uudesta taulukosta voidaan tarkistaa jälleen, onko sitä vastaava käypä kantaratkaisu optimaalinen. Jos ratkaisu ei ole optimaalinen, luodaan taas uusi taulukko.

Esimerkki 2.2. Ratkaistaan seuraava ongelma käyttäen simplex-menetelmää:

$$\begin{aligned} \min z = -x_1 - 3x_2 - 5x_3 \\ 2x_1 - 5x_2 + 3x_3 \leq 3 \\ x_1 + 4x_2 \leq 5 \\ x_1, x_2, x_3 \geq 0. \end{aligned}$$

Ongelma tulee muuttaa aluksi kanonisessa muodossa olevaksi maksimointiongelmaksi, jotta simplex-menetelmää voidaan hyödyntää. Saadaan

$$\begin{aligned} \max -z &= x_1 + 3x_2 + 5x_3 \\ 2x_1 - 5x_2 + 3x_3 + x_4 &= 3 \\ x_1 + 4x_2 + x_5 &= 5 \\ x_1, x_2, x_3 &\geq 0. \end{aligned}$$

Kirjoitetaan ongelma taulukkomuotoon:

	x_1	x_2	x_3	x_4	x_5	z	
x_4	2	-5	3	1	0	0	3
x_5	1	4	0	0	1	0	5
	-1	-3	-5	0	0	1	0

Taulukon mukainen käypä kantaratkaisu on $\mathbf{x}^T = \begin{bmatrix} 0 & 0 & 0 & 3 & 5 \end{bmatrix}$, mutta huomataan, ettei tämä ratkaisu ole optimaalinen, sillä kohderivillä on negatiivisia alkioita. Koska -5 on alimman rivin pienin arvo, valikoituu x_3 tulevaksi muuttujaksi. Tämän uudeksi arvoksi määräytyy lukujen b_i ja a_it pienin positiivinen suhde, joten saadaan

$$x_3 = \frac{3}{3} = 1.$$

Koska luku 3 esiintyi kantamuuttujan x_4 osoittamalla rivillä, on x_4 lähtevä muuttuja. Uusi käypä kantaratkaisu on siis $\mathbf{x}^T = \begin{bmatrix} 0 & 0 & 1 & 0 & 5 \end{bmatrix}$. Nyt voidaan muodostaa uusi taulukko pivotoimalla:

	x_1	x_2	x_3	x_4	x_5	z	
x_3	$\frac{2}{3}$	$-\frac{5}{3}$	1	$\frac{1}{3}$	0	0	1
x_5	1	4	0	0	1	0	5
	$\frac{7}{3}$	$-\frac{34}{3}$	0	$\frac{5}{3}$	0	1	5

Pivotoinnissa taulukon vasemman reunan kantamuuttujista toinen vaihtui. Koska kantamuuttujan kerroin on oltava omalla rivillään 1 ja muilla riveillä 0, ylempi rivi on kerrottu luvulla $\frac{1}{3}$ ja se on lisätty kohderiviin kerrottuna luvulla $\frac{5}{3}$. Muuttujan x_5 määräämään riviin ei tarvitse tehdä muutoksia, sillä molempia kantamuuttujia vastaavilla sarakkeilla on tällä rivillä luku 0.

Taulukon kohderiviltä huomataan, ettei tämäkään ole vielä optimaalinen ratkaisu. Määritetään siis jälleen tuleva ja lähtevä muuttuja, jolloin uudeksi taulukoksi saadaan:

	x_1	x_2	x_3	x_4	x_5	z	
x_3	$\frac{13}{12}$	0	1	$\frac{1}{3}$	$\frac{5}{12}$	0	$\frac{37}{12}$
x_2	$\frac{1}{4}$	1	0	0	$\frac{1}{4}$	0	$\frac{5}{4}$
	$\frac{31}{6}$	0	0	$\frac{5}{3}$	$\frac{17}{6}$	1	$\frac{115}{6}$

Taulukon mukaan $\mathbf{x}^T = \begin{bmatrix} 0 & \frac{5}{4} & \frac{37}{12} & 0 & 0 \end{bmatrix}$ on ongelman käypä optimaalinen ratkaisu. Näin ollen kohdefunktion maksimiarvo on $\frac{115}{6}$, joten alkuperäisen minimointiongelman optimaaliseksi arvoksi saadaan $-\frac{115}{6}$.

2.1.2 Dual simplex -menetelmä

Eräs variaatio simplex-menetelmästä on nimeltään dual simplex -menetelmä. Se muistuttaa vaiheiltaan simplex-meneltelmää. Dual simplex -menetelmä perustuu lineaaristen optimointiongelmien kaksijakoisuuteen.

Määritelmä 2.5. Olkoon lineaarinen optimointiongelma muotoa

$$\begin{aligned} \min z &= \mathbf{c}^T\mathbf{x} \\ A\mathbf{x} &\leq \mathbf{b} \\ \mathbf{x} &\geq 0. \end{aligned}$$

Tällöin ongelmaa

$$\begin{aligned} \max z &= \mathbf{b}^T\mathbf{w} \\ A^T\mathbf{w} &\geq \mathbf{c} \\ \mathbf{w} &\geq 0, \end{aligned}$$

kutsutaan alkuperäisen ongelman eli *primaaliongelman duaaliongelmaksi*.

Lause 2.3. *[18, s. 174]*

a) Jos joko primaali- tai duaaliongelmalla on käypä ratkaisu, joka antaa äärellisen optimaalisen arvon kohdefunktiolle, myös toisella ongelmalla on olemassa käypä ratkaisu, joka antaa saman kohdefunktion arvon.

b) Primaaliongelmalla on olemassa optimaalinen ratkaisu $\mathbf{x}_0$ ja duaaliongelmaalla on olemassa optimaalinen ratkaisu $\mathbf{w}_0$, jos sekä primaali- että duaaliongelmalla on käypä ratkaisu. Tämän lisäksi $\mathbf{c}^T\mathbf{x}_0 = \mathbf{b}^T\mathbf{w}_0$.

Todistus. Kts. [18, s. 174] $\square$

Dual simplex -menetelmässä ratkaisun optimaalisuus säilyy eri vaiheiden välillä, mutta nämä ratkaisut eivät kaikki ole käypiä. Tavoitteena on siis saavuttaa optimaalinen ratkaisu, joka on käypä. Tavallisessa simplex-menetelmässä puolestaan etsitään optimaalista ratkaisua käypien ratkaisujen joukosta. Dual simplex -menetelmä on esimerkiksi hyödyllinen tilanteissa, joissa ratkaistuun ongelmaan lisätään uusia rajoitteita, joiden vuoksi vanha ratkaisu ei ole enää käypä. Dual simplex -menetelmän avulla ratkaisun käypyys voidaan tällaisessa tilanteessa palauttaa. [18]

Kuten tavallisessa simplex-menetelmässä, dual simplex -menetelmä voidaan toteuttaa taulukoiden avulla. Dual simplex -taulukon yleinen muoto on seuraava:

		c_1	c_2	$\cdots$	c_n	c_{n+1}	c_{n+2}	$\cdots$	c_{n+m}	
$\mathbf{c}_B$		x_1	x_2	$\cdots$	x_n	x_{n+1}	x_{n+2}	$\cdots$	x_{n+m}	$\mathbf{x}_B$
c_{i1}	x_{n+1}	a_{11}	a_{12}	$\cdots$	a_{1n}	1	0	$\cdots$	0	x_{i1}
c_{i2}	x_{n+2}	a_{21}	a_{22}	$\cdots$	a_{2n}	0	1	$\cdots$	0	x_{i2}
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
c_{im}	x_{n+m}	a_{m1}	a_{m2}	$\cdots$	a_{mn}	0	0	$\cdots$	1	x_{im}
		$-c_1$	$-c_2$	$\cdots$	$-c_1$	0	0	$\cdots$	0	

Taulukko on lähes samanlainen kuin simplex-menetelmässä. Erona on taulukon vasemmassa reunassa oleva vektori $\mathbf{c}_B$ sekä ylin rivi, jossa esiintyy vektorin $\mathbf{c}^T$ alkiot. Vektori $\mathbf{c}_B$ sisältää vektorin $\mathbf{c}^T$ alkiot, jotka vastaavat kantavektoreita. Dual simplex -taulukkoa voidaan käyttää myös simplex-menetelmässä.

Dual simplex -menetelmän voidaan ajatella koostuvan kahdesta osasta. Ensimmäisessä osassa tarkistetaan saadun optimaalisen ratkaisun käypyys. Toisessa osassa puolestaan muokataan ratkaisua lähemmäksi optimaalista käypää ratkaisua.

Menetelmän ensimmäisessä vaiheessa etsitään sellainen initiaalinen kantaratkaisu, joka muodostaa taulukon kohderiville vain ei-negatiivisia arvoja, ja jonka ainakin yhden kantamuuttujan arvo on negatiivinen. Tämän jälkeen valitaan taulukosta lähtevä muuttuja valitsemalla kantamuuttujista negatiivisin. Lähtevän muuttujan rivistä tulee pivot-rivi.

Tuleva muuttuja valitaan kohderivin alkioiden ja niitä vastaavien negatiivisten pivot-rivin alkioiden suhteiden perusteella. Mikäli pivot-rivillä ei ole negatiivisia alkioita, ongelmalla ei ole käypää ratkaisua. Suurinta negatiivista suhdetta vastaava alkio määrää tulevan muuttujan, jota vastaava sarake on pivot-sarake. Jos kaksi tai useampi suhdetta on samat, voi valinnan tehdä näiden välillä mielivaltaisesti.

Pivotointi suoritetaan tämän jälkeen samalla tavalla kuin simpex-menetelmässä, mikä johtaa uuteen taulukkoon. Kohderivi voidaan laskea

$$z_j - c_j = \mathbf{c}_B^T \mathbf{t}_j - c_j,$$

jossa yhtälön vasen puoli vastaa kohderivin alkioiden arvoja ja $\mathbf{t}_j$ on j:nnes sarake.

Optimaalinen käypä ratkaisu saavutetaan, kun kaikkien kantamuuttujien arvot ovat ei-negatiivisia. Jos kaikki kantamuuttujat eivät ole ei-negatiivisia, siirrytään takaisin menetelmän ensimmäiseen vaiheeseen.

Esimerkki 2.3. Ratkaistaan seuraava ongelma käyttäen dual simplex-menetelmää:

$$\begin{aligned} \min z &= x_1 + 2x_2 \\ -x_1 + 2x_2 - x_3 &\leq 3 \\ -2x_1 - x_2 + x_3 &\leq -6 \\ x_1, x_2, x_3 &\geq 0. \end{aligned}$$

Muutetaan ongelma maksimointiongelmaksi js lisätään pelivaramuuttujat, jolloin saadaan

$$\begin{aligned} \max -z &= -x_1 - 2x_2 \\ -x_1 + 2x_2 - x_3 + x_4 &= -4 \\ -2x_1 - x_2 + x_3 + x_5 &= -6 \\ x_1, x_2, x_3, x_4, x_5 &\geq 0. \end{aligned}$$

Muodostetaan ongelmasta dual simplex -taulukko:

		-1	-2	0	0	0	
$\mathbf{c}_B$		x_1	x_2	x_3	x_4	x_5	$\mathbf{x}_B$
0	x_4	-1	2	-1	1	0	-4
0	x_5	-2	-1	1	0	1	-6
		1	2	0	0	0	0

Taulukon mukainen ratkaisu on $\mathbf{x}^T = \begin{bmatrix} 0 & 0 & 0 & -4 & -6 \end{bmatrix}$ ei ole käypä ratkaisu, joten siirrytään menetelmän seuraavaan vaiheeseen. Valitaan lähteväksi muuuttujaksi pienin negatiivisista kantamuuttujista, joka on tässä tapauksessa x_5. Tämän muuttujan osoittama rivi on täten pivot-rivi. Tulevaksi muuttujaksi valitaan suurin seuraavista negatiivisista suhteista:

$$\frac{1}{-2} = -\frac{1}{2}, \; \frac{2}{-1} = -2.$$

Näistä ensimmäinen eli $-\frac{1}{2}$ on suurempi, joten x_1 valitaan tulevaksi muuttujaksi. Suoritetaan pivotointi ja saadaan:

		-1	-2	0	0	0	
$\mathbf{c}_B$		x_1	x_2	x_3	x_4	x_5	$\mathbf{x}_B$
0	x_4	0	$\frac{5}{2}$	$-\frac{3}{2}$	1	$-\frac{1}{2}$	-1
-1	x_1	1	$\frac{1}{2}$	$-\frac{1}{2}$	0	$-\frac{1}{2}$	3
		0	$\frac{3}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$	-3

Saatu ratkaisu $\mathbf{x}^T = \begin{bmatrix} 3 & 0 & 0 & -1 & 0 \end{bmatrix}$ ei ole edelleenkään käypä, sillä $x_4 < 0$. Suoritetaan pivotointi uudelleen ja luodaan uusi taulukko:

		-1	-2	0	0	0	
$\mathbf{c}_B$		x_1	x_2	x_3	x_4	x_5	$\mathbf{x}_B$
0	x_3	0	$-\frac{5}{3}$	1	$-\frac{2}{3}$	$\frac{1}{3}$	$\frac{2}{3}$
-1	x_1	1	$-\frac{1}{3}$	0	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{10}{3}$
		0	$\frac{7}{3}$	0	$\frac{1}{3}$	$\frac{1}{3}$	$-\frac{10}{3}$

(2.5)

Taulukon mukainen ratkaisu on $\mathbf{x}^T = \begin{bmatrix} \frac{10}{3} & 0 & \frac{2}{3} & 0 & 0 \end{bmatrix}$. Tämä ratkaisu on käypä, joten se on alkuperäisen ongelman optimaalinen käypä kantaratkaisu. Alkuperäisen minimointiongelman kohdefunktion optimaalinen arvo on $\frac{10}{3}$.

Tutkitaan vielä tilannetta, jossa lisätään ehto

$$-x_1 - x_3 \leq -6$$

ratkaistuun ongelmaan. Lisätään rajoite taulukkoon 2.5 ja saadaan:

		-1	-2	0	0	0	0	
$\mathbf{c}_B$		x_1	x_2	x_3	x_4	x_5	x_6	$\mathbf{x}_B$
0	x_3	0	$-\frac{5}{3}$	1	$-\frac{2}{3}$	$\frac{1}{3}$	0	$\frac{2}{3}$
-1	x_1	1	$-\frac{1}{3}$	0	$-\frac{1}{3}$	$-\frac{1}{3}$	0	$\frac{10}{3}$
0	x_6	-1	0	-1	0	0	1	-6
		0	$\frac{7}{3}$	0	$\frac{1}{3}$	$\frac{1}{3}$	0	$-\frac{10}{3}$

Kantamuuttujien määräämissä sarakkeissa kaikkien paitsi yhden alkion tulee olla nollia, joten muuttujan x_6 määräämään riviin on lisättävä muiden rivien moninkertoja, jotta taulukon mukaan muuttujat x_1 ja x_3 olisivat kantamuuttujia. Uudeksi taulukoksi saadaan:

$$\begin{array}{cc|cccccc|c}
 & & -1 & -2 & 0 & 0 & 0 & 0 & \\
\mathbf{c}_B & & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & \mathbf{x}_B \\
\hline
0 & x_3 & 0 & -\frac{5}{3} & 1 & -\frac{2}{3} & \frac{1}{3} & 0 & \frac{2}{3} \\
-1 & x_1 & 1 & -\frac{1}{3} & 0 & -\frac{1}{3} & -\frac{1}{3} & 0 & \frac{10}{3} \\
0 & x_6 & 0 & -2 & 0 & -1 & 0 & 1 & -2 \\
\hline
 & & 0 & \frac{7}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 & -\frac{10}{3}
\end{array} \tag{2.6}$$

Ratkaisu on taulukon 2.6 mukaan $\mathbf{x}^T = \begin{bmatrix} \frac{10}{3} & 0 & \frac{2}{3} & 0 & 0 & -2 \end{bmatrix}$, joka ei enää ole käypä, minkä aiheutti uuden ehdon lisääminen. Käytetään dual simplex -mentelmää käypyyden palauttamiseksi. Uudeksi taulukoksi saadaan:

$$\begin{array}{cc|cccccc|c}
 & & -1 & -2 & 0 & 0 & 0 & 0 & \\
\mathbf{c}_B & & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & \mathbf{x}_B \\
\hline
0 & x_3 & 0 & -\frac{1}{3} & 1 & 0 & \frac{1}{3} & -\frac{2}{3} & 2 \\
-1 & x_1 & 1 & \frac{1}{3} & 0 & 0 & -\frac{1}{3} & -\frac{1}{3} & 4 \\
0 & x_4 & 0 & 2 & 0 & 1 & 0 & -1 & 2 \\
\hline
 & & 0 & \frac{5}{3} & 0 & 0 & \frac{1}{3} & \frac{1}{3} & -4
\end{array}$$

Ratkaisuksi saadaan $\mathbf{x}^T = \begin{bmatrix} 4 & 0 & 2 & 2 & 0 & 0 \end{bmatrix}$, joka on käypä, joten se on uuden ongelman optimaalinen ratkaisu, jonka avulla saadaan kohdefunktion z minimiarvoksi 4.

2.1.3 Muita menetelmiä lineaariseen optimointiin

Simplex-menetelmät eivät ole ainoita lineaarisen optimoinnin ratkaisumenetelmiä. Koska niiden toimiminen vaatii melko tarkat olosuhteet, mainitaan myös muita menetelmiä. Simplex-menetelmien huono puoli on iteraatioiden mahdollinen eksponentiaalinen kasvu tilanteissa, joissa algoritmi käy läpi jokaisen käyvän alueen kärkipisteen. Tämä kasvattaa laskenta-aikaa. Sisäpistemenetelmät (engl. interior-point methods) ovat tällaisissa tapauksissa tehokkaampia. Vaikka sisäpistemenetelmien vaiheet ovat usein työläämpiä, on iteraatioiden määrä pahimmassa tapauksessa huomattavasti pienempi kuin simplex-menetelmillä. Nimensä mukaisesti sisäpistemenetelmät hakevat ratkaisua käyvän alueen sisäpisteistä lähestyen optimaalista kärkipistettä, kun esimerkiksi simplex-menetelmät liikkuvat kärkipisteestä toiseen. [21, s. 393] Esimerkkinä sisäpistemenetelmästä toimii Karmarkarin algoritmi (kts. [17]).

2.2 Kokonaislukuoptimointimenetelmiä

Usein konkreettisten optimointiongelmien, kuten aikataulutus- ja työnjako-ongelmien ratkaisut, voivat olla vain kokonaislukuarvoisia. Aikaisemmin esitellyt menetelmät lineaaristen optimointiongelmien ratkaisemiseksi eivät välttämättä siis toimi näiden ongelmien ratkaisemisessa, sillä niiden tuottamat ratkaisut eivät ole rajoitettu kokonaislukuarvoihin. Tässä alaluvussa esitellään *leikkaavien tasojen -menetelmien* sekä *Branch-and-Bound -menetelmien* toimintaa. Vaikka menetelmistä esitetään käsinlaskettavia esimerkkejä, on syytä muistaa, että konkreettisissa tilanteissa etenkin kokonaislukuoptimointi tehdään tietokoneen avulla. Luvussa perehdytään vain aitoihin kokonaislukuoptimointiongelmiin, joissa jokaisen muuttujan arvo on määrätty olemaan kokonaisluku.

2.2.1 Leikkaavien tasojen -menetelmät

Olkoon lineaarinen kokonaislukuoptimointiongelma muotoa

$$\min z = \mathbf{c}^T\mathbf{x} \tag{2.7a}$$

$$A\mathbf{x} = \mathbf{b} \tag{2.7b}$$

$$\mathbf{x} \geq \mathbf{0} \tag{2.7c}$$

$$x_j \in \mathbb{Z}. \tag{2.7d}$$

Ratkaistaan aluksi yhtälöiden (2.7a), (2.7b) ja (2.7c) määräämä lineaarinen optimointiongelma eli jätetään kokonaislukurajoitus huomioimatta. Tätä kutsutaan myös *relaksoiduksi ongelmaksi*. Käytetään tämän ratkaisemisessa dual simplex -menetelmää. Oletetaan, että ongelmalla on käypä ratkaisu ja on olemassa äärellinen optimaalinen ratkaisu. Saadun lopullisen dual simplex -taulukon i:nnes rajoite voidaan esittää muodossa

$$\sum_{j=1}^{n} t_{ij}x_j = x_{Bi}, \tag{2.8}$$

jossa t_{ij} kuvaa taulukon alkioita ja x_{Bi} on optimaalisen lineaarisen ongelman ratkaisun i:nnen kantamuuttujan arvo.

Olkoon $\lfloor a \rfloor = \max\{n \in \mathbb{Z} \,|\, n \leq a\}$ eli $\lfloor a \rfloor$ on luvun a lattiafunktio. On helposti havaittavissa, että $\lfloor t_{ij} \rfloor \leq t_{ij}$, joten yhtälö (2.8) saadaan muotoon

$$\sum_{j=1}^{n} \lfloor t_{ij} \rfloor x_j \leq x_{Bi}.$$

Tämän epäyhtälön vasen puoli on selvästi kokonaisluku, koska se koostuu kokonaislukujen tulojen summasta. Täten myös oikea puoli voidaan asettaa vastaamaan kokonaislukua eli saadaan

$$\sum_{j=1}^{n} \lfloor t_{ij} \rfloor x_j \leq \lfloor x_{Bi} \rfloor$$

ja pelivaramuuttujan $u_i \geq 0$ avulla saadaan edelleen

$$\sum_{j=1}^{n} \lfloor t_{ij} \rfloor x_j + u_i = \lfloor x_{Bi} \rfloor. \tag{2.9}$$

Yhtälöstä (2.9) voidaan päätellä, että pelivaramuuttujan u_i on myös oltava kokonaisluku. Oletetaan nyt, että x_{Bi} ei ole kokonaisluku. Tällöin on olemassa luvut $f_i \in (0, 1)$ ja $g_{ij} \in [0, 1)$, joiden avulla voidaan kirjoittaa

$$x_{Bi} = \lfloor x_{Bi} \rfloor + f_i$$

ja

$$t_{ij} = \lfloor t_{ij} \rfloor + g_{ij}.$$

Vähennetään nyt yhtälö (2.8) yhtälöstä (2.9) ja saadaan

$$\sum_{j=1}^{n} (\lfloor t_{ij} \rfloor - t_{ij}) x_j + u_i = \lfloor x_{Bi} \rfloor - x_{Bi},$$

joka voidaan sieventää muotoon

$$\sum_{j=1}^{n} (-g_{ij}) x_j + u_i = -f_i. \tag{2.10}$$

Yhtälöä (2.10) kutsutaan *leikkaavan tason rajoitteeksi*. Tämä rajoite poistaa optimaalisia ratkaisuja, jotka eivät koostu kokonaisluvuista. Toisaalta se ei kuitenkaan poista käypiä kokonaislukuratkaisuja.

Leikkaavien tasojen -menetelmät koostuvat seuraavista vaiheista:

1. Ratkaistaan relaksoitu ongelma käyttäen esimerkiksi dual simplex -menetelmää. Mikäli saadussa ratkaisussa esiinty vain kokonaislukuarvoja, se on haluttu ratkaisu. Jos taas optimaalisessa ratkaisussa on muitakin kuin kokonaislukuarvoja, siirrytään seuraavaan kohtaan.
2. Muodostetaan yhtälön (2.10) mukainen rajoite sen rivin suhteen, joka antaa suurimman f_i arvon.
3. Lisätään leikkaavan tason rajoite alkuperäiseen ongelmaan ja siirrytään takaisin ensimmäiseen kohtaan.

Esimerkki 2.4. Ratkaistaan optimointiongelma

$$\max z = 4x_1 + 2x_2 \tag{2.11a}$$
$$x_1 + 2x_2 \leq 3 \tag{2.11b}$$
$$6x_1 + 4x_2 \leq 10 \tag{2.11c}$$
$$x_1, x_2 \geq 0 \tag{2.11d}$$
$$x_1, x_2 \in \mathbb{Z}. \tag{2.11e}$$

Aloitetaan ongelman ratkaiseminen kokonaislukuoptimointiongelmaa vastaavan lineaarisen ongelman ratkaisemisella. Tämä voidaan ratkaista käyttämällä simplex-menetelmää. Alkuperäiseksi taulukoksi saadaan:

	x_1	x_2	x_3	x_4	z	
x_3	1	2	1	0	0	3
x_4	6	4	0	1	0	10
	-4	-2	0	0	1	0

Simplex-menetelmää hyödyntämällä taulukko saadaan muotoon:

	x_1	x_2	x_3	x_4	z	
x_3	0	$\frac{4}{3}$	1	$-\frac{1}{6}$	0	$\frac{4}{3}$
x_1	1	$\frac{2}{3}$	0	$\frac{1}{6}$	0	$\frac{5}{3}$
	0	$\frac{2}{3}$	0	$\frac{2}{3}$	1	$\frac{20}{3}$

(2.12)

Taulukosta 2.12 voidaan lukea relaksoidun ongelman optimaalinen käypä ratkaisu, mutta koska kantavektoreiden arvot eivät ole kokonaislukuja on luotava leikkaava tason rajoite. Yhtälön (2.10) mukaan tasoksi saadaan

$$-\frac{2}{3}x_2 - \frac{1}{6}x_4 + u_1 = -\frac{2}{3}. \tag{2.13}$$

Kuvataan taso muuttujien x_1 ja x_2 avulla hyödyntäen alkuehtoa (2.11c), johon on lisätty pelivaramuuttuja x_4. Sijoitetaan siis yhtälö

$$x_4 = 10 - 6x_1 - 4x_2$$

yhtälöön (2.13), jolloin saadaan suora

$$x_1 = 1,$$

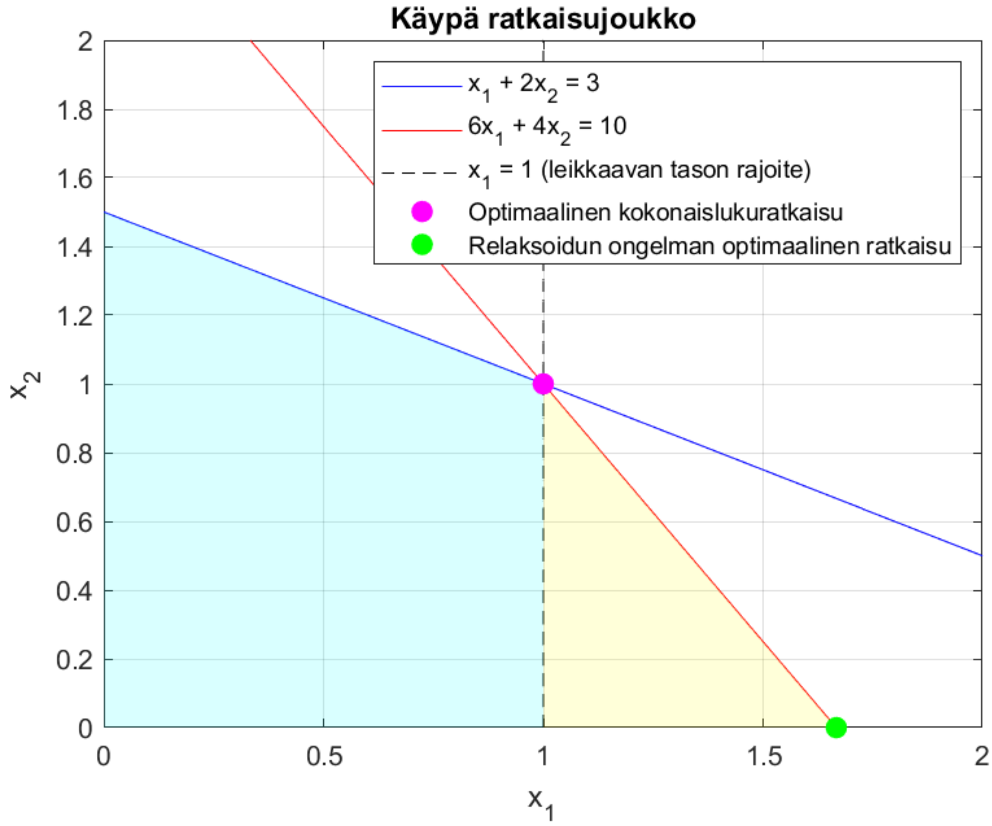
jota havainnolistetaan kuvassa 2.2. Lisätään yhtälö (2.13) taulukkoon 2.12 ja saadaan:

	x_1	x_2	x_3	x_4	u_1	z	
x_3	0	$\frac{4}{3}$	1	$-\frac{1}{6}$	0	0	$\frac{4}{3}$
x_1	1	$\frac{2}{3}$	0	$\frac{1}{6}$	0	0	$\frac{5}{3}$
u_1	0	$-\frac{2}{3}$	0	$-\frac{1}{6}$	1	0	$-\frac{2}{3}$
	0	$\frac{2}{3}$	0	$\frac{2}{3}$	0	1	$\frac{20}{3}$

Havaitaan, ettei saatu ratkaisu ole enää käypä, joten käytetään dual simplex -menetelmää käypyyden palauttamiseksi. Menetelmän avulla saadaan:

		4	2	0	0	0	
$\mathbf{c}_B$		x_1	x_2	x_3	x_4	u_1	$\mathbf{x}_B$
0	x_3	0	0	1	$-\frac{1}{2}$	2	0
4	x_1	1	0	0	0	1	1
2	x_2	0	1	0	$\frac{1}{4}$	$-\frac{3}{2}$	1
		0	0	0	$\frac{1}{2}$	1	6

Taulukosta optimaaliseksi ratkaisuksi voidaan lukea $\mathbf{x}_B^T = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \end{bmatrix}$. Tämä ratkaisu on kokonaislukuarvoinen, joten se on käypä.



Kuva 2.2. *Esimerkki 2.4 visualisoituna*

2.2.2 Branch-and-Bound -menetelmät

Ongelmissa, joissa käypien ratkaisujen joukko on rajoitettu, on selvästi äärellinen määrä kokonaislukuratkaisuja. Tämän vuoksi on mahdollista etsiä optimaalista ratkaisua käymällä läpi näitä käypiä ratkaisuja systemaattisesti. Branch-and-Bound -menetelmät perustuvat strategiaan, jossa ratkaisut jaetaan eri osajoukkoihin. Osajoukkoja voidaan hylätä implisiittisesti, jos tiedetään ettei niiden sisältämät ratkaisut voi olla optimaalisia. Näin ollen päädytään optimaaliseen ratkaisuun. Branch-and-Bound -menetelmien avulla voidaan vähentää pitkiä laskuja, joita esiintyy usein esimerkiksi leikkaavien tasojen -menetelmissä. Tämä voi puolestaan tehostaa optimaalisten ratkaisujen löytämistä. [18] Dakinin algoritmi on esimerkki Branch-and-Bound -menetelmästä. Se on systemaattinen tapa etsiä ratkaisuja kokonaislukuoptimointiongelmaan. [9]

Branch-and-Bound -menetelmiä kuvataan usein puukaaviona. Menetelmässä puukaavion solmut kuvaavat saatuja ratkaisuja ja vastaavasti särmät kuvaavat lisättyä ehtoa. Puukaavioon lisättävä solmu pitää sisällään kaikki sitä ennen samassa haarassa olleiden särmien rajoitteet. *Päätesolmulla* (engl. terminal node) tarkoitetaan solmua, josta haaroittuvat solmut eivät voi saada käypää optimaalista ratkaisua. Näin ollen tästä solmusta lähtevää haaraa ei tarvitse tutkia. *Roikkuvalla solmulla* (engl. dangling node) tarkoitetaan puolestaan sellaista solmua, jota tutkitaan myöhemmin.

Olkoon (2.7) jälleen ratkaistavana ongelmana. Branch-and-Bound -menetelmän vaiheet ovat seuraavat:

1. Etsitään initiaalinen ratkaisu ratkaisemalla kokonaislukuongelmaa vastaava lineaarinen ongelma. Käypä optimaalinen ratkaisu on löydetty, mikäli kaikki ratkaisun muuttujat $x_j \in \mathbb{Z}$. Jos näin ei ole, siirrytään seuraavaan vaiheeseen.
2. Tässä vaiheessa valitaan *haarautuva muuttuja* (engl. branching variable). Valinta tehdään niiden kantamuuttujien väliltä, joiden arvo ei ole kokonaisluku tässä vaiheessa käsiteltävässä ratkaisussa. Muuttujista valitaan usein se, jonka murto-osa on suurin. Luodaan tälle muuttujalle uudet rajoitteet. Tiedetään, että ei-kokonaislukuarvoisen kantamuuttujan arvo voidaan kuvata

   $$x_{Bi} = \lfloor x_{Bi} \rfloor + f_i,$$

   kun $0 < f_i < 1$. Koska muuttujan x_j arvon on oltava kokonaisluku, se ei voi olla välillä $(\lfloor x_{Bi} \rfloor, \lfloor x_{Bi} \rfloor + 1)$. On siis oltava joko

   $$x_j \leq \lfloor x_{Bi} \rfloor \tag{2.14}$$

   tai

   $$x_j \geq \lfloor x_{Bi} \rfloor + 1. \tag{2.15}$$

3. Edellisen vaiheen pohjalta luodaan kaksi solmua, joissa esiintyy uudet kokonaislukuongelmat. Toiseen solmuun on lisätty rajoite (2.14) ja toiseen rajoite (2.15). Näiden ongelmien käypyys voidaan palauttaa käyttämällä dual simplex -menetelmää.
4. Edellisen vaiheen solmut voidaan asettaa päätesolmuiksi, jos
   (a) solmut eivät olleet käypiä.
   (b) solmuissa kaikki $x_j \in \mathbb{Z}$.

   Jälkimmäisemmä tapauksessa on lisäksi syytä tutkia, onko kohdefunktion arvo suurempi kuin suurin tähän mennessä saatu arvo. Mikäli näin on, asetetaan kyseinen arvo parhaaksi arvoksi.
5. Edellisellä vaiheella on 3 mahdollista lopputulosta.
   (a) Jos molemmat solmut asetettiin päätesolmuiksi, siirrytään tutkimaan roikkuvien solmujen listaa järjestyksessä. Jos roikkuvan solmun kohdefunktion arvo on suurempi kuin senhetkinen paras arvo, siirrytään vaiheeseen 2. Listan ollessa tyhjä, optimaalinen ratkaisu on senhetkinen paras kohdefunktion arvo.
   (b) Jos vain toinen solmuista on päätesolmu, valitaan solmuista se, joka ei ole päätesolmu ja siirrytään vaiheeseen 2.

(c) Jos kumpikaan solmuista ei ole päätesolmu, valitaan lupaavampi eli se solmu, jonka kohdefunktion arvo on suurempi ja siirrytään vaiheeseen 2. Toinen solmuista lisätään roikkuvien solmujen listaan.

Esimerkki 2.5. Ratkaistaan kokonaislukuoptimointiongelma

$$\max z = 5x + 2y \tag{2.16a}$$

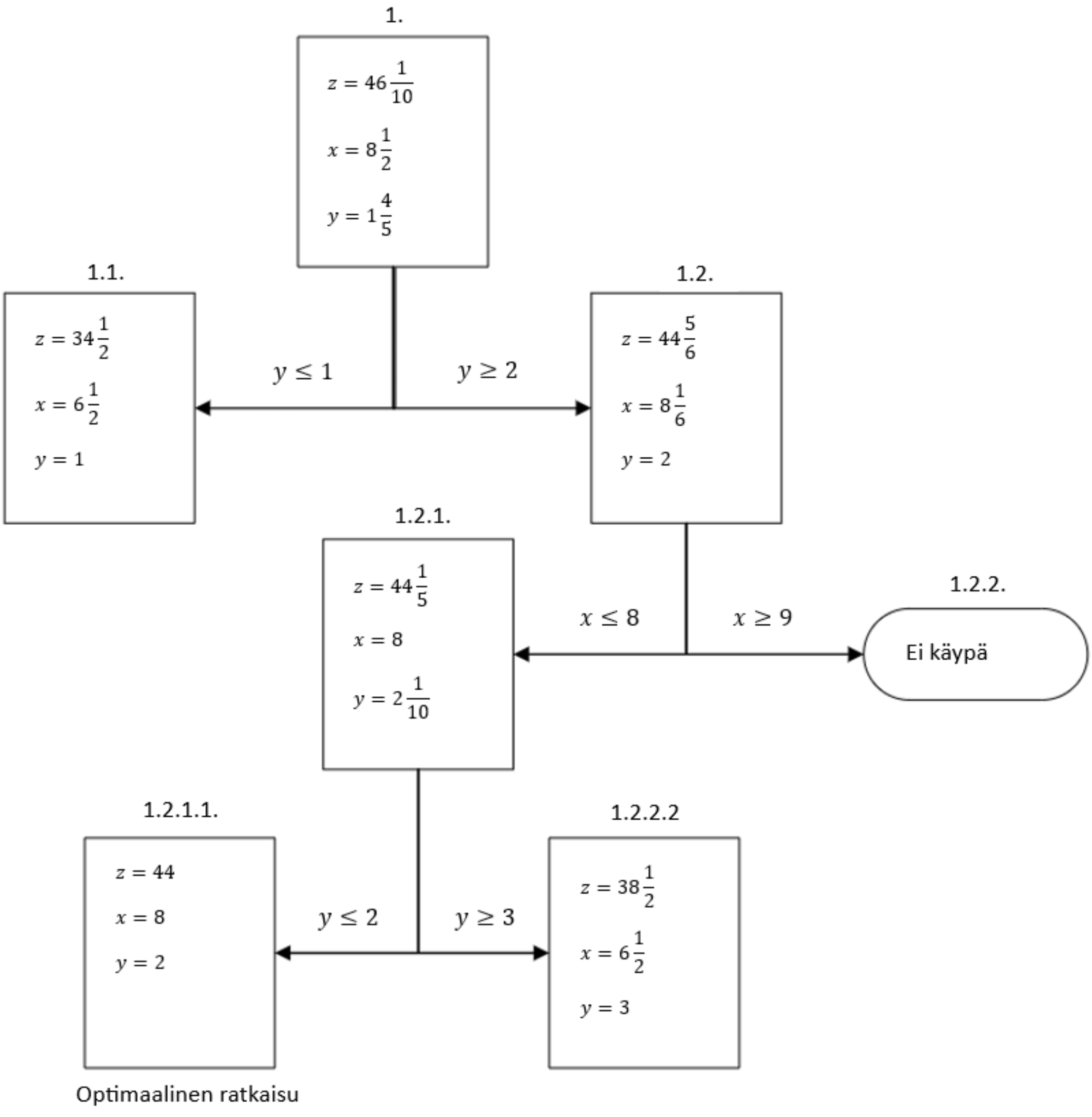
$$12x - 7y \leq 84 \tag{2.16b}$$

$$6x + 10y \leq 69 \tag{2.16c}$$

$$x, y \geq 0 \tag{2.16d}$$

$$x, y \in \mathbb{Z}. \tag{2.16e}$$

käyttäen Branch-and-Bound -menetelmää. Ongelman ratkaisu on esitetty kuvassa 2.3. Kuvassa solmu 1 kuvaa kokonaislukuongelmaa vastaavan lineaarisen ongelman initiaalista ratkaisua. Haarautuvaksi muuttujaksi valitaan y. Uusiksi rajoitteiksi saadaan $y \leq 1$ ja $y \geq 2$, joiden avulla saadaan uudet solmut 1.1 ja 1.2. Näistä solmuista lupaavampi on 1.2, joten valitaan solmusta haarautuva muuttuja, joka on x. Solmu 1.1 lisätään roikkuvien solmujen listaan. Muuttujan x uusiksi rajoitteiksi saadaan $x \leq 8$ ja $x \geq 9$. Näistä jälkimmäinen tapaus ei ole mahdollinen rajoitusten (2.16a) ja (2.16b) puitteissa. Näin ollen asetetaan solmu 1.2.2 päätesolmuksi. Solmu 1.2.1 ei puolestaan ole päätesolmu, joten valitaan jälleen haarautuva muuttuja ja saadaan rajoitteet $y \leq 2$ ja $y \geq 3$. Solmu 1.2.1.1 on päätesolmu, sillä sen muuttujien arvot ovat positiivisia kokonaislukuja. Solmu 1.2.1.2 on lisätään roikkuvien solmujen listaan. Koska roikkuvien solmujen kohdefunktioiden arvot ovat pienempiä kuin parhaan kokonaislukuarvoisen ratkaisun arvo, voidaan ne asettaa päätesolmuiksi. Optimaalinen ratkaisu on solmun 1.2.1.1 mukainen ratkaisu eli $x = 8$ ja $y = 2$, jolloin $z = 44$.



Kuva 2.3. *Ongelman 2.16 ratkaisun vaiheet esitettynä puukaaviossa. Kuva on muokattu lähteestä [18, s. 431].*

3. MATEMATIIKAN SÄHKÖISEEN TENTTIIN LIITTYVÄÄ TEORIAA

Tässä luvussa keskitytään sähköisiin tentteihin pedagogisesta näkökulmasta. Luvussa esitellään niihin liittyviä käsitteitä kuten formatiivinen ja summatiivinen arviointi sekä proseduraalinen ja konseptuaalinen tieto. Lisäksi pohditaan niiden etuja ja mahdollisuuksia sekä hyödyntämistä TUNI:ssa.

3.1 Formatiivinen ja summatiivinen arviointi

Formatiivinen arviointi on toimintaa, jolla opettajat ja opiskelijat arvioivat itseään ja toisiaan, antamalla palautetta, minkä avulla opetusta ja oppimista voidaan parantaa. Formatiivisen arvioinnin ensisijaisena tavoitteena onkin parantaa opiskelijan suoritusta. Formatiivinen arviointi tiedottaa myös opetuksen tavoitteista ja edistymisestä tähän tavoitteeseen nähden. Kysymysten esittäminen opiskelijoilta ja pistokokeiden pitäminen ovat esimerkkinä tällaisesta arvioinnista. [10] Formatiivista arviontia tapahtuu siis opetuksessa jatkuvasti.

Summatiivinen arvointi on puolestaan kumulatiivista arviontia, joka keskittyy arvosanoihin. Arvosanojen avulla opiskelijoita verrataan tavoitteiden määräämään tasoon ja toisiinsa. Toisin kuin formatiivisessa palautteessa summatiivisen arvioinnin tavoitteena on selvittää, kuinka paljon opiskelija tietää ja kuinka paljon hän on oppinut, eikä niinkään kehittää opiskelijan suoritusta. Summatiivista arviointia hyödynnetään vain esimerkiksi kurssin, lukuvuoden tai koulutuksen lopussa, mutta sillä on suuri merkitys. [10] Esimerkiksi ylioppilaskokeet ja pääsykokeet, joiden mukaan monet koulutuspaikat jaetaan, arvioidaan summatiivisesti.

3.2 Proseduraalinen ja konseptuaalinen tieto matematiikassa

Matematiikan opetuksen yhteydessä puhutaan usein proseduraalisesta ja konseptuaalisesta tiedosta. Näiden tärkeydestä toisiinsa nähden on keskusteltu vuosikymmenien ajan. Yleisimmin proseduraalista tietoa kuvataan taitona ja konseptuaalista tietoa ymmärryksenä, vaikka tämä ei ole kovin tarkka erottelu.

Hiebert [15, s. 3–8] määrittelee teoksessaan sekä proseduraalisen että konseptuaalisen tiedon matematiikassa. Proseduraalinen tieto koostuu ensinnäkin matematiikan kielen lukutaidosta. Proseduraaliseksi tiedoksi luokitellaan esimerkiksi eri symbolien tunteminen matematiikan kontekstissa ja hyväksyttävän laskutoimituksen kirjoittaminen. Proseduraalisen tiedon toinen puoli on algoritmien ja laskusääntöjen hallitseminen. Tämä koostuu oikeiden prosessien käytöstä oikeassa järjestyksessä, jotta päästään haluttuun lopputulokseen. Hiebertin mukaan etenkin koulumaailmassa nämä proseduurit ovat pääosin symbolisia laskutoimituksia. Laskutoimitusten symbolit voivat kuvata tällöin tosimaailman objekteja.

Konseptuaalisen tiedon Hiebert [15, s. 3–8] määrittelee puolestaan matematiikan käsitteiden suhteena. Eri käsitteiden ja tietojen väliset suhteet muodostavat tietoverkon, jonka jokainen osa liittyy johonkin toiseen jollakin tavalla. Näin ollen eristettyjä tiedon palasia ei ole vaan kaikki linkittyy toisiinsa. Toisin sanoen asiat voidaan selittää toistensa avulla. Konseptuaalista tietoa voidaan siis kasvattaa luomalla yhteyksiä asioiden välille.

3.3 Sähköisistä tenteistä yleisesti

Digitalisaation ja tietokoneiden yleistyttyä matematiikan taitojen vaatimukset ovat muuttuneet. Matematiikalla on tänä päivänä suora yhteys teknologiaan ja työelämässä tarvitaan tietotekniikkaan liittyviä matematiikan taitoja sekä kykyä ratkaista ongelmia. Lisäksi matematiikan osaamista tarvitaan enemmän kuin koskaan. [16]

Sähköinen tenttiminen on kehittynyt paljon vuosikymmenten aikana. Tietokoneita on käytetty tenttimiseen 1970–luvulta lähtien, jolloin tenteissä oli yksinkertaisia monivalintatehtäviä. 1990–luvulla sähköisissä tenteissä alkoi esiintyä yhä enemmän ongelmanratkaisutehtäviä. [24] Sähköisiä tenttejä ei voitu kuitenkaan ennen käyttää yhtä monipuolisesti kuin nykypäivänä. Nykyään sähköinen tenttiminen on alkanut syrjäyttämään paperitenttiä, minkä vuoksi paperitentit ovat vähentyneet [24]. Syynä tähän voidaan pitää sähköisten tenttien tehokkuutta ja joustavuutta sekä monipuolisempia mahdollisuuksia taitojen mittaamiseen.

Pedagogisesti sähköisten tenttien suurin hyöty on monipuolinen taitojen mittaaminen ja moniuloitteisempi taitojen arviointi [24]. Sähköiseen tenttiin on mahdollista luoda paljon enemmän eri tehtävätyyppejä, joilla opiskelijoiden osaamista mitataan, kuin perinteiseen paperitenttiin. Matematiikan paperitenteissä mahdollisia tehtävätyyppejä ovat yksinkertaiset laskutehtävät, sanalliset laskutehtävät sekä kuvaajan analysointi- tai piirtotehtävät. [16] Sähköisissä tenteissä voi näiden lisäksi olla interaktiivisia kuvaajatehtäviä sekä aineiston kuten videon tai animaation analysointia. [16] Kuvaajien tutkiminen sähköisesti on myös tarkempaa kuin paperilla, jossa kuvan tarkkuuden rajallisuus voi tuoda epätoivottua hajontaa hyväksyttäviin vastauksiin. Sähköiseen tenttiin on mahdollista luoda vaihtoehtoisia kysymyksiä, jotka vaihtuvat tenttimiskerroittain, joten eri opiskelijoilla on eri kysymyksiä tenteissä. Tämä helpottaa esimerkiksi uusintatentin järjestämistä ja vaikeuttaa tenttivilppiä. [19]

Sähköisten tenttien luotettavuutta verrattaen paperitenttiin on tutkittu paljon. Esimerkiksi Smolinsky ym. [25] tutkivat mittaavatko sähköinen tentti ja paperitentti samoja taitoja, ja voiko paperitenttiä korvata sähköisellä. Yliopistossa tehdyn tutkimuksen tuloksena oli, että sähköisen tentin tulokset olivat verrattavissa paperitenttiin nähden. Sähköisen tentin tulokset olivat myös johdonmukaisia keskenään. Bennettin ym. [7] artikkelissa puolestaan tutkittiin vuonna 2001 noin tuhatta kahdeksasluokkalaista. Heidän tuloksiaan vertailtiin kahdesta samasta matematiikan kokeesta, joista toinen oli toteutettu sähköisesti ja toinen paperisena. Tuloksena saattiin, että sähköisenä tehty koe oli tilastollisesti merkittävästi vaikeampi kuin paperikoe. Syyksi tähän epäiltiin oppilaiden vähäistä kokemusta tietokoneiden käytöstä. Koska tietokoneiden käyttö on yleistynyt huomattavasti 20 vuoden aikana, tämän tutkimuksen erojen voisi ajatella olevan pienempiä nykypäivänä.

Tutkimuksissa on havaittu joitakin sähköisten tenttien mahdollisia huonoja puolia. Esimerkiski Rønningin [22] tutkimuksessa havaittiin, että sähköisen kokeen automaattitarkisteisissa tehtävissä, joita opiskelija saa yrittää uudelleen, yritetään vain etsiä vastausta sen sijaan, että tehtävää yritettäisiin ratkaista.

3.4 Sähköiset tentit TUNI:ssa

Sähköinen tenttiminen suoritetaan Tampereen korkeakouluissa TUNI EXAM -palvelun avulla. Tentit suoritetaan EXAM-tiloissa, joita sijaitsee TUNI:n jokaisella kampuksella. EXAM-tilat ovat varustettu kulunvalvonnalla ja ne ovat kameravalvottuja, mikä estää tenttivilppiä. Tentit suoritetaan tiloissa olevilla tietokoneilla, joilla ei pääse verkkoon tai pysty kommunikoimaan toisten koneiden kanssa. [5]

Kuten sähköisten tenttien tapauksessa yleensä, EXAM-tentin suuri hyöty on sen joustavuus [19]. Opiskelija pystyy varaamaan itselleen sopivan tilan sekä tenttimisajankohdan tietyltä kurssikohtaiselta aikaväliltä. Näin päädytään harvemmin tilanteeseen, jossa opiskelija ei pääse osallistumaan alkuperäiseen tenttiin, jonka vuoksi hän joutuu osallistumaan uusintatenttiin useita viikkoja kurssin opetuksen loppumisen jälkeen. Tämä voi heikentää suoriutumista tentissä. EXAM-tentti voi näin ollen parantaa tulosten johdonmukaisuutta. Tentin järjestäjien näkökulmasta sähköinen tentti vaatii vähemmän resursseja, sillä kursseille, joiden osallistujamäärä on suuri, ei tarvitse varata useita tenttimistiloja ja tenttivalvojia [19].

Matematiikan EXAM-tentit erottuvat paperitenteistä usealla tavalla. Kun opiskelijalla saa olla paperitentissä mukana yleensä enintään laskin ja kaavakokoelma tai niin kutsuttu "lunttilappu", EXAM-tentissä opiskelijalla ei saa olla mukanaan mitään. [3] Tentin tietokoneissa on kuitenkin käytössä useita ohjelmistoja, joita saa tentin aikana vapaasti käyttää.

Ohjelmistoihin sisältyy muun muassa laskin, Matlab sekä Microsoft Word ja PowerPoint [4]. Lisäksi TAMK:n EXAM-tilassa tietokoneille on asennettu TI-Nspire -laskinohjelma, joka vastaa laskinta, jota suuri osa tekniikan opiskelijoista TAMK:ssa käyttää. Tentin laatija pystyy myös lisäämään tarvittaessa tentin liitteeksi tiedostoja, joita opiskelija saa hyödyntää tenttiä tehdessään.

Osassa TUNI EXAM -tiloja on käytössä kosketusnäyttöjä ja piirtoalustoja. Kosketusnäytöllä tarkoitetaan tietokoneen näyttöä, joka toimii myös kosketuksen avulla. Piirtoalusta on puolestaan erillinen piirtämiseen tarkoitettu näyttö, jolla voi piirtää tarkasti. TUNI pilotoi tällaisia piirtoalustoja TAMK:n EXAM-tiloissa.

3.5 STACK-tehtävät

STACK-tehtävät ovat tietokoneella suoritettavia automaattitarkisteisia tehtäviä. STACK-tehtäviä voidaan luoda esimerkiksi Moodle-alustalle. Automaattitarkisteisten tehtävien suuri hyöty on opettajien ajansäästö. Yhdellä opettajalla ei ole välttämättä aikaa tarkistaa useiden satojen oppilaiden kokeita tai harjoitustehtäviä. Jos puolestaan tarkistamiseen käytetään useampaa opettajaa, ei arviointi ole välttämättä johdonmukaista. [23, s. 1] STACK-tehtävävien pisteytys on puolestaan objektiivista, sillä arviointiin vaikuttaa vain vastauskenttään kirjoitettu ratkaisu. Tämä voidaan nähdä myös STACK-tehtävien heikkoutena, sillä arvioinnissa ei oteta huomioon opiskelijan ratkaisustrategiaa tai päättelyketjua.

Toisaalta STACK-tehtävistä on paljon hyötyä opiskelijoille, koska he saavat tekemistään tehtävistä välittömän palautteen. STACK-tehtävien palaute koostuu useasta osasta. Jo ennen vastauksen lukitsemista ohjelma kertoo, missä muodossa vastaus tulkitaan ja onko annettu syntaksi validi. Jos syntaksi ei ole validi, vastausta ei voi lukita. Vastauksen lukitsemisen jälkeen opiskelija saa mahdollisen formatiivisen arvioinnin yrityskertansa vastauksesta sekä summatiivisen arvioinnin eli pistemäärän yrityskerrastaan. Opettaja pystyy tehtäviä luodessaan vaikuttamaan, millaista palautetta opiskelija saa erilaisten vastausten jälkeen. Tehtävän laatija pystyy myös vaikuttamaan, missä muodossa tehtävän vastaus on annettava. [23, s. 9–14]

3.6 Satunnaistetut tehtävät

Tietokonetarkisteisten tehtävien ratkaisuna toimii yleensä vain matemaattinen notaatio vastauskentässä ja useimmissa tapauksissa oikea ratkaisu on yksikäsitteinen. Jos jokaisella opiskelijalla olisi identtiset tehtävät, plagioinnin havaitseminen olisi mahdotonta [23, s. 38] ja plagiointi olisi puolestaan todella helppoa esimerkiksi harjoitustehtäviä tehdessä. Toisaalta eri aikaan tentin suorittavat opiskelijat voisivat välittää toisilleen tentin kysymykset ja vastaukset. Tämän vuoksi satunnaistetut tehtävät ovat hyödyllisiä. Satunnaistetuilla tehtävillä tarkoitetaan tehtävää, jonka perusidea ja haluttu ratkaisustrategia pysyvät samoina, mutta eri opiskelijoille annetaan eri lähtöarvot eri suorituskertoina. Plagiointi on tällä keinolla haasteellisempaa kuin identtisten tehtävien tapauksessa. Satunnaistetut tehtävät mahdollistavat myös opiskelijoiden välisen yhteistyön harjoitellessa ilman pelkoa plagioinnista, sillä he voivat pohtia tehtävää yhdessä proseduraalisesta näkökulmasta ratkaisematta toistensa tehtäviä. [23, s. 38]

Satunnaistetun tehtävän luomiseen liittyy useita huomioitavia seikkoja. Tehtävää luodessa on huomioitava kenelle tehtävä kohdistetaan, saako tehtävässä käyttää apuvälineitä, miten tehtävä tarkoitetaan ratkaistavaksi, rajataanko alkuarvoja ja miten tehtävänanto asetellaan. [23, s. 39–42] Otetaan esimerkiksi tehtävä, jossa tulee ratkaista lineaarinen yhtälöryhmä, joka on muotoa

$$\begin{cases} a_{11}x + a_{12}y = b_1 \\ a_{12}x + a_{22}y = b_2. \end{cases}$$

Tällaisen tehtävän voi ratkaista usealla eri tavalla, kuten sijoitusmenetelmällä tai matriisien avulla. Matriisienkin avulla yhtälöryhmä voidaan ratkaista käänteismatriisin avulla tai Gauss-Jordanin eliminointimenetelmää hyödyntäen.

Tehtävänanto pystytään satunnaistamaan satunnaistamalla alkuarvot $a_{11}, a_{12}, a_{21}, a_{22}, b_1$ ja b_2. Alkuarvoja ei voida kuitenkaan valita täysin satunnaisesti, vaan täytyy huomioida, millainen yhtälöryhmän ratkaisuksi halutaan. Esimerkiksi yhtälöryhmän

$$\begin{cases} 3x + y = 2 \\ 3x + y = 2. \end{cases}$$

ratkaiseminen harjoitus- tai tenttitehtävänä ei välttämättä ole tarkoituksenmukaista, kun tarkoituksena oppia ratkaisemaan lineaarinen yhtälöryhmä. Jos yhtälöryhmälle halutaan esimerkiksi yksikäsitteinen ratkaisu tulee alkioiden a_{ij} muodostaman matriisin A rivien olla lineaarisesti riippumattomia ja nollavektorista poikkeavia. Alkuarvojen arpomista täytyy siis rajoittaa tämän ehdon mukaisesti. Lisäksi alkuarvojen suuruus on huomioitava, mikäli tehtävä on suunniteltu käsin laskettavaksi.

4. SATUNNAISEN MATRIISIN MÄÄRITTÄMINEN

Satunnainen $n \times n$ -matriisi voidaan luoda MATLAB:n avulla. Tavoitteena on luoda sellaisia neliömatriiseja, jotka toteuttavat halutut ehdot matematiikan sähköisiä tehtäviä varten. Pääasiallisesti matriiseja luodaan tehtävään, jossa halutaan ratkaista matriisiyhtälö

$$R\mathbf{x} = \mathbf{b}$$

käyttäen Gauss-Jordanin eliminointimenetelmää. Jotta tehtävään luotava matriisi R on pedagogisesti mielekäs, tulee sen luomiseen käytettävällä algoritmilla olla seuraavat ominaisuudet:

1. Matriisin alkioiden arvot voidaan määrätä olemaan kokonaislukuja välillä $[x_{\min}, x_{\max}]$.
2. Jokaisen alkion arvo esiintyy matriisissa enintään k kertaa.
3. Matriisin alkioiden suurimman arvon voidaan vaatia olevan vähintään s ja pienimmän arvon enintään l.
4. Tuotetut matriisit ovat toisiinsa nähden satunnaisia.
5. Matriisi suosii itseisarvoltaan pieniä lukuja.

Mikäli algoritmilla vaadittaisiin olevan vain ehdot 1 ja 4, algoritmin rakentaminen olisi suoraviivaista. Tällöin olisi mahdollista esimerkiksi luoda $n \times n$ -matriisi, johon MATLAB luo satunnaisia kokonaislukuarvoja väliltä $[x_{\min}, x_{\max}]$. Toisaalta olisi mahdollista myös asettaa jokaiselle muuttujalle rajoitukset $[x_{\min}, x_{\max}]$ ja käyttää satunnaista kohdefunktiota. Näillä keinoin ei kuitenkaan ole mahdollista toteuttaa muita vaadittavia ehtoja. Jotta ehdot 2 ja 3 voidaan toteuttaa, on voitava hallita matriisin R alkioiden jokaista mahdollista arvoa ja paikkaa. Ehtoja voidaan noudattaa samanaikaisesti hyödyntäen kokonaislukuoptimointia.

4.1 MATLAB-komento intlinprog

Kappaleessa 4.2 esiteltävän optimointiongelman ratkaisemiseen voidaan käyttää MATLAB-komentoa *intlinprog*, jonka avulla pystytään ratkaisemaan lineaarisia kokonaislukuoptimointiongelmia. Kyseinen komento pystyy ratkaisemaan muodossa

$$\min \mathbf{f}^T\mathbf{x} \tag{4.1a}$$

$$A\mathbf{x} \leq \mathbf{b} \tag{4.1b}$$

$$A_{eq}\mathbf{x} = \mathbf{b}_{eq} \tag{4.1c}$$

$$\mathbf{lb} \leq \mathbf{x} \leq \mathbf{ub} \tag{4.1d}$$

olevia ongelmia, joissa haluttujen muuttujien x_i tulee olla kokonaislukuja. Matriisi A sisältää epäyhtälöiden kertoimet ja matriisi A_{eq} sisältää vastaavasti yhtälöiden kertoimet. Komennon syöte on seuraavanlainen:

```
x=intlinprog(f,intcon,A,b,Aeq,beq,lb,ub,x0,options)
```

Parametrin *intcon* avulla voidaan määrittää, mitkä muuttujavektorin $\mathbf{x}$ arvot tarvitsevat olla kokonaislukuja ja parametri x_0 toimii alkuarvauksena. Parametrin *options* avulla voidaan säätää esimerkiksi komennon käyttämiä laskentamenetelmiä. Komennon syötteeksi ei välttämättä tarvita kaikkia näitä parametrejä, mikäli ongelma ei sitä vaadi. [26]

Komento käyttää 6-osaista strategiaa ongelman ratkaisemiseksi. Se ei kuitenkaan tarvitse kaikkia näitä menetelmiä ratkaisun löytämiseen, vaan optimaalinen ratkaisu voidaan löytää, missä tahansa vaiheessa laskentaa. [26] Prosessin ensimmäinen vaihe on lineaarisen ongelman esikäsittely. Vaiheen päätavoitteina on eliminoida mahdollisimman monta itseääntoistavaa rajoitetta, kiinnittää mahdollisimman monia muuttujia ja muokata muuttujien rajoja tarvittaessa. Rajoitteiden vähentämisen seurauksena vaiheessa myös vähennetään muuttujien lukumäärää. Vaiheen tarkoituksena on nopeuttaa erityisesti suurten optimointiongelmien ratkaisua. [20]

Ratkaisuprosessin toisessa vaiheessa ratkaistaan optimointiongelma ilman kokonaislukurajoitusta, jolloin saadaan relaksoidun ongelman optimaalinen ratkaisu $\mathbf{x}_{LP}$. Matlab käyttää relaksoidun ongelman ratkaisemiseen esimerkiksi työssä aiemmin luvussa 2.1.2 esiteltyä dual simplex -menetelmää. Koska kohdefunktio on sama alkuperäisellä ongelmalla ja relaksoidulla ongelmalla, mutta relaksoidussa ongelmassa on vähemmän rajoitteita,

$$\mathbf{f}^T\mathbf{x}_{LP} \leq \mathbf{f}^T\mathbf{x},$$

jossa $\mathbf{x}$ on alkuperäisen ongelman optimaalinen ratkaisu. [26]

Kolmas vaihe vastaa ensimmäistä vaihetta, mutta nyt esikäsittely tehdään lineaariselle kokonaislukuongelmalle. Tässä esikäsittelyssä tutkitaan epäyhtälöä (4.1c) kokonaislukurajoituksilla. Tämän tarkoituksena on ensinnäkin selvittää, onko olemassa käypää ratkaisua. Tässä vaiheessa selvitetään myös voiko rajoja tiukentaa, voiko epäyhtälöitä vahvistaa, voiko toistuvia rajoituksia poistaa ja voiko muuttujia kiinnittää. [26]

Kokonaislukuongelman esikäsittelyn jälkeen intlinprog luo leikkauksia (engl. cut generation). Leikkaukset ovat lisättyjä epäyhtälörajoituksia ongelmaan. Näiden tarkoituksena on pienentää lineaarisen ongelman käypää ratkaisualuetta, jotta ratkaisut lähestyisivät kokonaislukuja. MATLAB käyttää leikkausten tekemiseen useita erilaisia menetelmiä, kuten luvussa 2.2.1 esiteltyä menetelmää, joiden käyttöä on mahdollista hallita. [26]

Seuraavassa vaiheessa pyritään löytämään käypiä ratkaisuja heuristiikkojen (engl. heuristics) avulla, jotka ovat laskentamenetelmiä, joiden avulla ratkaisun löytäminen voi olla nopeampaa ennen Branch-and-bound -menetelmää tai sen aikana. Huomioitavaa on, että niiden toimivuus ei ole aina täysin varmaa. Heurestiikkavaihtoehtoja komennosta löytyy 9 erilaista, joita käytetään tilanteesta riippuen. [26] Paras vaihtoehto on aina tilannekohtainen.

Viimeisessä vaiheessa ongelma ratkaistaan Branch-and-bound -menetelmän avulla. Menetelmästä on kerrottu enemmän luvussa 2.2.2.

4.2 Optimointiongelman muodostaminen

Satunnaisen matriisin luomisesta voidaan muodostaa lineaarinen kokonaislukuoptimointiongelma. Jotta saadaan muodostettua halutut rajoitteet, luodaan jokaista matriisin mahdollista arvoa jokaisessa matriisin paikassa vastaavat muuttujat. Ongelmassa muuttujien lukumäärä on täten $n^2(m+1)$, jossa n on luotavan neliömatriisin R rivien sekä sarakkeiden lukumäärä ja m on kaikkien mahdollisten kokonaislukuarvojen lukumäärä. Esimerkiksi tilanteessa, jossa $x_{\min} = -5$ ja $x_{\max} = 5$, matriisi R voi sisältää 11 eri kokonaislukuarvoa eli $m = 11$.

Ensimmäiset n^2 muuttujaa vastaavat matriisin R alkioita. Merkitään näitä muuttujia $r_1, \ldots, r_{n^2}$. Seuraavat n^2 muuttujaa vastaavat puolestaan mahdollista kokonaislukuarvoa $x_{\min}$ matriisin R jokaisessa kohdassa ja niin edelleen. Merkitään yleisesti mahdollisten lukuarvojen sijainteja vastaavia muuttujia $x_{v,p}$, jossa $v \in [x_{\min}, x_{\max}]$ kuvaa mahdollista kokonaislukuarvoa ja $p \in [1, n^2]$ kuvaa paikkaa matriisissa R. Muuttujat $x_{v,p}$ voivat saada vain arvon 0 tai 1. Kun muuttujat kasataan matriiseiksi, saadaan

$$R = \begin{bmatrix} r_1 & r_{n+1} & \cdots & r_{n^2-n+1} \\ r_2 & r_{n+2} & \cdots & r_{n^2-n+2} \\ \vdots & \vdots & & \vdots \\ r_n & r_{2n} & \cdots & r_{n^2} \end{bmatrix} \text{ ja}$$

$$X_{x_{\min}} = \begin{bmatrix} x_{x_{\min},1} & x_{x_{\min},n+1} & \cdots & x_{x_{\min},n^2-n+1} \\ x_{x_{\min},2} & x_{x_{\min},n+2} & \cdots & x_{x_{\min},n^2-n+2} \\ \vdots & \vdots & & \vdots \\ x_{x_{\min},n} & x_{x_{\min},2n} & \cdots & x_{x_{\min},n^2} \end{bmatrix}, \ldots,$$

$$X_{x_{\max}} = \begin{bmatrix} x_{x_{\max},1} & x_{x_{\max},n+1} & \cdots & x_{x_{\max},n^2-n+1} \\ x_{x_{\max},2} & x_{x_{\max},n+2} & \cdots & x_{x_{\max},n^2-n+2} \\ \vdots & \vdots & & \vdots \\ x_{x_{\max},n} & x_{x_{\max},2n} & \cdots & x_{x_{\max},n^2} \end{bmatrix}.$$

Halutaan, että ensimmäiset n^2 muuttujaa eli muuttujat $r_1, \ldots, r_{n^2}$ vastaavat muita muuttujia. Toisin sanoen, mikäli eräs muuttuja $x_{v,p} = 1$, $r_p = v$. Tämä ehto toteutuu seuraavan yhtälöryhmän ollessa tosi:

$$\begin{aligned} r_1 - x_{\min} \cdot x_{x_{\min},1} - (x_{\min} + 1) \cdot x_{x_{\min}+1,1} - \cdots - x_{\max} \cdot x_{x_{\max},1} &= 0 \\ r_2 - x_{\min} \cdot x_{x_{\min},2} - (x_{\min} + 1) \cdot x_{x_{\min}+1,2} - \cdots - x_{\max} \cdot x_{x_{\max},2} &= 0 \\ &\vdots \\ r_{n^2} - x_{\min} \cdot x_{x_{\min},n^2} - (x_{\min} + 1) \cdot x_{x_{\min}+1,n^2} - \cdots - x_{\max} \cdot x_{x_{\max},n^2} &= 0. \end{aligned}$$

Koska tiedetään, että $x_{v,p} \in \{0, 1\}$ ja $\sum_{v=x_{\min}}^{x_{\max}} x_{v,p} = 1$, vastaa muuttujan r_p arvo muuttujan, jonka arvo on 1, kertoimen vastalukua. Esimerkiksi, jos $x_{x_{\min}+1,3} = 1$, saadaan yhtälö

$$r_3 - (x_{\min} + 1) = 0,$$

josta nähdään selvästi, että $r_3 = x_{\min} + 1$. MATLAB-koodissa yhtälöryhmän kertoimet on tallennettu matriiseihin

$$A_{eq} = \begin{bmatrix} I_{n^2} & -x_{\min} \cdot I_{n^2} & -(x_{\min} + 1) \cdot I_{n^2} & \cdots & -x_{\max} \cdot I_{n^2} \end{bmatrix} \text{ ja } \mathbf{b}_{eq}^T = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix}.$$

Jotta matriisin R alkiot pysyvät annettujen rajojen sisällä, täytyy varmistaa, että matriisin jokaisessa kohdassa on täsmälleen yhtä muuttujaa vastaava arvo. Tämä ehto toteutuu, kun

$$\begin{aligned} x_{x_{\min},1} + x_{x_{\min}+1,1} + \cdots + x_{x_{\max},1} &= 1 \\ x_{x_{\min}+1,2} + x_{x_{\min}+1,2} + \cdots + x_{x_{\max},2} &= 1 \\ &\vdots \\ x_{x_{\min},n^2} + x_{x_{\min}+1,n^2} + \cdots + x_{x_{\max},n^2} &= 1, \end{aligned}$$

koska muuttujat $x_{v,p}$ voivat saada arvoja vain joukosta $\{0,1\}$. Yhtälöryhmä ilmaistaan MATLAB-koodissa matriisimuodossa, jossa

$$A_{eq} = \begin{bmatrix} 0_{n^2} & I_{n^2} & I_{n^2} & \cdots & I_{n^2} \end{bmatrix} \text{ ja } \mathbf{b}_{eq}^T = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}.$$

Toisen ehdon mukaan sama luku saa esiintyä matriisissa R korkeintaan k kertaa. Toisin sanoen luku $v \in [x_{\min}, x_{\max}]$ voi olla korkeintaan k:ssa matriisin R paikassa samanaikaisesti. Edelleen tämä tarkoittaa sitä, että tietyllä indeksin v arvolla muuttujista $x_{v,p}$ vain k kappaletta voivat saada arvon 1. Yhtälöryhmäksi saadaan täten

$$\begin{aligned} x_{x_{\min},1} + x_{x_{\min},2} + \cdots + x_{x_{\min},n^2} &\leq k \\ x_{x_{\min}+1,1} + x_{x_{\min}+1,2} + \cdots + x_{x_{\min}+1,n^2} &\leq k \\ &\vdots \\ x_{x_{\max},1} + x_{x_{\max},2} + \cdots + x_{x_{\max},n^2} &\leq k. \end{aligned}$$

Koska tietty luku saa esiintyä matriisissa myös vähemmän kuin k kertaa, ehto muodostuu epäyhtälöistä. MATLAB-koodissa epäyhtälöryhmä tallennetaan muodossa

$$A = \begin{bmatrix} 0_{1,n^2} & J_{1,n^2} & 0_{1,n^2} & \cdots & 0_{1,n^2} \\ 0_{1,n^2} & 0_{1,n^2} & J_{1,n^2} & \cdots & 0_{1,n^2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0_{1,n^2} & 0_{1,n^2} & 0_{1,n^2} & \cdots & J_{1,n^2} \end{bmatrix} \text{ ja } \mathbf{b} = \begin{bmatrix} k \\ k \\ \vdots \\ k \end{bmatrix}.$$

Huomioitavaa on se, että epäyhtälöryhmät tallennetaan MATLAB-koodissa eri matriiseihin kuin yhtälöryhmät. (kts. 4.1).

Koska koodi suosii itseisarvoiltaan pieniä lukuja, on varmistettava, että matriisissa R esiintyy myös vähintään yksi luku, joka on lähempänä ylärajaa. Asetetaan alaraja $s \in$

$(x_{\min}, x_{\max})$ matriisin R suurimmalle alkiolle. Tätä ehtoa vastaa yhtälöryhmä

$$\begin{aligned} x_{s,1} + x_{s,2} + \cdots + x_{s,n^2} &\geq 1 \\ x_{s+1,1} + x_{s+1,2} + \cdots + x_{s+1,n^2} &\geq 1 \\ &\vdots \\ x_{x_{\max},1} + x_{x_{\max},2} + \cdots + x_{x_{\max},n^2} &\geq 1. \end{aligned}$$

MATLAB-komennon intlinprog toimivuuden vuoksi epäyhtälöiden suunnat on käännettävä, joten yhtälöryhmä tulee kertoa luvulla -1, jolloin saadaan

$$\begin{aligned} -x_{s,1} - x_{s,2} - \cdots - x_{s,n^2} &\leq -1 \\ -x_{s+1,1} - x_{s+1,2} - \cdots - x_{s+1,n^2} &\leq -1 \\ &\vdots \\ -x_{x_{\max},1} - x_{x_{\max},2} - \cdots - x_{x_{\max},n^2} &\leq -1. \end{aligned}$$

Jokaisessa yhtälössä siis muuttujien arvoista vähintään yhden on oltava 1, jotta epäyhtälöryhmä toteutuu. Matriisimuodossa

$$A = \begin{bmatrix} 0 & 0 & \cdots & 0 & -J_{1,x_{\max}-s} \end{bmatrix} \text{ ja } \mathbf{b} = \begin{bmatrix} -1 \end{bmatrix}.$$

Matriisin A kaikki alkiot, jotka vastaavat parametria s pienempiä arvoja, saavat siis arvon 0. Vastaavasti voidaan määrittää maksimi l matriisin minimiarvolle. Tällöin epäyhtälöryhmän

$$\begin{aligned} -x_{x_{\min},1} - x_{x_{\min},2} - \cdots - x_{x_{\min},n^2} &\leq -1 \\ -x_{x_{\min}+1,1} - x_{x_{\min}+1,2} - \cdots - x_{x_{\min}+1,n^2} &\leq -1 \\ &\vdots \\ -x_{l,1} - x_{l,2} - \cdots - x_{l,n^2} &\leq -1. \end{aligned}$$

on toteuduttava. Jälleen matriisimuodossa voidaan kirjoittaa

$$A = \begin{bmatrix} 0_{1,n^2} & -J_{1,l-x_{\min}} & 0 & 0 & \cdots & 0 \end{bmatrix} \text{ ja } \mathbf{b} = \begin{bmatrix} -1 \end{bmatrix}.$$

Minimoitavana kohdefunktiona on

$$\mathbf{f}^T \mathbf{x},$$

jossa kohdevektori $\mathbf{f} \in \mathbb{R}^{n^2(m+1)}$. Aiemmin esitettyjen ehtojen pohjalta tiedetään, että muuttujavektorin $\mathbf{x}$ ensimmäiset n^2 alkiota voi saada arvoja väliltä $[x_{\min}, x_{\max}]$, kun loput alkioista koostuvat luvuista 0 ja 1. Asetetaan kohdevektorin $\mathbf{f}$ ensimmäiset n^2 alkiota nolliksi, jotta muuttujat $r_1, r_2, \ldots, r_{n^2}$ voivat saada mitä tahansa arvoja määrätyltä väliltä. Tällöin kohdefunktio voidaan kirjoittaa muodossa

$$\mathbf{f}^T\mathbf{x} = \begin{bmatrix} f_{n^2+1} & f_{n^2+2} & \cdots & f_{n^2(m+1)} \end{bmatrix} \begin{bmatrix} x_{x_{\min},1} \\ x_{x_{\min},2} \\ \vdots \\ x_{x_{\max},n^2} \end{bmatrix},$$

Algoritmin ehtoina ovat satunnaisuus ja itseisarvoltaan pienien lukujen suosiminen. Algoritmi saadaan suosimaan itseisarvoltaan pieniä lukuja, kun asetetaan kohdevektorin $\mathbf{f}$ alkiot f_i, kun $i = n^2+1, \ldots, n^2(m+1)$, vastaamaan lukujen $[x_{\min}, x_{\max}]$ itseisarvoja. Näin ollen minimointiongelmassa itseisarvoltaan pieniä lukuja vastaavat muuttujat saavat todennäköisemmin arvon 1 kuin suuria lukuja vastaavat. Luotavat matriisit ovat satunnaisia kohdefunktion ollessa satunnainen. Kohdefunktion satunnaisuus saavutetaan lisäämällä jokaiseen kohdevektorin $\mathbf{f}$ alkioon f_i satunnainen luku t standardinormaalijakaumasta, joka on kerrottu säädettävällä parametrilla σ.

Kun huomioidaan sekä pienien lukujen suosiminen että kohdevektorin satunnaisuus, saadaan kohdevektorin alkioiden arvoksi

$$f_i = |v + \sigma t|,$$

jossa $i = n^2+1, \ldots, n^2(m+1)$, v on muuttujaa $x_{v,p}$ vastaava arvo ja t on satunnainen luku normaalijakaumasta.

Luotavan matriisin R kääntyvyyteen on myös mahdollista vaikuttaa. Jos halutaan, että matriisista tulee singulaarinen, on annettava ehto, jonka mukaan matriisin R sarakkeiden tulee olla kohtisuorassa singulaarisen matriisin $S \in \mathbb{R}^{n\times n}$ sarakkeiden kanssa. Matriisin S kaikki rivit on oltava lisäksi samassa tasossa toistensa kanssa. Tällainen matriisi voi olla esimerkiksi

$$S = J_{n\times n}.$$

Vähintään 2 matriisin R saraketta ovat tällöin samansuuntaiset eli ne ovat lineaarisesti riippuvia. Matriisien R ja S sarakkeiden kohtisuoruus saavutetaan, kun

$$\begin{aligned} s_1r_1 + s_2r_2 + \cdots + s_nr_n &= 0 \\ s_{n+1}r_{n+1} + s_{n+2}r_{n+2} + \cdots + s_{2n}r_{2n} &= 0 \\ &\vdots \\ s_{n^2-n+1}r_{n^2-n+1} + s_{n^2-n+2}r_{n^2-n+2} + \cdots + s_{n^2}r_{n^2} &= 0, \end{aligned}$$

jossa s_i kuvaa matriisin S alkioita. MATLAB-koodissa ehto voidaan tallentaa matriisimuodossa

$$A = \begin{bmatrix} J_{1,n} & 0_{1,n} & \cdots & 0_{1,n} & 0_{n\times n^2 m} \\ 0_{1,n} & J_{1,n} & \cdots & 0_{1,n} & 0_{n\times n^2 m} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0_{1,n} & 0_{1,n} & \cdots & J_{1,n} & 0_{n\times n^2 m} \end{bmatrix} \text{ ja } \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

4.3 Esimerkki 3×3 -matriisin luomisesta

Luodaan algoritmin avulla satunnaisia 3×3 -matriiseja. Aluksi on asetettava sopivat arvot parametreille, jotta saatavan matriiisin avulla voitaisiin luoda mielekäs tehtävä. Koska tehtävästä halutaan käsinlaskettava, luvut eivät saa olla itseisarvoltaan liian suuria, mutta kuitenkin riittävän suuria, jotta ratkaisut eivät ole triviaaleja. Asetetaan matriisin alkiot olemaan kokonaislukuja välillä $[-20, 20]$. Tämä ei kuitenkaan tarkoita, että suurin matriisissa esiintyvä luku tulee olemaan välttämättä 20 ja pienin -20. Valitaan matriisissa suurimman luvun alarajaksi 11 ja pienimmän luvun ylärajaksi -9.

Tehtävän mielekkyyden kannalta on myös tärkeää, ettei matriisissa ole liian paljon toistuvia lukuja. Koska kyseessä on 3×3 -matriisi, valitaan parametrin k arvoksi 2. Näin valittaessa matriisissa ei voi esiintyä esimerkiksi nollariviä tai -saraketta.

Parametrin σ sopiva suuruus on helpointa määrittää kokeilemalla. Jotta matriisissa esiintyy mahdollisimman eri suuruisia alkioita kuitenkin siten, että suurin osa alkioista on itseisarvoltaan pieniä, valitaan $\sigma = 3$. Mikäli parametrin σ arvoa kasvatettaisiin alkioiden itseisarvot kasvaisivat. Vastaavasti parametria pienennettäessä itseisarvot pienenevät. Tehdään 3 esimerkkiajoa kyseisillä parametreilla. Saadaan matriisit

$$\begin{bmatrix} 13 & 3 & -9 \\ -2 & -1 & -1 \\ 2 & -4 & -2 \end{bmatrix}, \begin{bmatrix} 0 & 0 & -1 \\ -1 & 16 & -2 \\ -3 & 3 & -9 \end{bmatrix}, \begin{bmatrix} 0 & -2 & -1 \\ -10 & 2 & 11 \\ -4 & 0 & 1 \end{bmatrix}.$$

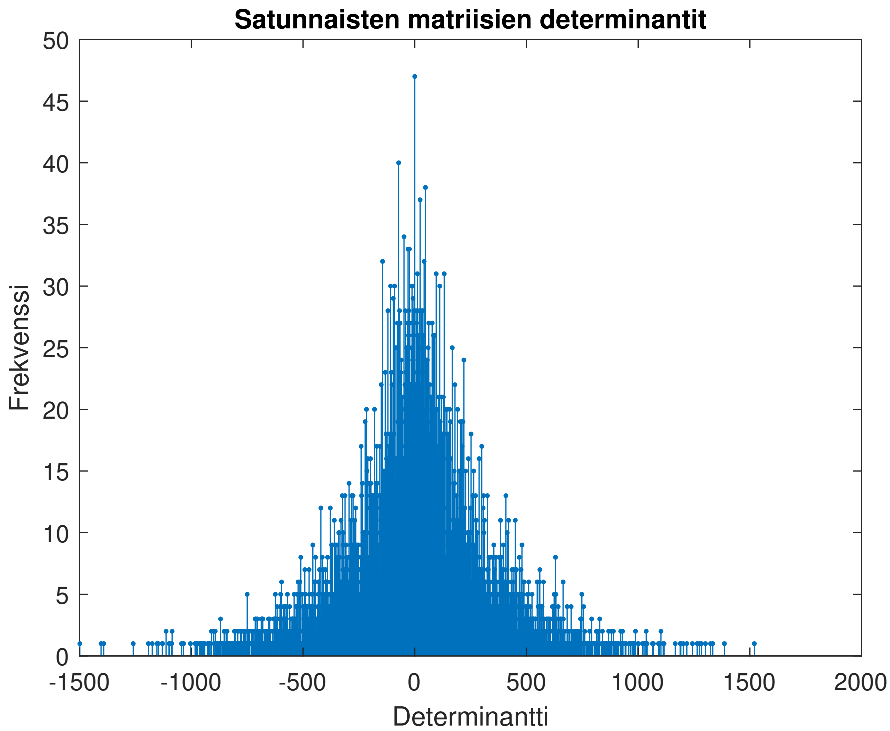
Matriiseista havaitaan kaikkien asetettujen ehtojen toteutuvan. Havaitaan myös, että annetuilla parametreilla kaikki paitsi rajoitetut alkiot ovat itseisarvoltaan melko pieniä.

Tuotettavia matriiseja on tarkoitus käyttää korkeakouluissa kurssien harjoitustehtävissä ja tenteissä. Usein kurssit ovat opiskelijamäärältään suuria ja esimerkiksi saman sähköisen tentin voi suorittaa yli 800 opiskelijaa. Näin ollen on tärkeää, ettei kahdelle opiskelijalle generoituisi samaa matriisia tenttitilanteessa. Tutkitaan 10 000:n luodun matriisin satunnaisuutta alkuperäisillä parametreilla. Näistä matriiseista yksikään ei ollut identtisiä toistensa kanssa vaan kaikki matriisit olivat jollakin tavalla toisistaan eroavia.

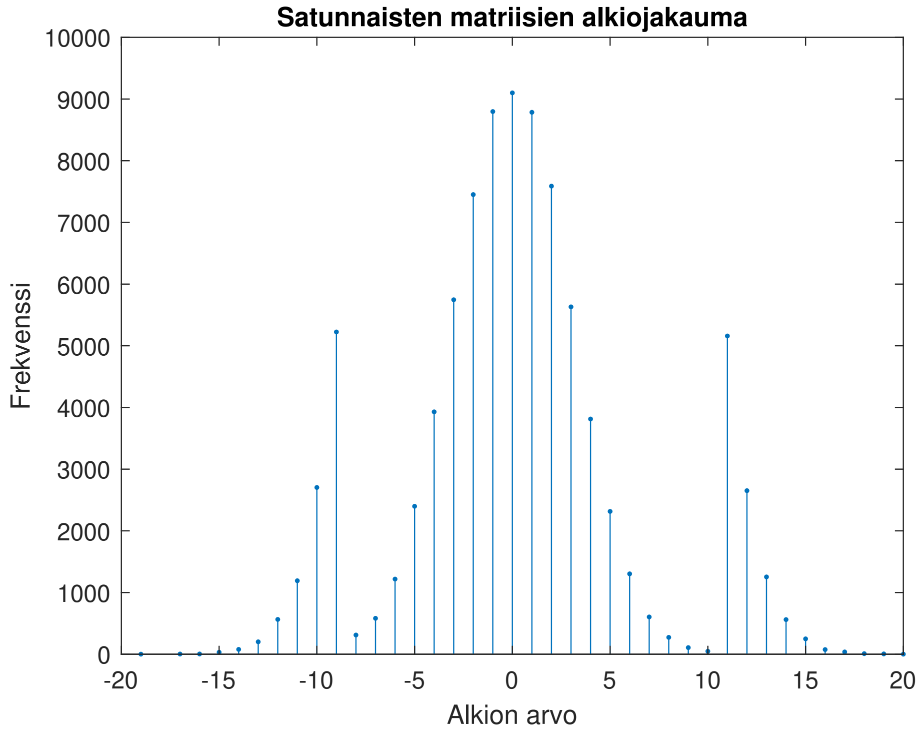
Tutkitaan seuraavaksi näiden matriisien determinantteja. Kuvasta 4.1 huomataan, että determinanttien, joiden itseisarvo on pieni, frekvenssi on suuri. Kuvaajassa korkein frekvenssi on determinantilla 0. Näitä arvoja vastaavat matriisit, jotka ovat singulaarisia. Singulaaristen matriisien frekvenssi on kuvaajassa 47, mikä tarkoittaa, että luoduista matriiseista 0.47% ei ole kääntyviä. Samalla tavalla voidaan tutkia luoduissa matriiseissa esiintyvien alkioiden yleisyyttä. Kuvasta 4.2 havaitaan, miten algoritmi suosii itseisarvoltaan pieniä lukuja. Huomioitavaa on myös, että algoritmi suosii näillä parametrien arvoilla lukuja -9 ja 11. Kun muistetaan, että $s = 11$ ja $l = -9$, voidaan todeta algoritmin suosivan näitä valittuja rajoja ja lukuja, jotka ovat itseisarvoltaan hieman näitä suurempia. Puolestaan lukujen, joiden itseisarvon suuruus on yli 15, esiintymistiheys on erittäin pieni.

Sama ilmiö voidaan havaita jakaumasta 4.3, jossa parametrien l ja s arvot on vaihdettu. Algoritmi suosii edelleen eniten itseisarvoltaan pieniä lukuja, mutta myös lukuja, jotka ovat vähän suurempia kuin s tai pienempiä kuin l. Kyseisten parametrien muutos ei vaikuta merkittävästi determinanttijakaumaan.

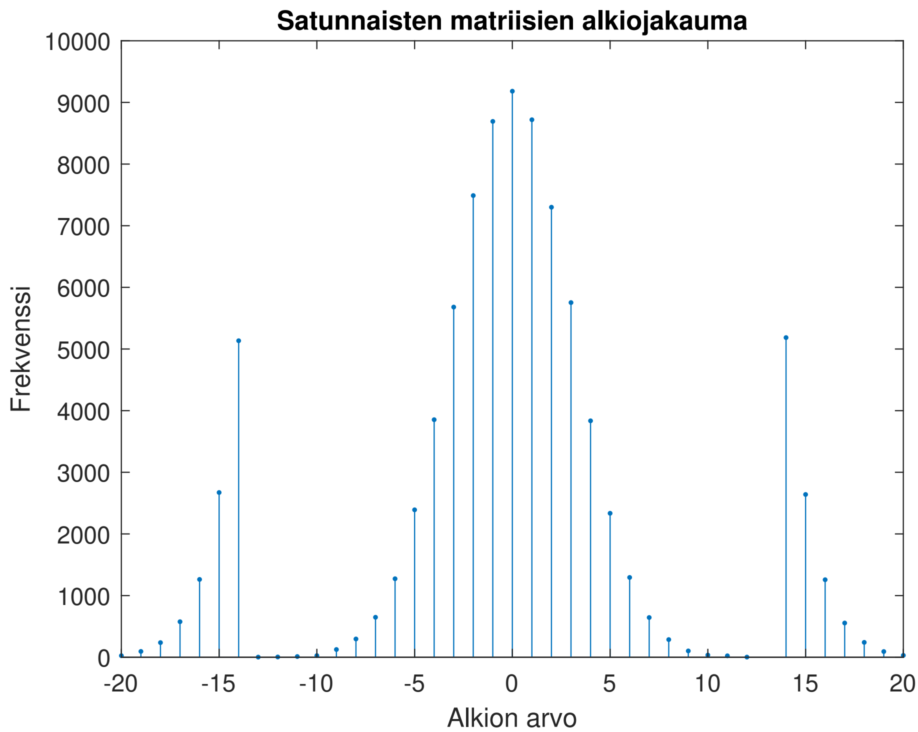
Mikäli alkioiden esiintymismääriä matriiseissa haluttaisiin tasoittaa, voitaisiin se tehdä kasvattamalla parametria σ. Kuvasta 4.4 nähdään, että itseisarvoltaan pieniä lukuja esiintyy tällöin huomattavasti vähemmän ja vastaavasti suurempia lukuja enemmän. Myös parametrien l ja s vaikutus alkioiden esiintyvyyteen pienenee. Itseisarvoltaan suurten alkioiden esiintymistiheyden kasvu aiheuttaa determinanttien arvojen kasvamista, mikä nähdään kuvasta 4.5.



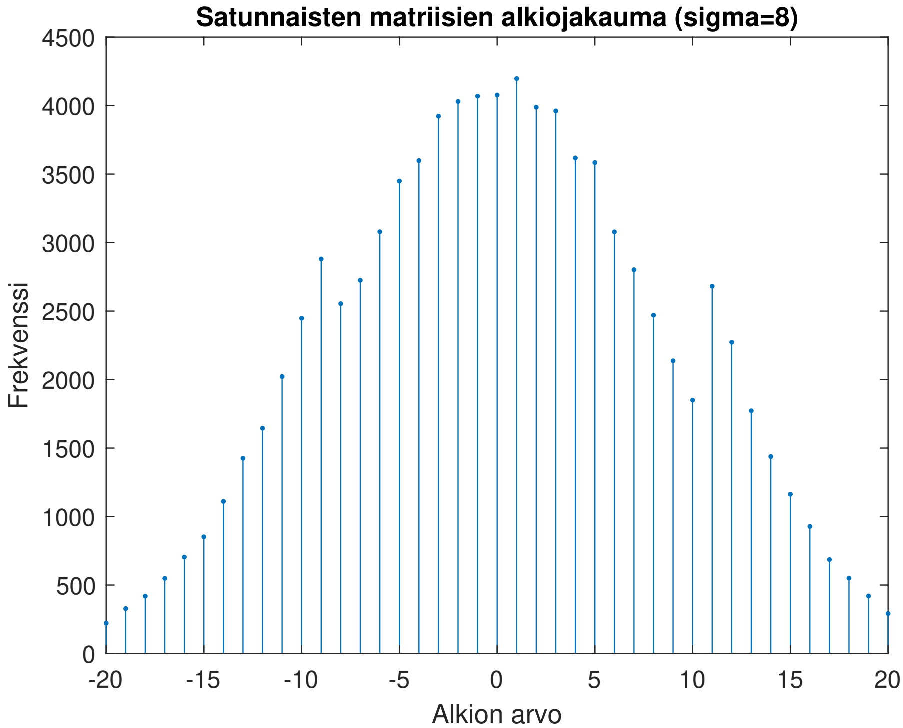
Kuva 4.1. *10 000:n luodun matriisin determinanttien frekvenssijakauma, kun* $\sigma = 3$, $l = -9$ *ja* $s = 11$



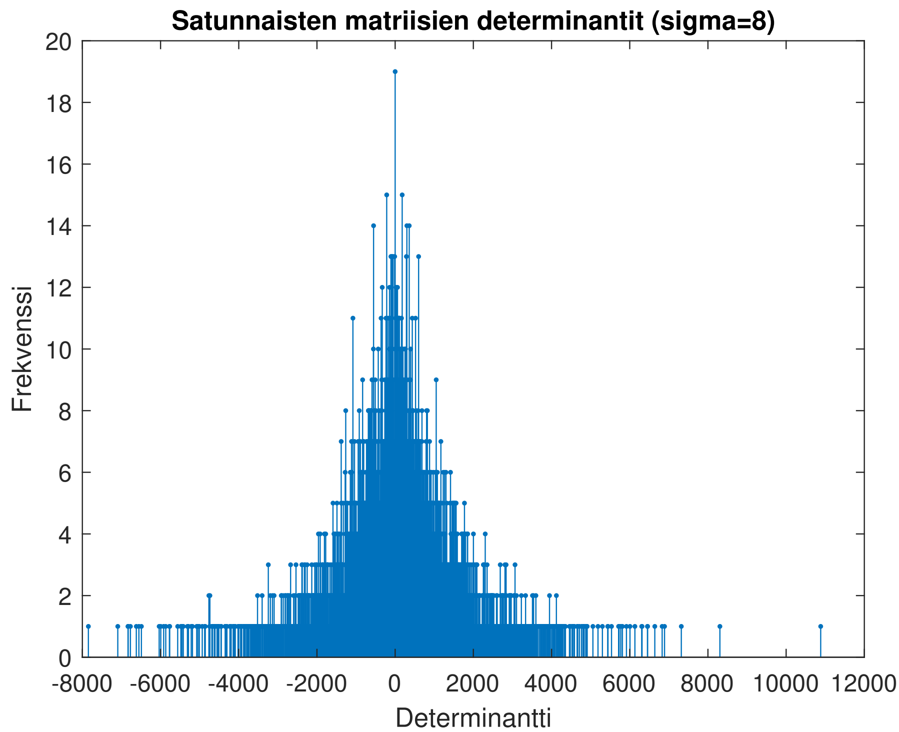
Kuva 4.2. *10 000:n luodun matriisin alkioiden frekvenssijakauma , kun* $\sigma = 3$*,* $l = -9$ *ja* $s = 11$



Kuva 4.3. *10 000:n luodun matriisin alkioiden frekvenssijakauma, kun* $\sigma = 3$*,* $l = -14$ *ja* $s = 14$

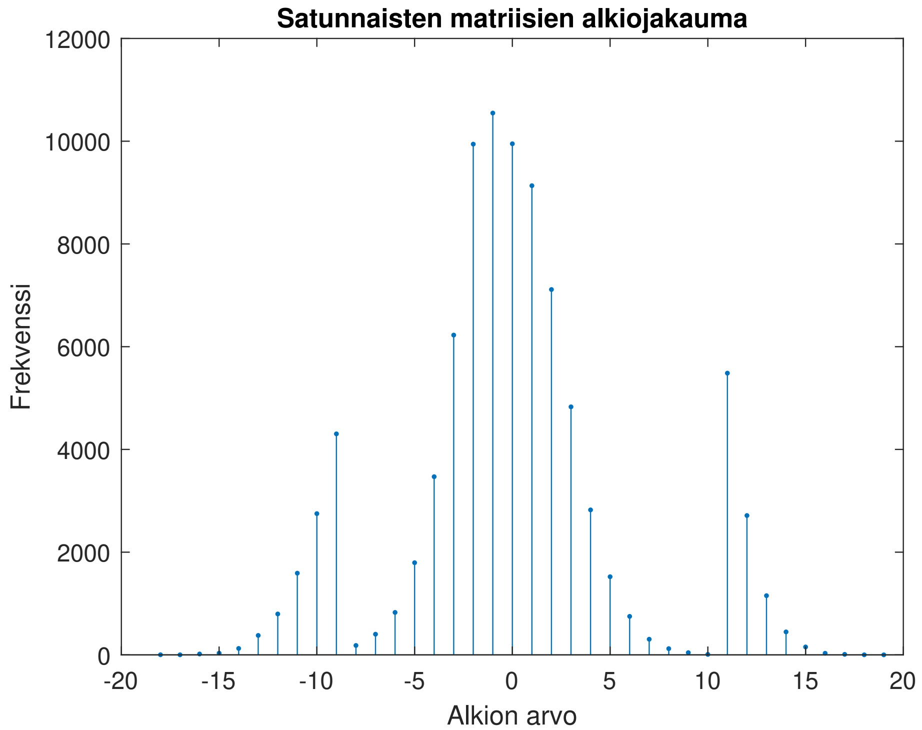


Kuva 4.4. *10 000:n luodun matriisin alkioiden frekvenssijakauma, kun* $\sigma = 8$, $l = -9$ *ja* $s = 11$



Kuva 4.5. *10 000:n luodun matriisin determinanttien frekvenssijakauma, kun* $\sigma = 8$, $l = -9$ *ja* $s = 11$

Matriiseilla ei havaittu toistuvuutta myöskään singulaarisuusehdon ollessa voimassa 10 000:n matriisin otoksella, kun $\sigma = 3,\ s = 11$ ja $l = -9$. Singulaarisuusehto ei vaikuta suuresti alkioiden jakaumaan, vaan se on muodoltaan vastaava kuin kuvassa 4.2 kuten huomataan kuvasta 4.6. Singulaarisuus ehtoa käyttäen kaikkien luotavien matriisien determinantti on tietysti 0.



Kuva 4.6. *10 000:n luodun singulaarisen matriisin alkioiden frekvenssijakauma, kun* $\sigma = 3$, $l = -9$ *ja* $s = 11$

5. TUTKIMUS

Sähköisiin tentteihin liittyvä tutkimus toteutettiin lukuvuoden 2022-2023 aikana Tampereen ammattikorkeakoulussa. Tutkimuksen tavoitteena oli selvittää tekniikan alojen matematiikan opiskelijoiden kokemuksia sähköisestä matematiikan tentistä. Tutkimukseen kerättiin aineistoa ensimmäisen vuoden opiskelijoilta. Matematiikan tenttejä ei ole ollut mahdollista järjestää kyseisessä koulussa laajamittaisesti, sillä opetukseen ei kuulu matemaattisten notaatioiden kirjoittaminen kaavaeditorilla. TUNI:ssa pilotoitavana olevia piirtoalustoja hyödyntämällä opiskelijoiden ei tarvitse käyttää tentissä kaavaeditoreja, vaan voivat kirjoittaa matemaattiset notaatiot samaan tapaan kuin paperille.

5.1 Tutkimuskysymykset

Tutkimuksessa halutaan selvittää opiskelijoiden mielipiteitä sähköisen tentin sujuvuudesta ja toimivuudesta. Koska piirtoalustat ovat pilotoitavana, on tärkeää selvittää, miten hyvin ne soveltuvat matematiikan sähköisten tenttien suorittamiseen ja aiheuttaako niiden käyttäminen ongelmia. Toisaalta tutkimuksessa pyritään selvittämään pitävätkö opiskelijat esimerkiksi piirtoalustaa, laskinta ja paperin käyttöä tärkeänä tenttiä tehdessä. Lisäksi opiskelijoilta halutaan tutkimuksessa kehitysehdotuksia EXAM-tentin suorittamiseen. Tutkimuskysymyksiksi muodostuivat seuraavat:

TK1 Kuinka käyttäjäystävällisenä opiskelijat pitävät matematiikan tenttiin vastaamista tietokoneeseen liitetyn piirtoalustan avulla?

TK2 Mitä asioita opiskelijat pitävät tärkeimpinä matematiikan tentissä?

TK3 Miten opiskelijat kehittäisivät sähköiseen tenttiin vastaamista?

TK4 Mikäli saisivat valita, tekisivätkö opiskelijat jatkossa matematiikan tentit sähköisinä EXAM-tentteinä vai perinteisinä paperitentteinä?

5.2 Ohjeistukset piirtoalustojen käyttöön

TAMK:ssa ei olla aikaisemmin käytetty piirtoalustoja matematiikan sähköisessä tentissä, joten suoritettavaa tutkimusta varten luotiin ohjeistus piirtoalustojen ja tarvittavien ohjelmien käyttöön. Kyseinen ohjeistus löytyy liitteestä C.

Oppilaitos	Piirtoalusta käytössä	Piirtoalustan ohjeistus sähköisesti saatavilla
Aalto-yliopisto	-	-
Helsingin yliopisto	X	X
Itä-Suomen yliopisto	-	-
Jyväskylän yliopisto	-	-
Lapin yliopisto	-	-
Lappeenrannan ja Lahden teknillinen yliopisto	X	X
Oulun yliopisto	X	-
Kauppakorkeakoulu Hanken	-	-
Taideyliopisto	-	-
Turun yliopisto	X	X
Vaasan yliopisto	-	-
Åbo Akademi	-	-

Taulukko 5.1. *Piirtoalustoja EXAM-tenteissä hyödyntävät yliopistot Suomessa*

Tampereen yliopiston lisäksi vastaavanlaisia piirtoalustoja on tiedettävästi käytössä myös usean suomalaisen yliopiston EXAM-tiloissa, kuten nähdään taulukosta 5.1. On mahdollista, että myös muissa yliopistoissa käytetään piirtoalustoja, mutta tieto niiden hyödyntämisestä ei ole internetissä helposti saatavilla oppilaitoksen ulkopuolisille. Lähes kaikkien yliopistojen, jotka käyttävät piirtoalustoja, internet-sivustoilla oli saatavana ohjeistus piirtoalustan käyttöön sähköisessä tentissä. Oulun yliopiston sivustojen mukaan ohjeistus löytyy fyysisenä tenttitilasta, vaikkei sitä sähköisenä olekaan saatavilla. [12] Saatavilla olevat ohjeistukset löytyvät tästä:

- Helsingin yliopisto
- Lappeenrannan ja Lahden teknillinen yliopisto (LUT)
- Turun yliopisto

LUT:n ja Turun yliopiston ohjeistukset piirtoalustan käyttöön ovat melko samanlaiset tätä tutkimusta varten tehtyyn verrattuna. Myös näissä ohjeistetaan piirtämiseen Word sekä PowerPoint ohjelmissa, ja ne sisältävät lyhyen käyttöönotto opastuksen. [2][6] Helsingin yliopiston ohjeistus on laajempi sisältäen ohjevideon piirtoalustan käytöstä ja tiedostojen palauttamisesta. Helsingin yliopiston ohjeistuksessa ohjataan opiskelijaa piirtämään Paint- tai Xournal++ -ohjelmien avulla. [1]

5.3 Aineiston kerääminen

Tutkimuksessa aineistoa kerättiin TAMK:n ensimmäisen vuoden tekniikan alan opiskelijoilta. Tutkimukseen osallistuneet opiskelijat suorittivat uusintatentin vapaaehtoisesti sähköisenä tenttinä, jossa oli käytössä piirtoalusta. Sähköiseen tenttiin vastanneet opiskelijat saivat täytettäväkseen liitteestä B löytyvän kyselyn. Kyselyn täyttäminen ei kuulunut tenttisuoritukseen, vaan opiskelijat täyttivät sen omalla ajallaan.

Kysely koostui opiskelijoiden taustaan ja sähköiseen tenttiin liittyviä kysymyksiä. Opiskelijoiden taustasta haluttiin tietää, mikä oli heidän toisen asteen koulutustaustansa ja oliko sähköinen tenttiminen heille entuudestaan tuttua sähköisen ylioppilaskokeen vuoksi. Kyselyn avulla haluttiin selvittää myös olivatko opiskelijat tottuneet kosketusnäyttöjen käyttöön tietokoneella, jotta tuloksia analysoidessa olisi mahdollista ottaa huomioon opiskelijoiden tottumattomuuden vaikutus. Kyselyssä pyrittiin selvittämään mahdollisimman monipuolisesti opiskelijoiden kokemuksia sähköisestä tentistä. Kysely sisälsi avoimia kysymyksiä, jotta vastauksia pystyttiin tutkimaan laadullisesti. Lisäksi tutkimuskysymyksiin TK1 ja TK2 etsittiin vastausta likert-asteikollisten kysymysten avulla.

5.4 Kyselyn tulokset

Kyselyyn vastasi yhteensä 16 TAMK:n opiskelijaa aikavälillä 11/2022–5/2023. Vastanneista puolet olivat suorittaneet lukion ennen ammattikorkeakouluopintoja, kun taas 6 opiskelijaa oli käynyt ammattikoulun. Loput 2 opiskelijaa olivat suorittaneet joko kolmois- tai kaksoistutkinnon. Matematiikan ylioppilaskokeen suorittaneita vastaajien joukossa oli 9 kappaletta, joista 7 oli kirjoittanut kokeen paperisena ja vain 2 sähköisenä. Suurin osa vastanneista ei ollut myöskään tottunut käyttämään tietokonetta kosketusnäytön avulla. Vain 5 opiskelijaa oli tottunut kosketusnäyttöjen käyttöön.

5.4.1 Millaiseksi opiskelijat kokivat sähköiseen tenttiin vastaamisen paperitenttiin verrattuna?

Kaikki 16 opiskelijaa vastasivat tähän kysymykseen. Näistä vastauksista suurin osa oli joko positiivisia tai neutraaleja matematiikan sähköistä tenttiä kohtaan. Positiivisiksi vastauksiksi tulkittiin sellaiset vastaukset, joissa esiintyi vain positiivista palautetta sähköisestä tentistä. Vastaavasti negatiiviseksi palautteeksi luettiin vastaukset, joissa ei esiintynyt positiivisia kommentteja. Neutraaleihin luettiin vastaukset, joissa oli sekä positiivisia että negatiivisia kommentteja tai kommentteja tenttimismuotojen erilaisuudesta ottamatta kantaa paremmuuteen. Kokonaisuudessaan eniten positiivista palautetta sai piirtonäytön käytön kätevyys sekä ajankohdan vapaavalintaisuus. Negatiivista palautetta eniten saivat tenttisuorituksen tekniset ongelmat ja tottumattomuus. Opiskelijat, jotka olivat kirjoittaneet matematiikan sähköisenä vastasivat kysymykseen positiivisesti. Vastaavasti opiskelijat, jotka olivat tottuneet käyttämään tietokonetta kosketusnäytön avulla, eivät myöskään kokeneet sähköisen tentin olleen negatiivinen kokemus. Muiden opiskelijoiden vastaukset vaihtelivat enemmän. Eräs opiskelija, joka ei ollut tottunut käyttämään tietokonetta kosketusnäytön avulla, vastasi kysymykseen näin:

> "Kokonaisuudessa jopa helpompaa tehdä sähköisenä. Piirtonäytön käytön oppi nopeasti ja pidin sillä kirjoittamista kätevänä."

Kaikkien vastanneiden keskuudessa sähköisen tentin suurimmaksi hankaluudeksi nousi tekniset ongelmat. Vastausten mukaan tällaisia ongelmia olivat Word-ohjelman "tilttaus", näytön pomppiminen ja esimerkiksi sähköisen pyyhekumin käytön hankaluus. Vastauksista ilmenee, että tekniset ongelmat eivät suoranaisesti vaikuttaneet vastauksiin ja tentin tekemiseen, mutta ne veivät turhaa aikaa ja aiheuttivat stressiä, toisin kuin paperitentissä.

5.4.2 Kokivatko opiskelijat minkäänlaisia ongelmia sähköisessä tentissä ja millaisia ongelmat olivat?

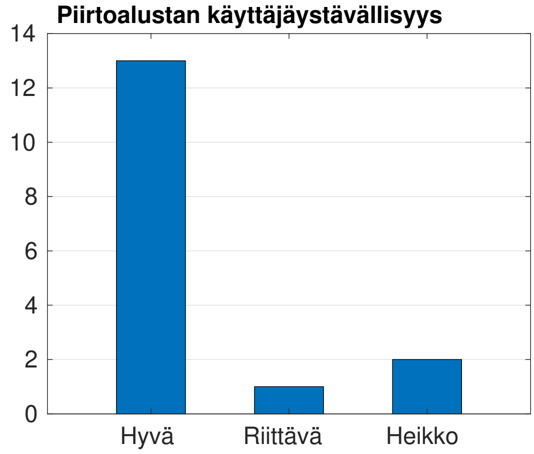
Vastanneista yhteensä 7 koki ongelmia tenttisuorituksessa. Ongelmia kokivat niin opiskelijat, jotka olivat tottuneet kosketusnäyttöjen käyttöön tietokoneilla sekä käyttöön tottumattomia opiskelijoita. Sähköisen ylioppilaskokeen suorittaneet opiskelijat eivät kokeneet ongelmia. Opiskelijoista 2 koki ongelmia tietokoneen laskimen käytössä. Niin ikään 2 opiskelijaa koki ongelmia navigoinnin kanssa tietokoneella ja moodlessa. Eräs opiskelija kuvaili ongelmiaan seuraavasti:

> ”Tentin aikana navigoinnissa oli vähän haasteita. Jostakin syystä palautuskansioita ei meinannut löytyä. Laskimen löytäminen koneelta oli haastavaa, kun ei muistanut laskimen nimeä. Myös laskimen käyttämisessä oli hieman ongelmia.”

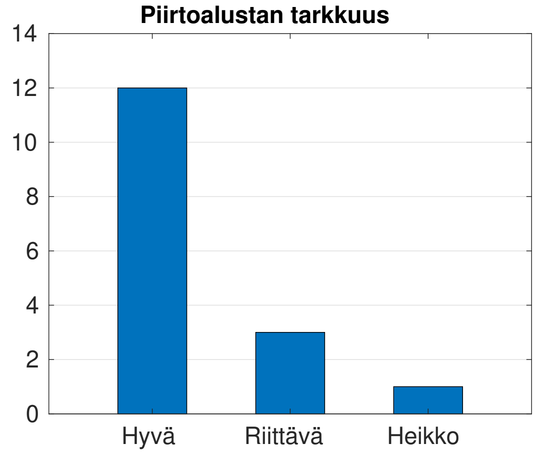
Kahdella opiskelijalla Word-ohjelma ei toiminut kunnolla, vaan opiskelijoiden mukaan ohjelma "tilttasi" tai "pimeni" välillä. Myöskään tiedostojen tallentaminen PDF-muodossa ei sujunut kaikilla ongelmitta. Eräs opiskelija koki ongelmia kynän toiminnan kanssa.

5.4.3 Mitä mieltä opiskelijat olivat piirtoalustasta ja ohjelmien käytettävyydestä?

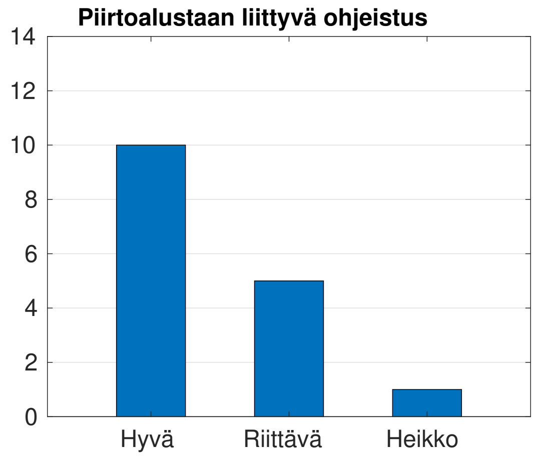
Kysymyksen tarkoituksena oli selvittää sähköisessä tentissä käytetyn piirtoalustan ja piirtämiseen käytettävien ohjelmien toimivuutta sekä luodun ohjeistuksen hyödyllisyyttä. Kysymyksen vastaukset on ilmoitettu kuvissa 5.1–5.4. Kuvaajista nähdään, että suurin osa vastaajista piti piirtoalustan käyttäjäystävällisyyttä ja tarkkuutta hyvänä. Myös ohjeistus oli lähes kaikkien mielestä hyvä tai riittävä. Yksikään opiskelija ei pitänyt ohjelmien käyttäjäystävällisyyttä heikkona.



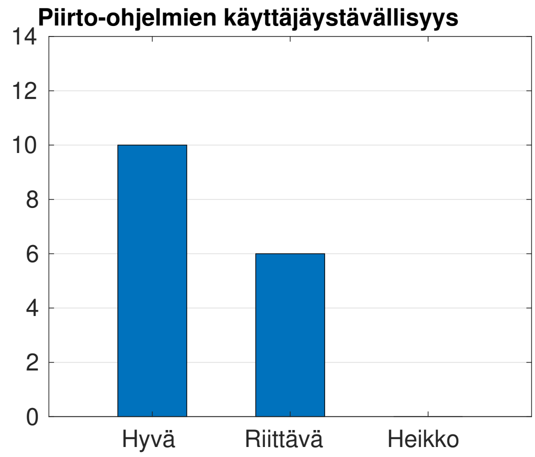
Kuva 5.1. *Kyselyn kohta 8.1*



Kuva 5.2. *Kyselyn kohta 8.2*



Kuva 5.3. *Kyselyn kohta 8.3*



Kuva 5.4. *Kyselyn kohta 8.4*

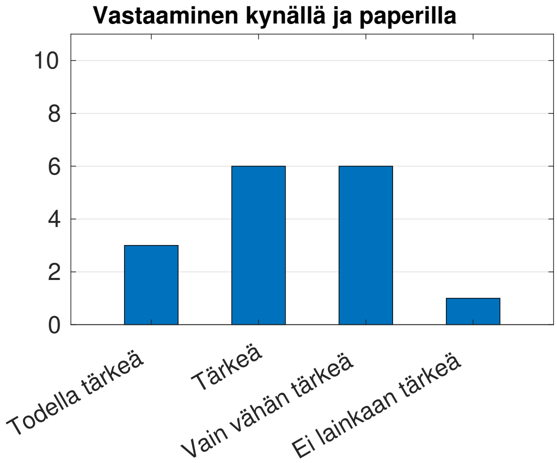
5.4.4 Kokivatko opiskelijat minkäänlaisia ongelmia piirtoalustan käytössä ja millaisia ongelmat olivat?

Kysymyksissä 9 ja 10 pyrittiin selvittämään piirtoalustan käyttöön liittyviä ongelmia. Opiskelijoista 6 koki ongelmia. Ongelmia kokeneet opiskelijat olivat sekä kosketusnäyttöihin tottuneita ja tottumattomia. Useammalla kuin yhdellä opiskelijalla ongelmia aiheuttivat piirtokynän heikko tunnistus, näytön "pomppailu" sivulla ja piirtoalustan liiallinen herkkyys. Kaksi jälkimmäistä ongelmaa liittyi eräässä tapauksessa toisiinsa opiskelijan paidan hihan osuessa piirtonäyttöön. Eräs opiskelija puolestaan piti piirtoalustan käyttöä epämukavana pöydällä olleen vähäisen tilan vuoksi. Ongelmia aiheutti myös kynän käyttöönottosovelluksen toimimattomuus ja sähköisen pyyhekumin käyttö.

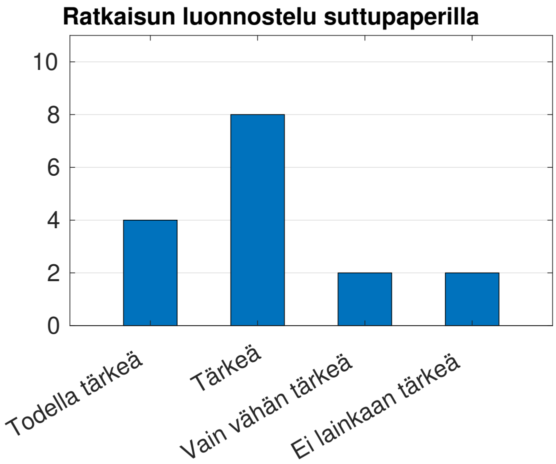
5.4.5 Miten tärkeänä opiskelijat pitävät matematiikan tenttisuoritukseen liittyviä seikkoja?

Kysymyksessä 11 opiskelijat arvioivat kahdeksan matematiikan tenttiin liittyvän tekijän tärkeyttä. Kuvissa 5.5–5.12 esitettyjen tulosten perusteella huomataan heti, että vastaajat pitivät tärkeimpinä vapaasti valittavaa tenttimisajankohtaa, mahdollisuutta käyttää piirtoalustaa, matematiikkaohjelmistoja ja taulukkokirjan käyttömahdollisuutta. Nämä kaikki olivat todella tärkeitä asioita suurimman osan mielestä. Vastaavasti vähiten tärkeinä pidettiin fyysisen laskimen käyttöä ja mahdollisuutta käyttää kynää ja paperia vastaamisessa. Huomioitavaa on kuitenkin, että suuri enemmistö ei pidä näitä asioita vähemmän tärkeinä, vaan mielipiteet ovat jakautuneet kaikkien vastausvaihtoehtojen kesken. Osa opiskelijoista piti siis näitäkin asioita todella tärkeinä. Myös STACK-tehtävien muodossa saatava välitön palaute jakaa mielipiteet vain vähän tärkeän ja tärkeän välillä.

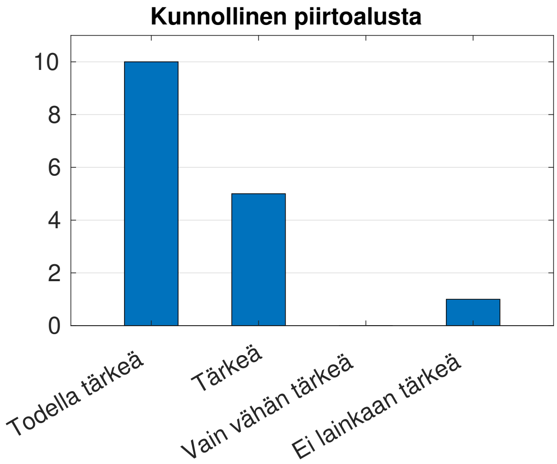
Kuva 5.5. *Kyselyn kohta 11.1*



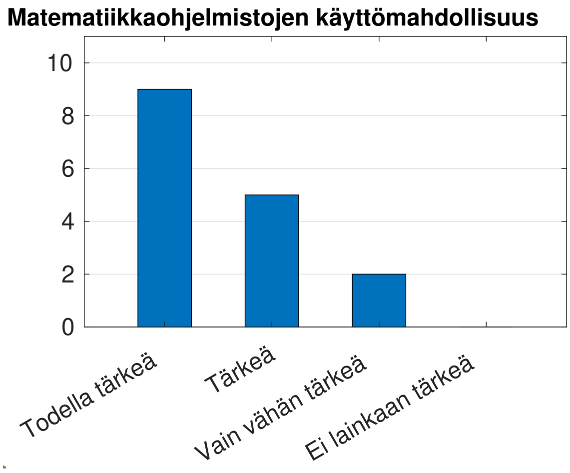
Kuva 5.6. *Kyselyn kohta 11.2*



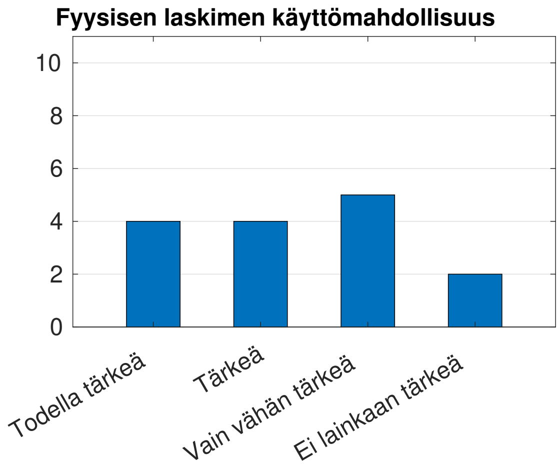
Kuva 5.7. *Kyselyn kohta 11.3*



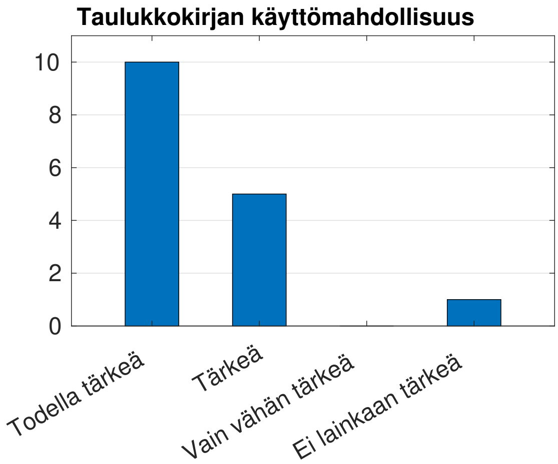
Kuva 5.8. *Kyselyn kohta 11.4*



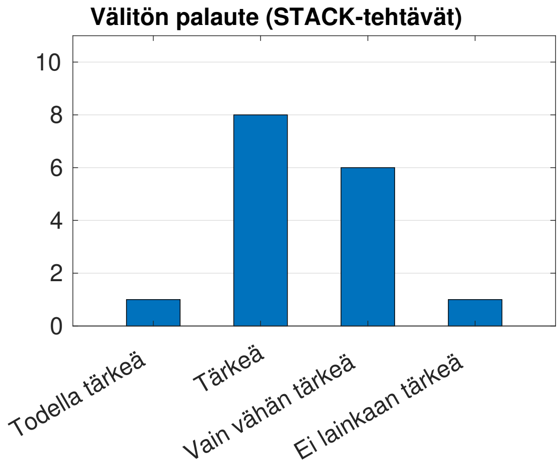
Kuva 5.9. *Kyselyn kohta 11.5*



Kuva 5.10. *Kyselyn kohta 11.6*



Kuva 5.11. *Kyselyn kohta 11.7*



Kuva 5.12. *Kyselyn kohta 11.8*

5.4.6 Miten opiskelijat helpottaisivat sähköiseen tenttiin vastaamista?

Vastaajista 4 oli sitä mieltä, että pidempi tenttimisaika paperitenttiin verrattu olisi tarpeellinen, sillä sähköisessä tentissä aikaa kuluu teknisiin ongelmiin ja valmisteluun. Yksi vastaaja ehdotti 10 minuutin valmisteluaikaa ennen tentin alkamista. Eräs ajanhallintaan liittyvä ehdotus oli puolestaan lisätä navigointipalkki Moodle-sivulle, jossa näkysi kaikki tehtävät.

> "Tenttiajan täytyisi olla pidempi, sillä tentissä voi menettää aikaa itsestä riippumattomista syistä, jolloin opiskelija ei ole tasa-arvoisessa asemassa paperitentin tehneisiin verrattuna. Koen että saamani koeaika on suhteutettuna lyhyempi kuin mitä se olisi ollut paperitenttiä tehtäessä."

Vastaavasti 4 opiskelijaa olisi toivonut fyysisen laskimen käyttömahdollisuutta. Syitä tähän oli tottumattomuus tietokonelaskimeen käyttöön ja vastaavasti oman tai edes tutun fyysisen laskimen nopeampi käyttö. Toisaalta myös toivottiin tietokonelaskimen käytön opettamista ennen koetta. Kuten fyysistä laskinta, 3 opiskelijaa olisi toivonut myös fyysistä suttupaperia.

Useampi opiskelija toivoi vastausohjeita STACK-tehtäviin. Vaikeuksia tuotti vastaamiseen käyttävien oikeiden notaatioiden muistaminen, pyöristystarkkuus ja matriisien muodostaminen. Eräs opiskelija piti koko Moodle-alustaa sekavana ja toivoi sen käyttöön ohjeistusta.

Eräs parannusehdotus liittyi tietokoneella piirtämiseen. Helpottava tekijä opiskelijan mukaan olisi käyttää paremmin piirtämiseen kohdistettua ohjelmaa vastausten antamiseen Wordin tai PowerPointin sijasta. Opiskelija toivoi myös, että vastausten tallentaminen olisi helpompaa ja reaaliaikaista.

EXAM-tilaan toivottiin avustajaa kuten opettajaa, joka osaa auttaa mikäli tietokone lakkaa esimerkiksi toimimasta. Tilan toivottiin olevan myös rauhallisempi sekä suurempi. Jälkimmäisestä epäselväksi jäi toivottiinko EXAM-luokan olevan suurempi tai väljempi vai oliko kyse yhden tenttijän tarvitsemasta tilasta.

6. YHTEENVETO

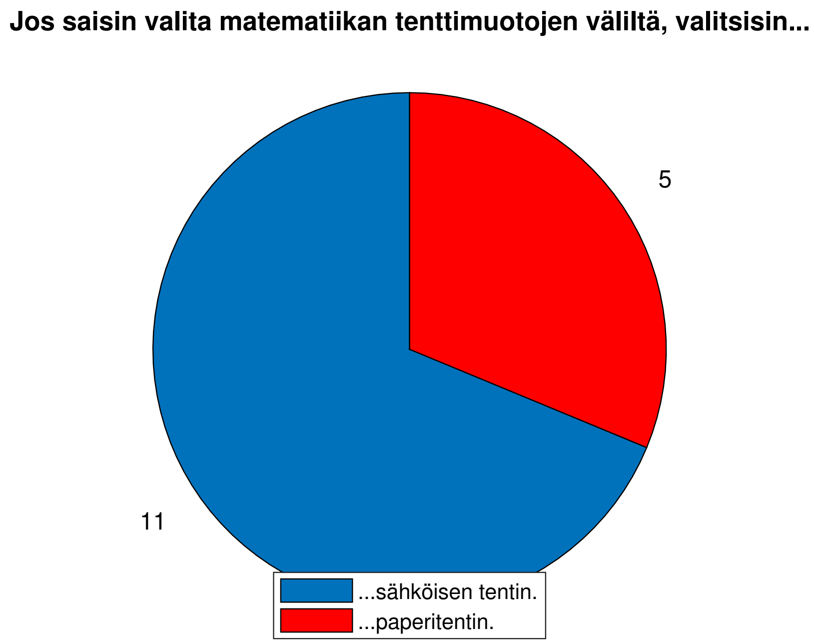
Tässä luvussa pohditaan syitä opiskelijoiden vastauksille ja tehdään niistä huomioita. Luvussa esitetään tutkimustulosten mukaisia pohdintoja tutkimuskysymyksiin. Lopuksi reflektoidaan tutkimuksen luotettavuutta.

6.1 Tulosten pohdintaa

Suurimman osan vastanneista mielestä piirtoalustat olivat kaikin puolin käyttäjäystävällisiä ja niiden hyödyntämistä pidettiin positiivisena. Suurimmat piirtoalustan aiheuttamat ongelmat liittyivät käyttökokemukseen enemmän kuin varsinaisesti tehtäviin vastaamiseen liittyviin ongelmiin. Voisi siis ajatella, että ongelmat liittyvät osaltaan kyseisten piirtoalustojen vähäiseen käyttökokemukseen. Tätä voisi verrata aiemmin tutkielmassa mainitussa Bennetin ym. [7] artikkelissa esiteltävään tutkimukseen. Riittävä ohjeistus piirtoalustan käyttöön on siis tärkeä, jotta käyttö olisi mahdollisimman vaivatonta ensimmäisestä kerrasta lähtien. Pieni työskentelytila häiritsi myös osaa opiskelijoista. On mahdollista, ettei kaikki opiskelijat huomanneet, että esimerkisi tietokoneen näppäimistöä ja hiirtä on mahdollista siirtää, jolloin piirtoalustan voi asettaa sopivalle paikalle. Piirtoalustan voi ottaa myös syliin tarvittaessa. Tämä ei ole välttämättä itsestään selvä asia ensimmäistä kertaa tenttiä piirtoalustaa sähköisessä tentissä käyttäville opiskelijoille, joten aiheesta voisi neuvoa esimerkiksi piirtoalustan ohjeistuksessa. Vaikka vastanneet olivat tyytyväisiä käytettäviin piirto-ohjelmistoihin, on huomioitava, ettei kyseiset ohjelmat ole täysin optimaalisia tarkoitukseen. Itse piirtäminen ei ollut opiskelijoille ongelma, mutta sivujen lisäämisen, pyyhekumin käytön ja tallentamisen hankaluudet ovat asioita, jotka hidastavat tentin suorittamista.

Kyselyyn vastanneet opiskelijat pitivät vapaasti valittavaa tenttiajankohtaa yhtenä tärkeimmistä seikoista matematiikan tenteissä. Myös kyselyn avointen kysymysten vastauksissa nostettiin vapaasti valittava ajankohta positiivisena seikkana. Joustava tenttimisaika voidaan tietenkin toteuttaa sähköisen EXAM-tentin avulla. Sähköisessä tentissä kunnollisen piirtoalustan käyttömahdollisuus on tärkeä opiskelijoiden mielestä, minkä voidaan ajatella johtuvan siitä, ettei TAMK:n opiskelijoilla ole valmiuksia kaavaeditorien sujuvaan käyttöön. Toisaalta piirtoalustaa käyttäen opiskelijat pystyvät perustelemaan vastauksensa piirrosten ja kuvien avulla sähköisessä tentissä samalla tavalla kuin paperitentissä. Näin ollen piirtoalustojen hyödyntämistä voidaan pitää perusteltuna. Opiskelijat pitivät sähköisessä tentissä matemaattisten ohjelmistojen hyödyntämistä tärkeänä, mutta vastausten perusteella osa olisi suosinut fyysisen laskimen käyttömahdollisuutta, mikä on ymmärrettävää, sillä opiskelijat ovat todennäköisesti tottuneet enemmän fyysiseen laskimeen. Useampi opiskelijoista olisi käyttänyt sähköisessä tentissä myös fyysistä suttupaperia. Näiden tuominen EXAM-tiloihin ei ole kuitenkaan mahdollista EXAM-tilojen sääntöjen [3] puitteissa, vaan ne kieltävät sekä fyysiset laskimet että suttupaperit tiloista. Tenttisuoritusta voisi helpottaa ohjeistus matematiikkaohjelmistojen käyttöön ja vinkit sähköisen suttupaperin käyttöön. Yksi tärkeimpinä pidetyistä seikoista oli taulukkokirjan käyttömahdollisuus, mikä on mahdollista sekä perinteisessä paperitentissä kuin sähköisessä tentissäkin.

Kyselyyn vastanneista opiskelijoista noin kaksi kolmasosaa tekisi tulevaisuudessa mielummin sähköisen tentin kuin paperitentin matematiikassa. Vaikka tutkimuksessa ei erikseen syitä tähän kysytty, syitä on mahdollista päätellä aiemmin esitettyjen vastausten perusteella. Joustava tenttimisaika on ehkä sähköisen tentin suurin etu paperitenttiin verrattuna. Moni opiskelija piti myös piirtoalustan käytön mielekkyyttä vähintään paperiin verrattavana, joten vastausten laatimisessa ei opiskelijoille ongelmia koitunut. Paperitentin kannattajien perusteina voidaan vastausten perusteella pitää teknisiä epävarmuustekijöitä ja käyttöön tottumattomuutta.



Kuva 6.1. *Yli kaksi kolmesta kyselyyn vastanneesta opiskelijasta tekisi mielummin matematiikan tentin sähköisenä.*

Tutkimuksen perusteella matematiikan sähköisten tenttien suorittamisen testaamista on syytä jatkaa eteenpäin. Vastausten perusteella piirtoalustat ovat toimiva apuväline sähköisissä tenteissä. Tulevissa tenteissä on syytä pohtia, tulisiko opiskelijoita valmistaa sähköiseen tenttiin esimerkiksi kokonaisvaltaisemman ohjeistuksen tai muun perehdytyksen avulla.

6.2 Tutkimuksen luotettavuus

Perustuen Eskolan ja Suorannan teokseen [11], laadullisessa tutkimuksessa luotettavuutta voidaan tarkastella neljästä näkökulmasta: uskottavuus, siirrettävyys, varmuus ja vahvistuvuus. Uskottavuudella tarkoitetaan, vastaako tutkijan päätelmät tutkittavien käsityksiä. Siirrettävyydellä tarkoitetaan voidaanko tulosta yleistää jollain tasolla. Jos tutkimus on varmuuden näkökulmasta luotettava, on siinä otettu huomioon ennakkoehtoja. Vahvistuvuudella puolestaan tarkoitetaan, ovatko muut vastaavanlaiset tutkimukset päässeet samankaltaiseen lopputulokseen.

Tutkimuksessa kartoitettiin opiskelijoiden näkemyksiä sekä avointen, että monivalintakysymysten avulla. Tutkimuksen kyselyyn vastasi 16 opiskelijaa, joten otoskoko tutkimuksessa oli melko pieni. Kuitenkin osallistujamäärään nähden avoimiin kysymyksiin saatiin runsaasti vastauksia. Kyselyn vastausaika oli keskimäärin lähes 9 minuuttia, mikä viittaa siihen, että kyselyyn on myös vastattu huolellisesti.

Tutkimuksen uskottavuutta lisää tutkijan omat kokemukset piirtoalustan käytöstä ja matematiikan sähköisten tenttien suorittamisesta. Näin ollen tutkijan on mahdollista ymmärtää esimerkiksi opiskelijoiden kohtaamat ongelmat tentin aikana. Pienen otoskoon vuoksi tutkimustulokset ei ole välttämättä yleistettävissä TAMK:n tekniikan alan opiskelijoihin. Siirrettävyyttä lisää kuitenkin vastanneiden opiskelijoiden erilaiset koulutustaustat. Tutkimus suoritettiin opiskelijoilla perustuen vain yhteen tenttimiskertaan, joka oli todennäköisesti suurelle osalle vastanneista ensimmäinen matematiikan sähköinen tentti. Yksittäisessä tenttisuorituksessa on mahdollista tapahtua odottamattomia teknisiä ongelmia, jotka vaikuttavat negatiivisesti kokemukseen. Tällaisena voidaan pitää esimerkiksi word ohjelman "tilttausta". Näin ollen tutkimusta ei voida pitää erityisen varmana.

Tutkimusta voidaan pitää onnistuneena, sillä kyselyn vastausten perusteella oli mahdollista selvittää vastauksia tutkimuskysymyksiin. Tutkimus suoritettiin asianmukaisesti: Tutkimukseen haettiin tutkimuslupa ja se toteutettiin tutkimussuunnitelman mukaisesti. Tutkimuksen suorittamisessa oli myös parannettavaa. Kyselyn monivalintakysymysten vastausvaihtoehdot olisivat voineet olla selkeämpiä: "Todella tärkeä" ja "Tärkeä" vaihtoehdot erityisesti olisivat voineet olla tarkemmin eroteltuja.

LÄHTEET

[1] Examinarium (n.d.) *Piirtäminen ja laskeminen*. URL: https://blogs.helsinki.fi/examinarium/piirtaminen-ja-laskeminen/ (viitattu 26. 09. 2023).

[2] LUT (n.d.) *Piirtonäytöt EXAM-tenttikoneilla*. URL: https://elut.lut.fi/sites/default/files/category-page/2021-12/Piirton%C3%A4yt%C3%B6t%20EXAM%20yleisohje.pdf (viitattu 26. 09. 2023).

[3] TUNI EXAM (n.d.) *EXAM-tilojen käyttösäännöt*. URL: https://sites.tuni.fi/exam/tilat/tilojen-kayttosaannot/ (viitattu 30. 07. 2023).

[4] TUNI EXAM (n.d.) *Ohjelmistot*. URL: https://sites.tuni.fi/exam/ohjeet/ohjelmistot/ (viitattu 03. 11. 2022).

[5] TUNI EXAM (n.d.) *Opiskelijan ohjeet*. URL: https://sites.tuni.fi/exam/ohjeet/opiskelijan-ohjeet/ (viitattu 03. 11. 2022).

[6] Turun yliopisto (n.d.) *Sähköinen tenttipalvelu: opiskelijan ohje*. URL: https://utuguides.fi/c.php?g=669085&p=4748232#s-lg-box-15329733 (viitattu 26. 09. 2023).

[7] Randy Elliot Bennett et al. ”Does it matter if I take my mathematics test on computer? A second empirical study of mode effects in NAEP”. eng. *The journal of technology, learning, and assessment* 6.9 (2008), s. 1–39. ISSN: 1540-2525.

[8] Der-San Chen. *Applied Integer Programming Modeling and Solution*. eng. Hoboken: Wiley, 2011. ISBN: 1-282-25370-0.

[9] R.J. DAKIN. ”A tree-search algorithm for mixed integer programming-problems”. eng. *Computer journal* 8.3 (1965), s. 250–253. ISSN: 0010-4620. DOI: 10.1093/comjnl/8.3.250.

[10] Dante D. Dixson ja Frank C. Worrell. ”Formative and Summative Assessment in the Classroom”. eng. *Theory into practice* 55.2 (2016), s. 153–159. ISSN: 0040-5841. DOI: 10.1080/00405841.2016.1148989.

[11] Jari. Eskola ja Juha. Suoranta. *Johdatus laadulliseen tutkimukseen*. fin. Tampere: Vastapaino, 1998. ISBN: 9789517685047.

[12] Hanna Fagerholm. *Sähköinen tenttijärjestelmä (EXAM) Opiskelijan ohje*. 2021. URL: https://ict.oulu.fi/5330/ (viitattu 05. 07. 2023).

[13] Archis Ghate ja Robert L. Smith. ”Characterizing extreme points as basic feasible solutions in infinite linear programs”. *Operations Research Letters* 37.1 (2009), s. 7–10. ISSN: 0167-6377. DOI: https://doi.org/10.1016/j.orl.2008.09.002. URL: https://www.sciencedirect.com/science/article/pii/S0167637708001089.

[14] Harold. Greenberg. *Integer programming*. eng. Mathematics in science and engineering ; 76. New York: Academic Press, 1971. ISBN: 0-12-299450-7.

[15] James Hiebert. *Conceptual and Procedural Knowledge: The Case of Mathematics*. eng. Taylor ja Francis, 2013. ISBN: 9781138148925.

[16] Kees Hoogland ja Dave Tout. ”Computer-based assessment of mathematics into the twenty-first century : pressures and tensions”. eng. *ZDM* 50.4 (2018), s. 675–686. ISSN: 1863-9690. DOI: 10.1007/s11858-018-0944-2.

[17] N. Karmarkar. ”A New Polynomial-Time Algorithm for Linear Programming”. Teoksessa: *Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing*. STOC ’84. New York, NY, USA: Association for Computing Machinery, 1984, s. 302–311. ISBN: 0897911334. DOI: 10.1145/800057.808695. URL: https://doi.org/10.1145/800057.808695.

[18] Bernard Kolman. *Elementary linear programming with applications*. eng. 2nd ed. Computer Science and Scientific Computing. San Diego, California ; Academic Press, 1995 - 1995. ISBN: 0-08-053079-6.

[19] Salla Koskinen et al. ”Sähköisen matematiikan tentin toteuttaminen ja opiskelijoiden kokemukset sähköisestä tentistä”. fin. Teoksessa: Asikainen, Mervi. 2017-10-31. URL: https://journal.fi/fmsera/article/view/60927.

[20] Cs. Mészáros ja U.H. Suhl. ”Advanced preprocessing techniques for linear and quadratic programming”. eng. *OR Spectrum* 25.4 (2003), s. 575–595. ISSN: 0171-6468. DOI: 10.1007/s00291-003-0130-x.

[21] Jorge. Nocedal. *Numerical Optimization*. eng. 2nd ed. 2006. Springer Series in Operations Research and Financial Engineering. New York, NY: Springer New York, 2006. ISBN: 0-387-40065-6.

[22] Frode Rønning. ”Influence of computer-aided assessment on ways of working with mathematics”. eng. *Teaching mathematics and its applications* 36.2 (2017), s. 94–107. ISSN: 0268-3679. DOI: 10.1093/teamat/hrx001.

[23] C. J. (Christopher J.) Sangwin. *Computer aided assessment of mathematics*. eng. Oxford: Oxford University Press, 2013. ISBN: 0-19-163586-3.

[24] V.J. Shute ja S. Rahimi. ”Review of computer-based assessment for learning in elementary and secondary education”. eng. *Journal of computer assisted learning* 33.1 (2017), s. 1–19. ISSN: 0266-4909. DOI: 10.1111/jcal.12172.

[25] Lawrence Smolinsky et al. ”Computer-Based and Paper-and-Pencil Tests: A Study in Calculus for STEM Majors”. *Journal of Educational Computing Research* 58.7 (2020), s. 1256–1278. DOI: 10.1177/0735633120930235.

[26] Inc. The MathWorks. *Mixed-Integer Linear Programming Algorithms*. 2023. URL: https://se.mathworks.com/help/optim/ug/mixed-integer-linear-programming-algorithms.html (viitattu 15. 06. 2023).

LIITE A: MATLAB-KOODI SATUNNAISEN MATRIISIN LUOMISEEN

Satunnaisen matriisin määrittäminen

Määritetään $n \times n$ -kokonaislukumatriisi R.

Matriisille luodaan seuraavat ominaisuudet.

1. Matriisin alkioiden arvot voidaan määrätä olemaan kokonaislukuja välillä $[\mathrm{xmin}, \mathrm{xmax}]$.
2. Jokainen arvo esiintyy korkeintaa k kertaa.
3. Vaaditaan, että suurin luku on vähintään s ja pienin luku on enintään l.
4. Luodut matriisit ovat satunnaisia toisiinsa nähden.
5. Matriisissa suositaan itseisarvoltaan pieniä lukuja.
6. On mahdollista asettaa matriisit singulaarisiksi.

```
n = 3;
k = 2;

xmin= -20;
xmax = 20;

s = 11;
l = -9;

sigma = 3;
painokerroin = 1;

x=xmin:xmax;
m=numel(x);
lkm=n*n*(1+m);
intcon = 1:lkm;
```

```
%Kohdefunktio
[~,~,F]=meshgrid(ones(1,n) ,ones(1,n),x);
f = painokerroin*abs([zeros(n^2,1);F(:)+sigma*randn(n*n*m,1)]);
intcon=1:lkm;
A=[]; b=[]; Aeq=[]; beq=[];

y=ones(size(f));
```

Muuttujien ala- ja ylärajat

```
LB=[repmat(xmin,n^2,1);zeros(n^2*m,1)];
UB=[repmat(xmax,n^2,1);ones(n^2*m,1)];
```

Jokaisessa paikassa tulee olla tasan yksi luku

```
Aeq=[Aeq;zeros(n*n) kron(ones(1,m),eye(n*n))];
beq=[beq;ones(n*n,1)];
```

Ensimmäinen matriisi A vastaa muita lukuja

```
Aeq=[Aeq;kron([1 -x],eye(n*n))];
beq=[beq;zeros(n*n,1)];
```

Jokainen luku esiintyy korkeintaa k kertaa.

```
A=[A;kron([zeros(m,1) eye(m)], ones(1,n*n))];
b=[b;k*ones(m,1)];
```

Suurin arvo on vähintää s.

```
isot = [zeros(n^2,1);F(:)]>(s-1);
A=[A;-isot.'];
b=[b;-1];
```

Pienin arvo on enintään l.

```
pienet = [zeros(n^2,1);F(:)]<(l+1);
A=[A;-pienet.'];
b=[b;-1];
```

Singulaarisuusehto

```
Aeq=[Aeq;[blkdiag(N(1,:),N(2,:),N(3,:)) zeros(n,n*n*m)]];
beq=[beq;zeros(n,1)];
```

Optimointi

```
options = optimoptions('intlinprog','Display','off');

[X,FVAL,EXITFLAG] = intlinprog(f,intcon,A,b,Aeq,beq,LB,UB,options);
R = reshape(X(1:n*n),n,n); %Huom. järjestetään sarakkeittain
R=sym(round(R))
Determinantti=det(R)
```

LIITE B: SÄHKÖISEEN TENTTIIN LIITTYVÄ TUTKIMUSKYSELY

Opiskelijoiden kokemukset sähköisestä tentistä Tampereen ammattikorkeakoulussa

Kysely liittyy tutkimukseen, jonka avulla kehitetään sähköisiä tenttejä. Tutkimuksen tietosuojailmoitus: https://tuni-my.sharepoint.com/:b:/g/personal/simo_ali-loytty_tuni_fi/EYLFZiE6xH5Bpt-dCPvssKIBYrUVS0_CEs5kA5EjPK_dSA?e=PdC6Mq

1. Saako vastauksiasi kyselyyn ja EXAM-tenttiin käyttää tutkimusaineistona? *

○ Kyllä

○ Ei

2. Mikä on koulutustaustasi ennen ammattikorkeakouluopiskelujasi? *

○ Lukio

○ Ammattikoulu

○ Kaksois-/kolmoistutkinto

○ Muu

3. Mikäli olet suorittanut ylioppilastutkinnon ja osallistunut matematiikan ylioppilaskokeeseen, kirjoititko matematiikan sähköisenä vai paperisena?

◯ Sähköisenä

◯ Paperisena

4. Oletko tottunut käyttämään tietokonetta kosketusnäytöllä? *

◯ Olen

◯ En ole

5. Millaiseksi koit sähköiseen tenttiin vastaamisen paperitenttiin verrattuna? *

Kirjoita vastaus

6. Koitko minkäänlaisia ongelmia sähköiseen tenttiin vastaamisessa tai sen suorittamisessa? *

◯ Koin

◯ En kokenut

7. Jos koit, niin millaisia ongelmat olivat?

Kirjoita vastaus

8. Mitä mieltä olit seuraavista asioista sähköisessä tentissä?
*

		Hyvä	Riittävä	Heikko
1.	Piirtoalustan käyttäjäystäv ällisyys	◯	◯	◯
2	Piirtoalustan tarkkuus	◯	◯	◯
3.	Piirtoalustaan liittyvä ohjeistus	◯	◯	◯
4.	Piirtämiseen käytettävien ohjelmien käyttäjäystäv ällisyys	◯	◯	◯

9. Koitko minkäänlaisia ongelmia piirtoalustan käytössä? *

◯ Koin

◯ En kokenut

10. Jos koit, niin millaisia ongelmat olivat?

Kirjoita vastaus

11. Miten tärkeänä pidät seuraavia matematiikan tenttisuoritukseen liittyviä seikkoja? *

	Todella tärkeä	Tärkeä	Vain vähän tärkeä	Ei lainkaan tärkeä
1. Vapaasti valittava tenttimisajan kohta	○	○	○	○
2. Mahdollisuus vastata kysymyksiin kynällä ja paperilla	○	○	○	○
3. Mahdollisuus luonnostella ratkaisua suttupaperilla, vaikka lopullinen vastaus palautettaisiinkin sähköisesti	○	○	○	○
4. Mahdollisuus käyttää kunnollista piirtoalustaa sähköisessä tentissä	○	○	○	○
5. Matematiikkaohjelmistojen käyttömahdollisuus sähköisessä tentissä (Matlab, TI-Nspire,...)	○	○	○	○

6. Fyysisen laskimen käyttömahdollisuus	○	○	○	○
7. Taulukkokirjan käyttömahdollisuus (Fyysinen tai sähköinen)	○	○	○	○
8. Välitön palaute (STACK-tehtävät)	○	○	○	○

12. Mitkä asiat helpottaisivat sähköiseen tenttiin vastaamista? *

Kirjoita vastaus

13. Jos saisin valita matematiikan tenttimuotojen väliltä, valitsisin... *

○ ...paperitentin.

○ ...sähköisen tentin.

14. Tähän voi kirjoittaa vapaan palautteen sähköiseen tenttiin liittyen.

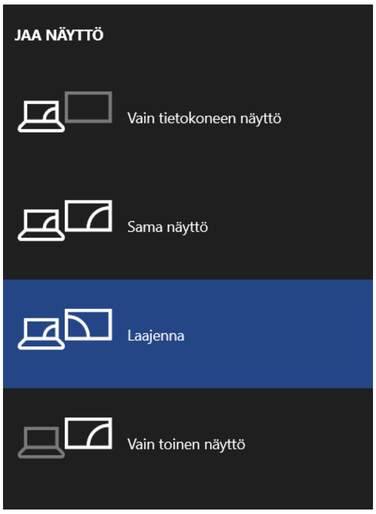
Kirjoita vastaus

LIITE C: SÄHKÖISEN TENTIN OHJEISTUS

Ohjeistus on luotu opiskelijoiden avuksi TAMK:n sähköisiin tentteihin, joissa on mahdollisuus käyttää piirtoalustaa. Ohjeistuksessa opastetaan piirto-ominaisuuden käyttöön Microsoft Word ja PowerPoint ohjelmilla. Ohjeitus sisältää myös ohjeet tiedostojen tallentamiseen PDF-muodossa.

Ohjeistus kosketusnäytön käyttöön EXAM -tentissä

Kosketusnäyttö on valmiiksi käyttövalmiina, kun kirjaudut koneelle. Kosketusnäyttö toimii näytön alta löytyvän kynän avulla. Jos kosketusnäyttö ei käynnisty automaattisesti, varmista, että *laajennettu* tai *sama näyttö* (*Extend* tai *Duplicate*) on valittuna näytön asetuksissa.



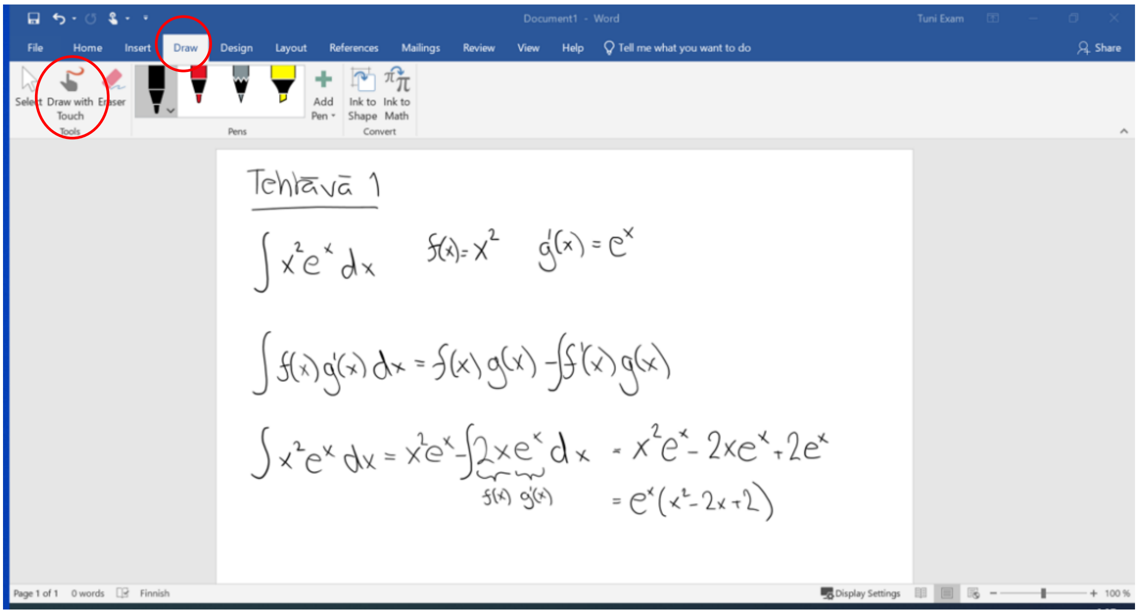
Microsoft Word

Avaa Microsoft Word.

Piirtotyökalun saa käyttöön Wordissa seuraavasti:

Valitse yläreunasta *Piirrä/Draw* → Aktivoi piirtäminen kohdasta *Piirrä koskettamalla/Draw with Touch*

Tämän jälkeen Word-sivulle on mahdollista piirtää kosketusnäytön avulla.

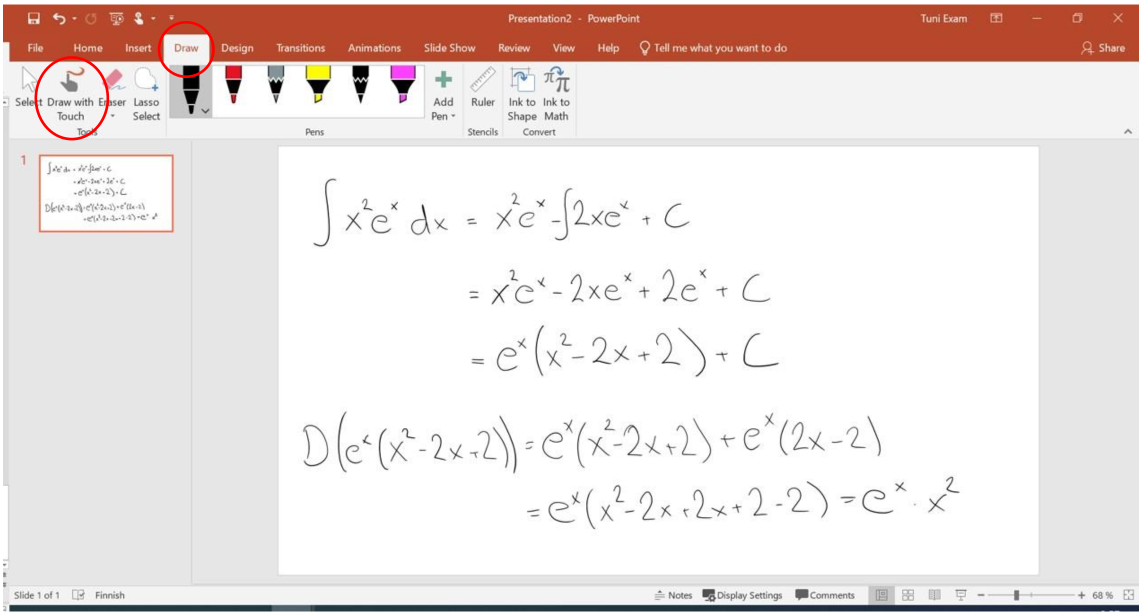


Microsoft PowerPoint

Avaa Microsoft PowerPoint.

PowerPointilla piirtotyökalu toimii samalla tavalla kuin Wordissa.

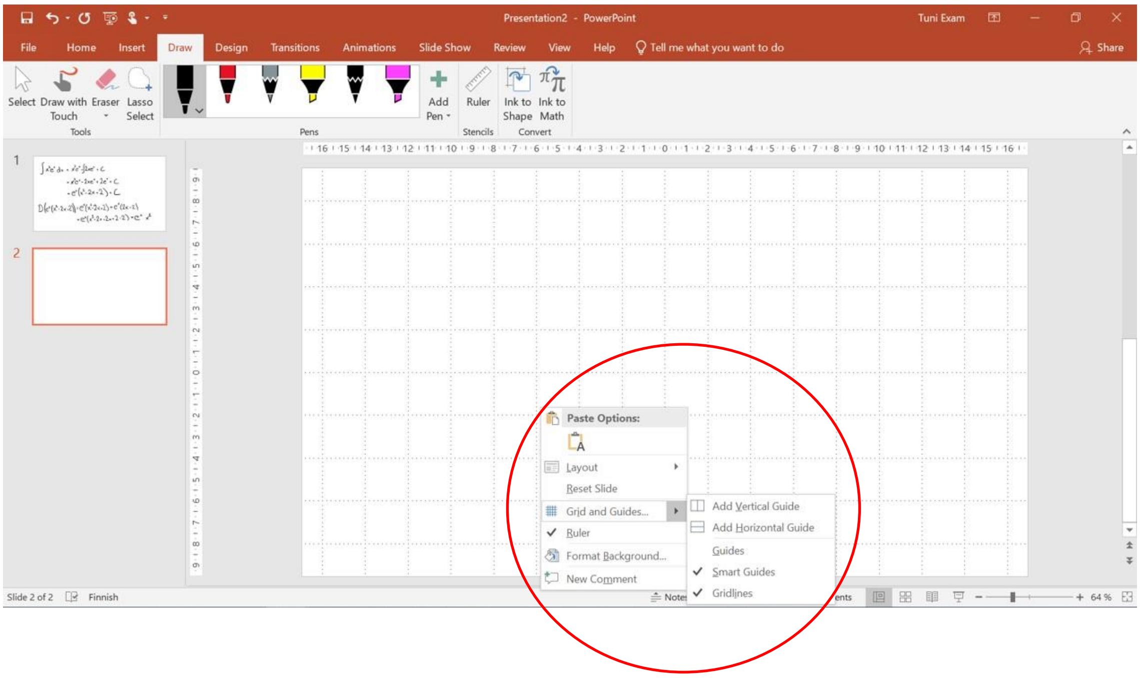
Valitse yläreunasta *Piirrä/Draw* --> Aktivoi piirtäminen kohdasta *Piirrä koskettamalla/Draw with Touch*



Powerpoint dioille on myös mahdollista lisätä ruudukko piirtämisen helpottamiseksi. (HUOM. Ruudukko ei näy tallennetussa PDF-tiedostossa.)

Ruudukon saa näkyviin seuraavasti:

Klikkaa hiiren oikealla painikkeella (mouse2) diaa → Valitse *Ruudukko/Grid and Guides* → Ota käyttöön *Ruudukko/Gridlines*

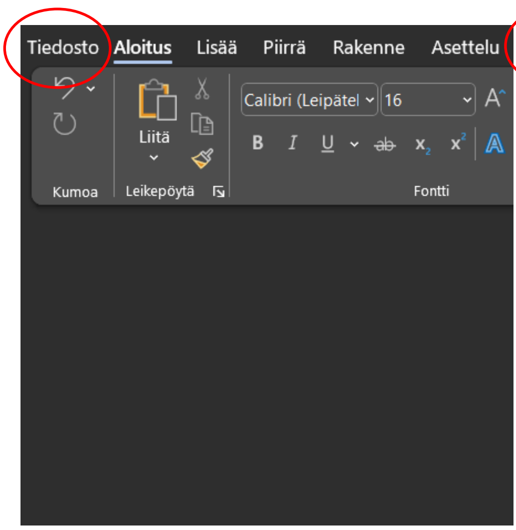


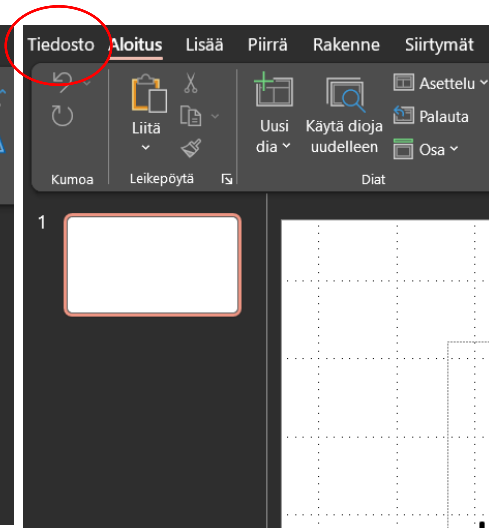
Tehtävän palautus

Ratkaisut palautetaan EXAM-tentissä PDF-muodossa. Ratkaisut kannattaa varmuuden vuoksi tallentaa myös Word/PowerPoint-tiedostona, jotta ratkaisuihin on helppo tehdä muutoksia tarvittaessa. Kaikki tiedostot kannattaa tallentaa työpöydälle. Tehtävien ratkaisut kannattaa palauttaa sitä mukaan, kun ne saa valmiiksi. Palauttaessasi ratkaisua tarkista aina, että PDF-tiedosto ovat muodostuneet oikein avaamalla tiedosto.

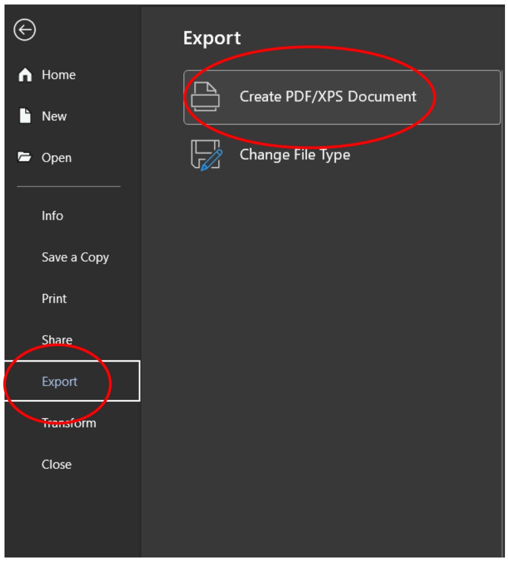
Tiedoston tallennus PDF-muodossa

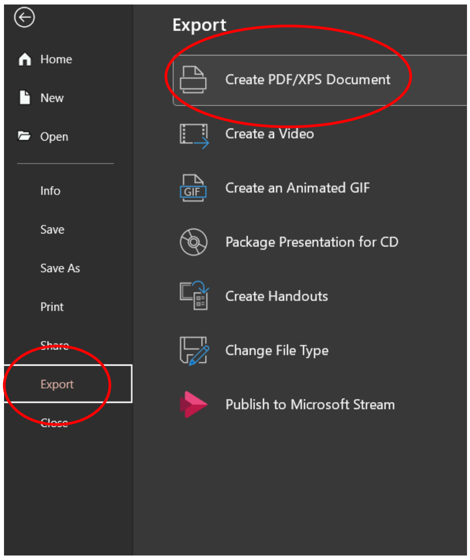
Valitse *Tiedosto/File* valintanauhasta





Valitse valikon vasemmasta reunasta *Vie/Export*, minkä jälkeen valitse *Luo PDF- tai XPS-tiedosto/Create PDF/XPS Document*





Nimeä tiedosto kuvaavasti (Esim. TEHTÄVÄ1) ja varmista, että tiedoston muoto on oikea (PDF). *Tallenna/Publish* tiedosto *Työpöydälle/Desktop,* josta se on helppo löytää.