



Vaasan yliopisto  
UNIVERSITY OF VAASA

MATTI LAAKSONEN

# Lineaarialgebran lisämateriaalia

VAASAN YLIOPISTON JULKAISUJA

---

OPETUSJULKAISUJA  
MATEMATIIKKA

**Julkaisija**

Vaasan yliopisto

**Julkaisuajankohta**

Elokuu 2014

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                                                                         |                       |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------|-----------------------|
| <b>Tekijä(t)</b><br>Matti Laaksonen                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | <b>Julkaisun tyyppi</b><br>Opetusmoniste                                |                       |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | <b>Julkaisusarjan nimi, osan numero</b><br>Vaasan yliopiston julkaisuja |                       |
| <b>Yhteystiedot</b><br>University of Vaasa<br>Department of Mathematics and Statistics<br>P.O.Box 700<br>FI-65101 Vaasa, Finland                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | <b>ISBN</b><br>1234567890                                               |                       |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | <b>ISSN</b><br>1234567890                                               |                       |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | <b>Sivumäärä</b><br>32                                                  | <b>Kieli</b><br>suomi |
| <b>Julkaisun nimike</b><br>Lineaarialgebran lisämateriaalia                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                         |                       |
| <b>Tiivistelmä</b><br>Opintojakson ”Lineaarialgebra”, math.1040 luentomateriaalia.<br><br>Tähän monisteeseen on kerätty todistuksia ja esimerkkejä, jotka ovat olleet opetusmonisteen ensimmäisessä versiossa. Koska materiaali alkoi paisua liian suureksi, osa siitä siirrettiin tähän lisämateriaaliin.<br>Käytännössä opetusmoniste on se materiaali, jonka läpikäynti riittää kokeissa menestymiseen. Lisämateriaalin sisältämät esimerkit saattavat helpottaa opetamonisteen asioiden omaksumista. Opiskelijan kannattaa tutustua lisämateriaaliin ja ottaa siitä se mikä tuntuu sopivan omaan opiskelutyyliin. |                                                                         |                       |
| <b>Asiasanat</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                                                         |                       |

# SISÄLTÖ

|          |                                                       |           |
|----------|-------------------------------------------------------|-----------|
| <b>1</b> | <b>Johdanto</b>                                       | <b>1</b>  |
| 1.1      | Ohjeita lisämateriaalin käyttöön <<< □ . . . . .      | 1         |
| 1.2      | Laskemisen historiaa . . . . .                        | 1         |
| 1.3      | Python, alkeita <<< □ . . . . .                       | 5         |
| <b>2</b> | <b>Trigonometriaa ja fysiikan vektoreita</b>          | <b>6</b>  |
| 2.1      | Tukivoimat ja kitka . . . . .                         | 6         |
| 2.2      | Ristikön tukivoimat . . . . .                         | 16        |
| <b>3</b> | <b>Lineaariset yhtälöryhmät</b>                       | <b>22</b> |
| 3.1      | Ratkaisun olemassaolo ja yksikäsitteisyys . . . . .   | 22        |
| 3.2      | Pyöristysvirheistä . . . . .                          | 27        |
| <b>4</b> | <b>Matriisit ja matriisilaskut</b>                    | <b>32</b> |
| 4.1      | Kitka-esimerkki . . . . .                             | 32        |
| 4.2      | Panos-tuotos -analyysi . . . . .                      | 35        |
| 4.2.1    | Kansantalouden rakenne . . . . .                      | 35        |
| 4.2.2    | Omakustannusarvo . . . . .                            | 39        |
| 4.3      | Yhtälöryhmän kantaratkaisut . . . . .                 | 43        |
| 4.4      | LP-mallin kantaratkaisut . . . . .                    | 49        |
| 4.4.1    | Pelivaramuuttujat ja varjohinnat . . . . .            | 54        |
| <b>5</b> | <b>Determinantit</b>                                  | <b>58</b> |
| 5.1      | Determinantti, perustelut ominaisuuksille . . . . .   | 58        |
| 5.1.1    | Permutaatiot . . . . .                                | 58        |
| 5.1.2    | Isot determinantit . . . . .                          | 61        |
| 5.2      | Minorikehitelmän perustelu . . . . .                  | 69        |
| 5.2.1    | Determinantin laskeminen minorikehitelmällä . . . . . | 70        |
| 5.3      | Piirianalyysi-esimerkki . . . . .                     | 73        |
| 5.4      | Matriisilaskut Excel-ohjelmalla . . . . .             | 76        |
| 5.5      | MatLab-ohjelmoinnin alkeita . . . . .                 | 83        |
| 5.5.1    | Rivikomentoja. . . . .                                | 83        |
| 5.5.2    | Komentojonotiedostot ja omat funktiot. . . . .        | 89        |

|           |                                                                  |            |
|-----------|------------------------------------------------------------------|------------|
| <b>6</b>  | <b>Pistetulon ja ristitulon sovelluksia</b>                      | <b>93</b>  |
| 6.1       | Pyörivistä kappaleista . . . . .                                 | 93         |
| 6.1.1     | Voiman momentti akselin suhteen avaruudessa . . . . .            | 93         |
| 6.2       | Liike pyörivässä koordinaatistossa . . . . .                     | 94         |
| 6.2.1     | Kehänopeus pyörittäessä tasossa. . . . .                         | 94         |
| 6.2.2     | Kehänopeus pyörittäessä akselin ympäri. . . . .                  | 96         |
| 6.2.3     | Liike pyörivässä koordinaatistossa, Corioliskiihtyvyys . . . . . | 97         |
| 6.2.4     | Matalapaineen pyörimissuunta . . . . .                           | 98         |
| <b>7</b>  | <b>Lineaariavaruuudet</b>                                        | <b>100</b> |
| 7.1       | Lisää kompleksilukulaskuja <<< □ . . . . .                       | 100        |
| 7.2       | Esimerkkejä Funktioavaruuksia <<< □ . . . . .                    | 100        |
| <b>8</b>  | <b>Kannat ja ortogonaalisuus</b>                                 | <b>101</b> |
| 8.1       | Esimerkkejä <<< □ . . . . .                                      | 102        |
| <b>9</b>  | <b>Ominaisarvotehtävä</b>                                        | <b>103</b> |
| 9.1       | Lineaarinen differentiaaliyhtälöryhmä <<< □ . . . . .            | 103        |
| 9.2       | Lisää matriisihajoitelmia <<< □ . . . . .                        | 103        |
| 9.3       | Kahden muuttujan sileän ääriarvon tutkiminen <<< □ . . . . .     | 103        |
| 9.4       | Kahden muuttujan sileän funktion Taylorin sarja <<< □ . . . . .  | 103        |
| 9.5       | Kvadraattinen optimointi <<< □ . . . . .                         | 103        |
| <b>10</b> | <b>Analyttistä geometriaa</b>                                    | <b>104</b> |
| 10.0.1    | 2D plottaus <<< □ . . . . .                                      | 104        |
| 10.0.2    | Jacobin determinantti <<< □ . . . . .                            | 104        |
| 10.0.3    | 3D plottaus <<< □ . . . . .                                      | 104        |
| 10.0.4    | Kartioleikkaukset; ellipsi, paraabeli, hyperbeli <<< □ . . . . . | 104        |
| 10.0.5    | Kappaleita ja pintoja; pallo, sylinteri, kartio <<< □ . . . . .  | 104        |
| 10.0.6    | Jacobin determinantti <<< □ . . . . .                            | 104        |
| 10.0.7    | Kamera <<< □ . . . . .                                           | 104        |
| 10.0.8    | Virtuaalikamera <<< □ . . . . .                                  | 104        |
| 10.1      | Avaruuden $\mathbb{R}^n$ analyttistä geometriaa <<< □ . . . . .  | 104        |
| 10.1.1    | Suorat ja hypertasot <<< □ . . . . .                             | 104        |
| 10.1.2    | Monistot <<< □ . . . . .                                         | 104        |
| 10.1.3    | Jacobin determinantti <<< □ . . . . .                            | 104        |

|                                                                         |            |
|-------------------------------------------------------------------------|------------|
| <b>11 Sovelluksia ja erillisaiheita</b>                                 | <b>105</b> |
| 11.1 Homogeeniset koordinaatit . . . . .                                | 105        |
| 11.1.1 Piirtelyharjoituksia . . . . .                                   | 105        |
| 11.1.2 Tason pisteiden homogeeniset koordinaatit. . . . .               | 108        |
| 11.1.3 Käytetyt koodit . . . . .                                        | 114        |
| 11.2 OpenGL <<< □ . . . . .                                             | 115        |
| 11.3 Dynaamisen mallin tasapainon stabiilisuus <<< □ . . . . .          | 115        |
| 11.4 Lineaarista optimointia <<< □ . . . . .                            | 115        |
| 11.5 Pienimmän neliösumman menetelmä . . . . .                          | 115        |
| 11.5.1 Approksimointi polynomilla . . . . .                             | 115        |
| 11.5.2 Lineaarisen mallin sovitus . . . . .                             | 119        |
| 11.5.3 Häiriöiden vaikutus . . . . .                                    | 120        |
| 11.6 Mallin nopea estimointi . . . . .                                  | 124        |
| 11.7 Sileän funktion $f(\vec{x})$ minimikohta <<< □ . . . . .           | 128        |
| 11.8 Analytic Hierarchy Process (AHP) <<< □ . . . . .                   | 128        |
| 11.9 Kalman suodin <<< □ . . . . .                                      | 128        |
| <b>12 Liitteet</b>                                                      | <b>130</b> |
| 12.1 Liite 1. Osaamistavoitteet . . . . .                               | 130        |
| 12.2 Liite 2. Todistuksia . . . . .                                     | 130        |
| 12.2.1 Trigonometriaa . . . . .                                         | 130        |
| 12.3 Liite 3. Esimerkkien ohjelmakoodit . . . . .                       | 132        |
| 12.3.1 Ohjelmointikielten Octave, Python, Java ja C vertailua . . . . . | 132        |
| 12.3.2 Esimerkkiohjelmat Python kielellä . . . . .                      | 133        |
| 12.3.3 Esimerkkiohjelmat Java kielellä . . . . .                        | 134        |
| 12.3.4 Esimerkkiohjelmat C kielellä . . . . .                           | 136        |
| 12.4 Liite 5. Lähdemateriaalia ohjelmointikielistä . . . . .            | 138        |
| 12.4.1 Octave . . . . .                                                 | 139        |
| 12.4.2 Python . . . . .                                                 | 139        |
| 12.4.3 C . . . . .                                                      | 139        |
| 12.4.4 Java . . . . .                                                   | 139        |
| lastpage . . . . .                                                      | 140        |
| <b>13 Kirjoittajalle</b>                                                | <b>141</b> |
| 13.1 Sisältö-suunnitelma . . . . .                                      | 141        |

## TODO LIST

|                                                                        |     |
|------------------------------------------------------------------------|-----|
| tarkista luvun numero, tai korvaa viittauksella lukulabeliin . . . . . | 23  |
| lisää linkki . . . . .                                                 | 29  |
| ohjeet octaven käytöstä . . . . .                                      | 142 |
| loppuun tehtäväsetti . . . . .                                         | 142 |

# 1 JOHDANTO

Ch:johdanto

## 1.1 Ohjeita lisämateriaalin käyttöön

## 1.2 Laskemisen historiaa

Sec:historiaa

Tässä matematiikan kehitysvaiheessa kaiken käytännöllisen laskemisen mallina on reaalilukujen laskutoimitukset. Eri lukualueet yleensä esitetään loogisessa järjestyksessä ( $\mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q} \subset \mathbb{R} \subset \mathbb{C}$ ), mutta niiden keksiminen ja käyttöönotto ei Euroopassa tapahtunut tässä järjestyksessä.

**Lukujärjestelmä.** Jo tuhansia vuosia on kaupankäynnin yhteydessä laskettu lukumäärillä ( $\mathbb{N}$ ) ja painoilla ( $\mathbb{R}_+$ ). Käytännön laskut pakottivat kehittämään murtoluvun käsitteen. Sumerilaiset ja babylonialaiset oppineet ryhtyivät käyttämään 60-järjestelmää lukujen ilmaisemiseen. Järjestelmä sopi hyvin heidän tarpeisiinsa, koska he jakoivat vuoden 360 päivään. Lisäksi 60 on jaollinen luvuilla 2, 3, 5, 6, 10, 12, 15, 20 ja 30. Tämä oli varsin hyödyllistä laskettaessa murtoluvuilla ja taito laskea murtoluvuilla kehittyi hämmästyttäväksi.

Vanhimmat tunnetut varsinaiset matemaattiset tekstit ovat nuolenpääkirjoituksella laaditut Tell-Harmalin savitaulut. Ne löydettiin Bagdadin seudulta 1970-luvulla ja ne ovat kirjoitetut ennen vuotta 2000 eKr. Ne sisältävät euklidisen geometrian pääperiaatteet (mm. myöhemmin Pythagoraan mukaan nimetyn lauseen). Algebran alalta niissä on esitetty mm. toisen asteen yhtälön ratkaisumenettely. Kirjoitusasusta on päätelty, että taulut ovat kopioita varhaisemmista sumerilaisista teksteistä, joten mainitut matemaattiset tehtävät on hallittu jo muinaisessa Sumerissa n. 2500 eKr.

**Murtoluvut.** Babylonialaiset katsoivat, että mitattavia suureita (pituuksia, painoja, yms.) voidaan mielivaltaisen tarkasti approksimoida murtolukujen avulla. Tämä on varsin lähellä nykyistä reaaliluvun käsitettä. Antiikin kreikkalainen Pythagoras (n. 580–500 eKr), joka oli keksinyt sävelasteikon lukusuhteet, julisti puolestaan tämän innoittamana, että kaikki olevainen on ilmaistavissa kokonaislukujen suhteina. Pythagoras perusti Etelä-Italian Krotoniin uskonnollis-siveellisen veljeskunnan, joka harrasti teollista tutkimusta ja tavoitteli poliittista valtaa. Pythagoraan opit tulivat laajalti tunnetuiksi. Siksi lienee ollut kiusallista, kun pythagoralaiset itse havaitsivat kokonaisluku-suhteiden riittämättömyyden.

Asia voidaan todeta seuraavasti: olkoon  $a$  neliön lävistäjä, kun neliön sivu on 1. Silloin Pythagoran lauseen nojalla  $a^2 = 2$ . Jos nyt kaikki voidaan ilmaista kokonaislukujen suhteina, niin  $a$  voidaan kirjoittaa supistettuun murtolukumuotoon  $a = m/n$ , missä  $m$ :llä ja  $n$ :llä ei ole yhteisiä tekijöitä. Nyt

$$\begin{aligned} \left(\frac{m}{n}\right)^2 &= 2 \\ \Leftrightarrow m^2 &= 2n^2 \end{aligned} \quad (1.1)$$

Siis  $m$  on parillinen, joten merkitsemme  $m = 2k$ . Nyt

$$\begin{aligned} (2k)^2 &= 2n^2 \\ \Leftrightarrow n^2 &= 2k^2 \end{aligned} \quad (1.2)$$

Siis  $n$  on parillinen! tämä on ristiriita, sillä  $m$ :llä ja  $n$ :llä ei pitänyt olla yhteisiä tekijöitä.

Pythagoran erehdys ja antiikin kreikkalaisten ankara loogisuus teki täysin selväksi, että lukuja on kahdenlaisia; rationaalisia ( $\mathbb{Q}$  eli murtoluvut) ja irrationaalisia ( $\mathbb{R} - \mathbb{Q}$  eli luvut, jotka eivät ole murtolukuja).

**Lukujärjestelmä.** Kymmenjärjestelmän mukainen lukujen merkitsemistapa kehitettiin ensin Intiassa n. vuonna 500. Arabialainen matemaatikko al-Khowarizmi Bagdadilainen kehitti n. vuonna 825 tätä järjestelmää ottamalla käyttöön nollaa merkitsevän symbolin. Kesti noin 200 vuotta, ennen kuin järjestelmä saapui Espanjaan, jossa se alkoi yleistyä eurooppalaisten kauppiaiden käytössä. Oppineet pitivät yhä 60-järjestelmää parempana ja siirtymäaika lieenee ollut hankala. 1200-luvun lopulla Firenzen kaupunkivaltio sääti lakeja, jotka kielsivät suosituiksi tulneiden kymmenkantaisien lukujen käytön. Tällä suojattiin vanhaan tottuneita kunnon kansalaisia sekaanuksilta, joita uuden järjestelmän käyttö aiheutti.

Tieteen maailmaan kymmenjärjestelmä murtautui kahden keksinnön myötä. Samarkandin astronomisen tutkimuslaitoksen johtaja al-Kashi oli ensimmäisiä, joka 1400-luvulla havaitsi, että negatiivinen eksponentti voidaan ilmaista kymmenkantaisessa järjestelmässä yhtä hyvin kuin 60-järjestelmässäkin. (Siis  $\frac{23}{100} = \frac{2}{10} + \frac{3}{100}$ .) Toinen ratkaisevan nerokas innovaatio tapahtui, kun Skotlantilainen John Napier vuonna 1617 (!) otti käyttöön desimaalipilkun sen nykyisessä merkityksessä. (Siis  $\frac{23}{100} = 0.23$ )

Desimaalipilkun käyttöönoton jälkeen 1600-luvulla laskeminen muuttui helpoksi, mikä välittömästi näkyy nopeana edistymisenä luonnontieteen historiassa. Murros on täysin verrattavissa tietokoneen vaikutukseen tällä vuosisadalla.

**Järjetömästä käsitteestä hyödylliseksi olioksi.** Seuraava lukukäsitteen laajennus tapahtui, kun eräät oppineet alkoivat puhua luvun  $-1$  neliöjuuresta. Suurin osa tutkijoista edusti kantaa, jonka mukaan  $\sqrt{-1}$  on käsitteenä järjetön. Siksi etsittiin yksinkertaista tapaa osoittaa, että lukua  $\sqrt{-1}$  ei ole olemassa. Vakiomenettely haluttaessa osoittaa, että jotakin ei ole olemassa, on olettaa sen olemassaolo ja johtaa tästä ristiriita (kuten Pythagoraan periaatteen falsifioinnissa yllä). Tutkijat siis olettivat, että on olemassa luku  $i = \sqrt{-1}$  ja alkoivat etsiä ristiriitaa. Ristiriitaa ei löytynyt!!! Mitä enemmän etsittiin, sitä enemmän tuli kauniita sovelluksia. Kompleksilukujen, lukujen  $a + ib$ ,  $a, b \in \mathbb{R}$ , joukko osoittautui erittäin hyödylliseksi. Koska ajatus luvusta  $i = \sqrt{-1}$  oli järjenvastainen, kehitettiin tulkinta, jossa kompleksiluku  $z = a + ib$  on lukupari  $(a, b)$  ja kompleksilukujen laskutoimitukset määritellään kaavoilla:

$$\begin{aligned}(a, b) + (c, d) &= (a + c, b + d) \\ (a, b) \cdot (c, d) &= (ac - bd, ad + bc)\end{aligned}\tag{1.3}$$

Tällä tavalla ”järjetön käsite” muuttui ”melko yksinkertaisiksi” olioiksi, joille on määriteltä ”aika erikoiset” laskutoimitukset. Havaittiin, että löytyi muitakin laskutoimituksilla varustettuja oliojoukkoja (funktiot, vektorit, matriisit yms.). Syntyi tarve systematisoida olioilla laskemista.

**Uudet ideat tuottavat lisää uusia ideoita.** Tutkiessaan korkeamman asteen yhtälöiden ratkeavuutta ranskalainen Evariste Galois (1811–32) ja norjalainen Niels Henrik Abel (1802–29) ottivat käyttöön ryhmän käsitteen. Abel onnistui osoittamaan, että viidennen asteen yhtälöä ei voi ratkaista juurilausekkeiden avulla. Tämä oli lopullinen ratkaisu jo sumerilaisia vaivanneeseen tuhansia vuosia vanhaan ongelmaan. Matematiikan nopea edistyminen ja Galois’n ja Abelin henkilöahmot ovat pitäneet yllä myyttiä siitä, että matemaatikot ovat nuoria.

Abelin ryhmää seurasi nopeasti uusia käsitteitä: renkaat, algebrat, lineaariavaruudet, operaattorialgebrat. Nämä ovat joukkoja, joissa on määriteltä tietyt ehdot toteuttavia laskutoimituksia. Kyse on siis edellä mainitusta ”olioilla laskemisesta”. Nämä abstraktit konstruktioit saivat 1900-luvun alussa erittäin käytännöllisiä sovelluksia, kun fyysikot niiden avulla kehittivät kvanttifysiikan ( $\rightarrow$  ydinvoima, elektroniikka, aineen rakenne) ja yleisen suhteellisuusteorian ( $\rightarrow$  atomipommi). Varsinkin Einstein tutki matematiikkaa kuin tavarataloa löytääkseen yleiselle suhteellisuusteorialle itseään tyydyttävän esitystavan.



**Kasvanut laskentavoima luo uusia haasteita.** Ohjelmoitavien koneiden aikakausi alkoi, kun englantilainen Gharles Babbage (1791–1871) alkoi vuonna 1823 rakentaa mekaanista ”differenssikone 1” -nimistä laitetta. Babbage ei koskaan nähnyt laitteen toimivan luotettavasti. Yli sata vuotta myöhemmin on voitu osoittaa, että laitteen suunnitelmassa ei ollut vikaa, vaan sen ohjelmointikieli oli Turing-täydellinen. Laitteessa oli yksinkertaisesti vain liian monta liikkuvaa osaa, eikä sitä saatu luotettavaksi. 1900-luvun alussa alettiin liikkuvia osia korvata sähköisillä releillä ja elektroniputkilla. Koneet tehtiin yleensä tiettyyn tarkoitukseen, jotka olivat tavallisesti sotilaallisia. Z3 (Saksa 1941), Colossus (Englanti 1944) avasi salakirjoitettuja viestejä, ENIAC (USA 1944) laski tykistön ampumataulukkoita. Toisen maailmansodan jälkeen iso joukko sodassa laskentatehtäviä tehneitä nuoria tutkijoita löysi uusia haasteita suurten yritysten laskentatehtävissä. Näistä markkinoista kiinnostuivat myös laitevalmistajat. Mm. IBM, joka siihen asti oli 70 vuoden ajan valmistanut mekaanisia ja sähkömekaanisia vaaakoja, laskukoneita ja kirjoituskoneita, ryhtyi 1950-luvulla valmistamaan ensimmäisiä sarjavalmistettavia suuria luotettavia tietokoneita.

Cern (European Organization for Nuclear Research) alkoi muotoutua 1954 ja sen ensimmäinen hiukkaskiihdytin aloitti toimintansa 1957. CERNin kaltaiset tutkimuslaitokset käyttävät aina paljon resursseja tietojärjestelmiensä kehittämiseen. Vuonna 2012 CERN uutisoi löytäneensä Higin hiukkasen. Sitä ennen CERN on muun ohessa esitellyt maailmalle World Wide Web -järjestelmän 1989 ja LAPACK -aliohjelmakirjaston 1992 (LAPACK = Linear Algebra Package). LAPACK on tänä päivänä käytännössä jokaisen matematiikka-ohjelmiston ytimessä.

Hiukkaskiihdyttimet saavat suuria rahasummia käyttöönsä, mutta yleensä tutkimus ei saa kaikkia haluamiaan resursseja. Siksi on perustettu kansallisia teholaskennan yksiköitä. Suomessa CSC (Center of Scientific Computing) vastaa akateemisen tutkimuksen laskentaresursseista ja myös myy teholaskentaa yrityksille. Yrityksiä kiinnostaa erityisesti simulointi ja data mining. (On halvempaa tehdä laajat simulaatiot laivan kelluntaominaisuuksista, kuin rakentaa viisi prototyyppiä.)

CSC:n tehokkain kone on tällä hetkellä Louhi (Cray XT4/XT5). Louhissa on tuhansia prosessoreita, joiden kellotaajuus on 2,3–2,7GHz. Kellotaajuus ei siis ole poikkeuksellinen. Prosessorit on yhdistetty ryhmiksi ja ryhmät on edelleen yhdistetty nopealla tiedonsiirtoväylällä. Kun Louhella ajetaan ohjelma, mukaan tulee vähintään 32 ja enintään 2048 prosessoria. Ohjelma, jonka ajo läppärillä kestää vuorokauden, saadaan Louhella ajettua minuutissa. Tällä on suuri merkitys tuotekehityksen simullaatioissa.

1980-luvulla kuuli sanottavan, että ”ohjelmointia ei enää tarvita, koska kaikki ohjelmat on jo tehty”. Sen sanojat olivat väärässä, ja jos sanoisivat sen tänään, olisivat taas väärässä. Aina halutaan saada parhaista koneista kaikki teho irti. Kun ennen algoritmeja optimoitiin tavoitteena vähentää prosessorin laskentatyötä, tänään pyritään vähentämään tiedon-siirtely -tarvetta. Monet algoritmit tullaan lähivuosina suunnittelemaan kokonaan uudelleen.

**Edellä kuvatun vaikutus nykymatematiikkaan ja tähän kurssiin.** Tänään ammatimatematiikat pohtivat rakenteisia joukkoja ja algoritmeja. Koulumatematiikan pohjalta syntyy mielikuva siitä, että matemaattinen taito on kykyä ratkaista yhtälö tai laskea funktion arvo. Todellisuudessa kumpikaan ei yleensä ole suuri ongelma. Käytännön ongelmat, optimointi, simulointi yms. ratkaistaan algoritmeilla, jotka toistavat yksinkertaista toimintoa tuhansia kertoja. Oleellista on nyt pystyä selvittämään, millä ehdoilla algoritmi antaa halutun tuloksen. Esimerkiksi Newtonin algoritmi epälineaarisen yhtälön ratkaisemiseksi toimii usein hienosti, mutta ei se aina toimi.

Perinteisesti lineaarialgebran kurssilla on opetettu käsitteet ja esitelty esimerkkejä, jotka on kohtuullisen helppo laskea käsin kynällä ja paperilla. Tällä kurssilla opetetaan samat käsitteet, mutta entisen lisäksi näytetään esimerkkejä, joiden läpilasku vaatii satoja tai tuhansia toistoja. Toistot on pakko tehdä tietokoneella.

### 1.3 Python, alkeita

## 2 TRIGONOMETRIA JA FYSIIKAN VEKTOREITA

Ch:TrigFys

### 2.1 Tukivoimat ja kitka

Sec:TukiKitka

Tukivoimien ja kitkan yleiset periaatteet opitaan fysiikan kurssilla. Seuraavassa lasketaan muutama perusesimerkki. Ne kaikki liittyvät periaatteeseen, jonka mukaan kappale on levossa (eli sen liiketila ei muutu), jos siihen vaikuttavien voimien summa on nollavektori, ja siihen vaikuttavien momenttien summa on nollavektori.

$$\sum_{i=1}^n \vec{F}_i = \vec{0}, \quad (2.1)$$

$$\sum_{i=1}^n \vec{M}_i = \vec{0}. \quad (2.2)$$

$$(2.3)$$

Fysiikan kursseilla esimerkkejä tulee lisää.

**Esimerkki 1 (Kappale vinolla tasolla.)** Tarkastellaan kappaletta vinolla tasolla. Kappaleen massa on 12,25 kg. Tasosta tunnetaan kolme pistettä:  $P = (2.00, 0.00, 0.00)$ ,  $Q = (0.00, 3.00, 0.00)$  ja  $R = (0.00, 0.00, 1.00)$ .

(a) Jaa kappaleen paino tason suuntaiseen ja tasoa vastaan kohtisuoraan komponenttiin.

(b) Laske myös alustan ja kappaleen välisen tukivoiman ja kitkavoiman suuruudet, jos kappale pysyy paikallaan.

Exa:Kvt1

(c) Miten suuri tulee kitkakertoimen vähintään olla, jotta kappale pysyisi paikallaan?

Ratkaisu: Kuvassa [fig:kvt01](#) on piirretty tilanne sivusta nähtynä. Kuvaan on piirretty  $z$ -akseli,

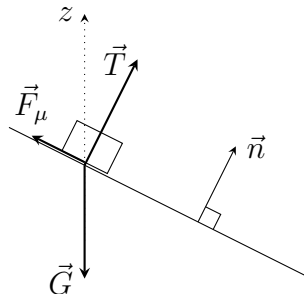


fig:kvt01

**Kuva 1.** Kappale vinolla tasolla.

joka on vastakkaissuuntainen painolle  $\vec{G}$ .  $x$ - ja  $y$ -akseleita ei ole piirretty kuvaan, koska ne eivät ole kuvatason suuntaisia.

(a) Tason normaalivektori  $\vec{n}$  on kohtisuorassa vektoreita  $\vec{PQ} = -2\vec{i} + 3\vec{j}$  ja  $\vec{PR} = -2\vec{i} + \vec{k}$  vastaan, joten yksikkö-normaalivektori on

$$\vec{n} = \frac{1}{|\vec{PQ} \times \vec{PR}|} \vec{PQ} \times \vec{PR}.$$

Painon  $\vec{G}$  tasoa vastaan kohtisuora komponentti on sen vektoriprojektio normaalin suuntaan, eli

$$\vec{G}_{\perp\text{taso}} = \vec{G}_n = (\vec{G} \cdot \vec{n})\vec{n}.$$

Tason suuntainen komponentti on

$$\vec{G}_{\parallel\text{taso}} = \vec{G} - \vec{G}_{\perp\text{taso}} = \vec{G} - (\vec{G} \cdot \vec{n})\vec{n}.$$

Octavella laskettuna:

```
octave:1> P = [2;0;0];
octave:2> Q = [0;3;0];
octave:3> R = [0;0;1];
octave:4> PQ = Q-P;
octave:5> PR = R-P;
octave:6> n = (cross(PQ,PR)) / (norm(cross(PQ,PR)))
n =

    0.42857
    0.28571
    0.85714

octave:7> G = [0;0;-12.25];
octave:8> Gkohtis = dot(G,n)*n
Gkohtis =

   -4.5000
   -3.0000
   -9.0000

octave:9> Gyhdens = G - Gkohtis
Gyhdens =

    4.5000
    3.0000
   -3.2500
```

Siis  $\vec{G}_{\perp\text{taso}} = (-4.50\vec{i} - 3.00\vec{j} - 9.00\vec{k})\text{N}$  ja  $\vec{G}_{\parallel\text{taso}} = (4.50\vec{i} + 3.00\vec{j} - 3.25\vec{k})\text{N}$ .

(b)

$$\begin{aligned}
 |\vec{T}| &= |-\vec{G}_{\perp\text{taso}}| = |4.50\vec{i} + 3.00\vec{j} + 9.00\vec{k}| \text{ N} \\
 &= \sqrt{4.50^2 + 3.00^2 + 9.00^2} \text{ N} = 10.50 \text{ N} \\
 |\vec{F}_\mu| &= |-\vec{G}_{\parallel\text{taso}}| = |-4.50\vec{i} - 3.00\vec{j} + 3.25\vec{k}| \text{ N} \\
 &= \sqrt{(-4.50)^2 + (-3.00)^2 + 3.25^2} \text{ N} = 6.31 \text{ N}
 \end{aligned}$$

(c) Kitkakertoimen tulee olla vähintään

$$\mu \geq \frac{|\vec{F}_\mu|}{|\vec{T}|} = \frac{6.31 \text{ N}}{10.50 \text{ N}} = 0.60.$$

□

**Esimerkki 2 (Kappale vinolla tasolla, jatkuu.)** Tarkastellaan vielä edellisen esimerkin tasoa. Määritä tason ja vaakatason välinen kulma.

Exa:Kvt2

**Ratkaisu:** Tason ja vaakatason välinen kulma  $\alpha$  on tason normaalin  $\vec{n}$  ja pystysuoran  $\vec{k}$  välinen kulma

$$\begin{aligned}
 \alpha &= \arccos\left(\frac{\vec{n} \cdot \vec{k}}{|\vec{n}| |\vec{k}|}\right) = \arccos(\vec{n} \cdot \vec{k}) \\
 &= \arccos\left((0.42857\vec{i} + 0.28571\vec{j} + 0.85714\vec{k}) \cdot (\vec{k})\right) \\
 &= \arccos(0.85714) = 0.5411 \text{ (rad)} = 31.00^\circ
 \end{aligned}$$

□

**Esimerkki 3 (Palkin nosto.)** Vaakatasossa lepäävä palkki, jonka massa on  $m = 500.00 \text{ kg}$  ja pituus on  $L = 4,000 \text{ m}$ , nostetaan pystyyn. Nostovaijerin yläpää on suoraan palkin maahan jäävän pää yläpuolella korkeudella  $h = 7.35 \text{ m}$ . Nosturi kelaa vaijeria niin, että vaijeri lyhenee, mutta vaijerin yläpää pysyy paikallaan. Palkin toinen pää nousee vaijerin nostamana, ja tavoitteena on saada palkki pystyyn niin, että palkin alapää ei liukahda noston aikana paikaltaan. Tarkastellaan kuvan  <sup>Fig:Pnostol</sup> 2 mukaista noston vaihetta. (a) Laske palkin alapäähän kuvan tilanteessa vaikuttavan kitkavoiman suuruus. (b) Millä ehdolla palkki ei liukahda noston aikana, kun alustan ja palkin välinen lepokitkakerroin on 0.50?

Exa:Pnostol

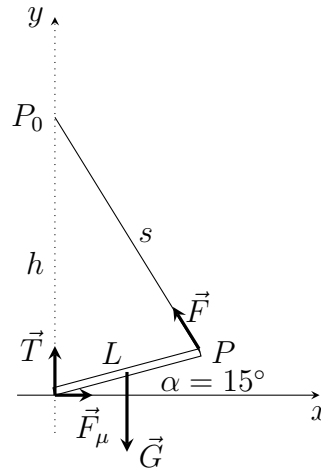


fig:Pnostol

**Kuva 2.** Palkin nosto.

(a) Palkin nousevan päään koordinaatit ovat

$$x_P = L \cos \alpha = 4.000 \text{ m} \cos 15^\circ = 3.86370 \text{ m} \quad \text{ja}$$

$$y_P = L \sin \alpha = 4.000 \text{ m} \sin 15^\circ = 1.03528 \text{ m}.$$

Vaijerin pituus on  $s = \sqrt{x_P^2 + (h - y_P)^2} = \sqrt{(3.86370 \text{ m})^2 + (6.31472 \text{ m})^2} = 7.40296 \text{ m}$

Vaijerin suuntavektori  $\vec{e}$  on

$$\begin{aligned} \vec{e} &= \frac{\overrightarrow{PP_0}}{s} = \frac{-x_P \vec{i} + (h - y_P) \vec{j}}{s} = \frac{-3.86370 \text{ m} \vec{i} + 6.31472 \text{ m} \vec{j}}{7.40296 \text{ m}} \\ &= -0.521913 \vec{i} + 0.852999 \vec{j}. \end{aligned}$$

Vaijerin tukivoima on

$$\vec{F} = F \vec{e} = \underbrace{-0.521913 F \vec{i}}_{=F_x} + \underbrace{0.852999 F \vec{j}}_{=F_y}.$$

Paino on

$$\vec{G} = -mg \vec{j} = -500,00 \text{ kg} \cdot 9.81 \frac{\text{m}}{\text{s}^2} = -4905.00 \text{ N} \vec{j}$$

Voimien momentit origon suhteen ovat

$$\begin{aligned} \vec{M}_F &= (x_P \vec{i} + y_P \vec{j}) \times (F_x \vec{i} + F_y \vec{j}) = (x_P F_y - y_P F_x) \vec{k} \\ &= (3.86370 \text{ m} \cdot 0.852999 F - 1.03528 \text{ m} \cdot (-0.521913 \text{ m}) F) \vec{k} = F \cdot 3.83606 \text{ m} \vec{k}, \end{aligned}$$

$$\begin{aligned} \vec{M}_G &= \left( \frac{x_P}{2} \vec{i} + \frac{y_P}{2} \vec{j} \right) \times (-mg \vec{j}) = -\frac{x_P mg}{2} \vec{k} \\ &= -\frac{3.86370 \text{ m} \cdot 4905.00 \text{ N}}{2} \vec{k} = -9475.72 \text{ Nm} \vec{k} \end{aligned}$$

$$\vec{M}_T = \vec{0} \text{ (koska varsi on nolla),}$$

$$\vec{M}_{F_\mu} = \vec{0} \text{ (koska varsi on nolla).}$$

Tasapainoedosta seuraa nyt, että (1) momenttien summa on nollavektori, (2) voimien pystysuorien komponenttien summa on nolla, ja (3) voimien vaakasuorien komponenttien summa on nolla.

$$\begin{cases} \vec{M}_F + \vec{M}_G = \vec{0} \\ F_y + T - G = 0 \\ F_x + F_\mu = 0 \end{cases} \Leftrightarrow \begin{cases} 3.83606 \text{ m} \vec{k} \cdot F = 9475.72 \text{ Nm} \vec{k} \\ 0.85300 \cdot F + T = 4905 \text{ N} \\ -0.52192 \cdot F + F_\mu = 0 \end{cases} \begin{matrix} (1) \\ (2) \\ (3) \end{matrix}$$

$$(1) \Rightarrow F = \frac{9475.72 \text{ N}}{3.83606} = 2470.17 \text{ N}$$

$$(2) \Rightarrow T = 4905.00 \text{ N} - 0.852999 \cdot 2470.17 \text{ N} = 2797.95 \text{ N}$$

$$(3) \Rightarrow F_\mu = 0.521913 \cdot 2470.17 \text{ N} = 1289.23 \text{ N}$$

Siis kitkavoima on noin 1 290 N.

Kitkavoiman suhde normaali-tukivoimaan on

$$\frac{1289 \text{ N}}{2798 \text{ N}} = 0.46 < \mu.$$

Palkki siis pysyy tässä asennossa ilman, että maahan tukeutuva palkin pää alkaisi liukua alustalla.

(b) Se, että näillä arvoilla palkki ei liu'u alustalla, ei vielä riitä takaamaan noston onnistumista alusta loppuun saakka. Edellä laskimme momentit suoraviivaisesti käyttäen tehtävässä annettuja suureita. Nyt otamme noston vaihetta kuvaaviksi muuttujiksi vaijerin pituuden  $s$  ja vaijerin ja pystysuunnan välisen kulman  $\beta$ . (Ks. kuva 3.)

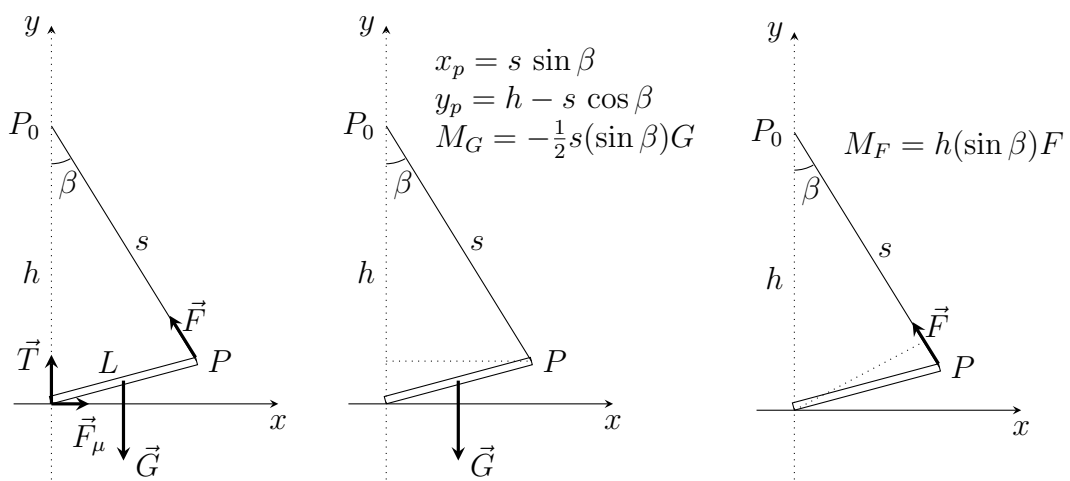


fig:Pnosto2

**Kuva 3.** Momentit palkin nostossa.

Momenttiehto saa nyt muodon:

$$\begin{aligned} M_F + M_G &= 0 \\ \Leftrightarrow F \cdot h \sin \beta &= \frac{1}{2} G \cdot s \sin \beta \\ \Leftrightarrow F &= \frac{s}{2h} G \end{aligned}$$

Kaikkien voimien summa on nolla:

$$\begin{aligned} \vec{F}_\mu + \vec{F}_x &= \vec{0} \quad \text{vaakasuorat komponentit} \\ \Rightarrow F_\mu &= -F_x = F \sin \beta = \frac{s \sin \beta}{2h} G = \frac{x_P}{2h} G \end{aligned}$$

$$\begin{aligned} \vec{T} + \vec{F}_y + \vec{G} &= \vec{0} \quad \text{pystysuorat komponentit} \\ \Rightarrow T &= G - F \cos \beta = G - \frac{s \cos \beta}{2h} G \\ &= \left( \frac{2h}{2h} - \frac{(h - y_P)}{2h} \right) G = \frac{h + y_P}{2h} G \end{aligned}$$

Kitkan ja tukivoiman suhde on siis

$$\frac{F_\mu}{T} = \frac{x_P}{h + y_P}.$$

Noston aikana tämän suhteen osoittaja pienenee ja nimittäjä kasvaa, joten suurimman arvonsa suhde saa noston alussa. Silloin

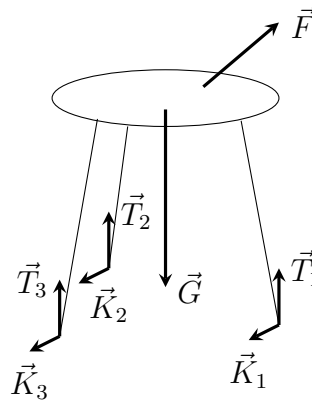
$$\left( \frac{F_\mu}{T} \right)_{\text{alussa}} = \frac{L}{h}.$$

Nosto onnistuu suunnitellulla tavalla, jos  $L/h < \mu$ . Nyt esimerkin arvoilla  $L/h = 4/7.35 > 0.5$ . Ehto ei siis toteudu ja palkki liukuu paikaltaan. Nostoa varten tulee saada korkeampi nosturi, tai sitten palkin alapää tulee tukea noston aikana.  $\square$

**Esimerkki 4 (Jakkaran tasapaino.)** Kolmijalkaisen jakkaran jalat koskettavat vaakatasoa ( $x, y$ -taso) pisteissä  $P_i = (r \cos(\pi/18 + i \cdot \pi/3), r \sin(\pi/18 + i \cdot \pi/3), 0)$ ,  $i = 1, 2, 3$ , missä  $r = 20,0$  cm. Siis Kosketuspisteiden paikkavektorit ovat:

$$\begin{aligned} \vec{r}_1 &= 19.70 \text{ cm} \cdot \vec{i} + 3.47 \text{ cm} \cdot \vec{j} + 0 \cdot \vec{k} \\ \vec{r}_2 &= -12.86 \text{ cm} \cdot \vec{i} + 15.32 \text{ cm} \cdot \vec{j} + 0 \cdot \vec{k} \\ \vec{r}_3 &= -6.84 \text{ cm} \cdot \vec{i} - 18.79 \text{ cm} \cdot \vec{j} + 0 \cdot \vec{k} \end{aligned}$$





**Kuva 4.** Jakkaran kaato.

fig:Jakkara01

Jakkaran korkeus on 45 cm. Jakkaran massa on 3.25 kg, ja painopiste on kohdassa  $\vec{r}_{cm} = 40\text{cm}\vec{k}$ . Jakkaran pisteeseen  $Q = (5\text{cm}, 10\text{cm}, 45\text{cm})$  vaikuttaa voima  $\vec{F} = 4.00\text{N}\vec{i} + 3.00\text{N}\vec{j} + 7.00\text{N}\vec{k}$ . Tarkista, että jakkara ei kaadu voiman takia.

Käytännössä on mahdollista, että jakkara alkaa liukua alustalla. Liukumisen tutkiminen on jo hieman mutkikkaampi asia. Emme tee sitä nyt, mutta myöhemmin esimerkissä [Ex:Kitka](#) käsittelemme hieman yksinkertaisemmän liukumisongelman.

Koska muut voimat ovat pysty- tai vaakasuoria, jaamme myös vektorin  $\vec{F}$  pysty- ja vaakasuoraan komponenttiin.

$$\vec{F} = \underbrace{4.000\text{N}\vec{i} + 3.000\text{N}\vec{j}}_{\vec{F}_{\text{vaaka}}} + \underbrace{7.000\text{N}\vec{k}}_{\vec{F}_{\text{pysty}}}$$

Voiman vaakasuora komponentti voidaan myös kirjoittaa muotoon  $\vec{F}_{\text{vaaka}} = 5.000\text{N}\vec{e}$ , missä  $\vec{e} = 0.8000\vec{i} + 0.6000\vec{j}$ . Paino on  $\vec{G} = -3.25\text{kg} \cdot 9.81 \frac{\text{m}}{\text{s}^2} \vec{k} = -31.8825\text{N}\vec{k}$ .

Laskemme seuraavaksi miten suuria tukivoimien  $\vec{T}_1 = T_1\vec{k}$ ,  $\vec{T}_2 = T_2\vec{k}$ ,  $\vec{T}_3 = T_3\vec{k}$  tulee olla, jotta jakkara pysyy kaatutumatta. Kitkavoimia  $\vec{K}_1$ ,  $\vec{K}_2$ ,  $\vec{K}_3$  emme nyt ratkaise, mutta pidämme ne alussa mukana tarkastelussa, jotta varmasti kaikki tarvittava tulee huomioitua. Jos jokin tukivoima suuntautuu alaspäin ( $T_k < 0$ , jollekin  $k$ ), niin jakkara kaatuu. Jos kaikki tukivoimat suuntautuvat ylöspäin ( $T_k \geq 0, \forall k$ ), niin jakkara pysyy pystyssä.

Kitkavoimat  $\vec{K}_1$ ,  $\vec{K}_2$  ja  $\vec{K}_3$  ovat vaakasuoria ja vastakkaissuuntaisia vektorille  $\vec{e}$ . Siis  $\vec{K}_i = -K_i\vec{e}, i = 1, 2, 3$ . (Tämä ei ole triviaali asia, mutta voidaan perustella energiaperiaatteella.)

Voimia koskevat tasapainoehdot ovat

$$\begin{cases} \sum_k \vec{T}_k + \vec{F}_{pysty} + \vec{G} = \vec{0} \\ \sum_k \vec{K}_k + \vec{F}_{vaaka} = \vec{0} \end{cases} \Rightarrow \begin{cases} T_1\vec{k} + T_2\vec{k} + T_3\vec{k} = 24.8825\text{N}\vec{k} \\ -K_1\vec{e} - K_2\vec{e} - K_3\vec{e} = -5.000\text{N}\vec{e} \end{cases}$$

$$\Rightarrow \begin{cases} T_1 + T_2 + T_3 = 24.8825\text{N} \\ K_1 + K_2 + K_3 = 5.0000\text{N} \end{cases}$$

Valmistaudumme tarkistamaan tasapainon momenttiehtoa. Tätä varten lasketaan seuraavaksi kaikkien voimien momentit origon suhteen.

$$\begin{aligned} \vec{M}_F &= \vec{OQ} \times \vec{F} = (5\text{cm}\vec{i} + 10\text{cm}\vec{j} + 45\text{cm}\vec{k}) \times (4\text{N}\vec{i} + 3\text{N}\vec{j} + 7\text{N}\vec{k}) \\ &= -0.6500\text{Nm}\vec{i} + 1.4500\text{Nm}\vec{j} - 0.2500\text{Nm}\vec{k} \\ \vec{M}_G &= \vec{0} \end{aligned}$$

$$\begin{aligned} \vec{M}_{T_1} &= \vec{r}_1 \times \vec{T}_1 = (19.70\text{cm} \cdot \vec{i} + 3.47\text{cm} \cdot \vec{j}) \times T_1\vec{k} = T_1 \cdot 0.0347\text{m}\vec{i} - T_1 \cdot 0.1970\text{m}\vec{j} \\ \vec{M}_{T_2} &= \vec{r}_2 \times \vec{T}_2 = (-12.86\text{cm} \cdot \vec{i} + 15.32\text{cm} \cdot \vec{j}) \times T_2\vec{k} = T_2 \cdot 0.1532\text{m}\vec{i} + T_2 \cdot 0.1286\text{m}\vec{j} \\ \vec{M}_{T_3} &= \vec{r}_3 \times \vec{T}_3 = (-6.84\text{cm} \cdot \vec{i} - 18.79\text{cm} \cdot \vec{j}) \times T_3\vec{k} = T_3 \cdot (-0.1879\text{m})\vec{i} + T_3 \cdot 0.0684\text{m}\vec{j} \end{aligned}$$

$$\begin{aligned} \vec{M}_{K_1} &= \vec{r}_1 \times \vec{K}_1 = (19.70\text{cm} \cdot \vec{i} + 3.47\text{cm} \cdot \vec{j}) \times (-K_1 \cdot 0.8000\vec{i} - K_1 \cdot 0.6000\vec{j}) \\ &= -K_1 \cdot 0.09044\text{m}\vec{k} \\ \vec{M}_{K_2} &= \vec{r}_2 \times \vec{K}_2 = (-12.86\text{cm} \cdot \vec{i} + 15.32\text{cm} \cdot \vec{j}) \times (-K_2 \cdot 0.8000\vec{i} - K_2 \cdot 0.6000\vec{j}) \\ &= K_2 \cdot 0.19972\text{m}\vec{k} \\ \vec{M}_{K_3} &= \vec{r}_3 \times \vec{K}_3 = (-6.84\text{cm} \cdot \vec{i} - 18.79\text{cm} \cdot \vec{j}) \times (-K_3 \cdot 0.8000\vec{i} - K_3 \cdot 0.6000\vec{j}) \\ &= -K_3 \cdot 0.10928\text{m}\vec{k} \end{aligned}$$

Momenttiehto:

$$\begin{aligned} \vec{0} &= \vec{M}_F + \sum_k \vec{M}_{T_k} + \sum_k \vec{M}_{K_k} \\ &= (-0.6500\text{N} + 0.0347T_1 + 0.1532T_2 - 0.1879T_3)\vec{i} \\ &\quad + (1.4500\text{N} - 0.1970T_1 + 0.1286T_2 + 0.0684T_3)\vec{j} \\ &\quad + (-0.2500\text{N} - 0.09044K_1 + 0.19972K_2 - 0.10928K_3)\vec{k} \end{aligned}$$

Momenttien summa on nollavektori, jos  $\vec{i}$ :n kerroin on nolla,  $\vec{j}$ :n kerroin on nolla ja  $\vec{k}$ :n kerroin on nolla. Kun kokoamme kaikki saamamme yhtälöt yhteen, saamme viisi

yhtälöä

$$\left\{ \begin{array}{rrrrcl} 0.0347T_1 & + & 0.1532T_2 & - & 0.1879T_3 & = & 0.6500\text{N} \\ -0.1970T_1 & + & 0.1286T_2 & + & 0.0684T_3 & = & -1.4500\text{N} \\ T_1 & + & T_2 & + & T_3 & = & 24.8825\text{N} \\ -0.09044K_1 & + & 0.19972K_2 & - & 0.10928K_3 & = & 0.2500\text{N} \\ K_1 & + & K_2 & + & K_3 & = & 5.0000\text{N} \end{array} \right.$$

Tukivoimat ratkeavat kolmesta ensimmäisestä yhtälöstä. Yhtälöryhmien ratkaisemista opettelemme seuraavassa luvussa. Myöhemmin, kun olemme käyneet läpi yhtälöryhmien ratkaisemisen, voit palata tarkistamaan ratkaisun. Nyt otamme ratkaisun Octavela

```
octave:1> A = [0.0347 0.1532 -0.1879;
>              -0.1970 0.1286 0.0684;
>              1      1      1      ]
A =

    0.034700    0.153200   -0.187900
   -0.197000    0.128600    0.068400
    1.000000    1.000000    1.000000

octave:2> b = [0.6500; -1.4500; 24.8825]
b =

    0.65000
   -1.45000
   24.88250

octave:3> T = A\b
T =

   13.4297
    6.8484
    4.6045
```

Koska kaikki tukivoimat suuntautuvat ylöspäin, jakkara ei kaadu. □

**Esimerkki 5 (Vaijereihin ripustettu massa)** 10.00kg massa on ripustettu kiinnitysrenkaaseen, josta lähtee kolme vaijeria. Ensimmäisen vaijerin yläpää on kiinni pisteessä  $P_1 = (0\text{cm}, 225\text{cm}, 300\text{cm})$ , toisen vaijerin yläpää on kiinni pisteessä  $P_2 = (310\text{cm}, 0\text{cm}, 300\text{cm})$  ja kolmannen vaijerin yläpää on kiinnitetty pisteeseen  $P_3 = (0\text{cm}, 0\text{cm}, 300\text{cm})$ . Massa lasketaan roikkumaan vaijereiden varaan, jolloin kiinnitysrengas asettuu kohtaan  $P_0 = (112\text{cm}, 135\text{cm}, 252\text{cm})$ . Laske vaijereiden jännitykset.

Lasketaan ensin vaijereiden (tukivoimien) suuntavektorit

$$\begin{aligned}\vec{u}_1 &= \frac{\overrightarrow{P_0P_1}}{|\overrightarrow{P_0P_1}|} = \frac{-112\vec{i} + 90\vec{j} + 48\vec{k}}{\sqrt{112^2 + 90^2 + 48^2}} = -0.7393\vec{i} + 0.5941\vec{j} + 0.3169\vec{k} \\ \vec{u}_2 &= \frac{\overrightarrow{P_0P_2}}{|\overrightarrow{P_0P_2}|} = \frac{198\vec{i} - 135\vec{j} + 48\vec{k}}{\sqrt{198^2 + 135^2 + 48^2}} = 0.8101\vec{i} - 0.5524\vec{j} + 0.1964\vec{k} \\ \vec{u}_3 &= \frac{\overrightarrow{P_0P_3}}{|\overrightarrow{P_0P_3}|} = \frac{-112\vec{i} - 135\vec{j} + 48\vec{k}}{\sqrt{112^2 + 135^2 + 48^2}} = -0.6159\vec{i} - 0.7423\vec{j} + 0.2639\vec{k}\end{aligned}$$

Tasapainon voima-ehto on nyt

$$\begin{aligned}\vec{F}_1 + \vec{F}_2 + \vec{F}_3 + \vec{G} &= \vec{0} \\ \Leftrightarrow F_1\vec{u}_1 + F_2\vec{u}_2 + F_3\vec{u}_3 - mg\vec{k} &= \vec{0} \\ \Leftrightarrow F_1(-0.7393\vec{i} + 0.5941\vec{j} + 0.3169\vec{k}) \\ &\quad + F_2(0.8101\vec{i} - 0.5524\vec{j} + 0.1964\vec{k}) \\ &\quad + F_3(-0.6159\vec{i} - 0.7423\vec{j} + 0.2639\vec{k}) - 98.1\text{N}\vec{k} = \vec{0} \\ \Leftrightarrow (-0.7393F_1 + 0.8101F_2 - 0.6159F_3)\vec{i} \\ &\quad + (0.5941F_1 - 0.5524F_2 - 0.7423F_3)\vec{j} \\ &\quad + (0.3169F_1 + 0.1964F_2 + 0.2639F_3 - 98.1\text{N})\vec{k} = \vec{0} \\ \Leftrightarrow \begin{cases} -0.7393F_1 + 0.8101F_2 - 0.6159F_3 = 0 \\ 0.5941F_1 - 0.5524F_2 - 0.7423F_3 = 0 \\ 0.3169F_1 + 0.1964F_2 + 0.2639F_3 = 98.1\text{N} \end{cases}\end{aligned}$$

```
octave:4> P0 = [112; 135; 252];
octave:5> P1 = [ 0; 225; 300];
octave:6> P2 = [310; 0; 300];
octave:7> P3 = [ 0; 0; 300];
octave:8> u1 = (P1-P0)/norm(P1-P0)
u1 =

    -0.73934
     0.59411
     0.31686

octave:9> u2 = (P2-P0)/norm(P2-P0)
u2 =

     0.81014
    -0.55237
```

```

0.19640

octave:10> u3 = (P3-P0)/norm(P3-P0)
u3 =

-0.61586
-0.74233
0.26394

octave:11> A = [u1 u2 u3]
A =

-0.73934    0.81014   -0.61586
0.59411   -0.55237   -0.74233
0.31686    0.19640    0.26394

octave:12> b = [0; 0; 98.1]
b =

0.00000
0.00000
98.10000

octave:13> voimat = A\b
voimat =

185.760
180.464
14.387

```

Siis vaijereiden jännitykset ovat  $F_1 = 186\text{ N}$ ,  $F_2 = 180\text{ N}$  ja  $F_3 = 14.4\text{ N}$ . □

## 2.2 Ristikon tukivoimat

Sec:RistikkoI

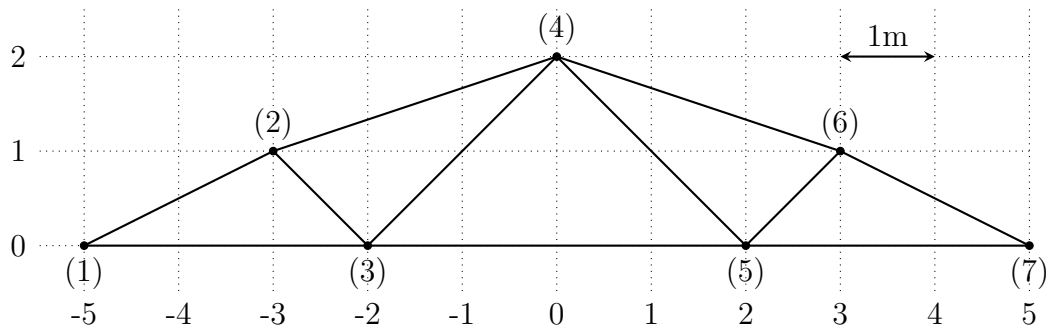
**Esimerkki 6 (Ristikko, symmetrinen taakka.))** Tarkastellaan kuvan [fig:RistikkoI1](#) mukaista katto-ristikkoa. Ristikon liitospisteiden koordinaatit ovat

$$\begin{aligned}
 P_1 &= (-5, 0) & P_7 &= (5, 0) \\
 P_2 &= (-3, 1) & P_6 &= (3, 1) \\
 P_3 &= (-2, 0) & P_5 &= (2, 0) \\
 P_4 &= (0, 2)
 \end{aligned}$$

Katolla on 50cm lunta. Lumen vesi-arvo on noin 150mm. Ristikko kantaa noin  $8\text{ m}^2$  kattopintaa, joten ristikon päällä olevan lumitaakan massa on noin 1200kg. Ristikon

kantama kattorakenteiden massa on noin 400kg. Lumen ja kattorakenteiden yhteenlaskettu paino on siis  $\vec{G} = -15\,700\text{N}\vec{j}$ . Ristikoon vaikuttavat ulkoiset taakat ja tukivoimat, sekä ristikon palkkien jännitysvoimat on piirretty kuvaan 6. Huomaa, että tukivoimat ja taakat on piirretty kuvaan erikoisella tavalla. Oikeastaan vektorit pitäisi piirtää alkamaan vaikutuspisteestä, mutta tukivoimat on piirretty vaikutussuoralleen ristikon alapuolelle, ja taakat on piirretty ristikon yläpuolelle. Laske palkkeihin vaikuttavat voimat.

Ex:ristikko1



Kuva 5. Kattoristikko.

fig:RistikkoI1

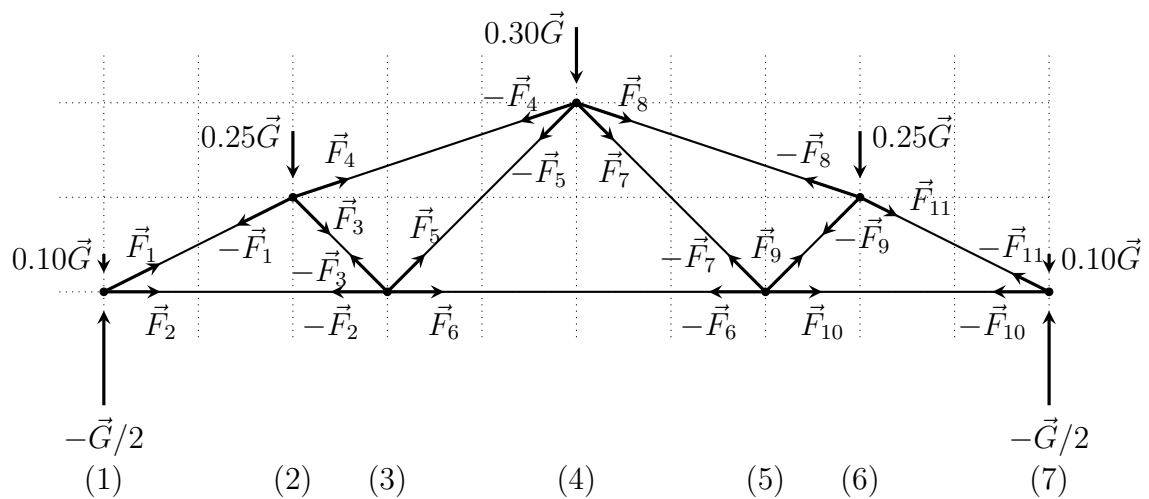
Jokaisessa ristikon liitospisteessä vaikuttavien voimien summa on nollavektori. Syntyy siis yksi vektoriyhtälö jokaista liiipospistettä kohti.

$$\left\{ \begin{array}{ll} \vec{F}_1 + \vec{F}_2 & = 0.40\vec{G} \\ -\vec{F}_1 + \vec{F}_3 + \vec{F}_4 & = -0.25\vec{G} \\ -\vec{F}_2 - \vec{F}_3 + \vec{F}_5 + \vec{F}_6 & = \vec{0} \\ -\vec{F}_4 - \vec{F}_5 + \vec{F}_7 + \vec{F}_8 & = -0.30\vec{G} \\ -\vec{F}_6 - \vec{F}_7 + \vec{F}_9 + \vec{F}_{10} & = \vec{0} \\ -\vec{F}_8 - \vec{F}_9 + \vec{F}_{11} & = -0.25\vec{G} \\ -\vec{F}_{10} - \vec{F}_{11} & = 0.40\vec{G} \end{array} \right.$$

Kaikkien yhtälöiden summa on triviaali  $\vec{0} = \vec{0}$ , joten yksi yhtälöistä voidaan poistaa ryhmästä. Poistamme neljännen yhtälön. Kolmesta ensimmäisestä yhtälöstä ratkaisemme voimat  $\vec{F}_1, \dots, \vec{F}_6$ . Loput voimat saadaan symetrian perusteella

Lasketaan ensin voimien  $\vec{F}_1, \dots, \vec{F}_6$  yksikkösuuntavektorit

$$\begin{aligned} \vec{F}_1 &= F_1(u_1\vec{i} + v_1\vec{j}) = F_1(0.894427\vec{i} + 0.447214\vec{j}) \\ \vec{F}_2 &= F_2(u_2\vec{i} + v_2\vec{j}) = F_2\vec{i} \\ \vec{F}_3 &= F_3(u_3\vec{i} + v_3\vec{j}) = F_3(0.707107\vec{i} - 0.707107\vec{j}) \\ \vec{F}_4 &= F_4(u_4\vec{i} + v_4\vec{j}) = F_4(0.948683\vec{i} + 0.316228\vec{j}) \\ \vec{F}_5 &= F_5(u_5\vec{i} + v_5\vec{j}) = F_5(0.707107\vec{i} + 0.707107\vec{j}) \\ \vec{F}_6 &= F_6(u_6\vec{i} + v_6\vec{j}) = F_6\vec{i} \end{aligned}$$



**Kuva 6.** Kattoristikon voimat.

fig:RistikkoI2

Hajoitamme kolme ensimmäistä vektori yhtälöä yhtälöä vaakasuoria ja pystysuoria komponentteja koskeviin osiin. Syntyvät kuusi yhtälöä antavat kuuden voiman suuruudet.

$$\left\{ \begin{array}{lcl} u_1 F_1 + F_2 & = & 0 \\ v_1 F_1 & = & -0.4G \\ -u_1 F_1 + u_3 F_3 + u_4 F_4 & = & 0 \\ -v_1 F_1 + v_3 F_3 + v_4 F_4 & = & 0.25G \\ -F_2 - u_3 F_3 + u_5 F_5 + F_6 & = & 0 \\ -v_3 F_3 + v_5 F_5 + F_6 & = & 0 \end{array} \right.$$

Ratkaisemme yhtälöryhmän Octavella

```
octave:1> P1 = [-5 0]';
octave:2> P2 = [-3 1]';
octave:3> P3 = [-2 0]';
octave:4> P4 = [0 2]';
octave:5> P5 = [2 0]';
octave:6> w1 = (P2-P1)/norm(P2-P1);
octave:7> w2 = (P3-P1)/norm(P3-P1);
octave:8> w3 = (P3-P2)/norm(P3-P2);
octave:9> w4 = (P4-P2)/norm(P4-P2);
octave:10> w5 = (P4-P3)/norm(P4-P3);
octave:11> w6 = (P5-P3)/norm(P5-P3);
octave:12> z = [0; 0];
octave:13> A = [
> w1 w2 z z z z;
> -w1 z w3 w4 z z;
> z -w2 -w3 z w5 w6]
A =
```

```

0.89443  1.00000  0.00000  0.00000  0.00000  0.00000
0.44721  0.00000  0.00000  0.00000  0.00000  0.00000
-0.89443  0.00000  0.70711  0.94868  0.00000  0.00000
-0.44721  0.00000 -0.70711  0.31623  0.00000  0.00000
0.00000 -1.00000 -0.70711  0.00000  0.70711  1.00000
0.00000 -0.00000  0.70711  0.00000  0.70711  0.00000

octave:14> G = 15700;
octave:15> b = [ 0 -0.4*G 0 0.25*G 0 0]';
octave:16> F = A\b
F =

-1.4043e+04
 1.2560e+04
-1.9428e+03
-1.1791e+04
 1.9428e+03
 9.8125e+03

```

Voimat ovat siis

$$\begin{aligned}
 F_1 &= -14000\text{N}, & (\text{puristus}), \\
 F_2 &= 12600\text{N}, & (\text{venytys}), \\
 F_3 &= -1900\text{N}, & (\text{puristus}), \\
 F_4 &= -11800\text{N}, & (\text{puristus}), \\
 F_5 &= 1900\text{N}, & (\text{venytys}), \\
 F_6 &= 9800\text{N}, & (\text{venytys}).
 \end{aligned}$$

□

Edellä käytimme symmetriaa kahdella tavalla. Symmetrian takia momenttien summa on nollavektori, joten momentteja ei tarvinnut tarkastella. Toisaalta symmetrian avulla voitiin tuntemattomien muuttujien määrä lähes puolittaa. Jos lumi on kasautunut toiselle lappeelle (sama kokonaismassa), niin symmetria rikkoutuu.

**Esimerkki 7 (Ristikko, epäsymmetrinen taakka.)** Tarkastellaan uudelleen esimerkin [Ex:ristikko1](#) 6 ristikkoa, mutta tilanteessa, jossa lumi on kasautunut epätasaisesti. Laske ristikon palkkeihin vaikuttavat voimat, kun ristikon taakan jakauma on:

| Liitospiste | taakka        | varsitukivoima  |
|-------------|---------------|-----------------|
| (1)         | $0.05\vec{G}$ | $-t\vec{G}$     |
| (2)         | $0.10\vec{G}$ |                 |
| (4)         | $0.35\vec{G}$ |                 |
| (6)         | $0.40\vec{G}$ |                 |
| (7)         | $0.10\vec{G}$ | $-(1-t)\vec{G}$ |



Ratkaisemme ensin tukivoimat momenttiehdosta. Koska palkkeihin vaikuttavat voimat ovat pareittain vastakkaisia, on niiden kokonaismomentti nollavektori. Momenttiehto koskee siis taakkoja ja tukivoimia. Lasketaan ulkoisten voimien momentit

| Liitospiste | varsi $\vec{r}$    | voima $\vec{F}$      | momentti $\vec{r} \times \vec{F}$ |
|-------------|--------------------|----------------------|-----------------------------------|
| taakat      |                    |                      |                                   |
| (1)         | $-5\vec{m}\vec{i}$ | $-0.05G\vec{j}$      | $0.25G\vec{m}\vec{k}$             |
| (2)         | $-3\vec{m}\vec{i}$ | $-0.10G\vec{j}$      | $0.30G\vec{m}\vec{k}$             |
| (4)         | $\vec{0}$          | $-0.35G\vec{j}$      | $\vec{0}$                         |
| (6)         | $3\vec{m}\vec{i}$  | $-0.40G\vec{j}$      | $-1.20G\vec{m}\vec{k}$            |
| (7)         | $5\vec{m}\vec{i}$  | $-0.10G\vec{j}$      | $-0.50G\vec{m}\vec{k}$            |
| tukivoimat  |                    |                      |                                   |
| (1)         | $-5\vec{m}\vec{i}$ | $tG\vec{j}$          | $-5tG\vec{m}\vec{k}$              |
| (7)         | $5\vec{m}\vec{i}$  | $(1-t)G\vec{j}$      | $(5G-5tG)\vec{m}\vec{k}$          |
| yhteensä:   |                    | $0 = \sum \vec{M} =$ | $(3.85 - 10t)G\vec{m}\vec{k}$     |

Siis  $t = 0.385$  ja  $(1 - t) = 0.615$ .

Teemme ongelman ratkaisusta lyhyen Octave-ohjelman

```
# Pisteiden koordinaatit
P = [-5 0; -3 1; -2 0; 0 2; 2 0; 3 1; 5 0]';
# pisteiden lukumaara on npi
[c,npi] = size(P);
# -----
# Palkkien paat (alkupiste, loppupiste)
B = [1 2; 1 3; 2 3; 2 4; 3 4; 3 5;
     4 5; 4 6; 5 6; 5 7; 6 7]';
# palkkien lukumaara on npa
[n, npa] = size(B);
# palkkien suuntavektorit
w = zeros(c, npa);
for i=1:npa
    v = P(:, B(2, i)) - P(:, B(1, i));
    w(:, i) = v / norm(v);
endfor
# -----
# Kokonaistaakka
G = 15700;
# Taakat
taakka = [0 -0.05*G; 0 -0.10*G; 0 0;
          0 -0.35*G; 0 0; 0 -0.40*G; 0 -0.10*G]';
# Tukivoimat
tuki = [0 0.385*G; 0 0; 0 0; 0 0; 0 0; 0 0; 0 0.615*G]';
# -----
```

```

# Rakennetaan ratkaistavan yhtaloryhman kerroinmatriisi
A = zeros(c*npi, npa);
for p=1:npa
    a = B(1,p);
    b = B(2,p);
    A((a*c-c+1):(a*c),p) = w(:,p);
    A((b*c-c+1):(b*c),p) = -w(:,p);
endfor
#-----
# rakennetaan yhtaloryhman RHS
b = zeros(2*npi,1);
for i=1:npi
    b(2*i-1) = taakka(1,i) + tuki(1,i);
    b(2*i)    = taakka(2,i) + tuki(2,i);
endfor
#-----
# ratkaistaan yhtaloryhma
F = A\b

```

ohjelma tulostaa konsolille voimat (Newtonina)

```

F =

-1.1761e+04
 1.0519e+04
 1.9428e+02
-1.1233e+04
-1.9428e+02
 1.0794e+04
 3.8023e+03
-1.4212e+04
-3.8023e+03
 1.6171e+04
-1.8080e+04

```

Siis

$$\begin{aligned}
 F_1 &= -11\,800\text{ N}, & (\text{puristus}), \\
 F_2 &= 10\,500\text{ N}, & (\text{venytys}), \\
 F_3 &= 200\text{ N}, & (\text{venytys}), \\
 F_4 &= -11\,200\text{ N}, & (\text{puristus}), \\
 F_5 &= -200\text{ N}, & (\text{puristus}), \\
 F_6 &= 10\,800\text{ N}, & (\text{venytys}), \\
 F_7 &= 3\,800\text{ N}, & (\text{venytys}), \\
 F_8 &= -14\,200\text{ N}, & (\text{puristus}), \\
 F_9 &= -3\,800\text{ N}, & (\text{puristus}), \\
 F_{10} &= 16\,200\text{ N}, & (\text{venytys}), \\
 F_{11} &= -18\,100\text{ N}, & (\text{puristus}).
 \end{aligned}$$

□

### 3 LINEAARISET YHTÄLÖRYHMÄT

Ch:YhtRyh

#### 3.1 Ratkaisun olemassaolo ja yksikäsitteisyys

Sec:YRrlkm

Edellä olleet esimerkit osoittavat, että yhtälöryhmällä ei aina ole ratkaisua ja eräissä tapauksissa ratkaisuja on ääretön määrä.

Vaikka Gaussin algoritmi ratkaisee kaikki tapaukset, niin on tilanteita, joissa rivioperaatioiden käyttö on hankalaa.

esim1\_c2s4

**Esimerkki 8 Kansantulo ja verotus.** Kuvataan kansantulon yhteyttä kulutukseen ja investointeihin muutamalla yhtälöllä. Yhtälöissä esiintyvät kirjaimet ovat seuraavat:

- $x_n$  = kulutus vuonna  $n$ ,
- $y_n$  = kansantulo vuonna  $n$ ,
- $z_n$  = investoinnit vuonna  $n$ ,
- $I_0$  = keskimääräinen investointitaso (oletetaan vakioksi),
- $s$  = säästämisaste ( $0 \leq s \leq 1$ , oletetaan vakioksi),
- $t$  = veroaste ( $0 \leq t \leq 1$ , oletetaan vakioksi),
- $k$  = investointien herkkyys kulutuksen muutokselle, (oletetaan vakioksi)

Yhtälöt ovat

$$\begin{cases} y_n = x_n + z_n \\ x_n = (1-s)(1-t)y_{n-1} \\ z_n = I_0 + ty_{n-1} + k(x_n - x_{n-1}) \end{cases}$$

Jos taloudessa ei esiinny vakavia häiriöitä, niin vuotuiset arvot alkavat lähestyä tasapainoarvoja;  $x_n, x_{n-1} \rightarrow x$ ,  $y_n, y_{n-1} \rightarrow y$  ja  $z_n, z_{n-1} \rightarrow z$ . Tasapainoa siis kuvaa yhtälöt

$$\begin{cases} y = x + z \\ x = (1-s)(1-t)y \\ z = I_0 + ty \end{cases} \Leftrightarrow \begin{cases} -x + y - z = 0 \\ x - \alpha\beta y = 0 \\ -ty + z = I_0 \end{cases}, \quad \begin{pmatrix} \text{missä} \\ (1-s) = \alpha, \\ (1-t) = \beta \end{pmatrix}$$

Heti, kun kertoimien  $s$  ja  $t$  numeroarvot tiedetään, on yhtälöryhmän ratkaiseminen rivioperaatioiden avulla helppo rutiinitehtävä.

Yhtälöryhmän ratkaiseminen tässä yleisessä muodossa, ilman tietoa  $s$ :n ja  $t$ :n numeroarvoista, on edelleen mahdollista tehdä rivioperaatioiden avulla, mutta silloin tulee toimia hyvin tarkasti. Myöhemmin opimme Cramerin kaavat, joiden avulla voidaan tätä tyyppiä oleva yhtälöryhmä ratkaista suoraviivaisesti ja ilman vaikeuksia. Palaamme esimerkkiin uudelleen Cramerin kaavojen yhteydessä.

**Esimerkki 9 Markkinaosuus tasapainossa.** Kauppiaalla on alueellaan 1000 taloutta, joista  $x_t$  taloutta voidaan hetkellä  $t$  luokitella asiakastalouksiksi ja loput  $y_t = 1000 - x_t$  eivät ole asiakastalouksia. Markkinointi hoidetaan lehtimainoksilla ja postin mukana jaettavilla tarjousilmoituksilla. Viikon mainosten jälkeen tapahtuu asiakaskunnassa muutoksia seuraavin todennäköisyyksin:

| alkutila   | lopputila  | todennäköisyys |
|------------|------------|----------------|
| asiakas    | asiakas    | $p_{11}$       |
| asiakas    | ei-asiakas | $p_{12}$       |
| ei-asiakas | asiakas    | $p_{21}$       |
| ei-asiakas | ei-asiakas | $p_{22}$       |

Jos tilanne jatkuu samanlaisena riittävän kauan, saavutetaan tasapaino ( $x_t \rightarrow x$  ja  $y_t \rightarrow y$ ), jota kuvaa yhtälöryhmä

$$\begin{cases} p_{11}x + p_{12}y = x \\ p_{21}x + p_{22}y = y \\ x + y = 1000 \end{cases} \Leftrightarrow \begin{cases} (p_{11} - 1)x + p_{12}y = 0 \\ p_{21}x + (p_{22} - 1)y = 0 \\ x + y = 1000 \end{cases}.$$

Yhtälöryhmässä on kaksi muuttujaa ja kolme yhtälöä, jolloin Cramerin kaavat eivät toimi. Jokainen yhtälö sisältää tärkeitä tietoa, eikä mitään niistä voi poistaa. Myöhemmin opimme muodostamaan yhtälöryhmään liittyvän ”pseudoinverssin”, jonka avulla ratkaisu saadaan taas helposti.

Edellä esitetyt esimerkit voi taitava laskija ratkaista ilman erikoista teorian kehittelyä, käsittelemällä yhtälöitä luovalla tavalla. Kun vastaavan kaltaisissa yhtälöryhmissä on kymmeniä muuttujia, niin taitavuus ei enää yksin riitä, vaan tarvitaan systemaattisia menetelmiä, jotka voidaan kuvata yleisesti.

Jatkossa palaamme yhtälöryhmiin kolme kertaa:

(1) Tapaus: ”**Muuttujia on yhtä monta kuin yhtälöitä.**” Kun olemme luvussa 3 esitelleet matriisit ja determinantit, on helppo antaa yksinkertainen ehto ratkaisun yksikäsitteisyydelle. Esittelemme kaksi uutta yhtälöryhmän ratkaisumenetelmää. Erityisen tarkkaan käsittelemme homogeenisen yhtälöryhmän käsittelyn. (Homogeenisessa yhtälöryhmässä jokaisen yhtälön RHS on nolla.)

(2) Tapaus: ”**Yhtälöitä on vähemmän kuin muuttujia.**” Tapaus käsitellään luvussa 4.4 sivut 49-.

(3) Tapaus: ”**Yhtälöitä on enemmän kuin muuttujia.**” Luvussa 8 käsitellään lineaarikuvauksia ja matriisihajoitelmia. Uusilla käsitteillä saadaan kuvattua välttämätön ehto ratkaisun yksikäsitteisyydelle, sekä tehokas ratkaisumenetelmä.

tarkista luvun numero, tai korvaa viittauksella lukulabeliin

Todistuksen merkinnät voivat tuntua kummallisilta, joten otamme konkreettisen esimerkin

**Esimerkki 11** Yhtälöryhmällä

$$\begin{cases} x + 2y + 3z = 4 \\ -x + 3y + 2z = 1 \\ 2x - y + z = 3 \end{cases} \quad (3.6)$$

on ainakin ratkaisut

$$\begin{cases} x = 0 \\ y = -1 \\ z = 2 \end{cases} \quad \text{ja} \quad \begin{cases} \hat{x} = 1 \\ \hat{y} = 0 \\ \hat{z} = 1 \end{cases}$$

Lauseen todistuksen merkinnöillä tästä seuraa, että

$$\begin{aligned} 0 \cdot \vec{a}_{\bullet 1} + (-1) \cdot \vec{a}_{\bullet 2} + 2 \cdot \vec{a}_{\bullet 3} &= 1 \cdot \vec{a}_{\bullet 1} + 0 \cdot \vec{a}_{\bullet 2} + 1 \cdot \vec{a}_{\bullet 3} \\ \Leftrightarrow -\vec{a}_{\bullet 1} - \vec{a}_{\bullet 2} + \vec{a}_{\bullet 3} &= \vec{0} \\ \Leftrightarrow \vec{a}_{\bullet 3} &= \vec{a}_{\bullet 1} + \vec{a}_{\bullet 2} \end{aligned}$$

Siis kolmas kerroin sarake on kahden ensimmäisen kerroinsarakkeen summa. Kun tarkistaa asian alkuperäisestä yhtälöryhmästä, huomaa että näin todella on asian laita.

**Lause 12** Olkoon  $j$ :s sarake  $a_{\bullet j}$  kerroinkaaviossa <sup>kk</sup>(3.2) lineaarikombinaatio muista sarakkeista. Kun kaaviolle tehdään mikä tahansa Gaussin algoritmista käytetty rivio-peraatio, niin muuttuneen kaavion  $j$ :s sarake on edelleen lineaarikombinaatio samoista sarakkeista. (Jopa samoin kertoimin.)

lause20

**Todistus:** Harjoitustehtävä. □

**Lause 13** Jos  $m = n$ , niin kerroinkaaviossa <sup>kk</sup>(3.2) voidaan jokin sarake lausua muiden sarakkeiden lineaarikombinaationa, jos ja vain jos jokin kerroinkaavion rivi voidaan lausua muiden rivien lineaarikombinaationa.

lause21

**Todistus:**  $\Rightarrow$  Keskitymme ensin osoittamaan ”vain jos”-osan, joka sanoo, että ”Jos jokin sarake voidaan lausua muiden sarakkeiden lineaarikombinaationa, niin jokin rivi voidaan lausua muiden rivien lineaarikombinaationa”.

Oletamme, että sarake  $a_{\bullet k}$  on sarake, joka voidaan lausua sitä edeltäneiden sarakkeiden  $\{a_{\bullet 1}, \dots, a_{\bullet (k-1)}\}$  lineaarikombinaationa. (HT: miksi oletus voidaan tehdä?) Ryhdymme tekemään Gaussin algoritmin mukaisia riviooperaatioita.

(a) Jos kierrosten  $1 \dots (k-1)$  aikana pivotointi epäonnistuu jossakin sarakkeessa, niin viimeisellä kierroksella pivot-alkio ei ole paikassa  $kk$ , vaan sen yläpuolella. Siis viimeisessä Gaussin algoritmin mukaisessa kaaviossa alin rivi on nolla-rivi. Tämä nollarivi

on lineaarikombinaatio alkuperäisen kerroinkaavion riveistä, joten lauseen <sup>lause12</sup>(12) mukaan jokin rivi voidaan lausua muiden rivien lineaarikombinaationa.

(b) Jos kierrosten  $1 \dots (k-1)$  aikana pivotointi on onnistunut joka kierroksella, niin nyt olemme pivotoimassa pivot-alkiona alkio paikassa  $kk$ . Koska  $k$ :s sarake on edelleen lauseen <sup>lause20</sup>(12) mukaan lineaarikombinaatio edellisistä sarakkeista, ja edellisissä sarakkeissa on kaikissa riveillä  $k \dots n$  vain nollia, niin myös  $k$ :nnessa sarakkeessa on riveillä  $k \dots n$  pelkkiä nollia. Siis pivotointi epäonnistuu ja tästä seuraa samoin kuin edellä, että jokin alkuperäisen kerroinkaavion rivi on muiden lineaarikombinaatio.

⇐ Seuraavaksi osoitamme ”jos”-osan, joka sanoo, että ”Jos jokin rivi voidaan lausua muiden rivien lineaarikombinaationa, niin jokin sarake voidaan lausua muiden sarakkeiden lineaarikombinaationa.” Oletamme, että kerroinkaavion  $k$ :s rivi on lineaarikombinaatio edeltävistä riveistä. Käännetään kerroinkaavio niin, että  $j$ :nnessa rivistä tehdään uuden kaavion  $j$ :s sarake. ”jos vain”-kohdan mukaan käännetyn kaavion jokin rivi on lineaarikombinaatio muista käännetyn kaavion riveistä. Siis alkuperäisen kaavion jokin sarake on lineaarikombinaatio muista sarakkeista. □

**Lause 14** Jos  $m = n$ , niin yhtälöryhmällä <sup>ryhymä</sup>(3.1) on yksi yksikäsitteinen ratkaisu, jos ja vain jos mitään kerroinkaavion saraketta ei voida lausua muiden sarakkeiden lineaarikombinaationa.

**Todistus:** ⇒ Oletamme, että yhtälöryhmällä <sup>ryhymä</sup>(3.1) on yksi yksikäsitteinen ratkaisu. Osoitamme ensin, että mitään yhtälöryhmän yhtälöä ei saada rivioperaatioilla muotoon, jossa LHS olisi nolla. Jos jokin yhtälö saadaan rivioperaatioilla muotoon  $0 = b, b \neq 0$ , niin yhtälöryhmän ratkaisujoukko on tyhjä, mikä on vastoin oletusta. Jos jokin yhtälö saadaan rivioperaatioilla muotoon  $0 = 0$ , niin kyseinen yhtälö voidaan poistaa yhtälöryhmästä ratkaisun muuttumatta. Poiston jälkeen yhtälöryhmässä on muuttujia enemmän kuin yhtälöitä, joten Gaussin algoritmi johtaa ratkaisuun, jossa on monta ratkaisua tai tyhjä ratkaisujoukko. Tämä on taas vastoin oletusta. Ainoa vaihtoehto joka jää jäljelle on se, että oletuksen ollessa totta, ei mitään kerroinkaavion riviä voida rivioperaatioiden avulla muuttaa nollariviksi. Siis mitään riviä ei voida lausua muiden rivien lineaarikombinaationa ja lauseen <sup>lause21</sup>(13) mukaa ei siis mitään sarakettakaan voida lausua muiden sarakkeiden lineaarikombinaationa.

⇐ Oletamme, että mitään kerroinkaavion saraketta ei voida lausua muiden sarakkeiden lineaarikombinaationa. Silloin lauseen <sup>lause21</sup>(13) mukaan ei Gaussin algoritmin kulessa kerroinkaavion LHS koskaan mene nollavektoriksi. Jos pivot-kertoimen paikalla on nolla, niin nollan oikealla puolella, samalla rivillä, on siis nollasta eroava kerroin. Vaihtamalla nämä sarakkeet keskenään saadaan uusi kaavio, jossa muuttujien järjestys on muuttunut, mutta ratkaisu on sama alkuperäisen kaavion kanssa. Pivotointi onnistuu siis aina ja algoritmi antaa yksikäsitteisen ratkaisun. □

## 3.2 Pyöristysvirheistä

Sec:YRRoundErr

Tämän kurssin esimerkit ovat usein epärealistisia, koska esimerkkien kertoimet ovat pieniä kokonaislukuja, joilla on helppo laskea tarkasti. Seuraavissa kahdessa esimerkissä ratkaistaan yhtälöryhmä, jonka kertoimet tunnetaan neljän merkitsevän numeron tarkkuudella. Seuraavassa merkitä  $a = a_0 \pm \varepsilon$  tarkoittaa, että  $a_0 - \varepsilon \leq a \leq a_0 + \varepsilon$ . Tiedämme siis välin, johon  $a$ :n arvo varmuudella kuuluu.

Jos  $a = a_0 \pm \varepsilon$ ,  $b = b_0 \pm \mu$  ja  $c = c_0 \pm \nu$ , niin

$$\begin{aligned} a + b &= (a_0 + b_0) \pm (\varepsilon + \mu) \\ a - b &= (a_0 - b_0) \pm (\varepsilon + \mu) \\ a \cdot b &= (a_0 \cdot b_0) \pm (|b_0|\varepsilon + |a_0|\mu) \\ \frac{1}{b} &= \frac{1}{b_0 - \frac{\mu^2}{b_0}} \pm \frac{\varepsilon}{|b_0^2 - \mu^2|} \approx \frac{1}{b_0} \pm \frac{\mu}{b_0^2} \\ \frac{a}{b} &\approx \frac{a_0}{b_0} \pm \left( \frac{|b_0|\varepsilon + |a_0|\mu}{b_0^2} \right) \\ \frac{a}{b} \cdot c &= \frac{a_0}{b_0} \cdot c_0 \pm \left( \frac{|a_0 b_0|\nu + |b_0 c_0|\varepsilon + |c_0 a_0|\mu}{b_0^2} \right) \end{aligned}$$

**Esimerkki 15** Tarkastellaan yhtälöryhmää

$$\begin{cases} (2.135 \pm \varepsilon)x - (7.095 \pm \varepsilon)y - (2.346 \pm \varepsilon)z = -5.000 \pm \varepsilon \\ -(1.237 \pm \varepsilon)x + (4.108 \pm \varepsilon)y - (2.315 \pm \varepsilon)z = 14.000 \pm \varepsilon \\ (2.351 \pm \varepsilon)x - (1.306 \pm \varepsilon)y + (3.562 \pm \varepsilon)z = -11.000 \pm \varepsilon \end{cases}$$

Kertoimet ovat anturiarvoja, joiden toleranssi on  $\varepsilon = 0.0005$ . Ratkaistaan yhtälöryhmä ensin *Gauss-Jordan algoritmin* versiolla I (sivu <sup>Alg:GaussJordanI</sup>??) ja toiseksi algoritmin versiolla III (sivu <sup>Alg:GaussJordanIII</sup>??). Laskemme jokaisen laskutoimituksen kolmen desimaalin tarkkuudella (toleranssirajoja arvioimme neljän desimaalin tarkkuudella).

Es:yr1

$$\left( \begin{array}{ccc|c} [2.135 \pm \varepsilon] & (-7.095 \pm \varepsilon) & (-2.346 \pm \varepsilon) & (-5.000 \pm \varepsilon) \\ (-1.237 \pm \varepsilon) & (4.108 \pm \varepsilon) & (-2.315 \pm \varepsilon) & (14.000 \pm \varepsilon) \\ (2.351 \pm \varepsilon) & (-1.306 \pm \varepsilon) & (3.562 \pm \varepsilon) & (-11.000 \pm \varepsilon) \end{array} \right)$$

*Ratkaisu algoritmin versiolla I:* Ensimmäinen pivot-termi on ensimmäisen yhtälön ensimmäinen termi. Pivot-rivi kerrotaan luvulla

$$a = (1.237 \pm 0.0005) / (2.135 \pm 0.0005) = (0.579 \pm 0.0004)$$



ja lisätään riviin 2. Toiseksi pivot-rivi kerrotaan luvulla

$$b = (2.351 \pm 0.0005)/(2.135 \pm 0.0005) = (1.101 \pm 0.0005)$$

ja vähennetään rivistä 3.

$$\sim \left( \begin{array}{ccc|c} [2.135 \pm \varepsilon] & (-7.095 \pm \varepsilon) & (-2.346 \pm \varepsilon) & (-5.000 \pm \varepsilon) \\ (-1.237 \pm \varepsilon) & (4.108 \pm \varepsilon) & (-2.315 \pm \varepsilon) & (14.000 \pm \varepsilon) \\ (2.351 \pm \varepsilon) & (-1.306 \pm \varepsilon) & (3.562 \pm \varepsilon) & (-11.000 \pm \varepsilon) \end{array} \right) \begin{array}{l} *a \\ \leftarrow + \\ \leftarrow - \end{array}$$

$$\sim \left( \begin{array}{ccc|c} (2.135 \pm \varepsilon) & (-7.045 \pm 0.0005) & (-2.346 \pm 0.0005) & (-5.000 \pm 0.0005) \\ 0 & (0.029 \pm 0.0036) & (-3.673 \pm 0.0017) & (11.105 \pm 0.0028) \\ 0 & (6.451 \pm 0.0046) & (6.145 \pm 0.0022) & (-5.495 \pm 0.0036) \end{array} \right)$$

Seuraava pivot-alkio on rivillä 2 sarakkeessa 2. Kerrotaan pivot-rivi luvulla

$$c = (6.451 \pm 0.0046)/(0.029 \pm 0.0036) = (222.4 \pm 27.77)$$

ja vähennetään tulo rivistä 3:

$$\left( \begin{array}{ccc|c} (2.135 \pm \varepsilon) & (-7.045 \pm 0.0005) & (-2.346 \pm 0.0005) & (-5.000 \pm 0.0005) \\ 0 & (0.029 \pm 0.0036) & (-3.673 \pm 0.0017) & (11.105 \pm 0.0028) \\ 0 & 0 & (823.2 \pm 102.39) & (-2476 \pm 309.04) \end{array} \right)$$

Palautuvien sijoitusten avulla muuttujille saadaan arvot:

$$\begin{cases} x = 0.840 \pm 315.8 \\ y = 1.966 \pm 95.44 \\ z = -3.008 \pm 0.750 \end{cases}$$

Toleranssien perusteella tulos näyttää arvottomalta, mutta virherajat kertovat ”pahimman mahdollisen” virheen (jos kaikki pyöristyksen vaikuttavat samaan suuntaan). Käytännössä osa pyöristyksistä muuttavat tulosta ylöspäin ja osa alaspäin. Siten lopputulos on tavallisesti lähempänä oikeata kuin virherajojen perusteella odottaisi. Silti joskus todella tapahtuu se, että tulos on juuri niin huono kuin virherajojen mukaan on mahdollista.

*Ratkaisu algoritmin versiolla III:*

Algoritmin suoritus alkaa sillä, että sarakkeen 1 itseisarvoltaan suurin alkio vaihdetaan riville 1. Siis rivit 1 ja 3 vaihtavat paikkaa. Tämän jälkeen ensimmäinen pivot-termi on ensimmäisen yhtälön ensimmäinen termi. Pivot-rivi kerrotaan luvulla

$$a = (1.237 \pm 0.0005)/(2.351 \pm 0.0005) = (0.526 \pm 0.0003)$$

ja lisätään riviin 2. Toiseksi pivot-rivi kerrotaan luvulla

$$b = (2.135 \pm 0.0005)/(2.351 \pm 0.0005) = (0.908 \pm 0.0004)$$

ja vähennetään rivistä 3.

$$\begin{aligned}
 & \left( \begin{array}{ccc|c} (2.135 \pm \varepsilon) & (-7.095 \pm \varepsilon) & (-2.346 \pm \varepsilon) & (-5.000 \pm \varepsilon) \\ (-1.237 \pm \varepsilon) & (4.108 \pm \varepsilon) & (-2.315 \pm \varepsilon) & (14.000 \pm \varepsilon) \\ (2.351 \pm \varepsilon) & (-1.306 \pm \varepsilon) & (3.562 \pm \varepsilon) & (-11.000 \pm \varepsilon) \end{array} \right) \begin{array}{l} \text{vaihda} \\ \\ \text{vaihda} \end{array} \\
 \sim & \left( \begin{array}{ccc|c} (2.351 \pm \varepsilon) & (-1.306 \pm \varepsilon) & (3.562 \pm \varepsilon) & (-11.000 \pm \varepsilon) \\ (-1.237 \pm \varepsilon) & (4.108 \pm \varepsilon) & (-2.315 \pm \varepsilon) & (14.000 \pm \varepsilon) \\ (2.135 \pm \varepsilon) & (-7.095 \pm \varepsilon) & (-2.346 \pm \varepsilon) & (-5.000 \pm \varepsilon) \end{array} \right) \begin{array}{l} *a \\ \leftarrow + \\ \leftarrow - \end{array} \\
 \sim & \left( \begin{array}{ccc|c} (2.351 \pm \varepsilon) & (-1.306 \pm 0.0005) & (3.562 \pm 0.0005) & (-11.000 \pm 0.0005) \\ 0 & (3.421 \pm 0.0012) & (-0.441 \pm 0.0018) & (8.214 \pm 0.0041) \\ 0 & (-5.859 \pm 0.0015) & (-5.580 \pm 0.0024) & (4.988 \pm 0.0054) \end{array} \right)
 \end{aligned}$$

Seuraavaksi vaihdamme rivit 2 ja 3. Sen jälkeen pivot-alkio on rivillä 2 sarakkeessa 2. Pivot-rivi kerrotaan luvulla

$$c = (3.421 \pm 0.0012) / (5.859 \pm 0.0015) = (0.584 \pm 0.0004)$$

ja tulo lisätään riviin 3

$$\begin{aligned}
 \sim & \left( \begin{array}{ccc|c} (2.351 \pm \varepsilon) & (-1.306 \pm 0.0005) & (3.562 \pm 0.0005) & (-11.000 \pm 0.0005) \\ 0 & (-5.859 \pm 0.0015) & (-5.580 \pm 0.0024) & (4.988 \pm 0.0054) \\ 0 & (3.421 \pm 0.0012) & (-0.441 \pm 0.0018) & (8.214 \pm 0.0041) \end{array} \right) \begin{array}{l} \\ *c \\ \leftarrow + \end{array} \\
 \sim & \left( \begin{array}{ccc|c} (2.351 \pm \varepsilon) & (-1.306 \pm 0.0005) & (3.562 \pm 0.0005) & (-11.000 \pm 0.0005) \\ 0 & (-5.859 \pm 0.0015) & (-5.580 \pm 0.0024) & (4.988 \pm 0.0054) \\ 0 & 0 & (-3.700 \pm 0.0054) & (11.127 \pm 0.0092) \end{array} \right)
 \end{aligned}$$

Palautuvien sijoitusten avulla muuttujille saadaan arvot:

$$\begin{cases} x = 0.995 \pm 0.017 \\ y = 2.012 \pm 0.009 \\ z = -3.007 \pm 0.007 \end{cases}$$

Jos laskemme ilman pyöristyksiä tehtävässä annetuilla arvoilla, ratkaisuksi tulee  $x = 0.99610$ ,  $y = 2.01310$  ja  $z = -3.00750$ . Kahdesta algoritmiversiosta jälkimmäinen (versio III) toimi selvästi paremmin.

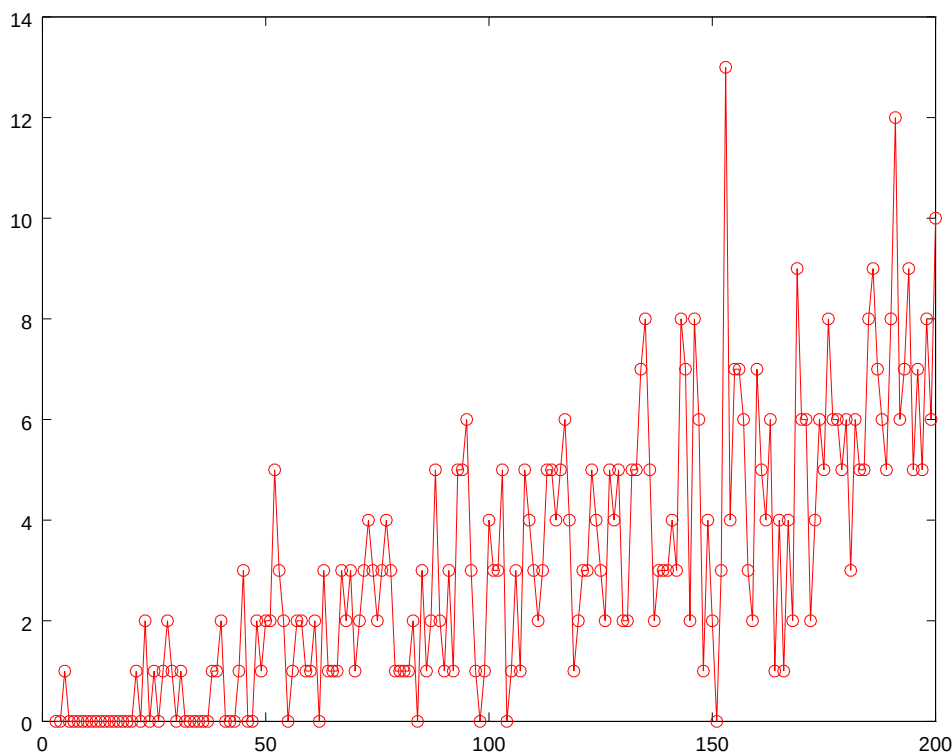
Esimerkit ovat vähän keinotekoisia siinä mielessä, että laskimme kolmen desimaalin tarkkuudella. Tietokoneet laskevat yleensä huomattavasti tarkemmin. Normaalisti Laskeessa yksinkertaisen tarkkuuden luvuilla Octavessa pyöristysvirhe on suuruusluokkaa  $\varepsilon = 10^{-8}$ . Tarkkuus on siis huomattavasti parempi kuin edellisessä esimerkissä.

Ei ole lainkaan helppoa mallintaa yhtälöryhmän ratkaisun aikana syntyvien pienten pyöristysvirheiden syntymistä ja kumuloitumista. Pyöristysvirheiden vaikutus tulokseen riippu paljon yhtälöryhmän erikoispiirteistä. Eri yhtälöryhmien herkkyyttä virheiden kasvulle käsitellään myöhemmin lisää luvussa ().

[lisää linkki](#)

Kuva 7 on piirretty seuraavasti: olkoon  $n$  yhtälöiden lukumäärä yhtälöryhmässä. Jokaisella  $n$ :n arvolla  $n = 3 \dots 200$  on arvottu tuhat yhtälöryhmää. Jokainen arvottu yhtälöryhmä on ratkaistu käyttäen yksinkertaista laskentatarkkuutta ja ratkaistu vielä uudelleen käyttäen kaksinkertaista tarkkuutta. Jos kahden ratkaisun suhteellinen ero on

enemmän kuin  $10^{-3}$ , niin yksinkertaisen tarkkuuden ratkaisussa on pyöristyksestä johtuvaa virhettä, joka näkyy vastauksessa kolmannen desimaalin vääränä arvona. Kuvaan on piirretty tieto siitä, miten monta näin suurta virhettä havaittiin tuhannessa toistossa. Kuvasta voimme vetää sen johtopäätöksen, että siinä yhtälöryhmien luokassa, josta



**Kuva 7.** Virheellisten ratkaisujen määrä tuhannessa toistossa. Vaaka-akseli kertoo yhtälöiden lukumäärän yhtälöryhmässä.

Fig:054

arvonta suoritettiin, pyöristysvirhe vaikuttaa ratkaisuun melko harvoin, mutta vaikutus on mahdollinen jo aivan pienilläkin yhtälöryhmillä. Nyt emme ryhdy tutkimaan asiaa enempää. Palaamme asiaan, kun olemme oppineet lisää vektori- ja matriisi-normeista.

Lyhyesti vertaamme vielä <sup>Es:yr1</sup>esimerkin 15 kahta ratkaisua. Ratkaisut eroavat toisistaan, mutta varsinkin virhearviot eroavat paljon. Tämä selittyy seuraavasti: algoritmiversiossa I toisen yhtälön  $y:n$  kertoimen suhteellinen tarkkuus alussa on

$$stol_1 = \frac{0.0005}{4.108} \approx 0.00012.$$

Ensimmäisen pivotoinnin jälkeen samassa paikassa olevan termin suhteellinen tarkkuus on

$$stol_2 = \frac{0.0036}{0.029} \approx 0.12.$$

Suhteellinen virhemahdollisuus kasvoi siis tuhat-kertaiseksi yhdellä pivointikierröksellä. Algoritmi valitsee tämän kertoimen pivot-alkioksi, jolloin sen romahtanut tarkkuus vaikuttaa tasan kaikkiin laskuihin tästä eteenpäin! Tällaista ei saisi tapahtua laskennan aikana.

Algoritmiversio I ei ole ammattimaista tasoa (professionaali)! Edellä kuvattu suhteellisen tarkkuuden romahdus tapahtuu yleensä siten, että lasketaan kahden melkein saman kokoisen luvun erotus. Näin saattaa tapahtua rivioperaatiossa. Tällaista alkia ei saa valita pivot-alkioksi. Algoritmiversio III välttää sen valitsemalla pivot-alkioksi itseisarvoltaan mahdollisimman suuren kertoimen.

## 4 MATRIISIT JA MATRIISILASKUT

Ch:Matrix

### 4.1 Kitka-esimerkki

Sec:EsKitkaI

Edellä opimme, että yhtälöryhmä voidaan kirjoittaa matriisi-yhtälöksi. Jos yhtälöryhmä on vapaa ja kvadraattinen, eli kerroinkaavio on säännöllinen neliömatriisi, niin yhtälöryhmän ratkaisu saadaan käänteismatriisin avulla. On tärkeitä huomata, ettei asia aina ole näin.

Yhtälöryhmien käsittelyä opeteltaessa syntyy helposti vääriä mielikuvia siksi, että ensimmäisten esimerkkien tulee aina olla helppolukuisia ja pieniä. Tekniikassa usein yhtälöryhmät ovat kolmen muuttujan ja kolmen yhtälön ryhmiä. Tämäkään ei aina ole totta.

**Esimerkki 16** Kolme massaa lepää vaakasuoralla alustalla ( $u, v$ -tasolla). Kappaleiden massat ja paikat ovat

$$\begin{aligned} m_1 &= 2.00 \text{ kg} & (u_1, v_1) &= (2.17 \text{ cm}, -5.42 \text{ cm}), \\ m_2 &= 1.00 \text{ kg} & (u_2, v_2) &= (-1.35 \text{ cm}, -9.12 \text{ cm}), \\ m_3 &= 4.00 \text{ kg} & (u_3, v_3) &= (5.15 \text{ cm}, 15.10 \text{ cm}). \end{aligned}$$

Kappaleiden päällä on kevyt levy, joka on kiinnitetty kappaleisiin niin, että kappaleet ja levy muodostavat yhden kiinteän systeemin. Levyn kohdistuu pisteeseen  $(0, 0)$  kymmenen Newtonin voima suuntaan  $\vec{i}$ . Laske kappaleiden ja alustan väliset kitkavoimat, kun oletamme että systeemi ei liiku. Pysykö systeemi paikallaan, jos kappaleiden ja alustan välinen lepokitkakerroin on  $\mu = 0.30$ ?

Ex:Kitka1

Sovimme ensin joistakin merkinnöistä. (Osa merkinnöistä on nyt tarpeettomia, mutta auttavat liittämään esimerkkisarjan osia toisiinsa.) Systeemiin vaikuttava ulkoinen voima on

$$\vec{F} = F\vec{i} = 10.000 \text{ N}\vec{i}.$$

Tukivoimat ovat

$$\begin{aligned} \vec{T}_1 &= T_1\vec{k} = 2.00 \text{ kg} \cdot 9.81 \frac{\text{m}}{\text{s}^2} \vec{k} = 19.62 \text{ N}\vec{k}, & T_1 &= 19.62 \text{ N}, \\ \vec{T}_2 &= T_2\vec{k} = 1.00 \text{ kg} \cdot 9.81 \frac{\text{m}}{\text{s}^2} \vec{k} = 9.81 \text{ N}\vec{k}, & T_2 &= 9.81 \text{ N}, \\ \vec{T}_3 &= T_3\vec{k} = 4.00 \text{ kg} \cdot 9.81 \frac{\text{m}}{\text{s}^2} \vec{k} = 39.24 \text{ N}\vec{k}, & T_3 &= 39.24 \text{ N}. \end{aligned}$$

Kitkavoimat ja niiden momentit origon suhteen ovat

$$\begin{aligned}\vec{K}_1 &= K_1 \vec{i}, & M_1 &= -v_1 K_1 = 5.42 \text{ cm } K_1, \\ \vec{K}_2 &= K_2 \vec{i}, & M_2 &= -v_2 K_2 = 9.12 \text{ cm } K_2, \\ \vec{K}_3 &= K_3 \vec{i}, & M_3 &= -v_3 K_3 = -15.10 \text{ cm } K_3,\end{aligned}$$

Merkitsemme vielä  $K_1 = x$ ,  $K_2 = y$ ,  $K_3 = z$ . Tämä on nyt hyvin ”epä-ammattilinen” merkintä, mutta se auttaa hahmottamaan syntyvän yhtälöryhmän rakennetta. Näillä merkinöillä tasapainon voima- ja momentti-ehto saavat muodot

$$\begin{aligned}& \begin{cases} K_1 + K_2 + K_3 = -F \\ v_1 K_1 + v_2 K_2 + v_3 K_3 = 0 \end{cases} \\ \Leftrightarrow & \begin{cases} x + y + z = -10.000 \text{ N} \\ 5.42x + 9.12y - 15.10z = 0 \end{cases}\end{aligned}$$

Kaksi yhtälöä eivät nyt riitä kolmen voiman ratkaisemiseen.

Jotta saisimme järkevät arvot kitkavoimille, mallinamme kappaleiden ja alustan välistä kitkaa nai’villa mallilla: Tarkastellaan ensin kappaletta 1. Kappale koskettaa alustaa useassa pisteessä. Kosketuspisteiden lukumäärä  $n_1$  on suoraan verrannollinen tukivoiman suuruuteen. Siis  $n_1 = \alpha T_1$ . Jokaisessa kosketuspisteessä syntyy pieni jousivoima  $f = ks$ , missä  $k$  on pienen jousen jousivakio ja  $s$  on mikroskooppinen siirtymä. Kappaleeseen vaikuttava kokonaiskitkavoima ja jousiin sitoutuva potentiaalienergia ovat

$$\begin{aligned}K_1 &= n_1 ks = \alpha T_1 ks, & \rightarrow s &= \frac{K_1}{\alpha T_1 k} \\ E_1 &= n_1 \frac{1}{2} ks^2 = \alpha T_1 \frac{1}{2} k \left( \frac{K_1}{\alpha T_1 k} \right)^2 = \frac{K_1^2}{2\alpha k T_1}\end{aligned}$$

Vakioita  $\alpha$  ja  $k$  emme nyt tiedä, mutta oletamme, että ne ovat kaikkien kolmen kappaleen kohdalla samat. Silloin jousisysteemien potentiaalienergia tulee minimoitua, kun etsimme mahdollisimman pienen arvon lausekkeelle

$$\frac{K_1^2}{2T_1} + \frac{K_2^2}{2T_2} + \frac{K_3^2}{2T_3}.$$

Ratkaisemme siis optimointitehtävän

$$\begin{cases} \text{Min} & \frac{x^2}{2T_1} + \frac{y^2}{2T_2} + \frac{z^2}{2T_3} \\ \text{ehdoin} & \begin{aligned} x + y + z &= -F \\ v_1 x + v_2 y + v_3 z &= 0 \end{aligned} \end{cases}.$$

Tällainen yhtälörajoitteita sisältävä sileä optimointitehtävä on tapana ratkaista Lagrangen menetelmällä. Emme nyt selitä tekniikkaa. Jos haluat kuvauksen siitä, niin etsi tietoa avainsanoilla ”optimointi” ja ”Lagrange” (”optimization”, ”Lagrangian”).

Lagrangen funktio on

$$\mathcal{L} = \frac{x^2}{2T_1} + \frac{y^2}{2T_2} + \frac{z^2}{2T_3} + \lambda_1(x + y + z + F) + \lambda_2(v_1x + v_2y + v_3z)$$

Välttämätön ehto optimille on  $\frac{\partial \mathcal{L}}{\partial x} = 0$ ,  $\frac{\partial \mathcal{L}}{\partial y} = 0$ ,  $\frac{\partial \mathcal{L}}{\partial z} = 0$ ,  $\frac{\partial \mathcal{L}}{\partial \lambda_1} = 0$ ,  $\frac{\partial \mathcal{L}}{\partial \lambda_2} = 0$  eli.

$$\begin{cases} x/T_1 & & & + \lambda_1 & + v_1\lambda_2 & = & 0 \\ & y/T_2 & & + \lambda_1 & + v_2\lambda_2 & = & 0 \\ & & z/T_3 & + \lambda_1 & + v_3\lambda_2 & = & 0 \\ x & + & y & + & z & = & -F \\ v_1x & + & v_2y & + & v_3z & = & 0 \end{cases}$$

$$\Leftrightarrow \begin{pmatrix} 1/T_1 & 0 & 0 & 1 & v_1 \\ 0 & 1/T_2 & 0 & 1 & v_2 \\ 0 & 0 & 1/T_3 & 1 & v_3 \\ 1 & 1 & 1 & 0 & 0 \\ v_1 & v_2 & v_3 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ \lambda_1 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -F \\ 0 \end{pmatrix}$$

Sijoitamme numerot paikalleen ja ratkaisemme yhtälöryhmän

```
octave:2> F = 10;
octave:3> T1 = 19.62;
octave:4> T2 = 9.81;
octave:5> T3 = 39.24;
octave:6> v1 = -5.42;
octave:7> v2 = -9.12;
octave:8> v3 = 15.10;
octave:9> A = [
> 1/T1 0 0 1 v1;
> 0 1/T2 0 1 v2;
> 0 0 1/T3 1 v3;
> 1 1 1 0 0;
> v1 v2 v3 0 0]
A =

    0.05097    0.00000    0.00000    1.00000   -5.42000
    0.00000    0.10194    0.00000    1.00000   -9.12000
    0.00000    0.00000    0.02548    1.00000   15.10000
    1.00000    1.00000    1.00000    0.00000    0.00000
```

```

-5.42000   -9.12000   15.10000   0.00000   0.00000
octave:10> b = [0; 0; 0 ; -F; 0 ]
b =
      0
      0
      0
     -10
      0

octave:11> xyz = inv(A)*b
xyz =
-4.4342336
-2.4776848
-3.0880816
 0.1870968
-0.0071788

```

Kitkavoimat ja niiden suhteet tukivoimiin ovat siis

$$\begin{aligned}
 K_1 &= -4.434\text{ N}, & |K_1/T_1| &= 0.226, \\
 K_2 &= -2.478\text{ N}, & |K_2/T_2| &= 0.253, \\
 K_3 &= -3.088\text{ N}, & |K_3/T_3| &= 0.079,
 \end{aligned}$$

Kitka riittää siis pitämään systeemin liikkumattomana. □

## 4.2 Panos-tuotos -analyysi

Sec:InOutAnal

Panos-tuotos-analyysi kehitettiin alunperin kansantaloutta silmällä pitäen. Silloin pyrittiin ennustamaan USA:n kansantalouden kehitystä. Ennusteet perustuivat edeltävinä vuosina USA:n kansantalouden rakenteesta kerättyihin tietoihin. Kävi ilmeiseksi, että vaikka kansantalouden tila muuttuisi nopeastikin, niin eräät sen rakennetta kuvaavat lukusuhteet muuttuvat silti hitaasti. Tämä antaa hyvän pohjan ennusteiden tekemiselle. Alkuperäisen kansantaloutta kuvaavan sovelluksen kehitti aikanaan amerikkalainen Wassily W. Leontief (1905-1999).

([http://en.wikipedia.org/wiki/Input-output\\_model](http://en.wikipedia.org/wiki/Input-output_model))

### 4.2.1 Kansantalouden rakenne

Panos-tuotos-menetelmä jakaa talouden toimialoihin ja tarkastelee sitä, miten eri toimialojen tuotokset ovat panoksia toisille toimialoille. Allaoleva panos-tuotos-matriisi



(input-output-matrix) esittää taloutta joka on jaettu kolmeen toimialaan T1, T2 ja T3. Toimialaa T1 vastaava rivi kertoo, että toimialan T1 tuotoksesta 240 yksikköä on välituotteita, jotka käytetään panoksena jatkotuotantoon. 180 yksikköä T1:n tuotoksesta myydään toimialan T2 yrityksille, jotka käyttävät sen panoksena omaan tuotantonsa. 144 yksikköä T1:n tuotoksesta myydään toimialan T3 yrityksille ja 36 yksikköä toimialan T1 tuotoksesta päätyy lopputuotteena kuluttajille. Siten toimialan T1 kokonaistuotos on  $240 + 180 + 144 + 36 = 600$ . Toimialan T2 kokonaistuotos on vastaavasti 360 ja toimialan T3 kokonaistuotos on 480.

|                      |    | käyttö/toimiala |     |     | loppu-<br>tuotteet |
|----------------------|----|-----------------|-----|-----|--------------------|
|                      |    | T1              | T2  | T3  |                    |
| Tuotanto<br>toimiala | T1 | 240             | 180 | 144 | 36                 |
|                      | T2 | 120             | 36  | 48  | 156                |
|                      | T3 | 120             | 72  | 48  | 240                |
| perus-<br>panokset   | p1 | 120             | 36  | 240 |                    |
|                      | p2 | 60              | 18  | 96  |                    |

Sarakkeista voimme lukea, mistä kukin toimiala hankkii panoksensa. Peruspanokset sisältävät työn, luonnonvarat ja tarkasteltavan talouden ulkopuolelta tulevat panokset.

Seuraavassa taulukossa jätämme lopputuotteet pois ja jaamme jokaisen sarakkeen luvut saraketta vastaavan toimialan kokonaistuotannolla. Saamme kerroin-matriisin (input-output coefficients) josta näemme, mitä panoksia eri toimialat tarvitsevat valmistaessaan yhden tuotosyksikön. Kerroin-matriisin yläosa ( $3 \times 3$ ) kuvaa eri toimialojen keskinäisiä riippuvuuksia. Muodostamme tästä talukon yläosasta matriisin **A**, jota sanomme ”toimialojen panos-tuotos -matriisiksi” tai teknologia-matriisiksi <sup>1</sup> (technology matrix). Taulukon alaosasta saamme ”tuotannontekijöiden panos-tuotos -matriisin” **B**, joka kertoo mitä tuotannontekijöitä tarvitaan kunkin tuotoksen tekemiseen.

|    | T1      | T2      | T3      |   |    | T1  | T2   | T3  |
|----|---------|---------|---------|---|----|-----|------|-----|
| T1 | 240/600 | 180/360 | 144/480 | → | T1 | 0.4 | 0.5  | 0.3 |
| T2 | 120/600 | 36/360  | 48/480  |   | T2 | 0.2 | 0.1  | 0.1 |
| T3 | 120/600 | 72/360  | 48/480  |   | T3 | 0.2 | 0.2  | 0.1 |
| p1 | 120/600 | 36/360  | 240/480 |   | p1 | 0.2 | 0.1  | 0.5 |
| p2 | 60/600  | 18/360  | 96/480  |   | p2 | 0.1 | 0.05 | 0.2 |

<sup>1</sup> Teknologia-matriisi (Technology matrix) on niin iskevä sana, että se on luultavasti keksitty ja otettu käyttöön useita kertoja hieman toisistaan poikkeavissa tarkoituksissa. Kannattaa siis olla tarkka, kun siteeraa eri tekstejä.

$$\mathbf{A} = \begin{pmatrix} 0.4 & 0.5 & 0.3 \\ 0.2 & 0.1 & 0.1 \\ 0.2 & 0.2 & 0.1 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0.2 & 0.1 & 0.5 \\ 0.1 & 0.05 & 0.2 \end{pmatrix}$$

Merkitään toimialojen kokonaistuotoksia muuttujilla  $x_1$ ,  $x_2$ , ja  $x_3$ . Ja otetaan kerroinmatriisien  $\mathbf{A}$  ja  $\mathbf{B}$  lisäksi käyttöön kokonaistuotos-vektori  $\vec{x}$  ja kysyntävektori  $\vec{d}$ , jotka saavat nyt arvot

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 600 \\ 360 \\ 480 \end{pmatrix}, \quad \vec{d} = \begin{pmatrix} 36 \\ 156 \\ 240 \end{pmatrix}$$

Kun toimiala T1 tyydyttää kaiken tuotteisiinsa kohdistuvan kysynnän, niin T1:n kokonaistuotos = T1:n tarvitsema panos + T2:n tarvitsema panos + T3:n tarvitsema panos + T1:n lopputuotteiden kysyntä. Vastaavat yhtälöt saadaan toimialojen T2 ja T3 kokonaistuotannolle. Siis

$$\begin{cases} x_1 = 0.4x_1 + 0.5x_2 + 0.3x_3 + 36 \\ x_2 = 0.2x_1 + 0.1x_2 + 0.1x_3 + 156 \\ x_3 = 0.2x_1 + 0.2x_2 + 0.1x_3 + 240 \end{cases} \Rightarrow \vec{x} = \mathbf{A}\vec{x} + \vec{d}.$$

Saadusta matriisiyhtälöstä saadaan yhteys kokonaistuotannon ja kysynnän välille seuraavasti

$$\begin{aligned} \vec{x} &= \mathbf{A}\vec{x} + \vec{d} \\ \Leftrightarrow \mathbf{I}\vec{x} - \mathbf{A}\vec{x} &= \vec{d} \\ \Leftrightarrow (\mathbf{I} - \mathbf{A})\vec{x} &= \vec{d} \end{aligned}$$

Matriisi  $(\mathbf{I} - \mathbf{A})$  on Leontief'n matriisi. Jos Leontief'n matriisi on säännöllinen, niin kokonaistuotanto saadaan kaavasta

$$\vec{x} = (\mathbf{I} - \mathbf{A})^{-1} \vec{d}$$

Panos-tuotos -analyysin ydin-ajatus tiivistyy seuraavaan

- Kysyntä  $\vec{d}$  voi muuttua nopeasti ja kokonaistuotanto  $\vec{x}$  muuttuu vastaavasti.
- Teknologiamatriisi  $\mathbf{A}$  ja vastaava Leontief'n matriisi  $(\mathbf{I} - \mathbf{A})$  muuttuvat hitaasti, joten ne voidaan estimoida edellisen kauden tiedoista.
- Kun tiedossa on perusteltu ennuste kysynnästä  $\vec{d}$  ja hyvä estimaatti Leontief'n matriisille  $(\mathbf{I} - \mathbf{A})$ , niin saamme hyvän ennusteen tuotannolle  $\vec{x}$ .

**Esimerkki 1.** Kappaleen alussa esitetyn talouden lopputuotteiden kysyntä oli  $\vec{d}_0 = (36 \ 156 \ 240)^T$ , Kokonaistuotanto oli  $\vec{x}_0 = (600 \ 360 \ 480)^T$  ja resurssitarve-vektori oli

$$\begin{aligned}\vec{r}_0 &= \mathbf{B}\vec{x}_0 \\ &= \begin{pmatrix} 0.2 & 0.1 & 0.5 \\ 0.1 & 0.05 & 0.2 \end{pmatrix} \begin{pmatrix} 600 \\ 360 \\ 480 \end{pmatrix} = \begin{pmatrix} 396 \\ 174 \end{pmatrix}.\end{aligned}$$

Jos loppukysyntä muuttuu siten, että uusi kysyntävektori on  $\vec{d}_1 = (77 \ 154 \ 231)$ , niin muuttuneet kokonaistuotanto- ja resurssitarve-vektorit ovat

$$\begin{aligned}\vec{x}_1 &= (\mathbf{I} - \mathbf{A})^{-1} \vec{d}_1 \\ &= \begin{pmatrix} 0.6 & -0.5 & -0.3 \\ -0.2 & 0.9 & -0.1 \\ -0.2 & -0.2 & 0.9 \end{pmatrix}^{-1} \begin{pmatrix} 77 \\ 154 \\ 231 \end{pmatrix} \\ &= \frac{1}{0.308} \begin{pmatrix} 0.79 & 0.51 & 0.32 \\ 0.20 & 0.48 & 0.12 \\ 0.22 & 0.22 & 0.44 \end{pmatrix} \begin{pmatrix} 77 \\ 154 \\ 231 \end{pmatrix} = \begin{pmatrix} 692.5 \\ 380 \\ 495 \end{pmatrix} \\ \vec{r}_1 &= \begin{pmatrix} 0.2 & 0.1 & 0.5 \\ 0.1 & 0.05 & 0.2 \end{pmatrix} \begin{pmatrix} 692.5 \\ 380 \\ 495 \end{pmatrix} = \begin{pmatrix} 424 \\ 187.25 \end{pmatrix}.\end{aligned}$$

Lopputuotteiden kysynnän muutos

$$\Delta \vec{d} = \vec{d}_1 - \vec{d}_0 = \begin{pmatrix} 77 \\ 154 \\ 231 \end{pmatrix} - \begin{pmatrix} 36 \\ 156 \\ 240 \end{pmatrix} = \begin{pmatrix} 41 \\ -2 \\ -9 \end{pmatrix}$$

saa aikaan kokonaistuotannossa ja resurssitarpeessa muutokset

$$\Delta \vec{x} = \vec{x}_1 - \vec{x}_0 = \begin{pmatrix} 692.5 \\ 380 \\ 495 \end{pmatrix} - \begin{pmatrix} 600 \\ 360 \\ 480 \end{pmatrix} = \begin{pmatrix} 92.5 \\ 20 \\ 15 \end{pmatrix}$$

$$\Delta \vec{r} = \vec{r}_1 - \vec{r}_0 = \begin{pmatrix} 424 \\ 187.25 \end{pmatrix} - \begin{pmatrix} 396 \\ 174 \end{pmatrix} = \begin{pmatrix} 28 \\ 13.25 \end{pmatrix}$$

Toimialan T1 kysyntä kasvoi, mutta toimialojen T2 ja T3 kysynät pienenevät. Silti kaikkien toimialojen kokonaistuotannot kasvoivat, koska toimiala T1 ”veti kahta muuta toimialaa perässään kasvuun”. Myös kummankin peruspanoksen kulutukset kasvoivat.

**Esimerkki 2.** Jatkamme edelleen edeltävän talouden analyysiä. Haluamme tietää min-kä verran lopputuotteet kuluttavat peruspanoksia.

$$\begin{aligned} \begin{pmatrix} r_1 \\ r_2 \end{pmatrix} = \vec{r} &= \mathbf{B}\vec{x} = \mathbf{B}(\mathbf{I} - \mathbf{A})^{-1}\vec{d} \\ &= \begin{pmatrix} 0.935 & 0.844 & 0.961 \\ 0.432 & 0.386 & 0.409 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix} \\ \Rightarrow \begin{cases} r_1 &= 0.935d_1 + 0.844d_2 + 0.961d_3 \\ r_2 &= 0.432d_1 + 0.386d_2 + 0.409d_3 \end{cases} \end{aligned}$$

Kertoimista näemme suoraan, että yksi T1:n tuotos tarvitsee 0.935 yksikköä perruspanos p1:tä ja 0.432 yksikköä peruspanos p2:ta. (Nämä ovat isompia lukuja kuin matriisin  $\mathbf{B}$  perusteella osaisi odottaa!)

#### 4.2.2 Omakustannusarvo

1950-luvun puolivälistä alkaen on panos-tuotos -mallia sovellettu yksittäisen yrityksen taloussuunnittelussa. Seuraavassa kuvataan juuri tällaista mikrotaloudellista sovellusta. Mallityyppi, jota käsitellään, on ns. Leontiefin avoin staattinen malli.

Mallin kohteena on yritys, joka jaetaan edelleen pienempiin alayksiköihin, joita kutsutaan seuraavassa osastoiksi. Se miten jako osastoihin suoritetaan, riippuu kulloisestakin yrityksestä. Seuraavassa oletamme osastojaon perustuvan siihen, että jokainen osasto tuottaa sille ominaista suoritetta. Suorite voi olla yrityksen myyntiohjelmaan kuuluva tuote, prosessissa edelleen jalostettava puolivalmiste tai prosessin ylläpitämiseen tarvittava energia tms. Kunkin osaston tuottama suorite voidaan käyttää kahdella eri tavalla:

- Se voidaan myydä yrityksen ulkopuolelle.
- Se voidaan käyttää muilla osastoilla tuotannontekijänä

**Esimerkki 3.** Tarkastellaan seuraavassa esimerkkiyritystä, jossa on neljä osastoa Os1, Os2, Os3 ja Os4, joista kukin tuottaa tiettyä suoritetta. Edellisen kauden tuotannosta ja sen jakautumisesta sekä käytetyistä tuotannontekijöistä on käytettävissä seuraavan taulukon tiedot.

Taulukon neljä ylintä riviä kertovat miten neljän osaston tuotantoa on käytetty oman ja toisten osastojen panoksina. Kuusi alinta riviä kertovat mitä muita tuotannontekijöitä osastot ovat käyttäneet panoksina.

|              |     | Os1  | Os2 | Os3  | Os4  | myynti | yhteensä |
|--------------|-----|------|-----|------|------|--------|----------|
|              | Os1 | 250  | 100 | 170  | 80   | 1400   | 2000     |
|              | Os2 | 1200 | 700 | 100  | 2600 | 400    | 5000     |
|              | Os3 | 7    | 1   | 3    | 2    | 12     | 25       |
|              | Os4 | 20   | 35  | 15   | 30   | 400    | 500      |
| raaka-aine 1 | tt1 | 200  | 0   | 80   | 120  |        | 400      |
| raaka-aine 2 | tt2 | 1400 | 800 | 2300 | 500  |        | 5000     |
| raaka-aine 3 | tt3 | 400  | 100 | 0    | 0    |        | 500      |
| energia      | tt4 | 50   | 30  | 80   | 40   |        | 200      |
| työvoima 1   | tt5 | 2700 | 900 | 100  | 2300 |        | 6000     |
| työvoima 2   | tt6 | 50   | 80  | 40   | 130  |        | 300      |

Laskemme ensin teknologia-matriisin  $\mathbf{A}$  ja panos-matriisin  $\mathbf{B}$ .

$$\mathbf{A} = \begin{pmatrix} 250/2000 & 100/5000 & 170/25 & 80/500 \\ 1200/2000 & 700/5000 & 100/25 & 2600/500 \\ 7/2000 & 1/5000 & 3/25 & 2/500 \\ 20/2000 & 35/5000 & 15/25 & 30/500 \end{pmatrix}$$

$$= \begin{pmatrix} 0.1250 & 0.0200 & 6.8000 & 0.1600 \\ 0.6000 & 0.1400 & 4.0000 & 5.2000 \\ 0.0035 & 0.0002 & 0.1200 & 0.0040 \\ 0.0100 & 0.0070 & 0.6000 & 0.0600 \end{pmatrix},$$

ja

$$\mathbf{B} = \begin{pmatrix} 200/2000 & 0/5000 & 80/25 & 120/500 \\ 1400/2000 & 800/5000 & 2300/25 & 500/500 \\ 400/2000 & 100/5000 & 0/25 & 0/500 \\ 50/2000 & 30/5000 & 80/25 & 40/500 \\ 2700/2000 & 900/5000 & 100/25 & 2300/500 \\ 50/2000 & 80/5000 & 40/25 & 130/500 \end{pmatrix}$$

$$= \begin{pmatrix} 0.1000 & 0.0000 & 3.2000 & 0.2400 \\ 0.7000 & 0.1600 & 92.0000 & 1.0000 \\ 0.2000 & 0.0200 & 0.0000 & 0.0000 \\ 0.0250 & 0.0060 & 3.2000 & 0.0800 \\ 1.3500 & 0.1800 & 4.0000 & 4.6000 \\ 0.0250 & 0.0160 & 1.6000 & 0.2600 \end{pmatrix}$$

Jos nyt kysyntä (myynti)  $\vec{d}$  muuttuu, niin näiden matriisien avulla voimme nyt laskea vastaavasti muuttuneen kokonais-tuotos -vektorin  $\vec{x} = (\mathbf{I} - \mathbf{A})^{-1} \vec{d}$  ja muuttuneen tuotantotekijöiden tarve -vektorin  $\vec{r} = \mathbf{B}(\mathbf{I} - \mathbf{A})^{-1} \vec{d}$ .

Yritys hinnoittelee eri osastojen tuotteet seuraavasti (indeksointi seuraa osastojen numerointia)

$$\vec{p}^T = (p_1 \ p_2 \ p_3 \ p_4)^T = (100.00 \ 10.00 \ 2000.00 \ 150.00)^T.$$

Tuotannontekijöiden yksikköhinnat ovat (indeksointi vastaa tuotannontekijöiden numerointia)

$$\vec{c}^T = (c_1 \ c_2 \ c_3 \ c_4 \ c_5 \ c_6)^T = (70.00 \ 10.00 \ 50.00 \ 8.00 \ 15.00 \ 25.00)^T.$$

Yrityksen kate saadaan selville, kun lasketaan tuotteiden euromääräiset myyntituotot ja tuotannontekijöiden ostot

|              |     | myynti(kpl) | yks.hinta | myynti(€) | yhteensä |
|--------------|-----|-------------|-----------|-----------|----------|
| tuote1       | Os1 | 1400        | 100       | 140 000   |          |
| tuote2       | Os2 | 400         | 10        | 4 000     |          |
| tuote3       | Os3 | 12          | 2000      | 24 000    |          |
| tuote4       | Os4 | 400         | 150       | 60 000    | 228 000  |
|              |     | ostot(kpl)  | yks.hinta | ostot(€)  | yhteensä |
| raaka-aine 1 | tt1 | 400         | 70        | 28 000    |          |
| raaka-aine 2 | tt2 | 5000        | 10        | 50 000    |          |
| raaka-aine 3 | tt3 | 500         | 50        | 25 000    |          |
| energia      | tt4 | 200         | 8         | 1 600     |          |
| työvoima 1   | tt5 | 6000        | 15        | 90 000    |          |
| työvoima 2   | tt6 | 300         | 25        | 7 500     | 202 100  |

Yrityksen myyntikate on

$$\frac{228\,000 - 202\,100}{228\,000} \cdot 100\% = 11.35\%$$

Seuraavana vuonna kysynnän rakenne muuttuu siten, että tuotteen 1 kysyntä pienenee ja tuotteen 4 kysyntä kasvaa niin, että uusi kysyntä-vektori on

$$\vec{d}_{uusi} = (500 \ 400 \ 12 \ 1300)^T.$$

Tuotannontekijöiden tarve -vektori on uudessa tilanteessa

$$\vec{r}_{uusi} = \mathbf{B}(\mathbf{I} - \mathbf{A})^{-1} \vec{d}_{uusi} = (578.2 \ 6666 \ 471.2 \ 304.8 \ 10635 \ 634.1)^T.$$

Uusi katelaskelma on

|              |     | myynti(kpl) | yks.hinta | myynti(€) | yhteensä |
|--------------|-----|-------------|-----------|-----------|----------|
| tuote1       | Os1 | 500         | 100       | 50 000    |          |
| tuote2       | Os2 | 400         | 10        | 4 000     |          |
| tuote3       | Os3 | 12          | 2000      | 24 000    |          |
| tuote4       | Os4 | 1300        | 150       | 195 000   | 273 000  |
|              |     | ostot(kpl)  | yks.hinta | ostot(€)  | yhteensä |
| raaka-aine 1 | tt1 | 578.2       | 70        | 40 476    |          |
| raaka-aine 2 | tt2 | 6666.2      | 10        | 66 662    |          |
| raaka-aine 3 | tt3 | 471.2       | 50        | 23 561    |          |
| energia      | tt4 | 304.8       | 8         | 2 438     |          |
| työvoima 1   | tt5 | 10635.0     | 15        | 159 526   |          |
| työvoima 2   | tt6 | 634.1       | 25        | 15 853    | 308 516  |

Kysynnän rakenteen muuttuminen pilasi kannattavuuden. Tämä johtui tietenkin siitä, että tuotteet oli hinnoiteltu huonosti. Tuotetta 1 myytiin ylihintaan ja tuotetta 4 myytiin alihintaan. Kun ylihintaisen myynnin osuus pieneni ja alihintaisen myynnin osuus kasvoi, niin kate romahti.

Seuraavaksi laskemme tuotteille omakustannusarvot. Periaate on seuraava.

$$\begin{aligned}\text{Tuotto} = R &= \vec{p}^T \vec{d} \\ \text{Kustannus} = C &= \vec{c}^T \vec{r} = \vec{c}^T \mathbf{B}(\mathbf{I} - \mathbf{A})^{-1} \vec{d}\end{aligned}$$

Jos nyt asetamme

$$\vec{p}^T = \vec{c}^T \mathbf{B}(\mathbf{I} - \mathbf{A})^{-1},$$

niin riippumatta kysynnän rakenteesta tuotto riittää kustannusten peittämiseen. Omakustannus -arvot edellisen yrityksen tapauksessa ovat

$$\vec{p}^T = \vec{c}^T \mathbf{B}(\mathbf{I} - \mathbf{A})^{-1} = (69.35 \quad 10.32 \quad 2153.50 \quad 187.59).$$

Omakustannushintoihin vielä lisätään haluttun kateprosentin (esim. 15%) suuruinen korotus, jolloin hinnoiksi tulee

$$\begin{aligned}\vec{p}^T &= 1.15 \cdot \vec{c}^T \mathbf{B}(\mathbf{I} - \mathbf{A})^{-1} = (79.75 \quad 11.87 \quad 2476.48 \quad 215.73) \\ &\approx (80.00 \quad 11.85 \quad 2475.00 \quad 216.00).\end{aligned}$$

Näillä hinnoilla edelliset kaksi katelaskelmaa menevät seuraaviin muotoihin

|              |     | myynti(kpl) | yks.hinta | myynti(€) | yhteensä |
|--------------|-----|-------------|-----------|-----------|----------|
| tuote1       | Os1 | 1400        | 80.00     | 112 000   |          |
| tuote2       | Os2 | 400         | 11.85     | 4 740     |          |
| tuote3       | Os3 | 12          | 2475.00   | 29 700    |          |
| tuote4       | Os4 | 400         | 216.00    | 86 400    | 232 840  |
|              |     | ostot(kpl)  | yks.hinta | ostot(€)  | yhteensä |
| raaka-aine 1 | tt1 | 400         | 70        | 28 000    |          |
| raaka-aine 2 | tt2 | 5000        | 10        | 50 000    |          |
| raaka-aine 3 | tt3 | 500         | 50        | 25 000    |          |
| energia      | tt4 | 200         | 8         | 1 600     |          |
| työvoima 1   | tt5 | 6000        | 15        | 90 000    |          |
| työvoima 2   | tt6 | 300         | 25        | 7 500     | 202 100  |

|              |     | myynti(kpl) | yks.hinta | myynti(€) | yhteensä |
|--------------|-----|-------------|-----------|-----------|----------|
| tuote1       | Os1 | 500         | 80.00     | 40 000    |          |
| tuote2       | Os2 | 400         | 11.85     | 4 740     |          |
| tuote3       | Os3 | 12          | 2475.00   | 29 700    |          |
| tuote4       | Os4 | 1300        | 216.00    | 280 800   | 355 240  |
|              |     | ostot(kpl)  | yks.hinta | ostot(€)  | yhteensä |
| raaka-aine 1 | tt1 | 578.2       | 70        | 40 476    |          |
| raaka-aine 2 | tt2 | 6666.2      | 10        | 66 662    |          |
| raaka-aine 3 | tt3 | 471.2       | 50        | 23 561    |          |
| energia      | tt4 | 304.8       | 8         | 2 438     |          |
| työvoima 1   | tt5 | 10635.0     | 15        | 159 526   |          |
| työvoima 2   | tt6 | 634.1       | 25        | 15 853    | 308 516  |

Nyt kate on halutun suuruinen riippumatta kysynnän rakenteesta. Tämä on tietenkin vasta lähtökohta laajemmalle analyysille, jossa huomioidaan myös markkinoiden kysyntäpuoli. Seuraavassa muutama helppo periaate:

- Tuotetta ei kannata valmistaa, jos se pitää myydä alle omakustannusarvon.
- Jos voitonmaksimointi -tarkastelu johtaa oleellisesti omakustannusarvoa korkeampaan hintaan, niin tuote on ”lypsylehmä” ja sitä tulee lypsää.
- Lypsylehmät houkuttelevat alalle uusia toimijoita, jolloin kilpailu kiristyy.
- Kilpailun kiristyessä tulee huolehtia kustannusten minimoinnista ja hinnoittelun tarkkuudesta.
- Uuden tuotteen tapauksessa tuotetta voidaan myydä alle omakustannusarvon, jos sillä tavoin saadaan vallattua markkinoita, ja jos uskotaan, että tulevaisuudessa tuote voidaan hinnoitella kallimmaksi tai tuotantokustannukset saadaan painettua alemmiksi niin, että tuotteesta tulee lypsylehmä.
- Hinnoittelu on käytännössä dynaaminen prosessi, jossa ratkaisevinta ei ole tämänhetkinen kate, vaan tuotannosta nyt ja tulevaisuudessa saatava nettotulovirta.

### 4.3 Yhtälöryhmän kantaratkaisut

Sec:YRkanta

Jos yhtälöryhmässä on muuttujia enemmän kuin yhtälöitä, niin tavallisesti sillä on useita ratkaisuja. Näin ei tietenkään tarvitse aina olla. Esimerkki puheena olevan kaltaisesta yhtälöryhmästä, jolla ei ole yhtään ratkaisua, on seuraava:

$$\begin{cases} x + y + z = 2 \\ 2x + 2y + 2z = 1 \end{cases}.$$



Ryhmällä ei tietenkään ole ratkaisua.

Yleisesti tosi ja pätevä väite tässä tilanteessa on seuraava.

**Lause 17** (*Juurien lukumäärä, kun yhtälöryhmä on leveä.*) Jos lineaarisessa yhtälöryhmässä on muuttujia enemmän kuin yhtälöitä, ja yhtälöryhmälle löytyy ratkaisu, niin sille löytyy myös muita ratkaisuja.

*Perustelu:* Asia on selvä, jos tuntee Gauss-Jordan algoritmin vaiheet tässä tapauksessa. Se, että ryhmällä on yksi ratkaisu, merkitsee sitä, että yhtälöryhmän täydennetty kerroinkaavio saadaan algoritmista kolmiomuotoon. Koska muuttujia on enemmän kuin yhtälöitä, on selvää, että osa muuttujista siirretään RHS:iin, jolloin ratkaisuja on paljon.  $\square$

Tässä kappaleessa tarkastelemme yhtälöryhmää

$$\mathbf{Ax} = \mathbf{b}, \tag{4.1} \quad \text{eq\_orig}$$

jossa on  $m$  yhtälöä ja  $n$  muuttujaa. Muuttujat ovat nyt  $x_1, x_2, \dots, x_n$  ja niitä on siis enemmän kuin yhtälöitä;  $n > m$ . Kerroinmatriisi  $\mathbf{A}$  on  $m \times n$ -matriisi,  $\mathbf{x} \in \mathbb{R}^n$  ja  $\mathbf{b} \in \mathbb{R}^m$ . Valitsemme muuttujien joukosta  $m$  muuttujaa ja muodostamme niistä joukon

$$BV = \{x_{k_1}, x_{k_2}, \dots, x_{k_m}\},$$

jota sanomme kannaksi. Kanta voidaan valita

$$\binom{n}{m} = \frac{n!}{m! \cdot (n-m)!}$$

tavalla.<sup>2</sup> Kantoja on käytännössä yleensä paljon. (Jos  $n = 10$  ja  $m = 7$ , niin  $\frac{10!}{7! \cdot (10-7)!} = \frac{10 \cdot 9 \cdot 8}{3 \cdot 2} = 120$ .) Jos on olemassa yhtälöryhmän (4.1) <sup>eq\\_orig</sup> ratkaisu, jossa kaikkien kantaan  $BV$  kuulumattomien muuttujien arvot ovat nollia niin sanomme, että ratkaisu on kantaan  $BV$  liittyvä kantaratkaisu. Kaikkiin kantoihin ei liity kantaratkaisua, mutta koska kantoja on paljon, myös kantaratkaisuja yleensä on runsaasti. Kaikkien kantaratkaisujen laskeminen on suoraviivainen ohjelmointitehtävä. Ainoa asiaa hieman mutkistava asia on se, että kantaan voi liittyä useita kantaratkaisuja. Seuraava esimerkki havainnollistaa asiaa.

<sup>2</sup>Binomikerroin kertoo miten monella tavalla voidaan valita  $m$  palloa  $n$  pallon joukosta, kun järjestyksellä ei ole väliä. Nyt pallojen sijasta poimitaan muuttujia.

**Esimerkki 18** Määritämme yhtälöryhmän

$$\begin{cases} x + y + z + w = 4 \\ 2x + 2y - w = 2 \\ 2y + 2z + 3w = 2 \end{cases} \quad (4.2) \quad \text{YRkanta_eq1}$$

kaikki kantaratkaisut. Merkitsemme yhtälöryhmän kerroinmatriisia ja RHS:ää kirjaimilla

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 0 & -1 \\ 0 & 2 & 2 & 3 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 4 \\ 2 \\ 2 \end{pmatrix}.$$

Muuttujien joukosta  $\{x, y, z, w\}$  voimme valita kolmen kantamuuttujan osajoukkoja neljällä tavalla.  $BV_1 = \{x, y, z\}$ ,  $BV_2 = \{x, y, w\}$ ,  $BV_3 = \{x, z, w\}$  ja  $BV_4 = \{y, z, w\}$ . Seuraavaksi laskemme kantaan  $BV_1$  liittyvän kantaratkaisun. Kun ensimmäinen kantaratkaisu on saatu, laskemme muiden kolmen kannan kantaratkaisut vähemmin selityksin.

Aloitamme tarkastelun kannasta  $BV_1 = \{x, y, z\}$ . Sanomme, että muuttujat  $x$ ,  $y$  ja  $z$  ovat kantamuuttujia. Kantaan kuulumatomalle muuttujalle  $w$  asetetaan arvoksi 0. Kantamuuttujien arvot asetetaan siten, että alkuperäinen yhtälöryhmä toteutuu (kun  $w = 0$ ). Koska  $w = 0$  voidaan yhtälöiden  $w$ -termit jättää pois. Silloin kantaratkaisun arvot saadaan yhtälöryhmästä

$$\begin{aligned} \begin{cases} x + y + z = 4 \\ 2x + 2y = 2 \\ 2y + 2z = 2 \end{cases} &\Leftrightarrow \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ 2 & 2 & 0 \\ 0 & 2 & 2 \end{pmatrix}}_{\mathbf{B}_{\{x,y,z\}}} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \\ 2 \end{pmatrix} \\ \Leftrightarrow \begin{pmatrix} x \\ y \\ z \end{pmatrix} &= \mathbf{B}_{\{x,y,z\}}^{-1} \begin{pmatrix} 4 \\ 2 \\ 2 \end{pmatrix} = \begin{pmatrix} 3 \\ -2 \\ 3 \end{pmatrix}, \quad \longrightarrow \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = \begin{pmatrix} 3 \\ -2 \\ 3 \\ 0 \end{pmatrix} \end{aligned}$$

Kantamuuttujien kertoimista muodostettu kantamatriisi  $\mathbf{B}_{BV}$  muodostuu kerroinmatriisin  $\mathbf{A}$  kanta-muuttujien sarakkeista. Jos kantamatriisi on säännöllinen, niin saamme yksikäsitteisen kantaratkaisun. Tässä tapauksessa  $(x \ y \ z \ w)^T = (3 \ -2 \ 3 \ 0)^T$ .

Jos kantamatriisi ei ole säännöllinen (eli  $\text{Det}(\mathbf{B}_{BV}) = 0$ ), niin on kaksi mahdollista tapusta. Joko kannan mukaisella yhtälöryhmällä ei ole yhtään ratkaisua, jolloin kantaratkaisua ei siis ole. Tai sitten yhtälöryhmällä on useita ratkaisuja. Jos kantaratkaisuja on useampia, niin kanta-yhtälöryhmästä  $\mathbf{B}_{BV}\mathbf{x}_{BV} = \mathbf{b}$  voidaan yksi yhtälö eliminoida rivioperaatioilla. Tämän jälkeen ryhmä on ”leveä” ja sille voidaan määrittää  $m - 1$  muuttujan kannat ja niihin liittyvät kantaratkaisut. (Ts. jos kantaan liittyy useita kantaratkaisuja, niin ne ovat osa-kantoihin liittyviä kantaratkaisuja.) Emme nyt pohdi tämän tapauksen käsittelyä, ettei esimerkki paisuisi liikaa. Tutkimme seuraavaksi loput kolme kanta.

*Tapaus*  $BV_2 = \{x, y, w\}$  ( $z = 0$ ).

$$\begin{cases} x + y + w = 4 \\ 2x + 2y - w = 2 \\ 2y + 3w = 2 \end{cases} \Leftrightarrow \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ 2 & 2 & -1 \\ 0 & 2 & 3 \end{pmatrix}}_{\mathbf{B}_{\{x,y,w\}}} \begin{pmatrix} x \\ y \\ w \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \\ 2 \end{pmatrix}$$

$$\Leftrightarrow \begin{pmatrix} x \\ y \\ w \end{pmatrix} = \mathbf{B}_{\{x,y,w\}}^{-1} \begin{pmatrix} 4 \\ 2 \\ 2 \end{pmatrix} = \begin{pmatrix} 4 \\ -2 \\ 2 \end{pmatrix}, \quad \longrightarrow \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = \begin{pmatrix} 4 \\ -2 \\ 0 \\ 2 \end{pmatrix}$$

*Tapaus*  $BV_3 = \{x, z, w\}$  ( $y = 0$ ).

$$\begin{cases} x + z + w = 4 \\ 2x - w = 2 \\ 2z + 3w = 2 \end{cases} \longrightarrow \text{ei ratkaisua!}$$

$\longrightarrow$  ei kantaratkaisua!

*Tapaus*  $BV_4 = \{y, z, w\}$  ( $x = 0$ ).

$$\begin{cases} y + z + w = 4 \\ 2y - w = 2 \\ 2y + 2z + 3w = 2 \end{cases} \Leftrightarrow \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ 2 & 0 & -1 \\ 2 & 2 & 3 \end{pmatrix}}_{\mathbf{B}_{\{y,z,w\}}} \begin{pmatrix} y \\ z \\ w \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \\ 2 \end{pmatrix}$$

$$\Leftrightarrow \begin{pmatrix} y \\ z \\ w \end{pmatrix} = \mathbf{B}_{\{y,z,w\}}^{-1} \begin{pmatrix} 4 \\ 2 \\ 2 \end{pmatrix} = \begin{pmatrix} -2 \\ 12 \\ -6 \end{pmatrix}, \quad \longrightarrow \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = \begin{pmatrix} 0 \\ -2 \\ 12 \\ -6 \end{pmatrix}$$

Saimme siis kolme kantaratkaisua

$$\mathbf{u}_1 = \mathbf{u}_{\{x,y,z\}} = \begin{pmatrix} 3 \\ -2 \\ 3 \\ 0 \end{pmatrix}, \quad \mathbf{u}_2 = \mathbf{u}_{\{x,y,w\}} = \begin{pmatrix} 4 \\ -2 \\ 0 \\ 2 \end{pmatrix} \text{ ja } \mathbf{u}_3 = \mathbf{u}_{\{y,z,w\}} = \begin{pmatrix} 0 \\ -2 \\ 12 \\ -6 \end{pmatrix}.$$

□

YRkanta\_esim2

**Esimerkki 19** *Jatkamme esimerkin (18) yhtälöryhmän tutkimista määrittämällä siihen liittyvän "homogeenisen" yhtälöryhmän*

$$\begin{cases} x + y + z + w = 0 \\ 2x + 2y - w = 0 \\ 2y + 2z + 3w = 0 \end{cases} \quad (4.3) \quad \text{YRkanta_eq2}$$

HomogYRmcsYRl1 esim1  
 kaikki ratkaisut. Esimerkin (??, s. ??) mukaisesti kaikki homogeenisen ryhmän ratkaisut saadaan muutaman perusratkaisun ('virittäjä-vektorin') lineaarikombinaationa. Koska olemme jo käsitelleet yhtälöryhmän kerroinkaaviota edellisessä esimerkissä tiedämme, että pivotointi tulee onnistumaan x-, y- ja z-sarakkeissa. RHS:iin siirtyy yksi muuttuja ja homogeenisen yhtälöryhmän ratkaisujoukon virittää siis yksi ratkaisuvektori.

$$\begin{aligned}
 & \left( \begin{array}{cccc|c} [1] & 1 & 1 & 1 & 0 \\ 2 & 2 & 0 & -1 & 0 \\ 0 & 2 & 2 & 3 & 0 \end{array} \right) \left| \begin{array}{l} *2 \\ \leftarrow - \end{array} \right. \\
 \Leftrightarrow & \left( \begin{array}{cccc|c} [1] & 1 & 1 & 1 & 0 \\ 0 & 0 & -2 & -3 & 0 \\ 0 & 2 & 2 & 3 & 0 \end{array} \right) \left| \begin{array}{l} \\ \text{vaihda} \\ \text{vaihda} \end{array} \right. \\
 \Leftrightarrow & \left( \begin{array}{cccc|c} [1] & 1 & 1 & 1 & 0 \\ 0 & 2 & 2 & 3 & 0 \\ 0 & 0 & -2 & -3 & 0 \end{array} \right) \Leftrightarrow \begin{cases} x + y + z = -w \\ 2y + 2z = -3w \\ -2z = 3w \end{cases} \\
 \Leftrightarrow & \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = a \cdot \begin{pmatrix} 1 \\ 0 \\ -3 \\ 2 \end{pmatrix} = a\mathbf{v}, \quad a \in \mathbb{R}.
 \end{aligned}$$

YRkanta\_esim3

**Esimerkki 20** Kahden kantaratkaisun erotus on homogeenisen yhtälöryhmän ratkaisu. esimerkin (4.2) kantaratkaisuille  $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$  ja vastaavan homogeenisen yhtälöryhmän (4.3) ratkaisujoukon virittäjälle  $\mathbf{v}$  on voimassa:

$$\begin{aligned}
 \mathbf{u}_1 - \mathbf{u}_2 &= \begin{pmatrix} 3 \\ -2 \\ 3 \\ 0 \end{pmatrix} - \begin{pmatrix} 4 \\ -2 \\ 0 \\ 2 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 3 \\ -2 \end{pmatrix} = -\mathbf{v}, \\
 \mathbf{u}_1 - \mathbf{u}_3 &= \begin{pmatrix} 3 \\ -2 \\ 3 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ -2 \\ 12 \\ -6 \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \\ -9 \\ 6 \end{pmatrix} = 3\mathbf{v}, \\
 \mathbf{u}_2 - \mathbf{u}_3 &= \begin{pmatrix} 4 \\ -2 \\ 0 \\ 2 \end{pmatrix} - \begin{pmatrix} 0 \\ -2 \\ 12 \\ -6 \end{pmatrix} = \begin{pmatrix} 4 \\ 0 \\ -12 \\ 8 \end{pmatrix} = 4\mathbf{v}.
 \end{aligned}$$

Esimerkin (18) yhtälöryhmällä on ääretön määrä ratkaisuja, mutta kantaratkaisuja sillä on vain kolme. Seuraavan lauseen mukaan nämä kolme kantaratkaisua luonnehtivat kaikki ratkaisut.

**Lause 21** Olkoon yhtälöryhmän  $\mathbf{Ax} = \mathbf{b}$  kantaratkaisut  $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_q$ .

(a) Jokainen kantaratkaisujen painotettu summa, jossa painojen summa on 1,

$$\mathbf{x} = c_1 \mathbf{u}_1 + c_2 \mathbf{u}_2 + \dots + c_q \mathbf{u}_q, \quad \sum_{k=1}^q c_k = 1,$$

on yhtälöryhmän ratkaisu.

(b) Jos yhtälöryhmällä on niin monta kantaratkaisua, että jokainen vastaavan homogeeniyhtälön ratkaisujoukon virittäjä on kantaratkaisujen lineaarikombinaatio, niin yhtälöryhmän jokainen ratkaisu voidaan lausua kantaratkaisujen painotettuna summana, jossa painokertoimien summa on 1.

Yrkanta\_lause1

Todistus: (a) A-kohta on läpilasku

$$\begin{aligned} \mathbf{Ax} &= \mathbf{A}(c_1 \mathbf{u}_1 + c_2 \mathbf{u}_2 + \dots + c_q \mathbf{u}_q) \\ &= (c_1 \mathbf{A}\mathbf{u}_1 + c_2 \mathbf{A}\mathbf{u}_2 + \dots + c_q \mathbf{A}\mathbf{u}_q) \\ &= (c_1 \mathbf{b} + c_2 \mathbf{b} + \dots + c_q \mathbf{b}) \\ &= (c_1 + c_2 + \dots + c_q) \mathbf{b} \\ &= \mathbf{b}. \end{aligned}$$

(b) Olkoon  $\mathbf{x}$  jokin yhtälöryhmän ratkaisu. Silloin  $\mathbf{x} - \mathbf{u}_1$  on vastaavan homogeenisen yhtälöryhmän ratkaisu, joka siis voidaan oletuksen mukaan lausua kantaratkaisujen lineaarikombinaationa

$$\begin{aligned} \mathbf{x} - \mathbf{u}_1 &= a_1 \mathbf{u}_1 + a_2 \mathbf{u}_2 + \dots + a_q \mathbf{u}_q \\ \Leftrightarrow \mathbf{x} &= (1 + a_1) \mathbf{u}_1 + a_2 \mathbf{u}_2 + \dots + a_q \mathbf{u}_q. \end{aligned}$$

Viimeisellä rivillä olevan lineaarikombinaation kertoimien summan on pakko olla 1, muuten  $\mathbf{x}$  ei olisi yhtälöryhmän ratkaisu.  $\square$

Yhdistämällä esimerkiksi (18-20) saamme edellisen lauseen mukaan tuloksen

$$\begin{aligned} &\begin{cases} x + y + z + w = 4 \\ 2x + 2y - w = 2 \\ 2y + 2z + 3w = 2 \end{cases} \\ \Leftrightarrow &\begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = c_1 \cdot \begin{pmatrix} 3 \\ -2 \\ 3 \\ 0 \end{pmatrix} + c_2 \cdot \begin{pmatrix} 4 \\ -2 \\ 0 \\ 2 \end{pmatrix} + c_3 \cdot \begin{pmatrix} 0 \\ -2 \\ 12 \\ -6 \end{pmatrix}, \quad (4.4) \\ &\text{missä } c_1 + c_2 + c_3 = 1. \end{aligned}$$

Ääretön määrä ratkaisuja saatiin siis puristettua kolmeen kantaratkaisuun, joiden painotettuna summana kaikki ratkaisut saadaan.

Ratkaisu saadaan vieläkin tiiviimpään pakettiin, jos käytämme homogeenisen yhtälöryhmän ratkaisujoukon virittäjiä. Ratkaisuvektori (4.4) voidaan kirjoittaa

$$\begin{aligned} \mathbf{x} &= (1 - c_2 - c_3)\mathbf{u}_1 + c_2\mathbf{u}_2 + c_3\mathbf{u}_3 \\ &= \mathbf{u}_1 + c_2(\mathbf{u}_2 - \mathbf{u}_1) + c_3(\mathbf{u}_3 - \mathbf{u}_1) \\ &= \mathbf{u}_1 + a \cdot \mathbf{v}, \quad a \in \mathbb{R}. \end{aligned}$$

Siis

$$\begin{cases} x + y + z + w = 4 \\ 2x + 2y - w = 2 \\ 2y + 2z + 3w = 2 \end{cases} \Leftrightarrow \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = \begin{pmatrix} 3 \\ -2 \\ 3 \\ 0 \end{pmatrix} + a \cdot \begin{pmatrix} 1 \\ 0 \\ -3 \\ 2 \end{pmatrix}, \quad a \in \mathbb{R}. \quad (4.5)$$

Koska virittäjiä on yleensä vähemmän kuin kantaratkaisuja, on esitys (4.5) tehokkaampi kuin esitys (4.4). Toisaalta monessa sovelluksessa juuri kantaratkaisut ovat tärkeitä, jolloin esitys (4.4) on kiistatta parempi. Seuraavassa kappaleessa sovellamme tämän ja edellisen kappaleen tietoja lineaariseen optimointiin. Siinä sovellustilanteessa juuri kantaratkaisut saavat tärkeän roolin.

*Huomautus:* Lauseen (21) b-kohdan ehto: ”Jos yhtälöryhmällä on niin monta kantaratkaisua, että jokainen vastaavan homogeeniyhtälön ratkaisujoukon virittäjä on kantaratkaisujen lineaarikombinaatio, niin...” ei ole mikään helposti hahmotettava asia. Sovelluksissa, joissa virittäjiä on vähän ja kantaratkaisuja on paljon, on helppo uskoa ehdon olevan totta. Palaamme asiaan myöhemmin uudelleen.

## 4.4 LP-mallin kantaratkaisut

Sec:LPkanta

Tämän kappaleen esimerkeissä ratkaistaan lineaarinen optimointitehtävä. Lineaarista optimointia on saatettu käsitellä lukion kursseilla. Esimerkit ovat melko helppoja ymmärtää vaikka et tuntisikaan asiaa ennalta.

LPkanta\_esim1

**Esimerkki 22** *Liisan Herkku leipoo tänään Punamarja-leivoksia ja Sinimarja-leivoksia. Leivokset leivotaan isoilla paisto-pelleillä. Yhden paistopellin kokoisen pohjan valmistamiseen kuluu 1,5kg voita 1kg sokeria 10 munaa ja 3kg jauhoja. Punamarja-leivoksen yhteen paistoerään kuluu 1kg puhaherukkaa, 1kg mustikkaa ja 3kg vadelmaa. Vastaavasti Sinimarja-leivoksen yhteen paistoerään kuluu 1kg puhaherukkaa, 2kg*

mustikkaa ja 1kg vadelmaa. Jokaisesta pellistä saatu paistos jaetaan 50 leivospalaan. Näin saadut leivospalat pakataan kuljetuslaatikoihin ja toimitetaan kauppoihin. Yksi Punamarja-leivospala myydään hintaan 1,25€ ja yksi Sinimarja-leivospala hintaan 0,95€.

Raaka-aineiden hinnat ovat 3,00€/kg (punaherukka), 7,00€/kg (mustikka) ja 10,00€/kg (vadelma). Leivospohjan raaka-aineiden hinnat ovat: voi 1,50€/kg, sokeri 0,50€/kg, muna 0,05€/muna, jauho 1,00€/kg. Raaka-aineiden käsittely, pohjan leipominen, paistaminen ja tuotteiden pakkaaminen aiheuttavat palkkakustannuksia noin 8,00€/pelti.

Voita, sokeria, munia ja jauhoa voidaan hankkia miten paljon tahansa, mutta marjoja on tällä hetkellä saatavilla vain 120kg punaherukkaa, 200kg mustikkaa ja 300kg vadelmaa.

Miten paljon Liisan herkun tulee leipoa Punamarja- ja Sinimarja-leivoksia, kun se haluaa käyttää raaka-aineet mahdollisimman tehokkaasti?

*Ratkaisu:* Varsinainen päätös nyt koskee sitä, miten monta pellillistä tehdään Punamarja-leivoksi ja miten monta pellillistä tehdään Sinimarja-leivoksia. Yksi Punamarja-pelti aiheuttaa tuottoja ja kustannuksia seuraavan taulukon mukaisesti.

|        |          |                      |              |
|--------|----------|----------------------|--------------|
| Tuotto |          | $50 \cdot 1,25€ =$   | 62,50€       |
| Kust.  |          |                      |              |
|        | voi      | $-1,5 \cdot 1,50€ =$ | -2,25€       |
|        | sokeri   | $-1 \cdot 0,50€ =$   | -0,50€       |
|        | munat    | $-10 \cdot 0,05€ =$  | -0,50€       |
|        | jauho    | $-3 \cdot 1,00€ =$   | -3,00€       |
|        | palkat   | $-1 \cdot 8,00€ =$   | -8,00€       |
|        | punaher. | $-1 \cdot 3,00€ =$   | -3,00€       |
|        | mustikka | $-1 \cdot 7,00€ =$   | -7,00€       |
|        | vadelma  | $-3 \cdot 10,00€ =$  | -30,00€      |
| Kate   |          |                      | 8,25€ /pelti |

Yksi Sinimarja-pelti aiheuttaa tuottoja ja kustannuksia seuraavan taulukon mukaisesti.

|        |          |                      |              |
|--------|----------|----------------------|--------------|
| Tuotto |          | $50 \cdot 0,95€ =$   | 47,50€       |
| Kust.  |          |                      |              |
|        | voi      | $-1,5 \cdot 1,50€ =$ | -2,25€       |
|        | sokeri   | $-1 \cdot 0,50€ =$   | -0,50€       |
|        | munat    | $-10 \cdot 0,05€ =$  | -0,50€       |
|        | jauho    | $-3 \cdot 1,00€ =$   | -3,00€       |
|        | palkat   | $-1 \cdot 8,00€ =$   | -8,00€       |
|        | punaher. | $-1 \cdot 3,00€ =$   | -3,00€       |
|        | mustikka | $-2 \cdot 7,00€ =$   | -14,00€      |
|        | vadelma  | $-1 \cdot 10,00€ =$  | -10,00€      |
| Kate   |          |                      | 6,25€ /pelti |

Merkitään muuttujalla  $x_1$  leivottavien Punamarja-leivos peltien lukumäärää ja merkitään muuttujalla  $x_2$  leivottavien Sinimarja-leivos peltien lukumäärää. Silloin tämän leivontapäivän aikana kertyvä kokonaiskate on

$$z = 8,25x_1 + 6,25x_2 \quad (\text{€}). \quad (4.6)$$

Punaherukan käyttöä koskee silloin ehto

$$1 \cdot x_1 + 1 \cdot x_2 \leq 120 \quad (\text{kg}), \quad (4.7)$$

sillä jokaiselle pellille tulee kilo punaherukkaa, ja käytettävissä on 120kg tätä marjaa.

Mustikan käyttöä rajoittaa ehto

$$1 \cdot x_1 + 2 \cdot x_2 \leq 200 \quad (\text{kg}), \quad (4.8)$$

sillä jokaiselle Punamarja-pellille tulee kilo mustikkaa, Sinimarja-pellille tulee kaksi kiloa mustikkaa ja käytettävissä on 200kg mustikkaa.

Vadelman käyttöä rajoittaa ehto

$$3 \cdot x_1 + 1 \cdot x_2 \leq 300 \quad (\text{kg}), \quad (4.9)$$

sillä jokaiselle Punamarja-pellille tulee kolme kiloa vadelmaa, Sinimarja-pellille tulee yksi kilo vadelmaa ja käytettävissä on 300kg vadelmaa. Jos päätösmuuttujien merkeistä ei sanota mitään, niin oletamme automaattisesti merkkiehtojen  $x_1 \geq 0$  ja  $x_2 \geq 0$ , olevan voimassa. Tuotantomäärät eivät siis saa olla negatiivisia.

Kun tavoitefunktio ja rajoitteet kootaan tiiviiseen esitykseen syntyy LP-malli:

$$\begin{array}{llll} \text{Max} & z = & 8,25x_1 & + & 6,25x_2 \\ \text{ehdoin} & & x_1 & + & x_2 \leq 120 \\ & & x_1 & + & 2x_2 \leq 200 \\ & & 3x_1 & + & x_2 \leq 300 \\ & & & & x_1, x_2 \geq 0 \end{array} \quad (4.10)$$

laLPkanta\_1

Piirrämme ensin graafisen ratkaisu  $x_1 \times x_2$ -tasoon ja sen jälkeen tutkimme optimiratkaisua matriisilaskennan keinoin.

**1. rajoite  $x_1 + x_2 \leq 120$**  Rajoitteeseen liittyy rajoitesuora  $L_1 : x_1 + x_2 = 120$ , joka kulkee pisteiden  $A = (120, 0)$  ja  $B = (0, 120)$  kautta. Rajoitesuoran  $L_1$  alapuolelta löytyvät rajoitteen toteuttavat tason pisteet. (ks. kuva 4.10 s. 51)

laLPkanta\_1p1

**2. rajoite  $x_1 + 2x_2 \leq 200$**  Rajoitteeseen liittyy rajoitesuora  $L_2 : x_1 + 2x_2 = 200$ , joka kulkee pisteiden  $C = (100, 50)$  ja  $D = (0, 100)$  kautta. Rajoitesuoran  $L_2$  alapuolelta löytyvät rajoitteen toteuttavat tason pisteet.



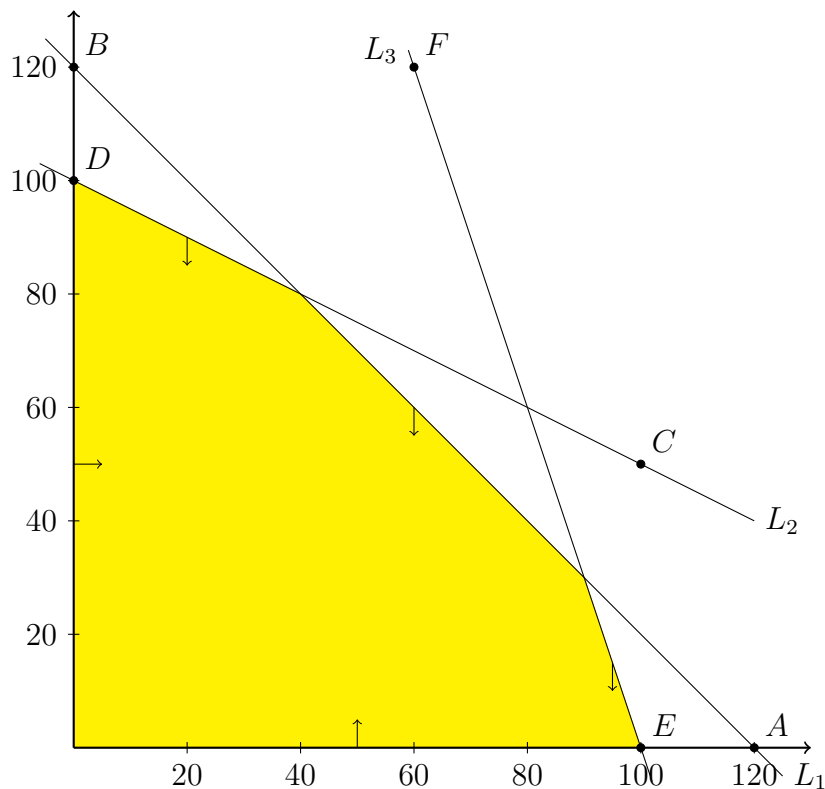
**3. rajoite  $3x_1 + x_2 \leq 300$**  Rajoitteeseen liittyy rajoitesuora  $L_3 : 3x_1 + x_2 = 300$ , joka kulkee pisteiden  $E = (100, 0)$  ja  $F = (60, 120)$  kautta. Rajoitesuoran  $L_3$  alapuolelta löytyvät rajoitteen toteuttavat tason pisteet.

**4. rajoite  $x_1 \geq 0$**  Rajoitesuora on nyt koordinaatiston pystyakseli ( $x_1 = 0$ ) ja sen oikealta puolelta löytyvät rajoitteen toteuttavat pisteet.

**5. rajoite  $x_2 \geq 0$**  Rajoitesuora on nyt koordinaatiston vaaka-akseli ( $x_2 = 0$ ) ja sen yläpuolelta löytyvät rajoitteen toteuttavat pisteet.

Kun edellä löydettyjen pisteiden avulla piirretään rajoitteiden rajoitesuorat ja huomioidaan millä puolella suoraa ovat käyvät pisteet, saadaan kuva 8 sivulla 52, johon käyvät pisteet on merkitty värittämällä.

Käyvän alueen pisteet  $(x_1, x_2)$  koordinaatit toteuttavat kaikki rajoiteepähtälöt (4.10).



**Kuva 8.** Esimerkin (22) rajoitteet ja käypä alue.

Seuraavaksi tutkimme tavoitefunktion käyvässä alueessa saamia arvoja. Käyvän alueen pisteessä  $(x_1, x_2) = (50, 50)$  tavoitefunktio saa arvon

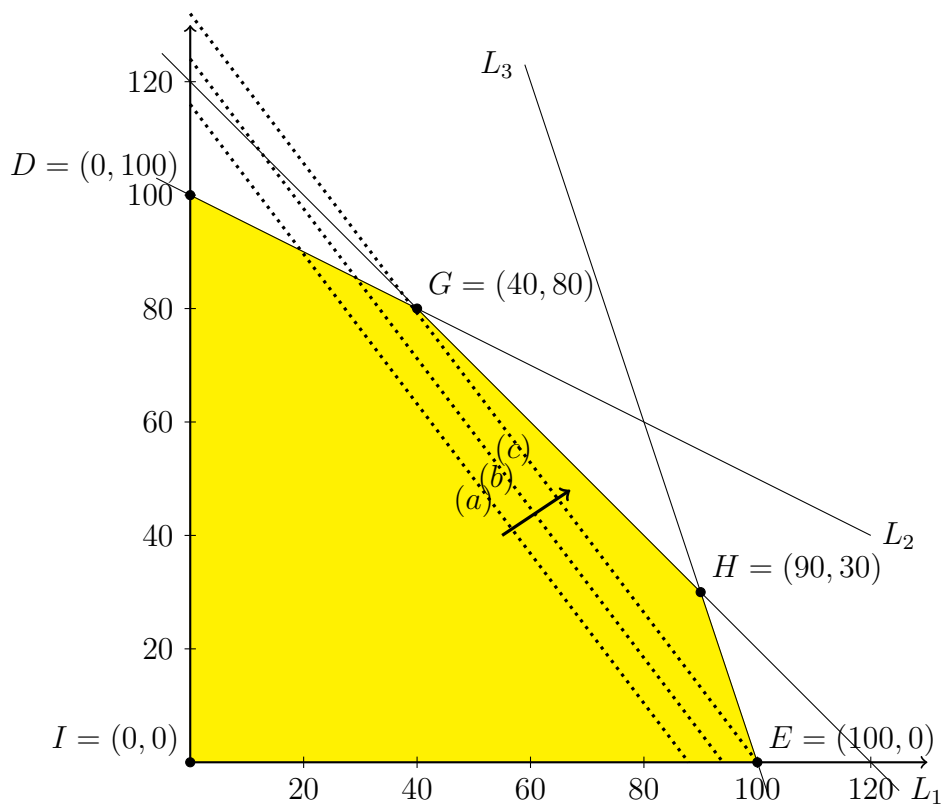
$$z(50, 50) = 8,25 \cdot 50 + 6,25 \cdot 50 = 725. \quad (\text{€})$$

Kaikki ne pisteet, joissa tavoitefunktio saa arvon 725 toteuttavat yhtälön

$$8,25x_1 + 6,25x_2 = 725$$

Tämä on jälleen suoran yhtälö. Suora kulkee pisteiden  $P = (725/8, 25; 0) = (87,9; 0)$  ja  $Q = (0; 725/6, 25) = (0; 116)$  kautta. Tämä tasa-arvosuora (a) on piirretty kuvaan 9. Samaan kuvaan on piirretty myös tavoitefunktion arvoon  $z = 775$  liittyvä tasa-arvosuora (b) sekä tavoitefunktion arvoon  $z = 825$  liittyvä tasa-arvosuora (c). LPkanta\_Fig2

Tasa-arvosuorat ovat yhdensuuntaisia, ja arvon kasvaessa ne siirtyvät (suuntansa säilyttäen) luoteeseen. Nyt on selvää, että käyvän alueen pisteistä paras, eli viimeinen tasa-arvosuoran alle jäävä piste, on nurkkapiste  $H$ . Nurkkapiste  $H$  on rajoitesuorien  $L_1$  ja  $L_3$  leikkauspiste.



**Kuva 9.** Esimerkin (22) tavoitesuora. LPkanta\_esim1

Optimipisteessä tavoitefunktio saa arvon

$$z(H) = z(90; 30) = 8,25 \cdot 90 + 6,25 \cdot 30 = 930,00. \quad (\text{€})$$

Liisan herkun leivontaongelman ratkaisu on siis seuraava:

Leivotaan 90 pellillistä Punamarja-leivoksia (4500 leivosta) ja 30 pellillistä Sinimarja-leivoksia (1500 leivosta). Leivoksista saatava myyntituotto on  $4500 \cdot 1.25 + 1500 \cdot 0.95 = 7050$  (€).

Marjoja käytetään seuraavasti: Punaherukkaa ostetaan  $1 \cdot 90 + 1 \cdot 30 = 120$  (kg), mustikkaa ostetaan  $1 \cdot 90 + 2 \cdot 30 = 150$  (kg) ja vadelmaa ostetaan  $3 \cdot 90 + 1 \cdot 30 = 300$  (kg). Kaikki tarjolla ollut punaherukka ja vadelma ostetaan siis sisään, mutta mustikka-resurssista jää 50 kg käyttämättä. Marjojen ostoista aiheutuneet kustannukset ovat  $120 \cdot 3,00 + 150 \cdot 7,00 + 300 \cdot 10,00 = 4410$  (€). Muut leivontakustannukset 120 pellin leipomisesta ovat  $120 \cdot 14,25 = 1710$  (€).

Leivonnan kate on nyt

$$7050\text{€} - 4410\text{€} - 1710\text{€} = 930\text{€}.$$

#### 4.4.1 Pelivaramuuttujat ja varjohinnat

Lisätään LP-malliin (4.10) pelivaramuuttujat  $s_k =$  resurssin  $k$  käyttämättä jäänyt resurssi. Siis  $s_1$  on ostamatta jätetyn punaherukan määrä (kg),  $s_2$  on ostamatta jätetyn mustikan määrä (kg) ja  $s_3$  on ostamatta jätetyn vadelman määrä (kg). Edellä löydetyssä optimiratkaisussa  $x_1 = 90$ ,  $x_2 = 30$ ,  $s_1 = 0$ ,  $s_2 = 50$  ja  $s_3 = 0$ .

Kun kirjoitamme rajoite-epäyhtälöt yhtälöiksi

$$\begin{cases} x_1 + x_2 + s_1 & = 120 \\ x_1 + 2x_2 + s_2 & = 200 \\ 3x_1 + x_2 + s_3 & = 300 \end{cases}, \quad (4.11)$$

niin on helppo nähdä, että edellä saatu optimiratkaisu on yhtälöryhmän (4.11) kantaan  $BV = \{x_1, x_2, s_2\}$  liittyvä kantaratkaisu.

Ratkaistaan muutkin rajoiteyhtälöryhmän kantaratkaisut

| kanta                         | YR                                                                                        | ratkaisu                                                          | $z$       |
|-------------------------------|-------------------------------------------------------------------------------------------|-------------------------------------------------------------------|-----------|
| $BV_1 = \{x_1, x_2, s_1\}$    | $\begin{cases} x_1 + x_2 + s_1 = 120 \\ x_1 + 2x_2 = 200 \\ 3x_1 + x_2 = 300 \end{cases}$ | $\begin{cases} x_1 = 80 \\ x_2 = 60 \\ s_1 = -20 \end{cases}$     | Ei käy!   |
| $BV_2 = \{x_1, x_2, s_2\}$    | $\begin{cases} x_1 + x_2 = 120 \\ x_1 + 2x_2 + s_2 = 200 \\ 3x_1 + x_2 = 300 \end{cases}$ | $\begin{cases} x_1 = 90 \\ x_2 = 30 \\ s_2 = 50 \end{cases}$      | $z = 930$ |
| $BV_3 = \{x_1, x_2, s_3\}$    | $\begin{cases} x_1 + x_2 = 120 \\ x_1 + 2x_2 = 200 \\ 3x_1 + x_2 + s_3 = 300 \end{cases}$ | $\begin{cases} x_1 = 40 \\ x_2 = 80 \\ s_3 = 100 \end{cases}$     | $z = 830$ |
| $BV_4 = \{x_1, s_1, s_2\}$    | $\begin{cases} x_1 + s_1 = 120 \\ x_1 + s_2 = 200 \\ 3x_1 = 300 \end{cases}$              | $\begin{cases} x_1 = 100 \\ s_1 = 20 \\ s_2 = 100 \end{cases}$    | $z = 825$ |
| $BV_5 = \{x_1, s_1, s_3\}$    | $\begin{cases} x_1 + s_1 = 120 \\ x_1 = 200 \\ 3x_1 + s_3 = 300 \end{cases}$              | $\begin{cases} x_1 = 200 \\ s_1 = -80 \\ s_3 = -300 \end{cases}$  | Ei käy    |
| $BV_6 = \{x_1, s_2, s_3\}$    | $\begin{cases} x_1 = 120 \\ x_1 + s_2 = 200 \\ 3x_1 + s_3 = 300 \end{cases}$              | $\begin{cases} x_1 = 120 \\ s_2 = 80 \\ s_3 = -60 \end{cases}$    | Ei käy    |
| $BV_7 = \{x_2, s_1, s_2\}$    | $\begin{cases} x_2 + s_1 = 120 \\ 2x_2 + s_2 = 200 \\ x_2 = 300 \end{cases}$              | $\begin{cases} x_2 = 300 \\ s_1 = -180 \\ s_2 = -400 \end{cases}$ | Ei käy    |
| $BV_8 = \{x_2, s_1, s_3\}$    | $\begin{cases} x_2 + s_1 = 120 \\ 2x_2 = 200 \\ x_2 + s_3 = 300 \end{cases}$              | $\begin{cases} x_2 = 100 \\ s_1 = 20 \\ s_3 = 200 \end{cases}$    | $z = 625$ |
| $BV_9 = \{x_2, s_2, s_3\}$    | $\begin{cases} x_2 = 120 \\ 2x_2 + s_2 = 200 \\ x_2 + s_3 = 300 \end{cases}$              | $\begin{cases} x_2 = 120 \\ s_2 = -40 \\ s_3 = 180 \end{cases}$   | Ei käy    |
| $BV_{10} = \{s_1, s_2, s_3\}$ | $\begin{cases} s_1 = 120 \\ s_1 + s_2 = 200 \\ s_1 + s_2 + s_3 = 300 \end{cases}$         | $\begin{cases} s_1 = 120 \\ s_2 = 200 \\ s_3 = 300 \end{cases}$   | $z = 0$   |

Kantoja on  $\frac{5!}{3!(5-3)!} = \frac{5 \cdot 4}{2 \cdot 1} = 10$  erilaista. Näistä 5 on käypiä siinä mielessä, että kanta-muuttujat saavat merkkirajoitteiden mukaiset ei-negatiiviset arvot. Käyvät kantaratkaisut vastaavat graafisessa ratkaisussa piirrettyjä käyvän alueen nurkkapisteitä kuvassa 9 LPkanta\_Fig2  
seuraavasti.

| kanta                         | kantaratkaisu<br>( $x_1, x_2, s_1, s_2, s_3$ ) | ( $x_1, x_2$ ) | piste |
|-------------------------------|------------------------------------------------|----------------|-------|
| $BV_2 = \{x_1, x_2, s_2\}$    | (90, 30, 0, 50, 0)                             | (90, 30)       | $H$   |
| $BV_3 = \{x_1, x_2, s_3\}$    | (40, 80, 0, 0, 100)                            | (40, 80)       | $G$   |
| $BV_4 = \{x_1, s_1, s_2\}$    | (100, 0, 20, 100, 0)                           | (100, 0)       | $E$   |
| $BV_8 = \{x_2, s_1, s_3\}$    | (0, 100, 20, 0, 200)                           | (0, 100)       | $D$   |
| $BV_{10} = \{s_1, s_2, s_3\}$ | (0, 0, 120, 200, 300)                          | (0, 0)         | $I$   |

Optimikannan kantaratkaisulle ja tavoitefunktion arvolle siinä pätee matriisiyhtälöt

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 3 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ s_2 \end{pmatrix} = \begin{pmatrix} 120 \\ 200 \\ 300 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

$$z = (8,25 \quad 6,25 \quad 0) \begin{pmatrix} x_1 \\ x_2 \\ s_2 \end{pmatrix}$$

Siiis

$$z = (8,25 \quad 6,25 \quad 0) \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 3 & 1 & 0 \end{pmatrix}^{-1} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \quad (4.12)$$

$$= (8,25 \quad 6,25 \quad 0) \begin{pmatrix} -0,5 & 0 & 0,5 \\ 1,5 & 0 & -0,5 \\ -2,5 & 1 & 0,5 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

$$= (5,25 \quad 0 \quad 1,00) \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \quad (4.13)$$

Kaavassa <sup>LPkanta\_eq2</sup>(4.13) esiintyvää vaakavektoria sanotaan ”varjohinta vektoriksi” (shadow price) ja sitä useimmissa oppikirjoissa merkitään symbolilla  $\boldsymbol{\pi} = (\pi_1 \quad \pi_2 \quad \pi_3)$ . Tällä merkinnällä kaava <sup>LPkanta\_eq2</sup>(4.13) voidaan kirjoittaa lyhyesti  $z = \boldsymbol{\pi b}$ . Erityisesti esimerkin tapauksessa, jos resursseissa tapahtuu pieniä muutoksia (kuitenkin niin pieniä, ettei optimikanta vaihdu) saadaan tavoitefunktion arvon muutos lausekkeesta

$$\begin{aligned} \Delta z = \boldsymbol{\pi \Delta b} &= \pi_1 \cdot \Delta b_1 + \pi_2 \cdot \Delta b_2 + \pi_3 \cdot \Delta b_3 \\ &= 5,25 \cdot \Delta b_1 + 0 \cdot \Delta b_2 + 1,00 \cdot \Delta b_3. \end{aligned}$$

Kertoimet tarkoittavat sitä, että jos Liisan herkku saa mahdollisuuden ostaa litran lisää punaherukkaa (normaaliin hintaan), niin kate kasvaa 5,25€. Jos Liisan herkku saa mahdollisuuden ostaa lisälitran mustikkaa (normaalihintaan), niin optimiratkaisun kate ei kasva lainkaan. Tämä selittyy sille, että optimiratkaisussa ei osteta enempää kuin 150kg mustikkaa, joten lisälitran ostomahdollisuutta ei optimiratkaisussa käytetä. Jos Liisan herkku saa mahdollisuuden ostaa lisälitran vadelmaa (normaalihintaan), niin optimiratkaisun kate kasvaa eurolla.

Usein lisäresurssia ei saada ostettua normaalihintaan  $p_{0,k}$ , vaan lisäyksiköstä on maksettava korkeampi hinta  $p_k = p_{0,k} + \Delta p_k$ . Silloin yhden yksikön lisäresurssista saatava tavoitefunktion parannus on

$$\Delta kate = \pi_k - \Delta p_k,$$

eli katteen parannus on varjohinta miinus lisähinta. Varjohinta antaa siis ylärajan sille, miten paljon lisä-resurssiyksiköstä voi maksaa ylihintaa ilman, että kate huononee.<sup>3</sup> Varjohinnat ovat tärkeä ja luonnollinen päätöksenteon apuväline kun toimitaan rajallisin resurssein. Siksi yritykset mielellään maksavat pientä korvausta sille, joka pystyy ne laskemaan.

Vielä on syytä korostaa, että varjohinta liittyy aina kantaan. Esimerkissä Liisan herkun optimiratkaisu on kantaan  $BV = \{x_1, x_2, s_2\}$  liittyvä kantaratkaisu. Varjohinta-vektori laskettiin kaavalla

$$\pi = (8,25 \quad 6,25 \quad 0) \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 3 & 1 & 0 \end{pmatrix}^{-1}.$$

Tässä vaakavektori sisältää optimikannan kantamuuttujien kertoimet tavoitefunktion lausekkeessa ja käännettävän matriisin sarakkeet ovat optimikannan kantamuuttujien sarakkeet rajoiteyhtälöryhmän kerroinkaaviossa. Varjohinnat ovat siis helpot laskea, kunhan vain tiedämme optimikannan. Optimikannan määrittystä pohditaan lisää 'Operaatioanalyysin' opintojaksolla (orms.1020).

---

<sup>3</sup>Huomaa, että joskus on aiheellista hankkia lisäresurssia, vaikka kate siitä huononisikin. Halutaan olla luotettava toimittaja lopputuotemarkkinoilla; halutaan ylläpitää täys-työllisyyttä, ...

## 5 DETERMINANTIT

Ch:Determ

### 5.1 Determinantti, perustelut ominaisuuksille

Sec:supDetDef

c\_permutaatiot

#### 5.1.1 Permutaatiot

Diskreetin matematiikan kurssilla tulemme tutkimaan äärellisten joukkojen erilaisia järjestyksiä eli permutaatioita. äärellisen joukon prototyypinä käytämme nyt joukkoa  $\mathbb{N}_n = \{1, 2, \dots, n\}$ . Koulukurssissa on jo todettu, että  $\mathbb{N}_n$  voidaan järjestää eli permutoida  $n!$  tavalla.

**Määritelmä 23** Joukon  $\mathbb{N}_n = \{1, 2, \dots, n\}$  bijektio itselleen on  $\mathbb{N}_n$ -permutaatio. Merkitään  $S_n = \{\varphi : \mathbb{N}_n \rightarrow \mathbb{N}_n \text{ on bijektio}\} = \mathbb{N}_n\text{-permutaatioiden joukko}$ . Permutaation  $\varphi$  inversioiden lukumäärä  $p_\varphi$  on niiden parien  $(j, k) \in \mathbb{N}_n \times \mathbb{N}_n$  lukumäärä, joille  $1 \leq j < k \leq n$  ja  $\varphi(j) > \varphi(k)$ . Vaihto  $\alpha_{jk} \in S_n$  on permutaatio, jolle

$$\begin{cases} \alpha(j) = k \\ \alpha(k) = j \\ \alpha(i) = i, \text{ kun } i \neq j, k \end{cases}.$$

Permutaatio on siis äärellisen joukon bijektio itselleen. Pienten joukkojen tapauksessa merkitsemme permutaatiota sen arvoluettelolla  $\varphi = (\varphi(1), \varphi(2), \dots, \varphi(n))$ . Merkintä on periaatteessa huono, mutta paljon käytetty. Muista, että oikeasti permutaatio on kuvaus, eikä jono. Usein onkin pienten permutaatioiden tapauksessa havainnollisempaa piirtää kuvauksen nuolikaavio.

**Esimerkki 24** (1) Jos  $\varphi \in S_3$  ja  $\varphi(1) = 2$ ,  $\varphi(2) = 3$  ja  $\varphi(3) = 1$ , niin se voidaan esittää

”arvojonona” tai ”nuolikaaviona”

$$\varphi = (2, 3, 1) \leftrightarrow \begin{array}{ccc} 1 & \searrow & 1 \\ 2 & \nearrow & 2 \\ 3 & \searrow & 3 \end{array} \quad \varphi$$

(2) Luetteloiden avulla on helppo esittää kaikki pienet permutaatiot.

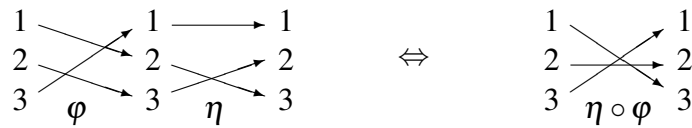
$$S_1 = \{(1)\}$$

$$S_2 = \{(1, 2), (2, 1)\}$$

$$S_3 = \{(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1)\}$$

(3) permutaation  $\varphi = (2, 3, 1)$  inversioiden lukumäärän  $p_{(2,3,1)} = 2$  näkee helposti kohdan (1) nuolikaaviosta.

(4) Kahdesta permutaatiosta yhdistämällä saatu kuvaus on myös permutaatio



Jonomerkinnällä sama asia merkitään.

$$(1, 3, 2) \circ (2, 3, 1) = (3, 2, 1)$$

(5) Edellä  $(1, 3, 2) \circ (2, 3, 1)$  lasketaan seuraavasti:

”Sisäfunktio kuvaa ykkösen kakkoselle (#1), jonka ulkofunktio kuvaa kolmoselle (#2). Siis yhdistetty kuvaus kuvaa ykkösen kolmoselle (#3).”

$$(1, \searrow \#2, 3, 2) \circ (2, \searrow \#1, 3, 1) = (3, \searrow \#3, , )$$

”Sisäfunktio kuvaa kakkosen kolmoselle (#4), jonka ulkofunktio kuvaa kakkoselle (#5). Siis yhdistetty kuvaus kuvaa kakkosen kakkoselle (#6).”

$$(1, 3, \searrow \#5, 2) \circ (2, \searrow \#4, 3, 1) = (3, \searrow \#6, 2, )$$

”Sisäfunktio kuvaa kolmosen ykköselle (#7), jonka ulkofunktio kuvaa ykköselle (#8). Siis yhdistetty kuvaus kuvaa kolmosen ykköselle (#9).”

$$(1, \searrow \#8, 3, 2) \circ (2, \searrow \#7, 3, 1) = (3, \searrow \#9, 2, 1)$$

(6) Vaihto toimii eri tavoin oikealta ja vasemmalta

$$\begin{array}{ccc} 1 & \searrow & 1 \\ 2 & \searrow & 2 \\ 3 & \searrow & 3 \end{array} \Leftrightarrow \alpha_{12} \circ (2, 3, 1) = (1, 3, 2)$$

Jonokielellä: ”Ykkönen ja kakkonen vaihtavat paikkaa.”

Kuvauskielellä: ”Ykkösen ja kakkosen alkukuvat vaihtuvat.”





*Jonokielellä: "Ensimmäinen ja toinen vaihtavat paikkaa."*

*Kuvauskielellä: "Ykkösen ja kakkosen kuvat vaihtuvat."*

**Määritelmä 25** Permutaatio  $\varphi \in S_n$  on parillinen, jos inversioiden määrä  $p_\varphi$  on parillinen ja vastaavasti  $\varphi$  on pariton, jos inversioiden määrä  $p_\varphi$  on pariton. Permutaation  $\varphi \in S_n$  merkki, eli signum, on luku

$$\text{sign } \varphi = \text{sgn } \varphi = \prod_{1 \leq j < k \leq n} \frac{\varphi(k) - \varphi(j)}{k - j}$$

### Esimerkki 26

$$\begin{aligned} \text{sgn}(2, 3, 1) &= \frac{(\varphi(2) - \varphi(1))}{(2 - 1)} \cdot \frac{(\varphi(3) - \varphi(1))}{(3 - 1)} \cdot \frac{(\varphi(3) - \varphi(2))}{(3 - 2)} \\ &= \frac{(3 - 2)}{(2 - 1)} \cdot \frac{(1 - 2)}{(3 - 1)} \cdot \frac{(1 - 3)}{(3 - 2)} = +1 \end{aligned}$$

$$\begin{aligned} \text{sgn}(2, 1, 3) &= \frac{(\varphi(2) - \varphi(1))}{(2 - 1)} \cdot \frac{(\varphi(3) - \varphi(1))}{(3 - 1)} \cdot \frac{(\varphi(3) - \varphi(2))}{(3 - 2)} \\ &= \frac{(1 - 2)}{(2 - 1)} \cdot \frac{(3 - 2)}{(3 - 1)} \cdot \frac{(3 - 1)}{(3 - 2)} = -1 \end{aligned}$$

**Lause 27** Olkoon  $\varphi, \eta \in S_n$  permutaatioita ja  $\alpha_{jk} \in S_n$  vaihto. Silloin

- (1)  $\text{sgn } \varphi = (-1)^{p_\varphi} = \begin{cases} +1, & \text{kun } \varphi \text{ on parillinen} \\ -1, & \text{kun } \varphi \text{ on pariton} \end{cases}$
- (2)  $\text{sgn}(\varphi \circ \eta) = \text{sgn } \varphi \cdot \text{sgn } \eta$
- (3)  $\text{sgn } \alpha_{jk} = -1$

**Todistus:** (1) Olkoon joukon  $\mathbb{N}_n = \{1, 2, \dots, n\}$  kahden alkion osajoukkojen joukko  $P = \{\{i, j\} \subset \mathbb{N}_n \mid i \neq j\}$ . Koska  $\varphi$  on bijektio, niin myös kuvaus  $P \rightarrow P, \{j, k\} \mapsto$

$\{\varphi(j), \varphi(k)\}$  on hyvin määritelty ja bijektio. Siis merkin määritelmässä esiintyy osoittajan tekijöinä itseisarvoiltaan samat luvut kuin nimittäjässä. Osoittajan tekijä  $(\varphi(j) - \varphi(k))$  on kuitenkin  $p_\varphi$  kertaa eri merkinen kuin nimittäjässä. Siis  $\text{sgn } \varphi = (-1)^{p_\varphi}$ .

(2) Merkitään  $\eta(j) = j^*$ .

$$\begin{aligned} \text{sgn}(\varphi \circ \eta) &= \prod_{1 \leq j < k \leq n} \frac{\varphi(\eta(k)) - \varphi(\eta(j))}{k - j} \\ &= \prod_{\{j,k\} \in P, j < k} \left( \frac{\varphi(\eta(k)) - \varphi(\eta(j))}{\eta(k) - \eta(j)} \right) \left( \frac{\eta(k) - \eta(j)}{k - j} \right) \\ &= \prod_{\{j^*, k^*\} \in P, j^* < k^*} \left( \frac{\varphi(k^*) - \varphi(j^*)}{k^* - j^*} \right) \prod_{\{j,k\} \in P, j < k} \left( \frac{\eta(k) - \eta(j)}{k - j} \right) \\ &= \text{sgn } \varphi \cdot \text{sgn } \eta \end{aligned}$$

(3) Merkitään  $n$ :llä erotusta  $k - j$ . Todistamme väitteen induktiolla  $n$ :n suhteen:

Induktion lähtökohta: Kun  $n = k - j = 1$  inversioiden määrä on 1 ja väite seuraa kohdasta (1).

Induktio-oletus: Väite on tosi, kun  $k - j < n$ .

Induktioväite: Väite on tosi, kun  $k - j = n$ .

Induktiotodistus: Valitaan indeksi  $p$  indeksien  $j$  ja  $k$  välistä (siis  $j < p < k$ ). Silloin

$$\alpha_{jk} = \alpha_{jp} \circ \alpha_{pk} \circ \alpha_{jp}$$

Siis lauseen kohdan (2) ja induktio-oletuksen nojalla

$$\text{sgn}(\alpha_{jk}) = \text{sgn}(\alpha_{jp}) \cdot \text{sgn}(\alpha_{pk}) \cdot \text{sgn}(\alpha_{jp}) = (-1) \cdot (-1) \cdot (-1) = -1$$

□

Seuraavaksi palaamme determinantin yleiseen määritelmään.

### 5.1.2 Isot determinantit

det\_isot

**Määritelmä 28**  $n \times n$ -matriisin  $\mathbf{A} = (a_{ij})$  determinantti on luku

$$\det \mathbf{A} = \sum_{\varphi \in S_n} (\text{sgn } \varphi \cdot a_{\varphi(1)1} a_{\varphi(2)2} \cdots a_{\varphi(n)n}),$$

jolle käytämme myös merkintää

$$\det \mathbf{A} = \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$$

Ensimmäiseksi tarkistamme, ettei uusi määritelmä ole ristiriidassa aiemmin antamiemme kaksi- ja kolmirivisten determinantin kaavojen kanssa.

### Esimerkki 29

$$\begin{aligned} \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} &= \operatorname{sgn}(1, 2)a_{11}a_{22} + \operatorname{sgn}(2, 1)a_{21}a_{12} \\ &= a_{11}a_{22} - a_{21}a_{12} \end{aligned}$$

*Siis kaksirivisen determinantin tapauksessa saadaan sama lauseke kuin määritelmässä*  
~~(def\_kaksiriviset\_kaksi\_det~~  
 (??) sivulla ??.

$$\begin{aligned} \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} &= \operatorname{sgn}(1, 2, 3)a_{11}a_{22}a_{33} + \operatorname{sgn}(1, 3, 2)a_{11}a_{32}a_{23} \\ &\quad + \operatorname{sgn}(3, 1, 2)a_{31}a_{12}a_{23} + \operatorname{sgn}(2, 1, 3)a_{21}a_{12}a_{33} \\ &\quad + \operatorname{sgn}(2, 3, 1)a_{21}a_{32}a_{13} + \operatorname{sgn}(3, 2, 1)a_{31}a_{22}a_{13} \\ &= a_{11}a_{22}a_{33} - a_{11}a_{32}a_{23} \\ &\quad + a_{31}a_{12}a_{23} - a_{21}a_{12}a_{33} \\ &\quad + a_{21}a_{32}a_{13} - a_{31}a_{22}a_{13} \\ &= a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \end{aligned}$$

*Siis myös kolmirivisen determinantin tapauksessa saadaan sama lauseke kuin määritelmässä*  
~~(def\_kolme\_riviset\_kolme\_det~~  
 (??) sivulla ??.

Seuraavat lauseet ovat determinantin ymmärtämisen kannalta hyvin tärkeät. Ne tulee ymmärtää hyvin. Siksi niistä esitetään sekä sanallinen (mahdollisimman havainnollinen), että kaavanomainen (mahdollisimman täsmällinen) muoto. Jotta kaavat olisivat helppo muodostaa, sovimme epästandardin, vain tässä yhteydessä käytettävän merkin.

Olkoon tarkasteltavina  $n \times n$ -matriisi  $\mathbf{A} = (a_{ij})$  ja  $n$ -alkioinen pystyvektori  $\vec{u} = (u_1 \ u_2 \ \cdots \ u_n)^T$ . Sovimme, että tässä kappaleessa (ja myöhemmin Cramerin kaavojen yhteydessä) merkintä

$$\mathbf{A}[k : \vec{u}]$$

tarkoittaa matriisia, joka saadaan, kun  $\mathbf{A}$ :n  $k$ :s sarake korvataan pystyvektorilla  $\vec{u}$ . Jos siis esimerkiksi

$$\mathbf{A} = \begin{pmatrix} 2 & -1 & 3 \\ 1 & 2 & 4 \\ 5 & -3 & 1 \end{pmatrix}, \quad \text{ja} \quad \vec{u} = \begin{pmatrix} 7 \\ 8 \\ 9 \end{pmatrix},$$

niin

$$\mathbf{A}[2 : \vec{u}] = \begin{pmatrix} 2 & 7 & 3 \\ 1 & 8 & 4 \\ 5 & 9 & 1 \end{pmatrix}$$

Lisäksi käytämme paljon käytettyä merkintää, jossa matriisin  $\mathbf{A} = (a_{ij})$   $k$ :tta saraketta merkitään pystyvektorilla

$$\vec{a}_{\bullet k} = \begin{pmatrix} a_{1k} \\ a_{2k} \\ \vdots \\ a_{nk} \end{pmatrix}$$

th\_det1

**Lause 30** Tarkastellaan  $n \times n$ -matriisia  $\mathbf{A}$ .

(Sanallinen muoto:) Jos neliömatriisin  $\mathbf{A}$  jonkin sarakkeen kaikki alkiot ovat nollia, niin matriisin determinanti on nolla.

(Kaavana:)

$$\vec{a}_{\bullet k} = \vec{0} \quad \Rightarrow \quad \det(\mathbf{A}) = 0$$

(esimerkki:)

$$\begin{vmatrix} 2 & 0 & 1 \\ 7 & 0 & -2 \\ 3 & 0 & 5 \end{vmatrix} = 0$$

**Todistus:**

$$\det(\mathbf{A}) = \sum_{\varphi \in S_n} \operatorname{sgn}(\varphi) a_{\varphi(1)1} \cdots \underbrace{a_{\varphi(k)k}}_{=0, \forall k} \cdots a_{\varphi(n)n} = 0$$

□

th\_det2

**Lause 31** Tarkastellaan  $n \times n$ -matriisia  $\mathbf{A}$ .

(Sanallinen muoto:) Jos neliömatriisin  $\mathbf{A}$  kaksi saraketta vaihtavat paikkaa, niin syntyvän matriisin determinanti on alkuperäisen vastaluku.

(Kaavana:)

$$\mathbf{B} = \mathbf{A}[j : \vec{a}_{\bullet k}][k : \vec{a}_{\bullet j}] \quad \Rightarrow \quad \det(\mathbf{B}) = -\det(\mathbf{A})$$

(Esimerkki:)

$$\begin{vmatrix} 2 & 1 & 3 \\ -5 & 0 & 4 \\ 0 & 0 & 7 \end{vmatrix} = - \begin{vmatrix} 1 & 2 & 3 \\ 0 & -5 & 4 \\ 0 & 0 & 7 \end{vmatrix}$$

**Todistus:** Jos  $\varphi \in S_n$  on  $n$ -permutaatio, niin merkitsemme  $\hat{\varphi} = \varphi \circ \alpha_{jk}$ . Nyt

$$\begin{aligned} \det(\mathbf{B}) &= \sum_{\varphi \in S_n} \operatorname{sgn}(\varphi) b_{\varphi(1)1} \cdots b_{\varphi(j)j} \cdots b_{\varphi(k)k} \cdots b_{\varphi(n)n} \\ &= \sum_{\varphi \in S_n} \operatorname{sgn}(\varphi) a_{\varphi(1)1} \cdots a_{\varphi(j)k} \cdots a_{\varphi(k)j} \cdots a_{\varphi(n)n} \\ &= \sum_{\varphi \in S_n} -\operatorname{sgn}(\hat{\varphi}) a_{\hat{\varphi}(1)1} \cdots a_{\hat{\varphi}(k)k} \cdots a_{\hat{\varphi}(j)j} \cdots a_{\hat{\varphi}(n)n} \\ &= -\det(\mathbf{A}) \end{aligned}$$

□

th\_det3

**Lause 32** Tarkastellaan  $n \times n$ -matriisia  $\mathbf{A}$ .(Sanallinen muoto:) Jos neliömatriisissa  $\mathbf{A}$  on kaksi identtistä saraketta, niin matriisin determinantti on 0.

(Kaavana:)

$$j \neq k, \text{ ja } \vec{a}_{\bullet j} = \vec{a}_{\bullet k} \Rightarrow \det(\mathbf{A}) = 0$$

(Esimerkki:)

$$\begin{vmatrix} 2 & 2 & 3 \\ -5 & -5 & 4 \\ 0 & 0 & 7 \end{vmatrix} = 0$$

**Todistus:** Tämä on suora seuraus edellisestä lauseesta. Nyt nimittäin  $\mathbf{A} = \mathbf{A}[j : \vec{a}_{\bullet k}][k : \vec{a}_{\bullet j}]$ , joten edellisen lauseen mukaan  $\det(\mathbf{A}) = -\det(\mathbf{A})$  □

th\_det4

**Lause 33** Tarkastellaan  $n \times n$ -matriisia  $\mathbf{A}$ .(Sanallinen muoto:) Neliömatriisin  $\mathbf{A}$  determinantin lasku on  $k$ :nnen sarakkeen suhteen lineaarinen operaatio.

(Kaavana:)

$$\begin{aligned} \det(\mathbf{A}[k : \vec{u} + \vec{v}]) &= \det(\mathbf{A}[k : \vec{u}]) + \det(\mathbf{A}[k : \vec{v}]), \quad \text{ja} \\ \det(\mathbf{A}[k : \lambda \cdot \vec{u}]) &= \lambda \cdot \det(\mathbf{A}[k : \vec{u}]) \end{aligned}$$

(Esimerkki:)

$$\begin{vmatrix} 2 & 5+1 & 3 \\ -5 & 2+9 & 4 \\ 0 & 3-2 & 7 \end{vmatrix} = \begin{vmatrix} 2 & 5 & 3 \\ -5 & 2 & 4 \\ 0 & 3 & 7 \end{vmatrix} + \begin{vmatrix} 2 & 1 & 3 \\ -5 & 9 & 4 \\ 0 & -2 & 7 \end{vmatrix}$$

$$\begin{vmatrix} 2 & 100 \cdot 5 & 3 \\ -5 & 100 \cdot 2 & 4 \\ 0 & 100 \cdot 3 & 7 \end{vmatrix} = 100 \cdot \begin{vmatrix} 2 & 5 & 3 \\ -5 & 2 & 4 \\ 0 & 3 & 7 \end{vmatrix}$$

**Todistus:**

$$\begin{aligned}
\det \mathbf{A}[k : \vec{u} + \vec{v}] &= \sum_{\varphi \in S_n} (\operatorname{sgn} \varphi \cdot a_{\varphi(1)1} \cdots (u_{\varphi(k)} + v_{\varphi(k)}) \cdots a_{\varphi(n)n}) \\
&= \sum_{\varphi \in S_n} (\operatorname{sgn} \varphi \cdot a_{\varphi(1)1} \cdots u_{\varphi(k)} \cdots a_{\varphi(n)n}) \\
&\quad + \sum_{\varphi \in S_n} (\operatorname{sgn} \varphi \cdot a_{\varphi(1)1} \cdots v_{\varphi(k)} \cdots a_{\varphi(n)n}) \\
&= \mathbf{A}[k : \vec{u}] + \mathbf{A}[k : \vec{v}]
\end{aligned}$$

$$\begin{aligned}
\det(\mathbf{A}[k : \lambda \cdot \vec{u}]) &= \sum_{\varphi \in S_n} (\operatorname{sgn} \varphi \cdot a_{\varphi(1)1} \cdots (\lambda \cdot u_{\varphi(k)}) \cdots a_{\varphi(n)n}) \\
&= \lambda \cdot \sum_{\varphi \in S_n} (\operatorname{sgn} \varphi \cdot a_{\varphi(1)1} \cdots u_{\varphi(k)} \cdots a_{\varphi(n)n}) \\
&= \lambda \cdot \det(\mathbf{A}[k : \vec{u}])
\end{aligned}$$

th\_det5

**Lause 34** Tarkastellaan  $n \times n$ -matriisia  $\mathbf{A}$ .

(Sanallinen muoto:) Jos neliömatriisin  $\mathbf{A}$  sarake kerrotaan reaalityylillä  $\lambda$  ja näin saatu sarake lisätään toiseen  $\mathbf{A}$ :n sarakkeeseen, niin syntyvän uuden matriisin determinantti on sama, kuin alkuperäisen matriisin determinantti.

(Kaavana:)

$$j \neq k \quad \Rightarrow \quad \det(\mathbf{A}[k : \vec{a}_{\bullet k} + \lambda \vec{a}_{\bullet j}]) = \det(\mathbf{A})$$

(Esimerkki:)

$$\begin{vmatrix} 2 & 1 & 3 \\ -5 & 1 & 4 \\ 0 & 0 & 7 \end{vmatrix} = \begin{vmatrix} 7 & 1 & 3 \\ 0 & 1 & 4 \\ 0 & 0 & 7 \end{vmatrix} = \det \mathbf{A}[1 : \vec{a}_{\bullet 1} + 5 \cdot \vec{a}_{\bullet 2}]$$

**Todistus:** Kahden edeltävän lauseen mukaan

$$\det(\mathbf{A}[k : \vec{a}_{\bullet k} + \lambda \vec{a}_{\bullet j}]) = \underbrace{\det(\mathbf{A}[k : \vec{a}_{\bullet k}])}_{=\det(\mathbf{A})} + \underbrace{\lambda \cdot \det(\mathbf{A}[k : \vec{a}_{\bullet j}])}_{=0} = \det(\mathbf{A})$$

□

th\_det6

**Lause 35** Tarkastellaan  $n \times n$ -matriisia  $\mathbf{A}$ .

(Sanallinen muoto:) Jos jokin neliömatriisin  $\mathbf{A}$  sarake voidaan lausua muiden sarakkeiden lineaarikombinaationa, niin matriisin determinantti on nolla.

(Esimerkki:)

$$\begin{vmatrix} 2 & 1 & 4 \\ -5 & 1 & -3 \\ 0 & 1 & 2 \end{vmatrix} = 0, \quad \text{sillä } \vec{a}_{\bullet 3} = \vec{a}_{\bullet 1} + 2 \cdot \vec{a}_{\bullet 2}$$

**Todistus:** Jos  $\vec{a}_{\bullet n} = \sum_{k=1}^{n-1} c_k \cdot \vec{a}_{\bullet k}$ , niin edellisen lauseen toistuva soveltaminen johtaa tulokseen

$$\det(\mathbf{A}) = \det(\mathbf{A}[n : \vec{a}_{\bullet n} - \sum_{k=1}^{n-1} c_k \cdot \vec{a}_{\bullet k}]) = \det(\mathbf{A}[n : \vec{0}]) = 0$$

□

th\_det6b

**Lause 36** Tarkastellaan  $n \times n$ -matriisia  $\mathbf{A}$ .

(Sanallinen muoto:) Neliömatriisilla  $\mathbf{A}$  ja sen transpoosilla  $\mathbf{A}^T$  on samat determinantit.

(Kaavana:)

$$\det(\mathbf{A}^T) = \det(\mathbf{A})$$

(Esimerkki:)

$$\begin{vmatrix} 2 & -5 & 0 \\ 1 & 1 & 0 \\ 3 & 4 & 7 \end{vmatrix} = \begin{vmatrix} 2 & 1 & 3 \\ -5 & 1 & 4 \\ 0 & 0 & 7 \end{vmatrix}$$

**Todistus:** Olkoon  $\psi = \phi^{-1}$ . Silloin  $\text{sgn} \psi = \text{sgn} \phi$  ja

$$a_{\psi(1)1} a_{\psi(2)2} \cdots a_{\psi(n)n} = a_{1\phi(1)} a_{2\phi(2)} \cdots a_{n\phi(n)}$$

sillä tekijät ovat samat vaikka ne ovatkin eri järjestyksessä. Nyt

$$\begin{aligned} \det(\mathbf{A}^T) &= \sum_{\phi \in S_n} \text{sgn} \phi \cdot (\mathbf{A}^T)_{\phi(1)1} (\mathbf{A}^T)_{\phi(2)2} \cdots (\mathbf{A}^T)_{\phi(n)n} \\ &= \sum_{\phi \in S_n} \text{sgn} \phi \cdot a_{1\phi(1)} a_{2\phi(2)} \cdots a_{n\phi(n)} \\ &= \sum_{\psi \in S_n} \text{sgn} \psi \cdot a_{\psi(1)1} a_{\psi(2)2} \cdots a_{\psi(n)n} \\ &= \det(\mathbf{A}) \end{aligned}$$

□

th\_det7

**Lause 37** Lauseissa ~~(30), (31), (32), (33), (34) ja (35)~~ <sup>th\_det1 th\_det2 th\_det3 th\_det4 th\_det5 th\_det6</sup> matriisin sarakkeita koskevat tulokset ovat voimassa myös matriisin riveille.

**Todistus:** Seuraa edellisestä lauseesta.

□

th\_det8

**Lause 38** Tarkastellaan  $n \times n$ -matriiseja  $\mathbf{A}$  ja  $\mathbf{B}$ .

(Sanallinen muoto:) Tulon determinantti on determinanttien tulo.

(Kaavana:)

$$\det(\mathbf{AB}) = \det(\mathbf{A}) \det(\mathbf{B})$$

**Todistus:** Olkoon  $\mathbf{B} = (b_{ij})$  ja  $\mathbf{A} = (a_{ij}) = (\vec{a}_{\bullet 1} \quad \vec{a}_{\bullet 2} \quad \cdots \quad \vec{a}_{\bullet n})$ . Tulon  $\mathbf{AB}$   $j$ :s sarake on silloin

$$\begin{aligned} (\mathbf{AB})_{\bullet j} &= \begin{pmatrix} (\mathbf{AB})_{1j} \\ (\mathbf{AB})_{2j} \\ \vdots \\ (\mathbf{AB})_{nj} \end{pmatrix} = \begin{pmatrix} a_{11}b_{1j} + a_{12}b_{2j} + \cdots + a_{1n}b_{nj} \\ a_{21}b_{1j} + a_{22}b_{2j} + \cdots + a_{2n}b_{nj} \\ \vdots \\ a_{n1}b_{1j} + a_{n2}b_{2j} + \cdots + a_{nn}b_{nj} \end{pmatrix} \\ &= b_{1j} \begin{pmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{n1} \end{pmatrix} + b_{2j} \begin{pmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{n2} \end{pmatrix} + \cdots + b_{nj} \begin{pmatrix} a_{1n} \\ a_{2n} \\ \vdots \\ a_{nn} \end{pmatrix} \\ &= \sum_{k=1}^n b_{kj} \vec{a}_{\bullet k} \end{aligned}$$

Siis

$$\mathbf{AB} = \left( \sum_{k_1=1}^n b_{k_1 1} \vec{a}_{\bullet k_1} \quad \sum_{k_2=1}^n b_{k_2 2} \vec{a}_{\bullet k_2} \quad \cdots \quad \sum_{k_n=1}^n b_{k_n n} \vec{a}_{\bullet k_n} \right)$$

Soveltamalla toistuvasti lausetta (33)<sup>th det 4</sup> saamme

$$\det(\mathbf{AB}) = \sum_{k_1=1}^n \sum_{k_2=1}^n \cdots \sum_{k_n=1}^n b_{k_1 1} b_{k_2 2} \cdots b_{k_n n} \det(\vec{a}_{\bullet k_1} \quad \vec{a}_{\bullet k_2} \quad \cdots \quad \vec{a}_{\bullet k_n})$$

Tässä summassa on  $n^n$  termiä, mutta niistä vain ne ovat nollasta poikkeavia, joissa luvut  $k_1, k_2, \dots, k_n$  ovat pareittain eri suuria eli muodostavat permutaation. Lauseen (32)<sup>th det 3</sup> mukaan nimittäin kaavion determinantti on nolla, jos sama sarake esiintyy kaaviossa kahdesti tai useammin. Kun luvut  $k_1, k_2, \dots, k_n$  muodostavat permutaation  $\varphi(j) = k_j$ , niin kaavio

$$(\vec{a}_{\bullet \varphi(1)} \quad \vec{a}_{\bullet \varphi(2)} \quad \cdots \quad \vec{a}_{\bullet \varphi(n)})$$

saadaan kaviosta

$$(\vec{a}_{\bullet 1} \quad \vec{a}_{\bullet 2} \quad \cdots \quad \vec{a}_{\bullet n}) = \mathbf{A}$$

suorittamalla  $p_\varphi$  sarakkeiden vaihtoa, joten

$$\det(\vec{a}_{\bullet \varphi(1)} \quad \vec{a}_{\bullet \varphi(2)} \quad \cdots \quad \vec{a}_{\bullet \varphi(n)}) = \operatorname{sgn} \varphi \cdot \det(\mathbf{A}).$$

Siis

$$\begin{aligned} \det(\mathbf{AB}) &= \sum_{\varphi \in S_n} b_{\varphi(1)1} b_{\varphi(2)2} \cdots b_{\varphi(n)n} \det(\vec{a}_{\bullet \varphi(1)} \quad \vec{a}_{\bullet \varphi(2)} \quad \cdots \quad \vec{a}_{\bullet \varphi(n)}) \\ &= \det(\mathbf{A}) \sum_{\varphi \in S_n} \operatorname{sgn} \varphi \cdot b_{\varphi(1)1} b_{\varphi(2)2} \cdots b_{\varphi(n)n} \\ &= \det \mathbf{A} \det(\mathbf{B}) \end{aligned}$$

□



th\_det9

**Lause 39** (Sanallinen muoto:) Yksikkömatriisin determinantti on 1.  
(Kaavana:)

$$\det(\mathbf{I}) = 1$$

**Todistus:**

$$\det(\mathbf{I}) = \sum_{\varphi \in S_n} \operatorname{sgn} \varphi \cdot \delta_{\varphi(1)1} \cdots \delta_{\varphi(n)n} = 1$$

sillä ainoa permutaatio, jolla kaikki Kroneckerin deltat ovat ykkösiä, on  $\varphi = (1, 2, \dots, n)$  ja sen merkki on 1.  $\square$

th\_det10

**Lause 40** Tarkastellaan  $n \times n$ -matriisia  $\mathbf{A}$ .  
(Sanallinen muoto:) Matriisi on säännöllinen (eli sille on olemassa käänteismatriisi), jos ja vain jos matriisin determinantti ei ole nolla.  
(Kaavana:)

$$\exists \mathbf{A}^{-1} \Leftrightarrow \det(\mathbf{A}) \neq 0$$

**Todistus:**  $\Rightarrow$  (eli  $\exists \mathbf{A}^{-1} \Rightarrow \det(\mathbf{A}) \neq 0$ )

Oletamme nyt, että käänteismatriisi on olemassa. Määritetään käänteismatriisi riviopeeraatioiden avulla Gaussin algoritmeilla. Aluksi laadimme kaavion, jonka vasen puoli on  $\mathbf{A}$ . Jokaisessa vaiheessa algoritmin aikana kaavion vasemman osan determinantti joko, säilyy muuttumattomana, vaihtaa merkkiä tai tulee kerrotuksi nolasta eroavalla luvulla. Viimeisen kaavion vasemman puolen determinantti on 1. Siis ensimmäisen kaavion vasemman osan determinantti ei voi olla nolla.

$\Leftarrow$  (eli  $\nexists \mathbf{A}^{-1} \Rightarrow \det(\mathbf{A}) = 0$ )

Oletamme nyt, että käänteismatriisia ei ole olemassa. Aloitetaan käänteismatriisin määrittäminen kuten edellä. Oletuksen mukaan Gaussin algoritmi ei onnistu, vaan pivotointi epäonnistuu jossakin vaiheessa. Tämä merkitsee, että viimeisessä kaaviossa on rivi, jonka vasen osa on täynnä nollia. Siis jokin  $\mathbf{A}$ :n rivi voidaan lausua muiden lineaarikombinaationa ja lauseiden (35) ja (37) mukaan determinantti on nolla.  $\square$

th\_det11

**Lause 41** Jos  $n \times n$ -matriisia  $\mathbf{A}$  on säännöllinen, niin

$$\det(\mathbf{A}^{-1}) = \frac{1}{\det(\mathbf{A})}$$

**Todistus:** Harjoitustehtävä.

□

Edellä oli suuri joukko lauseita. Tehdään vielä yhteenveto:

- (lause 30 sivu 63) Jos neliömatriisin  $A$  jonkin sarakkeen kaikki alkiot ovat nollia, niin matriisin determinanti on nolla.
- (lause 31 sivu 63) Jos neliömatriisin  $A$  kaksi saraketta vaihtavat paikkaa, niin syntyvän matriisin determinanti on alkuperäisen vastaluku.
- (lause 32 sivu 64) Jos neliömatriisissa  $A$  on kaksi identtistä saraketta, niin matriisin determinanti on 0.
- (lause 33 sivu 64) Neliömatriisin  $A$  determinantin lasku on  $k$ :nnen sarakkeen suhteen lineaarinen operaatio.
- (lause 34 sivu 65) Jos neliömatriisin  $A$  sarake kerrotaan reaaliluvulla  $\lambda$  ja näin saatu sarake lisätään toiseen  $A$ :n sarakkeeseen, niin syntyvän uuden matriisin determinanti on sama, kuin alkuperäisen matriisin determinanti.
- (lause 35 sivu 65) Jos jokin neliömatriisin  $A$  sarake voidaan lausua muiden sarakkeiden lineaarikombinaationa, niin matriisin determinanti on nolla.
- (lause 36 sivu 66) Neliömatriisilla  $A$  ja sen transpoosilla  $A^T$  on samat determinantit.
- (lause 37 sivu 66) Lauseissa (30), (31), (32), (33), (34) ja (35) matriisin sarakkeita koskevat tulokset ovat voimassa myös matriisin riveille.
- (lause 38 sivu 66)  $\det(AB) = \det(A) \det(B)$
- (lause 39 sivu 68)  $\det(I) = 1$
- (lause 40 sivu 68) Matriisi on säännöllinen (eli sille on olemassa käänteismatriisi), jos ja vain jos matriisin determinanti ei ole nolla.
- (lause 41 sivu 68) Jos  $n \times n$ -matriisia  $A$  on säännöllinen, niin

$$\det(A^{-1}) = \frac{1}{\det(A)}$$

## 5.2 Minorikehitelmän perustelu

ec:MinoriLasku

Opetusmonisteessa olemme oppineet minorikehitelmän. Seuraavassa perustelemme menetelmän.

## 5.2.1 Determinantin laskeminen minorikehitelmällä

**Määritelmä 42** Matriisin  $\mathbf{A}$  alimatriisi  $\mathbf{A}_{ij}$  saadaan poistamalla  $\mathbf{A}$ :sta  $i$ :s rivi ja  $j$ :s sarake. Alimatriisin determinantti  $\det(\mathbf{A}_{ij})$  on matriisin  $\mathbf{A}$  alkioon  $a_{ij}$  liittyvä minori (alideterminantti). Lukua  $(-1)^{i+j} \det(\mathbf{A}_{ij})$  sanotaan matriisin  $\mathbf{A}$  alkioon  $a_{ij}$  liittyväksi kofaktoriksi.

**Esimerkki 43** Jos

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 3 & 2 \\ 3 & 4 & 7 & 1 \\ -2 & 4 & 5 & 1 \\ 0 & 1 & 8 & 2 \end{pmatrix}$$

niin

$$\mathbf{A}_{23} = \left( \begin{array}{ccc|c} 1 & 0 & 3 & 2 \\ 3 & 4 & 7 & 1 \\ -2 & 4 & 5 & 1 \\ 0 & 1 & 8 & 2 \end{array} \right) = \begin{pmatrix} 1 & 0 & 2 \\ -2 & 4 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

Vastaavat alkioon  $a_{23}$  liittyvät minori ja kofaktori ovat

$$\begin{aligned} \text{minori} &= \det(\mathbf{A}_{23}) = 3 \\ \text{kofaktori} &= (-1)^{2+3} \det(\mathbf{A}_{23}) = -3 \end{aligned}$$

Seuraavien tarkastelujen aikana käytetään kahta merkintää, joita on jo käytetty aiemminkin, mutta kertaamme ne ensin:

(1) Käytämme merkintää  $\vec{e}_k$  pystyvektorille, jonka  $k$ :s koordinaatti on 1 ja muut koordinaatit ovat nollia (siis  $e_j = \delta_{jk}$ ).

(2) Käytämme merkintää  $\mathbf{A}[k : \vec{b}]$  matriisista, joka saadaan kun  $k$ :s sarake matriisissa  $\mathbf{A}$  korvataan vektorilla  $\vec{b}$ .

**Lause 44**  $n \times n$ -neliömatriisille  $\mathbf{A}$  on voimassa

$$\det(\mathbf{A}[n : \vec{e}_n]) = \det(\mathbf{A}_{nn})$$

eli

$$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1(n-1)} & 0 \\ a_{21} & a_{22} & \cdots & a_{2(n-1)} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{(n-1)1} & a_{(n-1)2} & \cdots & a_{(n-1)(n-1)} & 0 \\ a_{n1} & a_{n2} & \cdots & a_{n(n-1)} & 1 \end{vmatrix} = \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1(n-1)} \\ a_{21} & a_{22} & \cdots & a_{2(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ a_{(n-1)1} & a_{(n-1)2} & \cdots & a_{(n-1)(n-1)} \end{vmatrix}$$

**Todistus:** Jos  $n$ -permutaatiolle  $\varphi \in S_n$  on voimassa, että  $\varphi(n) = n$ , niin sen alkuosa indusoi  $(n-1)$ -permutaation  $\hat{\varphi} \in S_{n-1}$ ,  $\hat{\varphi}(k) = \varphi(k)$ ,  $0 \leq k < n$  ja  $\text{sgn } \hat{\varphi} = \text{sgn } \varphi$ . Siis

$$\begin{aligned} \det(\mathbf{A}[n : \vec{e}_n]) &= \sum_{\varphi \in S_n} \text{sgn } \varphi \cdot a_{\varphi(1)1} \cdot a_{\varphi(2)2} \cdots a_{\varphi(n-1)(n-1)} \cdot \delta_{\varphi(n)n} \\ &= \sum_{\hat{\varphi} \in S_{n-1}} \text{sgn } \hat{\varphi} \cdot a_{\varphi(1)1} \cdot a_{\varphi(2)2} \cdots a_{\varphi(n-1)(n-1)} \\ &= \det(\mathbf{A}_{nn}) \end{aligned}$$

□

**Lause 45**  $n \times n$ -neliömatriisille  $\mathbf{A}$  on voimassa

$$\det(\mathbf{A}[j : \vec{e}_i]) = (-1)^{i+j} \det(\mathbf{A}_{ij})$$

eli

$$\begin{vmatrix} a_{11} & \cdots & a_{1,j-1} & 0 & a_{1,j+1} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{i-1,1} & \cdots & a_{i-1,j-1} & 0 & a_{i-1,j+1} & \cdots & a_{i-1,n} \\ a_{i1} & \cdots & a_{i,j-1} & 1 & a_{i,j+1} & \cdots & a_{in} \\ a_{i+1,1} & \cdots & a_{i+1,j-1} & 0 & a_{i+1,j+1} & \cdots & a_{i+1,n} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{n,j-1} & 0 & a_{n,j+1} & \cdots & a_{nn} \end{vmatrix} = (-1)^{i+j} \det(\mathbf{A}_{ij})$$

**Todistus:** Determinanttikaaviota muutetaan siten, että vaihdetaan ensin rivit  $i$  ja  $(i+1)$  keskenään, sitten rivit  $(i+1)$  ja  $(i+2)$  keskenään jne. kunnes ykkönen on saatu alimmalle riville. Sen jälkeen vaihdetaan ensin sarakkeet  $j$  ja  $(j+1)$  keskenään, sitten sarakkeet  $(j+1)$  ja  $(j+2)$ , kunnes ykkönen on oikeassa alanurkassa. Vaihtoja tehdään

$(n-i) + (n-j)$  kappaletta. Jotta arvo olisi oikea, tulee viimeinen determinanttikaavio kertoa luvulla  $(-1)^{2n-i-j} = (-1)^{i+j}$ . Saadun kaavion determinantti on edellisen lauseen nojalla

$$\begin{aligned}
 & (-1)^{i+j} \begin{vmatrix} a_{11} & \cdots & a_{1,j-1} & a_{1,j+1} & \cdots & a_{1n} & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{i-1,1} & \cdots & a_{i-1,j-1} & a_{i-1,j+1} & \cdots & a_{i-1,n} & 0 \\ a_{i+1,1} & \cdots & a_{i+1,j-1} & a_{i+1,j+1} & \cdots & a_{i+1,n} & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{n,j-1} & a_{n,j+1} & \cdots & a_{nn} & 0 \\ a_{i1} & \cdots & a_{i,j-1} & a_{i,j+1} & \cdots & a_{in} & 1 \end{vmatrix} \\
 &= (-1)^{i+j} \begin{vmatrix} a_{11} & \cdots & a_{1,j-1} & a_{1,j+1} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots \\ a_{i-1,1} & \cdots & a_{i-1,j-1} & a_{i-1,j+1} & \cdots & a_{i-1,n} \\ a_{i+1,1} & \cdots & a_{i+1,j-1} & a_{i+1,j+1} & \cdots & a_{i+1,n} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots \\ a_{n1} & \cdots & a_{n,j-1} & a_{n,j+1} & \cdots & a_{nn} \end{vmatrix} = (-1)^{i+j} \det(\mathbf{A}_{ij})
 \end{aligned}$$

□

**Lause 46**  $n \times n$ -neliömatriisin  $\mathbf{A}$  determinantti saadaan laskettua kaavoilla

$$\det(\mathbf{A}) = \sum_{j=1}^n (-1)^{i+j} a_{ij} \det(\mathbf{A}_{ij}) \quad \text{kehitemä } i\text{:nnen rivin suhteen}$$

$$\det(\mathbf{A}) = \sum_{i=1}^n (-1)^{i+j} a_{ij} \det(\mathbf{A}_{ij}) \quad \text{kehitemä } j\text{:nnen sarakkeen suhteen}$$

**Todistus:** Perustellaan ensin jälkimmäinen kaava. Determinantin lineaarisuudesta sarakkeen suhteen seuraa, että

$$\det(\mathbf{A}) = \det(\mathbf{A}[j : \sum_{i=1}^n a_{ij} \vec{e}_i]) = \sum_{i=1}^n a_{ij} \det(\mathbf{A}[j : \vec{e}_i]) = \sum_{i=1}^n a_{ij} (-1)^{i+j} \det(\mathbf{A}_{ij}).$$

Toisen kaavan soveltaminen transpoosiin johtaa ensimmäiseen kaavaan. □

Determinantin kehittäminen kannattaa perustaa riviin tai sarakkeeseen, jolla on mahdollisimman monta nollaa.

**Esimerkki 47** (1) Kehitetaan determinantti ensimmäisen rivin suhteen

$$\begin{aligned}
 & \begin{vmatrix} \mathbf{1} & \mathbf{3} & \mathbf{-1} \\ 1 & 5 & 0 \\ -2 & 2 & -3 \end{vmatrix} \\
 &= (-1)^{1+1} \cdot 1 \cdot \begin{vmatrix} 5 & 0 \\ 2 & -3 \end{vmatrix} + (-1)^{1+2} \cdot 3 \cdot \begin{vmatrix} 1 & 0 \\ -2 & -3 \end{vmatrix} + (-1)^{1+3} \cdot (-1) \cdot \begin{vmatrix} 1 & 5 \\ -2 & 2 \end{vmatrix} \\
 &= 1 \cdot (-15 - 0) + (-1) \cdot 3 \cdot (-3 - 0) + (-1) \cdot (2 + 10) \\
 &= -18
 \end{aligned}$$

(2) Kehitetaan sama determinantti toisen rivin suhteen

$$\begin{aligned}
 & \begin{vmatrix} 1 & 3 & -1 \\ \mathbf{1} & \mathbf{5} & \mathbf{0} \\ -2 & 2 & -3 \end{vmatrix} \\
 &= (-1)^{2+1} \cdot 1 \cdot \begin{vmatrix} 3 & -1 \\ 2 & -3 \end{vmatrix} + (-1)^{2+2} \cdot 5 \cdot \begin{vmatrix} 1 & -1 \\ -2 & -3 \end{vmatrix} + (-1)^{2+3} \cdot 0 \cdot \begin{vmatrix} 1 & 3 \\ -2 & 2 \end{vmatrix} \\
 &= (-1) \cdot (-9 + 2) + 5 \cdot (-3 - 2) - 0 \\
 &= -18
 \end{aligned}$$

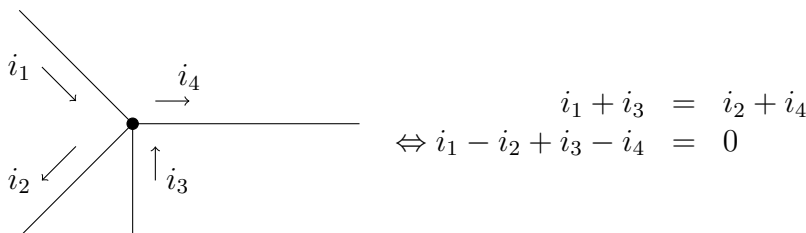
Käytännön laskemista helpottaa se, että kofaktoreiden merkkikaavio on varsin yksinkertainen:

$$\begin{pmatrix} + & - & + & - & + & \cdots \\ - & + & - & + & - & \cdots \\ + & - & + & - & + & \cdots \\ - & + & - & + & - & \cdots \\ + & - & + & - & + & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

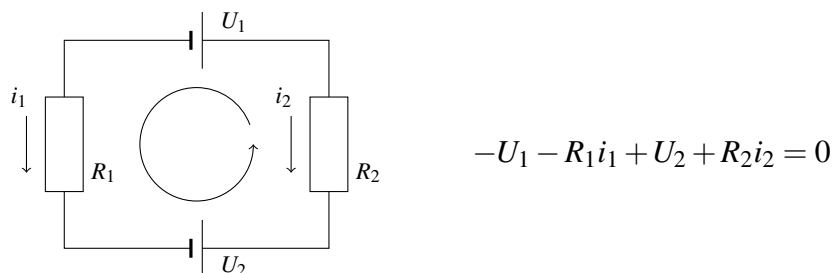
## 5.3 Piirianalyysi-esimerkki

Sec:Circuit

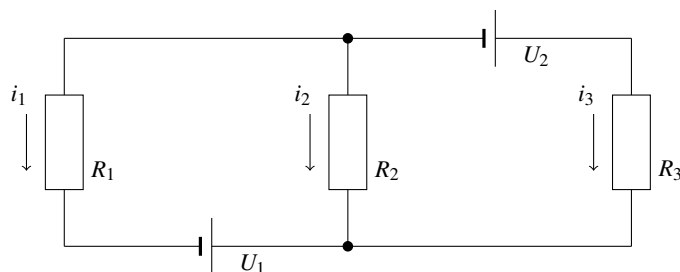
**Lause 48 Kirchhoffin ensimmäinen laki:** Missä tahansa virtapiirin liitoskohdassa pisteseen saapuvien sähkövirtojen summa on yhtä suuri kuin siitä poistuvien sähkövirtojen summa.



**Lause 49 Kirchhoffin toinen laki:** Suljetussa johdinsilmukassa järjestyksessä laskettu potentiaalierojen summa on nolla.



**Esimerkki 50** Laske virrat, kun oheisessa kytkennässä  $U_1 = 2.0\text{V}$ ,  $U_2 = 5.0\text{V}$ ,  $R_1 = 200\Omega$ ,  $R_2 = 500\Omega$  ja  $R_3 = 300\Omega$ .



Kirchhoffin ensimmäisestä laista saamme

$$i_1 + i_2 + i_3 = 0.$$

Kirchhoffin toisesta laista saamme

$$-U_2 - R_2 i_2 + R_3 i_3 = 0$$

$$-R_1 i_1 + U_1 + R_2 i_2 = 0$$

Sijoitetaan parametrien arvot  $U_1 = 2.0\text{V}$ ,  $U_2 = 5.0\text{V}$ ,  $R_1 = 200\Omega$ ,  $R_2 = 500\Omega$  ja  $R_3 = 300\Omega$  yhtälöihin

$$i_1 + i_2 + i_3 = 0,$$

$$-U_2 - R_2 i_2 + R_3 i_3 = 0,$$

$$-R_1 i_1 + U_1 + R_2 i_2 = 0,$$

ja järjestetään yhtälöt sieviksi:

$$\begin{cases} i_1 + i_2 + i_3 = 0 \\ -500\Omega i_2 + 300\Omega i_3 = 5.0\text{V} \\ -200\Omega i_1 + 500\Omega i_2 = -2.0\text{V} \end{cases}$$

Jaetaan vielä toinen ja kolmas yhtälö ohmilla, jolloin yhtälöryhmä menee muotoon:

$$\begin{cases} i_1 + i_2 + i_3 = 0 \\ -500i_2 + 300i_3 = 5.0\text{A} \\ -200i_1 + 500i_2 = -2.0\text{A} \end{cases}$$

$$D = \begin{vmatrix} 1 & 1 & 1 \\ 0 & -500 & 300 \\ -200 & 500 & 0 \end{vmatrix} = -310000$$

$$\begin{aligned} D_1 &= \begin{vmatrix} 0 & 1 & 1 \\ 5.0\text{A} & -500 & 300 \\ -2.0\text{A} & 500 & 0 \end{vmatrix} \\ &= +0 - 5.0\text{A} \cdot \begin{vmatrix} 1 & 1 \\ 500 & 0 \end{vmatrix} + (-2.0\text{A}) \cdot \begin{vmatrix} 1 & 1 \\ -500 & 300 \end{vmatrix} = 900\text{A} \end{aligned}$$

$$\begin{aligned} D_2 &= \begin{vmatrix} 1 & 0 & 1 \\ 0 & 5.0\text{A} & 300 \\ -200 & -2.0\text{A} & 0 \end{vmatrix} \\ &= -0 + 5.0\text{A} \cdot \begin{vmatrix} 1 & 1 \\ -200 & 0 \end{vmatrix} - (-2.0\text{A}) \cdot \begin{vmatrix} 1 & 1 \\ 0 & 300 \end{vmatrix} = 1600\text{A} \end{aligned}$$

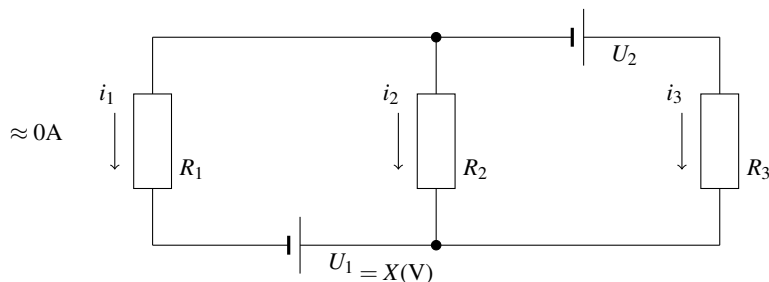
$$\begin{aligned} D_3 &= \begin{vmatrix} 1 & 1 & 0 \\ 0 & -500 & 5.0\text{A} \\ -200 & 500 & -2.0\text{A} \end{vmatrix} \\ &= +0 - 5.0\text{A} \cdot \begin{vmatrix} 1 & 1 \\ -200 & 500 \end{vmatrix} + (-2.0\text{A}) \cdot \begin{vmatrix} 1 & 1 \\ 0 & -500 \end{vmatrix} = -2500\text{A} \end{aligned}$$

Siis:

$$\begin{aligned} i_1 &= \frac{D_1}{D} = \frac{900\text{A}}{-310000} = -2.90\text{mA} \\ i_2 &= \frac{D_2}{D} = \frac{1600\text{A}}{-310000} = -5.16\text{mA} \\ i_3 &= \frac{D_3}{D} = \frac{-2500\text{A}}{-310000} = 8.06\text{mA} \end{aligned}$$

**Esimerkki 51** Edellisen esimerkin kytkennässä etsimme jännitelähteelle  $U_1$  sellainen arvo, että virta vastuksen  $R_1$  läpi on nolla (pieni). Merkitään  $U_1 = X$ . Tämä merkintä korostaa sitä, että olemme nyt etsimässä oikeata arvoa  $X$ :lle.





Yhtälöryhmä muuttuu muotoon

$$\begin{cases} i_1 + i_2 + i_3 = 0 \\ -500i_2 + 300i_3 = 5.0A \\ -200i_1 + 500i_2 = -XA \end{cases}$$

Ratkaisemme vain ensimmäisen muuttujan  $i_1 = D_1/D$ :n (muuta emme nyt tarvitse).  $D_1$  pitää laskea uudelleen, mutta  $D$  on sama kuin edellisessä esimerkissä.

$$\begin{aligned} D_1 &= \begin{vmatrix} 0 & 1 & 1 \\ 5.0A & -500 & 300 \\ -XA & 500 & 0 \end{vmatrix} \\ &= +0 - 5.0A \cdot \begin{vmatrix} 1 & 1 \\ 500 & 0 \end{vmatrix} + (-XA) \cdot \begin{vmatrix} 1 & 1 \\ -500 & 300 \end{vmatrix} \\ &= (2500 - 800X)A \end{aligned}$$

Siis

$$i_1 = \frac{D_1}{D} = \frac{(2500 - 800X)A}{-310000},$$

ja  $i_1 = 0A$ , jos

$$\begin{aligned} 2500 - 800X &= 0 \\ \Leftrightarrow X &= \frac{2500}{800} = 3.125(V) \end{aligned}$$

## 5.4 Matriisilaskut Excel-ohjelmalla

Sec:MatExcel

Excel on taulukkolaskenta -ohjelmisto, jonka vahvinta sovellusalueita on kirjanpidon ja budjetoinnin kaltaiset rahalaskut. Toisaalta kaikki laskeminen, johon liittyy numero- taulukoita tai -kaavioita soveltuu Excelillä tehtäväksi. Excelissä on useita tilastollisia

funktioita valmiina. Tässä lyhyessä esityksessä käydään läpi vain vektoreihin ja matriiseihin liittyviä toimintoja

**Kaavan sijoittaminen soluun.** Excel-työpohja (Worksheet) on suuri taulukko, jossa sarakkeet nimetään kirjaimin (A, B, C, ...) ja rivit nimetään numeroin (1, 2, 3, ...). Soluihin voidaan tallettaa lukuja, merkkijonoja ja kaavoja.

Olkoon työpohja seuraavan näköinen

|   | A | B | C  | D | E |
|---|---|---|----|---|---|
| 1 |   |   |    |   |   |
| 2 |   |   | 25 |   |   |
| 3 |   |   | 10 |   |   |
| 4 |   |   |    |   |   |
| 5 |   |   |    |   |   |

Siis solun C2 on sijoitettu luku 25, ja soluun C3 on sijoitettu luku 10 ja muut solut ovat tyhjiä. Soluun C4 sijoitetaan kaava `=C2+C3`. (Huomaa: kaava alkaa =-merkillä!) Kaavan tultua valmiiksi painetaan [Enter]-näppäintä, jolloin ohjelma laskee lausekkeen C2+C3 arvon ja sijoittaa sen soluun C4.

|   | A | B | C  | D | E |
|---|---|---|----|---|---|
| 1 |   |   |    |   |   |
| 2 |   |   | 25 |   |   |
| 3 |   |   | 10 |   |   |
| 4 |   |   | 35 |   |   |
| 5 |   |   |    |   |   |

Kun myöhemmin vie solukohdistimen soluun C4 ja painaa [F2]-näppäintä, pääsee tilaan, jossa kaavaa voi editoida.

|   | A | B | C      | D | E |
|---|---|---|--------|---|---|
| 1 |   |   |        |   |   |
| 2 |   |   | 25     |   |   |
| 3 |   |   | 10     |   |   |
| 4 |   |   | =C2+C3 |   |   |
| 5 |   |   |        |   |   |

**Riviooperaatiot.** Seuraavassa laskentapohjassa on yhtälöryhmän kerroinkaavio

|   | A | B | C | D | E | F  |
|---|---|---|---|---|---|----|
| 1 |   | 1 | 2 | 3 |   | -2 |
| 2 |   | 2 | 5 | 8 |   | <– |
| 3 |   |   |   |   |   |    |
| 4 |   |   |   |   |   |    |
| 5 |   |   |   |   |   |    |
| 6 |   |   |   |   |   |    |

Kopioidaan ensin pivot-rivi B1:D1 paikkaan B4:D4 copy-paste -toiminnoilla. Toinen tapa on toimia seuraavasti:

(1) Maalaa alue B4:D4

(2) Kirjoita kaava:

(3) Näppäile: [Ctrl]-[Shift]-[Enter]

Varsinainen rivi-operaatio saadaan aikaan seuraavasti:

(1) Maalaa alue B5:D5

(2) Kirjoita kaava:

(3) Näppäile: [Ctrl]-[Shift]-[Enter]

Näiden näppäilyjen jälkeen laskentapohja on seuraavan näköinen

|   | A | B | C | D | E | F  |
|---|---|---|---|---|---|----|
| 1 |   | 1 | 2 | 3 |   | -2 |
| 2 |   | 2 | 5 | 8 |   | <– |
| 3 |   |   |   |   |   |    |
| 4 |   | 1 | 2 | 3 |   |    |
| 5 |   | 0 | 1 | 2 |   |    |
| 6 |   |   |   |   |   |    |

Huomautus: Kun sijoitetaan kaava yhteen soluun, riittää näppäillä lopuksi [Enter]. Kun kaavan antama tulos on vektori tai matriisi, joka sijoitetaan monesta solusta muodostuvaan alueeseen, pitää lopuksi näppäillä [Ctrl]-[Shift]-[Enter]. (Paina ensin [Ctrl] ja [Shift] pohjaan ja pidä ne pohjassa samalla kun painat [Enter]-näppäintä.)

**Kahden matriisin summa.** Seuraavassa laskentapohjassa on kaksi  $2 \times 3$ -matriisia; alueessa B1:D2, ja alueessa F1:H3. Näiden matriisien summa on myös  $2 \times 3$ -matriisi.

|   | A | B | C  | D  | E | F | G | H  | I |
|---|---|---|----|----|---|---|---|----|---|
| 1 |   | 1 | -1 | -1 |   | 1 | 4 | 3  |   |
| 2 |   | 2 | 1  | 0  |   | 2 | 0 | -2 |   |
| 3 |   |   |    |    |   |   |   |    |   |
| 4 |   |   |    |    |   |   |   |    |   |
| 5 |   |   |    |    |   |   |   |    |   |
| 6 |   |   |    |    |   |   |   |    |   |

Lasketaan summa-matriisi ja sijoitetaan se paikkaan B4:D5 seuraavasti:

(1) Maalaa alue B4:D5

(2) Kirjoita kaava: `=B1:D2+F1:H2`

(3) Näppäile: [Ctrl]-[Shift]-[Enter]

Näiden näppäilyjen jälkeen laskentapohja on seuraavan näköinen

|   | A | B | C  | D  | E | F | G | H  | I |
|---|---|---|----|----|---|---|---|----|---|
| 1 |   | 1 | -1 | -1 |   | 1 | 4 | 3  |   |
| 2 |   | 2 | 1  | 0  |   | 2 | 0 | -2 |   |
| 3 |   |   |    |    |   |   |   |    |   |
| 4 |   | 2 | 3  | 2  |   |   |   |    |   |
| 5 |   | 4 | 1  | -2 |   |   |   |    |   |
| 6 |   |   |    |    |   |   |   |    |   |

**Matriisitulo MMULT.** Seuraavassa laskentapohjassa on kaksi matriisia;  $2 \times 3$ -matriisi alueessa B1:D2, ja  $3 \times 3$ -matriisi alueessa F1:H3. Näiden matriisien tulo on  $2 \times 3$ -matriisi.

|   | A | B | C  | D  | E | F | G | H  | I |
|---|---|---|----|----|---|---|---|----|---|
| 1 |   | 1 | -1 | -1 |   | 1 | 4 | 3  |   |
| 2 |   | 2 | 1  | 0  |   | 2 | 0 | -2 |   |
| 3 |   |   |    |    |   | 0 | 1 | 5  |   |
| 4 |   |   |    |    |   |   |   |    |   |
| 5 |   |   |    |    |   |   |   |    |   |
| 6 |   |   |    |    |   |   |   |    |   |

Lasketaan tulomatriisi ja sijoitetaan se paikkaan B4:D5 seuraavasti:

(1) Maalaa alue B4:D5

(2) Kirjoita kaava: `=MMULT(B1:D2;F1:H3)`

(3) Näppäile: [Ctrl]-[Shift]-[Enter]

Näiden näppäilyjen jälkeen laskentapohja on seuraavan näköinen

|   | A | B  | C  | D  | E | F | G | H  | I |
|---|---|----|----|----|---|---|---|----|---|
| 1 |   | 1  | -1 | -1 |   | 1 | 4 | 3  |   |
| 2 |   | 2  | 1  | 0  |   | 2 | 0 | -2 |   |
| 3 |   |    |    |    |   | 0 | 1 | 5  |   |
| 4 |   | -1 | 3  | 0  |   |   |   |    |   |
| 5 |   | 4  | 8  | 4  |   |   |   |    |   |
| 6 |   |    |    |    |   |   |   |    |   |

**Determinantti MDETERM.** Seuraavassa laskentapohjassa on  $3 \times 3$ -matriisi alueessa F1:H2.

|   | A | B | C | D | E | F | G | H  | I |
|---|---|---|---|---|---|---|---|----|---|
| 1 |   |   |   |   |   | 1 | 4 | 3  |   |
| 2 |   |   |   |   |   | 2 | 0 | -2 |   |
| 3 |   |   |   |   |   | 0 | 1 | 5  |   |
| 4 |   |   |   |   |   |   |   |    |   |
| 5 |   |   |   |   |   |   |   |    |   |

Lasketaan matriisin determinantti ja sijoitetaan se paikkaan C4 seuraavasti:

(1) Vie solukohdistin soluun C4

(2) Kirjoita kaava: `=MDETERM(F1:H3)`

(3) Näppäile: `[Enter]`

Näiden näppäilyjen jälkeen laskentapohja on seuraavan näköinen

|   | A | B | C   | D | E | F | G | H  | I |
|---|---|---|-----|---|---|---|---|----|---|
| 1 |   |   |     |   |   | 1 | 4 | 3  |   |
| 2 |   |   |     |   |   | 2 | 0 | -2 |   |
| 3 |   |   |     |   |   | 0 | 1 | 5  |   |
| 4 |   |   | -32 |   |   |   |   |    |   |
| 5 |   |   |     |   |   |   |   |    |   |

**Käänteismatriisi MINVERSE.** Seuraavassa laskentapohjassa on  $3 \times 3$ -matriisi alueessa F1:H2.

|   | A | B | C | D | E | F | G | H  | I |
|---|---|---|---|---|---|---|---|----|---|
| 1 |   |   |   |   |   | 1 | 4 | 3  |   |
| 2 |   |   |   |   |   | 2 | 0 | -2 |   |
| 3 |   |   |   |   |   | 0 | 1 | 5  |   |
| 4 |   |   |   |   |   |   |   |    |   |

Lasketaan matriisin käänteismatriisi ja sijoitetaan se paikkaan B4:D3 seuraavasti:

(1) Maalaa alue B1:D3

(2) Kirjoita kaava: `=MINVERSE(F1:H3)`

(3) Näppäile: [Ctrl]-[Shift]-[Enter]

Näiden näppäilyjen jälkeen laskentapohja on seuraavan näköinen

|   | A | B        | C        | D        | E | F | G | H  | I |
|---|---|----------|----------|----------|---|---|---|----|---|
| 1 |   | -0,06250 | 0,53125  | 0,25000  |   | 1 | 4 | 3  |   |
| 2 |   | 0,31250  | -0,15625 | -0,25000 |   | 2 | 0 | -2 |   |
| 3 |   | -0,06250 | 0,03125  | 0,25000  |   | 0 | 1 | 5  |   |
| 4 |   |          |          |          |   |   |   |    |   |

**Yhtälöryhmän ratkaisu käänteismatriisin avulla.** Seuraavassa laskentapohjassa on yhtälöryhmän kerroinkaavio.

|   | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 |   | 1 | 2 | 3 |   |   |
| 2 |   | 2 | 5 | 8 |   |   |
| 3 |   |   |   |   |   |   |
| 4 |   |   |   |   |   |   |
| 5 |   |   |   |   |   |   |
| 6 |   |   |   |   |   |   |

Ratkaistaan yhtälöryhmä periaatteella

$$\mathbf{A}\vec{x} = \vec{b} \quad \Leftrightarrow \quad \vec{x} = \mathbf{A}^{-1}\vec{b}$$

Lasketaan ratkaisuvektori  $\vec{x}$  ja sijoitetaan se alueeseen B4:B5 seuraavasti:

(1) Maalaa alue B4:B5

(2) Kirjoita kaava: `=MMULT(MINVERSE(B1:C2);D1:D2)`

(3) Näppäile: [Ctrl]-[Shift]-[Enter]

Näiden näppäilyjen jälkeen laskentapohja on seuraavan näköinen

|   | A | B  | C | D | E | F |
|---|---|----|---|---|---|---|
| 1 |   | 1  | 2 | 3 |   |   |
| 2 |   | 2  | 5 | 8 |   |   |
| 3 |   |    |   |   |   |   |
| 4 |   | -1 |   |   |   |   |
| 5 |   | 2  |   |   |   |   |
| 6 |   |    |   |   |   |   |

**Transpoosi.** Joskus on tarpeen vaihtaa taulukon rivit sarakkeiksi (ja sarakkeet riveiksi). Jos aineisto on iso, niin ainoa järkevä tapa toimia on käyttää transpoosi-komentoa.

|   | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 |   |   | 1 | 2 | 3 |   |
| 2 |   |   | 2 | 5 | 8 |   |
| 3 |   |   |   |   |   |   |
| 4 |   |   |   |   |   |   |
| 5 |   |   |   |   |   |   |
| 6 |   |   |   |   |   |   |

Transponoidaan kaavio C1:E2 ja sijoitetaan se alueeseen A3:B5 seuraavasti:

(1) Maalaa alue A3:B5

(2) Kirjoita kaava: `=TRANSPOSE(C1:E2)`

(3) Näppäile: [Ctrl]-[Shift]-[Enter]

Näiden näppäilyjen jälkeen laskentapohja on seuraavan näköinen

|   | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 |   |   | 1 | 2 | 3 |   |
| 2 |   |   | 2 | 5 | 8 |   |
| 3 | 1 | 2 |   |   |   |   |
| 4 | 2 | 5 |   |   |   |   |
| 5 | 3 | 8 |   |   |   |   |
| 6 |   |   |   |   |   |   |

## 5.5 MatLab-ohjelmoinnin alkeita

Sec:MatLab

Matlab-ohjelma on yhdysvaltalaisen yrityksen, MathWorks Inc., valmistama tuote. (ks. <http://www.mathworks.com/>) Tuote on ollut markkinoilla vuodesta 1984, ja siitä on kehittynyt yksi teknillisen laskennan ”standardi”. Matlab tarjoaa ohjelmointiympäristön, jonka avulla voidaan helposti käsitellä matriiseja.

Matlabiin on valmiiksi ohjelmoitu lähes kaikki kuviteltavissa olevat vektori- ja matriisilaskennan toiminnot. Lisäksi Matlabiin on tarjolla työkalupaketteja erilaisiin sovellustilanteisiin.

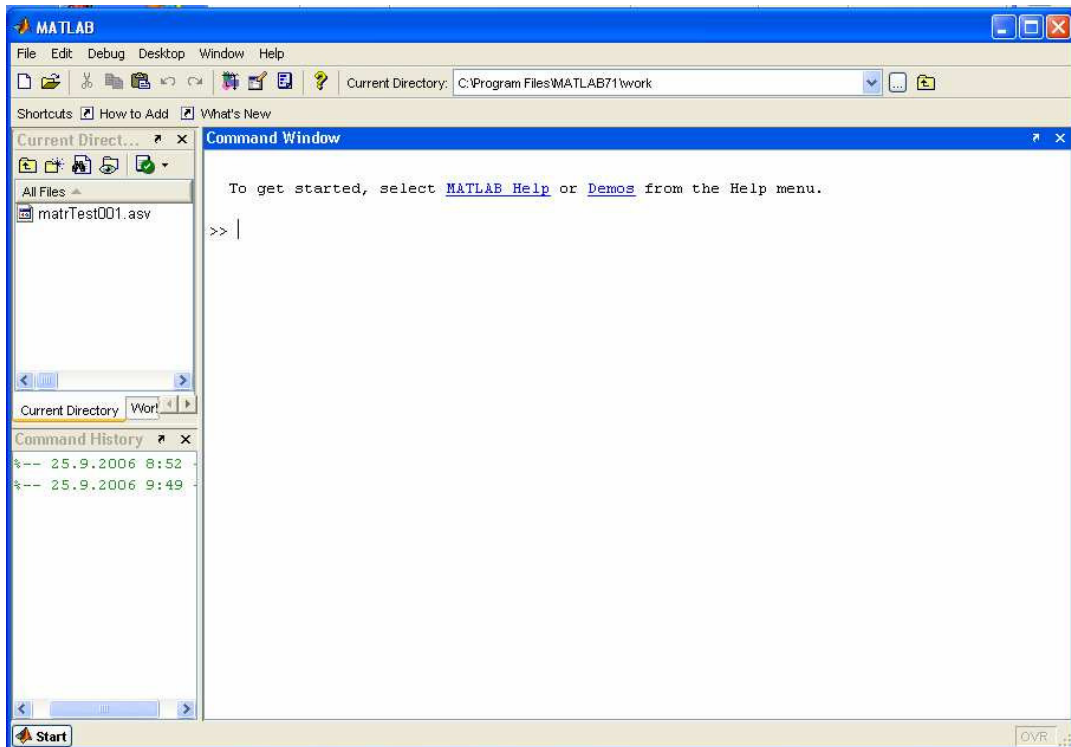
Tässä kappaleessa sivutaan vain pientä osaa Matlabin antamista mahdollisuuksista. Kirjallisuusluettelossa on mainittu joitakin Matlab-oppaita, joista tietoa löytyy enemmän. (Huomaa, että tämä tiedosto saattaa olla jo vanha lukiessasi tätä.) Tällä hetkellä kirjastosta löytyy esimerkiksi Simo K. Kivelän (1079); "MATLAB-opas, Matlabin version 4 mukaan". Kirja on käyttökelpoinen ainakin version 7 käyttäjille.

### 5.5.1 Rivikomentoja.

#### **Aloitus.**

Kun olet käynnistänyt ohjelman, näytölle ilmestyy komentoikkuna.





Tässä lyhyessä ohjeessa emme lainkaan käsittele, valikoita ja työkaluja, vaan annamme muutaman esimerkin komennoista. Komentoikkunaan (Command Window) vasempaan reunaan ilmestynvä ”>>”-merkki on ilmoitus siitä, että matlab on valmis ottamaan vastaan komentoja. Komento voi olla vaikkapa laskutehtävä:

```
>> 2+2
ans =
     4
>>
```

Ohjelma vastaa antamalla lausekkeen arvon. Komennolla voidaan asettaa vektorin tai matriisin alkiot. Tämän jälkeen matriisiin voidaan viitata nimellä. (Huomaa, että ”>>”-merkin perässä oleva on käyttäjän kirjoittama komento ja muut rivit ovat ohjelman reaktioita tässä dialogissa.)

```
>> A = [1 2 3 ; 4 5 6]
A =
     1     2     3
     4     5     6

>> B = [3 2;-1 0]
B =
```

|    |   |
|----|---|
| 3  | 2 |
| -1 | 0 |

```
>> A*B
??? Error using ==> mtimes Inner matrix dimensions must agree.
```

```
>> B*A
ans =
    11    16    21
    -1    -2    -3
>>
```

Jos haluaa, ettei ohjelma turhaan vastaa, niin voi komennon perään lisätä puolipisteen ”;”. Silloin ohjelman reaktio jää pois.

```
>> A = [1 2 3 ; 4 5 6];
>> B = [3 2;-1 0];
>> B*A
ans =
    11    16    21
    -1    -2    -3
>>
```

Käymme ensin lyhyesti läpi samat esimerkit kuin Excel-kappaleessa, sitten käsittelemme komentojono- ja funktio-tiedoston luomisen (M-file). M-tiedoston avulla voi pidemmän komentosarjan tallentaa tiedostoksi, jota voi editoida ja testata järkevällä tavalla. M-tiedostoja on myös helppo lähettää vaikkapa sähköpostilla.

**Rivioperaatiot.** Rivioperaatioiden tekeminen Matlabilla ei ole alkeisasia. Jos joudut tekemään rivioperaatioita Matlabilla, olet jo aika pitkällä matlab-ohjelmoinnissa. Nyt ei ole syytä korostaa rivioperaatioita sinänsä vaan sitä, että matriisin riviin on mahdollista viitata.

Esim:

```
>> A=[1 2 3 ; 2 5 8]
A =
     1     2     3
     2     5     8
```

```
>> A(2,3) % alkio rivillä 2 sarakkeessa 3
ans =
     8
```

```
>> A(:,3) % matriisin kolmas sarake
ans =
     3
     8
```

```
>> A(2,:) % matriisin toinen rivi
ans =
     2     5     8
```

```
>>
```

Ja sitten se rivioperaatio. Lisätään matriisiin toiseen riviin ensimmäinen rivi kerrottuna  $-2$ :lla:

```
>> A
A =
     1     2     3
     2     5     8
```

```
>> A(2,:) = A(2,:) - 2*A(1,:);
```

```
>> A
A =
     1     2     3
     0     1     2
```

```
>>
```

### Kahden matriisin summa.

```
>> A=[1 -1 -1 ; 2 1 9]
A =
     1    -1    -1
     2     1     9
```

```
>> B=[1 4 3 ; 2 0 -2]
```

```
B =
      1      4      3
      2      0     -2
```

```
>> A+B
ans =
      2      3      2
      4      1      7
>>
```

### Matriisitulo.

```
>> A=[1 -1 -1 ; 2 1 0]
A =
      1     -1     -1
      2      1      0
```

```
>> B=[1 4 3 ; 2 0 -2 ; 0 1 5]
B =
      1      4      3
      2      0     -2
      0      1      5
```

```
>> A*B
ans =
     -1      3      0
      4      8      4
>>
```

### Determinantti.

```
>> B=[1 4 3 ; 2 0 -2 ; 0 1 5]
B =
      1      4      3
      2      0     -2
      0      1      5
```

```
>> det(B)
ans =
    -32
>>
```

**Käänteismatriisi.**

```
>> B=[1 4 3 ; 2 0 -2 ; 0 1 5]
B =
     1     4     3
     2     0    -2
     0     1     5

>> inv(B)
ans =
    -0.0625    0.5313    0.2500
     0.3125   -0.1563   -0.2500
    -0.0625    0.0313    0.2500
>>
```

**Yhtälöryhmän ratkaisu käänteismatriisin avulla.**

ratkaistaan  $\vec{x}$  yhtälöryhmästä

$$\begin{pmatrix} 1 & 2 \\ 2 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 3 \\ 8 \end{pmatrix}$$

seuraavasti:

```
>> A=[1 2 ; 2 5]
A =
     1     2
     2     5

>> b=[3;8]
b =
     3
     8

>> inv(A)*b
ans =
    -1
     2
>>
```

**Matriisin transpoosi.**

Matriisin transpoosi merkitään heittomerkillä.

normaali notaatio:  $\mathbf{A}^T$       matlab-notaatio:  $\mathbf{A}'$

esimerkki:

```
>> A=[1 2 3 ; 4 5 6]
```

```
A =
```

```
    1    2    3
    4    5    6
```

```
>> A'
```

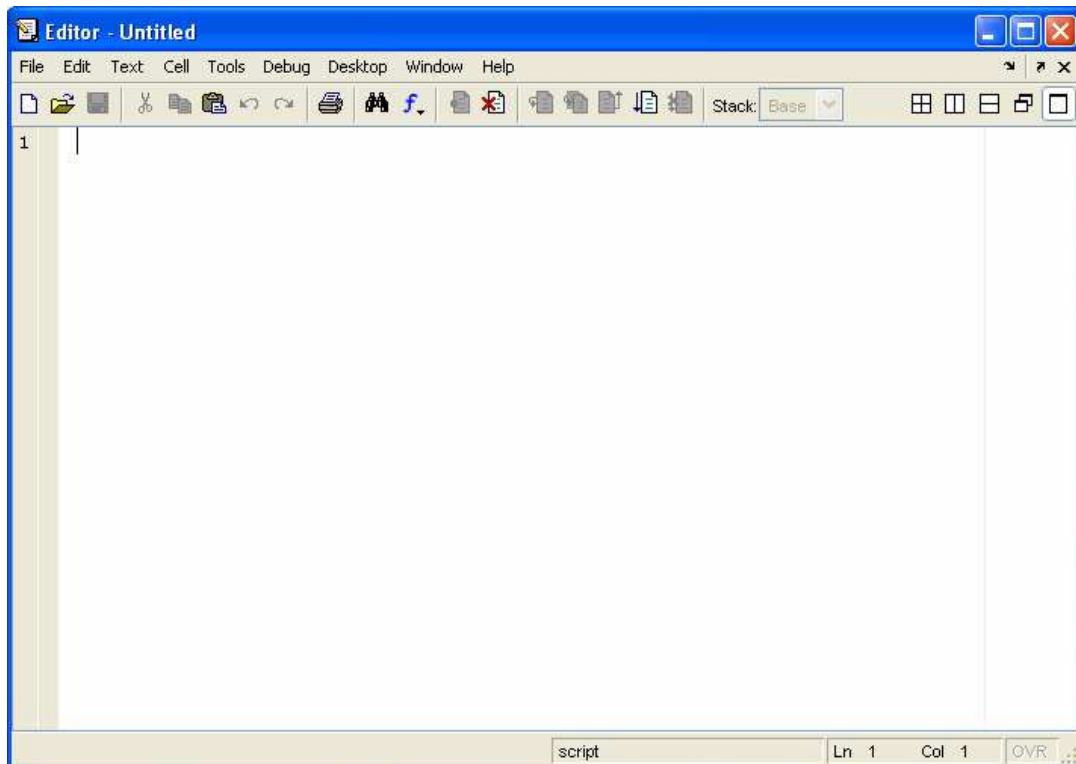
```
ans =
```

```
    1    4
    2    5
    3    6
```

```
>>
```

**5.5.2 Komentojonotiedostot ja omat funktiot.**

Valitsemalla komentoikkunan (Command Window) valikoista [File]-[New]-[M-file] avautuu Editori-ikkuna (Editor), johon voidaan kirjoittaa useita komentoja.



Kirjoita ensimmäisille riveille %-merkillä alkaville riveille kommentit, joista käy ilmi milloin ja mitä tarkoitusta varten tämä tiedosto on kirjoitettu.

**Esimerkki 52** Tutkitaan esimerkissä esim01\_Markov sivulla esim01\_Markov esiteltyä dynaamista systeemiä. Jos alkutila on  $(x_1, x_2, x_3)^T = (30, 20, 20)^T$ , niin muutaman seuraavan puolivuotisjakson jälkeiset tilat saadaan seuraavasti:

(1) Avataan uusi M-file ja kirjoitetaan sinne koodi:

```
% 25.9.2006 / math1040 (math1040c3s11e1.m)
% esimerkin 93 sivu 78 ratkaisu, kun
% alkutila on x_0 = [30 20 20]'
%
P=[0.80 0.05 0.10; 0.15 0.70 0.10; 0.05 0.25 0.80];
X=zeros(3,12); % luodaan matriisi, jossa on 3 rivia ja 12 saraketta.
% Alussa matriisi on täynnä nollia.
% Tämän matriisin sarakkeisiin talletetaan
% perakkaiset tilat
```

```

X(:,1) = [30 20 20]'; % asetetaan alkutila
%
for k=1:11 % lasketaan perakkaiset tilat
    X(:,k+1)=P*X(:,k);
end
X

```

(2) Valitaan editorin valikosta [Debug]-[Save and Run], jolloin komentoikkunaan tulee tulos:

X =

Columns 1 through 6

|         |         |         |         |         |         |
|---------|---------|---------|---------|---------|---------|
| 30.0000 | 27.0000 | 24.8750 | 23.3800 | 22.3343 | 21.6066 |
| 20.0000 | 20.5000 | 20.6500 | 20.6337 | 20.5493 | 20.4463 |
| 20.0000 | 22.5000 | 24.4750 | 25.9863 | 27.1164 | 27.9472 |

Columns 7 through 12

|         |         |         |         |         |         |
|---------|---------|---------|---------|---------|---------|
| 21.1023 | 20.7542 | 20.5147 | 20.3505 | 20.2382 | 20.1616 |
| 20.3481 | 20.2640 | 20.1961 | 20.1434 | 20.1036 | 20.0740 |
| 28.5496 | 28.9818 | 29.2892 | 29.5061 | 29.6583 | 29.7644 |

>>

Jos komentotiedoston loppuun vielä lisätään kaksi riviä:

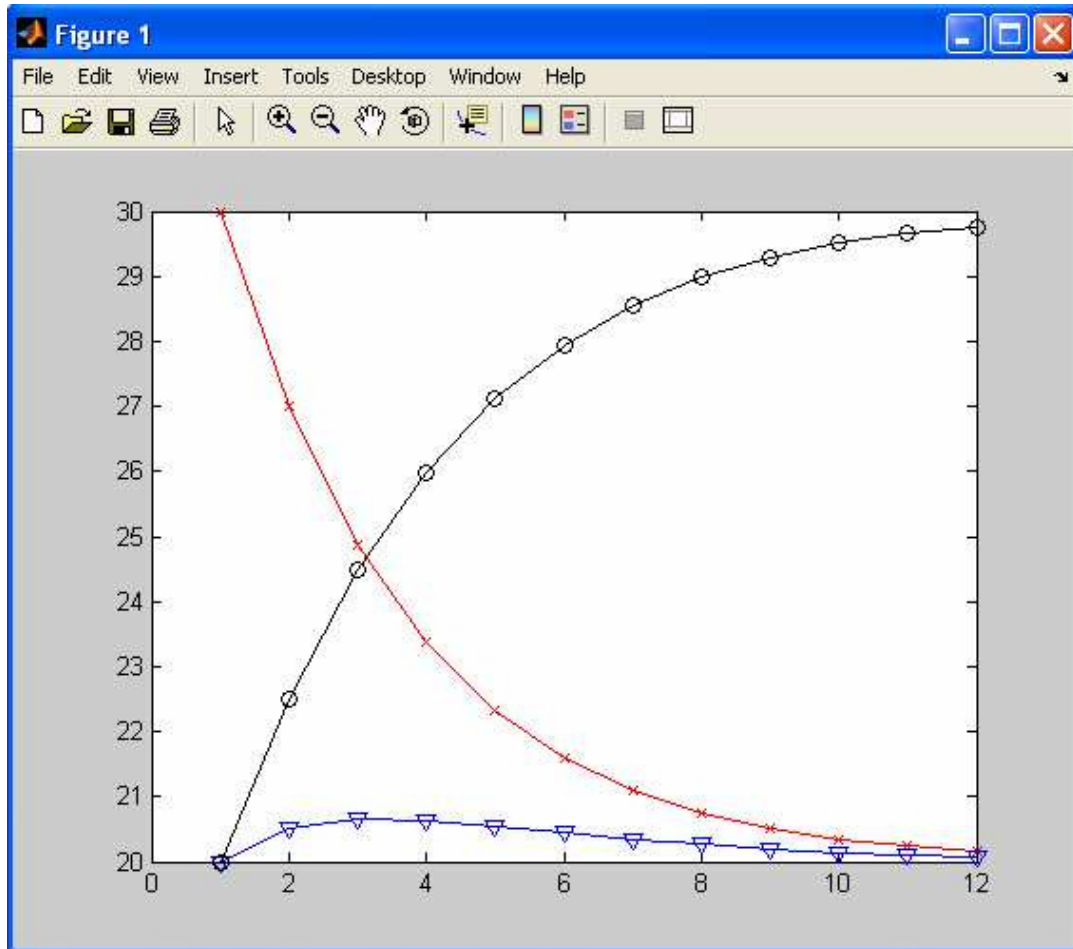
```

t=linspace(1,12,12); % vektori [1 2 3 4 ... 11 12]
plot(t,X(1,:), 'r-x', t,X(2,:), 'b-v', t,X(3,:), 'k-o')

```

niin saamme piirrettyä funktiot (murtoviivat)  $x_1(t)$ ,  $x_2(t)$  ja  $x_3(t)$ . Tulos ilmestyy omaan ikkunaan.





## 6 PISTETULON JA RISTITULON SOVELLUKSIA

Ch:DotCross

### 6.1 Pyörivistä kappaleista

c:RotatingBody

#### 6.1.1 Voiman momentti akselin suhteen avaruudessa

Ssec:FMomSpace

Seuraavassa pohdiskelussa kuvataan tutkittavaa systeemiä ehkä tarpeettomankin konkreettisesti. Tämä on tahallista. Myös lukijan on syytä visualisoida systeemi niin hyvin, kuin pystyy. Pikkutarkan nyperryksen jälkeen tuloksena on kaunis ja yksinkertainen kaava, joka tosin on kirjoitettu vektori-merkinnöin. Jos käytettävissä on Octaven kaltainen laskentaympäristö, niin vektorilausekke on helppo laskea.

Kun suureita ilmaistaan vektoreina, kulkevat momenttien merkit ja akselien positiiviset kiertosuunnat näitesti lausekkeissa ilman että niitä tarvitsisi erikseen pohtia!

Tarkastellaan laakeroituun akseliin kiinnitettyä kiinteää kappaletta (esimerkiksi sähkömoottorin roottori). Oletamme nyt, että kappaleen painopiste on akselilla, jolloin painolla ei ole momenttia. Akselista tiedetään yksi piste  $P_0$  ja yksikkö-suuntavektori  $\vec{a}$ . Vektori  $\vec{a}$  kiinnittää nyt akselin positiivisen kiertosuunnan oikean käden säännöllä: ”kun peukalo on  $\vec{a}$ :n suuntaan, niin muut sormet kiertävät akselia positiiviseen kiertosuuntaan”.

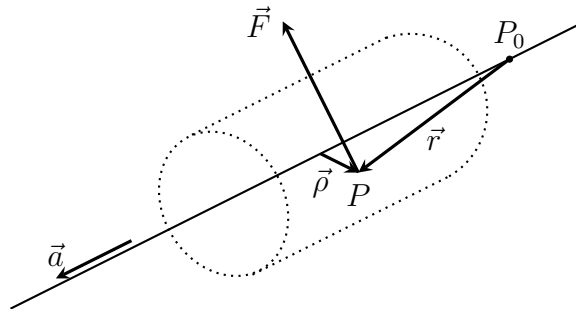


fig:vmA01

**Kuva 10.** Voiman momentti akselin suhteen.

Voima  $\vec{F}$  vaikuttaa kiinteän kappaleen pisteeseen  $P$ . Vektori  $\vec{r} = \overrightarrow{P_0P}$  on voiman varsi ja kohtisuora varsi  $\vec{\rho}$  on suuntajana, joka on kohtisuorassa akselia vastaan, sen alkupiste on akselilla ja loppupiste on  $P$ . Jos varren ja akselin välinen kulma on  $\alpha$ , niin kohtisuoran varren pituus on

$$\rho = r \sin \alpha = |\vec{a} \times \vec{r}|. \quad (6.1) \quad \text{eq:rb01}$$

Voima jaetaan kolmeen keskenään kohtisuoraan komponenttiin

$$\vec{F} = \vec{F}_a + \vec{F}_\rho + \vec{F}_\perp,$$

Ensimmäinen komponentti  $\vec{F}_a$  vaikuttaa kyllä kappaleeseen, mutta jos laakerit kestävät, niin kappale ei liiku tämän komponentin takia. Toinen komponentti  $\vec{F}_\rho$  suuntautuu suoraan akseliin päin (tai poispäin). Sekään ei aiheuta liikettä, jos akselin laakerointi kestää.

Kolmas komponentti  $\vec{F}_\perp$  vaikuttaa kappaleen pyörimiseen. Koska vektorit  $\vec{\rho}$  ja  $\vec{F}_\perp$  ovat keskenään kohtisuorassa ja kumpikin akselia vastaan kohtisuorassa, on pyörimiseen vaikuttava momentti

$$M_a = (\pm)\rho F_\perp. \quad (6.2) \quad \text{eq:rb02}$$

Lauseke on yksinkertainen, mutta usein epäkäytännöllinen. Siksi teemme sille vielä paremman muodon. Koska vektori  $\vec{F}_\perp$  on voiman vektoriprojektio vektorin  $\vec{a} \times \vec{\rho}$  suuntaan, ja toisaalta  $\vec{a} \times \vec{\rho} = \vec{a} \times (\vec{r} - (\vec{a} \cdot \vec{r})\vec{a}) = \vec{a} \times \vec{r}$ , saamme

$$F_\perp = |\vec{F}_{\vec{a} \times \vec{\rho}}| = |\vec{F}_{\vec{a} \times \vec{r}}| = \frac{|\vec{F} \cdot \vec{a} \times \vec{r}|}{|\vec{a} \times \vec{r}|} = \frac{|\vec{a} \cdot \vec{r} \times \vec{F}|}{|\vec{a} \times \vec{r}|}. \quad (6.3) \quad \text{eq:rb03}$$

Kun sijoitamme lausekkeet <sup>eq:rb02</sup>6.1 ja <sup>eq:rb03</sup>6.3 momentin lausekkeeseen <sup>eq:rb02</sup>6.2, saamme

$$M_a = (\pm)\rho F_\perp = (\pm)|\vec{a} \cdot \vec{r} \times \vec{F}| = \vec{a} \cdot \vec{r} \times \vec{F}.$$

Monissa sovellustilanteissa on luontevaa ja hyödyllistä ilmaista momentti vektorina (jolloin myös pyörimissuunta saa tulkinnan). Tässä tapauksessa on luonnollista valita suunnaksi akselin suunta. Siis

$$\vec{M}_a = M_a \vec{a} = (\vec{a} \cdot \vec{r} \times \vec{F}) \vec{a} = (\vec{r} \times \vec{F})_{\vec{a}},$$

eli voiman (vektori-)momentti akselin suhteen on voiman momentin vektoriprojektio akselin suuntaan.

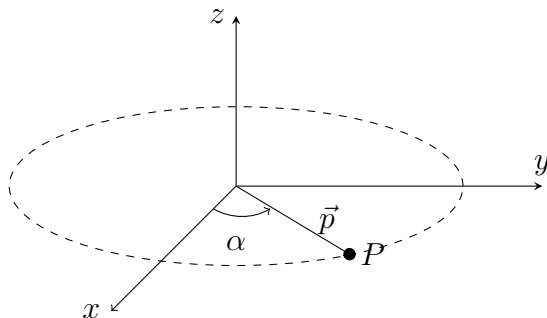
## 6.2 Liike pyörivässä koordinaatistossa

RotatingCoord

### 6.2.1 Kehänopeus pyörittäessä tasossa.

Sijoitetaan seuraavassa vektorit xyz-koordinaatistoon, jonka koordinaattiakselien suuntavektorit ovat  $\vec{i}$ ,  $\vec{j}$  ja  $\vec{k}$ . Tarkastellaan ensin suuntajanaa  $\overline{OP}$ , origosta xy-tason pisteeseen  $P$ . Hetkellä  $t = 0$  pisteen  $P$  koordinaatit ovat  $P = (r \cos \alpha, r \sin \alpha)$ . Siis pisteen  $P$  paikkavektori alkuhetkellä on

$$\vec{p} = r \cos \alpha \vec{i} + r \sin \alpha \vec{j}.$$



Tarkastellaan liikettä, jossa suuntajanan origon puoleinen pää pysyy paikallaan, mutta toinen pää kiertää origoa siten, että suuntajana pyörii kuin kellon viisari tasaisella kiertonopeudella. Jos suuntajana pyörii pituutensa säilyttäen origon ympäri kulmanopeudella  $\omega$ , niin silloin pisteen  $P$  paikkavektori hetkellä  $t$  on

$$\vec{p}(t) = r \cos(\alpha + \omega t) \vec{i} + r \sin(\alpha + \omega t) \vec{j}.$$

Pisteen  $P$  nopeus pyörimisliikkeessä on

$$\begin{aligned} \vec{v} &= \frac{d\vec{p}(t)}{dt} = \lim_{dt \rightarrow 0} \frac{\vec{p}(t+dt) - \vec{p}(t)}{dt} \\ &= \lim_{dt \rightarrow 0} \frac{(r \cos(\alpha + \omega(t+dt)) \vec{i} + r \sin(\alpha + \omega(t+dt)) \vec{j}) - (r \cos(\alpha + \omega t) \vec{i} + r \sin(\alpha + \omega t) \vec{j})}{dt} \end{aligned}$$

Yksinkertaistetaan lauseketta merkinnällä  $\alpha_t = \alpha + \omega t$  ja käytetään summakulman sinin ja kosinin kaavoja, jolloin saamme <sup>1</sup> (ks. taulukkokirja)

$$\begin{aligned} \vec{v} &= \lim_{dt \rightarrow 0} \frac{r[(\cos(\alpha_t + \omega dt) \vec{i} + \sin(\alpha_t + \omega dt) \vec{j}) - (\cos(\alpha_t) \vec{i} + \sin(\alpha_t) \vec{j})]}{dt} \\ &= r \lim_{dt \rightarrow 0} \frac{[\cos \alpha_t \cos \omega dt - \sin \alpha_t \sin \omega dt - \cos \alpha_t] \vec{i} + [\sin \alpha_t \cos \omega dt + \cos \alpha_t \sin \omega dt - \sin \alpha_t] \vec{j}}{dt} \\ &= r \lim_{dt \rightarrow 0} \left\{ \left[ \cos \alpha_t \frac{\cos \omega dt - 1}{dt} - \sin \alpha_t \frac{\sin \omega dt}{dt} \right] \vec{i} \dots \right. \\ &\quad \left. \dots + \left[ \sin \alpha_t \underbrace{\frac{\cos \omega dt - 1}{dt}}_{\rightarrow 0} + \cos \alpha_t \underbrace{\frac{\sin \omega dt}{dt}}_{\rightarrow \omega} \right] \vec{j} \right\} \\ &= \omega r (-\sin \alpha_t + \cos \alpha_t) \end{aligned}$$

Suora lasku osoittaa, että  $|\vec{v}(t)| = \omega r = \omega |\vec{p}(t)|$  ja  $\vec{v}(t) \perp \vec{p}$ . Lisäksi on selvää, että jos  $\omega > 0$ , niin vektorit  $\{\vec{k}, \vec{p}, \vec{v}\}$  muodostavat oikeakätisen systeemin, joten

$$\vec{v} = \omega \vec{k} \times \vec{p}.$$

<sup>1</sup>  $\cos(a+b) = \cos a \cos b - \sin a \sin b$  ja  $\sin(a+b) = \sin a \cos b + \cos a \sin b$

Tulos on edellä johdettu oikeaoppisesti vektorifunktion derivaattana. Toisaalta tulos on helppo nähdä koulufysiikan perusteella. Piste  $P$  kiertää ympyrärataa origon ympäri. Ympyräliikkeen kehänopeus on  $\omega r$  ja kehänopeus on radan tangentin suuntainen. Siis kohtisuorassa suuntajanaa  $\overrightarrow{OP}$  ja vektoria  $\vec{k}$  vastaan. Kolmikon  $\{\vec{k}, \vec{p}, \vec{v}\}$  oikeakätisyys on myös selvä. Siis

$$\vec{v} = \omega r \vec{k} \times \vec{p}^0 = \omega \vec{k} \times \vec{p}.$$

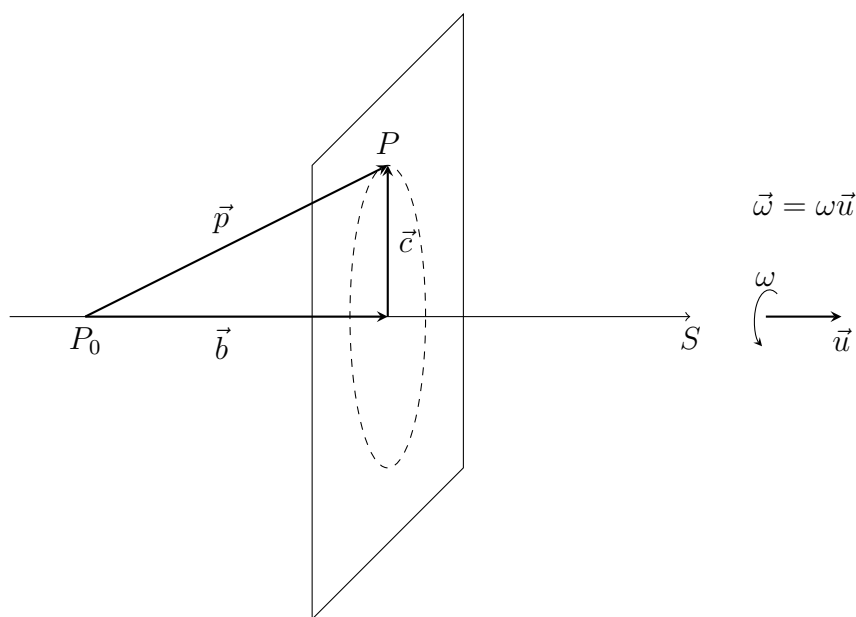
Pyörimisakselina oli nyt  $xyz$ -koordinaatiston  $z$ -akseli, joka kulkee origon kautta ja on vektorin  $\vec{k}$  suuntainen. Sovimme, että vektori  $\vec{\omega}$  on pituudeltaan  $\omega$  ja samansuuntainen pyörimisakselin  $\vec{k}$  kanssa. (Muista oikean käden sääntö: kun peukalo osoittaa pyörimisakselin  $\vec{\omega}$  suuntaan, niin muut sormet osoittavat positiivisen kiertosuunnan akselin ympäri.) Tällä merkinnällä kehänopeus saa yksinkertaisen muodon

$$\vec{v} = \vec{\omega} \times \vec{p}.$$

### 6.2.2 Kehänopeus pyörittäessä akselin ympäri.

Toiseksi tarkastelemme liikettä, joka syntyy kun piste  $P$  kiertää akselia (suoraa)  $S$  ympyrärataa akselia vastaan kohtisuolassa tasossa vakioetäisyydellä akselistä. Olkoon yksikkövektori  $\vec{u}$  pyörimisakselin  $S(P_0, \vec{u})$  suuntavektori. Kiinnitämme tarkastelun ajaksi origon akselin pisteeseen  $P_0$ , jolloin pisteen  $P$  paikkavektori on  $\vec{p} = \overrightarrow{P_0P}$ . Paikkavektori  $\vec{p}$  voidaan jakaa akselin suuntaiseen komponenttiin  $\vec{b}$  ja akselia vastaan kohtisuoraan komponenttiin  $\vec{c}$ .

$$\vec{p} = \vec{b} + \vec{c}, \quad \vec{b} \parallel \vec{u}, \quad \vec{c} \perp \vec{u}$$



Kun piste  $P$  kiertää akselia, kiertää  $P$  ja vektori  $\vec{c}$  akselin normaalitasossa. Edellisen luvun perusteella tiedämme, että pisteen  $P$  nopeus on

$$\vec{v} = \vec{\omega} \times \vec{c} = \vec{\omega} \times \vec{p}.$$

Viimeinen yhtäsuuruus perustuu siihen, että  $\vec{\omega} \times \vec{p} = \vec{\omega} \times (\vec{b} + \vec{c}) = \vec{\omega} \times \vec{b} + \vec{\omega} \times \vec{c} = \vec{\omega} \times \vec{c}$ .

### 6.2.3 Liike pyörivässä koordinaatistossa, Corioliskiihtyvyys

Tarkastellaan kahta ortonormeerattua koordinaatistoa (kantaa)  $E = \{\vec{e}_x, \vec{e}_y, \vec{e}_z\}$  ja  $W = \{\vec{w}_x, \vec{w}_y, \vec{w}_z\}$ , joilla on yhteinen origo, mutta  $E$  pysyy paikallaan ja  $W$  pyörii akselin  $\vec{\omega}$  ympäri. Pyörimisakseli  $\vec{\omega}$  pysyy paikallaan kummankin koordinaatiston suhteen! Tarkastellaan pistettä  $P$ , jonka paikkavektorilla on esitykset

$$\begin{aligned}\vec{r}(t) &= x(t)\vec{e}_x + y(t)\vec{e}_y + z(t)\vec{e}_z \\ &= \tilde{x}(t)\vec{w}_x + \tilde{y}(t)\vec{w}_y + \tilde{z}(t)\vec{w}_z\end{aligned}$$

Kun pisteen  $P$  nopeus määritetään  $E$ -koordinaatistossa, saadaan

$$\vec{v}(t) = \frac{d}{dt}(x(t)\vec{e}_x + y(t)\vec{e}_y + z(t)\vec{e}_z) = x'(t)\vec{e}_x + y'(t)\vec{e}_y + z'(t)\vec{e}_z.$$

Kun  $W$ -koordinaatiston mukana pyörivä tarkkailija määrittää pisteen  $P$  nopeusvektorin, niin tulos on

$$\vec{v}^*(t) = \frac{d}{dt}(\tilde{x}(t)\vec{w}_x + \tilde{y}(t)\vec{w}_y + \tilde{z}(t)\vec{w}_z) = \tilde{x}'(t)\vec{w}_x + \tilde{y}'(t)\vec{w}_y + \tilde{z}'(t)\vec{w}_z.$$

Edellä paikkavektorin  $\vec{r}(t)$  esitykset kahdessa eri koordinaatistossa esittivät samaa vektoria. On erittäin houkuttelevaa ajatella, että myös  $\vec{v}(t)$  ja  $\vec{v}^*(t)$  esittäisivät samaa universaalista siirtymisnopeusvektoria, mutta näin ei ole! Asia paljastuu, kun  $E$ -koordinaatiston mukana paikallaan pysyvä tarkkailija määrittää pisteen  $P$  nopeuden käyttäen  $E^*$ -koordinaatiston esitystä, tulokseksi tulee

$$\begin{aligned}\vec{v}(t) &= \frac{d}{dt}\{\tilde{x}(t)\vec{w}_x + \tilde{y}(t)\vec{w}_y + \tilde{z}(t)\vec{w}_z\} \\ &= \left(\frac{d\tilde{x}(t)}{dt}\vec{w}_x + \tilde{x}(t)\frac{d\vec{w}_x}{dt}\right) + \left(\frac{d\tilde{y}(t)}{dt}\vec{w}_y + \tilde{y}(t)\frac{d\vec{w}_y}{dt}\right) + \left(\frac{d\tilde{z}(t)}{dt}\vec{w}_z + \tilde{z}(t)\frac{d\vec{w}_z}{dt}\right) \\ &= \vec{v}^*(t) + \tilde{x}(t)\vec{\omega} \times \vec{w}_x + \tilde{y}(t)\vec{\omega} \times \vec{w}_y + \tilde{z}(t)\vec{\omega} \times \vec{w}_z \\ &= \vec{v}^*(t) + \vec{\omega} \times \vec{r}(t)\end{aligned}$$

Seuraavaksi  $E$  koordinaatiston suhteen levossa oleva tarkkailija määrittää pisteen  $P$  kiihtyvyyksvektorin käyttäen  $W$  koordinaatiston esitystä.

$$\begin{aligned}
 \vec{a}(t) &= \frac{d}{dt}(\vec{v}^*(t) + \vec{\omega} \times \vec{r}(t)) \\
 &= \frac{d}{dt}\{\dot{x}(t)\vec{w}_x + \dot{y}(t)\vec{w}_y + \dot{z}(t)\vec{w}_z\} + \vec{\omega} \times \frac{d\vec{r}(t)}{dt} \\
 &= \left(\ddot{x}(t)\vec{w}_x + \dot{x}(t)\frac{d\vec{w}_x}{dt}\right) + \left(\ddot{y}(t)\vec{w}_y + \dot{y}(t)\frac{d\vec{w}_y}{dt}\right) + \left(\ddot{z}(t)\vec{w}_z + \dot{z}(t)\frac{d\vec{w}_z}{dt}\right) \\
 &\quad + \vec{\omega} \times (\vec{v}^*(t) + \vec{\omega} \times \vec{r}(t)) \\
 &= \vec{a}^*(t) + 2\vec{\omega} \times \vec{v}^*(t) + \vec{\omega} \times (\vec{\omega} \times \vec{r})
 \end{aligned}$$

#### 6.2.4 Matalapaineen pyörimissuunta

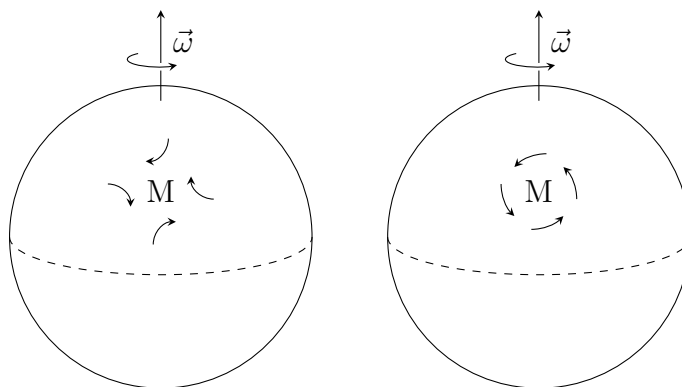
Tarkastellaan ilmamassojen liikettä matalapaineen keskuksen lähellä pohjoisella pallonpuoliskolla. Voimme kuvitella sääkarttaa, joka kuvaa matalapainetta Pohjois-Euroopan yllä. Paine-ero synnyttää ilmamassoihin voimavaikutuksen, joka saa massat liikkumaan kohden matalapaineen keskusta.

Merkitään lepokoordinaatistossa havaittua liikkeen kiihtyvyyttä vektorilla  $\vec{a}$  ja maapallon mukana pyörivän koordinaatiston suhteen mitattua nopeutta ja kiihtyvyyttä vektoreilla  $\vec{v}^*$  ja  $\vec{a}^*$ . Jos merkitsemme maapallon pyörimisakselia vektorilla  $\vec{\omega}$ , niin 'oikean käden säännön' mukaan pyörimisakselin suunta on etelänavalta pohjoisnavalle päin. Pyörivässä koordinaatistossa ja lepokoordinaatistossa havaittuja kiihtyvyyksiä sitoo kaava

$$\vec{a}^* = \overbrace{\vec{a} - 2\vec{\omega} \times \vec{v}^*(t)}^{\text{Coriolis-termi}} - \underbrace{\vec{\omega} \times (\vec{\omega} \times \vec{r})}_{\text{keskihaku-termi}}.$$

Right Hand Side'n ensimmäinen komponentti  $\vec{a}$  osoittaa matalapaineen keskusta kohden. Kolmas komponentti  $-\vec{\omega} \times (\vec{\omega} \times \vec{r})$  osoittaa kohtisuorasti pois päin pyörimisakselista, mikä on varsin luonnollista. Toisen komponentin  $-2\vec{\omega} \times \vec{v}^*(t)$  suunnan näkee siten, että asettaa peukalon vektorin  $\vec{\omega}$  suuntaan (pohjoisella pallonpuoliskolla se on likimain ylöspäin). Seuraavaksi asetetaan etusormi vektorin  $\vec{v}^*$  suuntaan, eli liikkeen suuntaan. Nyt keskisormi osoittaa vektorin  $\vec{\omega} \times \vec{v}^*$  suunnaksi liikesuuntaan nähden vasemmalle, joten  $-2\vec{\omega} \times \vec{v}^*$  osoittaa liikesuuntaan nähden oikealle. Näyttää siis siltä, kuin ilmamassat kokisivat voiman, joka kääntää niiden liikettä oikealle siitä, mikä tuntuisi luonnolliselta.

Kun alussa eri suunnista kohden matalapaineen keskusta liikkeelle lähteneet ilmassat kääntyvät kukin suunnastaan oikealle, niin lopputulos on se, että lopulta ilmassat kiertävät matalapaineen keskusta vastapäivään kiertäen. Hitaasti keskittyvässä spiraalivaiheessa matalapaineen imu, coriolisvoimat ja ilmassojen hitaus yhdessä pitävät spiraaliliikkeen vakaana ja matalapaineen keskuksen täyttyminen kestää useita päiviä.



**Kuva 11.** Matalapaineen pyörimissuunta pohjoisella pallonpuoliskolla.

6.3 Sylinteri- ja pallosymmetrisen funktion tilavuusintegraalista

6.4 Nablasta, divergenssistä ja roottorista



## 7 LINEAARIAVARUUDET

Ch:LinSpaces

7.1 Lisää kompleksilukulaskuja

7.2 Esimerkkejä Funktioavaruuksia

## 8 KANNAT JA ORTOGONAALISUUS

KantaOrthogon

## 8.1 Esimerkkejä

## 9 OMINAISARVOTEHTÄVÄ

Ch:EigSystem

9.1 Lineaarinen differentiaaliyhtälöryhmä

9.2 Lisää matriisihajoitelmia

9.3 Kahden muuttujan sileän ääriarvon tutkiminen

9.4 Kahden muuttujan sileän funktion Taylorin sarja

9.5 Kvadraattinen optimointi

## 10 ANALYYTTISTÄ GEOMETRIAA

Ch:AnalGeom

10.0.1 2D plottaus

10.0.2 Jacobin determinantti

10.0.3 3D plottaus

10.0.4 Kartioleikkaukset; ellipsi, paraabeli, hyperbeli

10.0.5 Kappaleita ja pintoja; pallo, sylinteri, kartio

10.0.6 Jacobin determinantti

10.0.7 Kamera

10.0.8 Virtuaalikamera

10.1 Avaruuden  $\mathbb{R}^n$  analyyttistä geometriaa

10.1.1 Suorat ja hypertasot

10.1.2 Monistot

10.1.3 Jacobin determinantti

## 11 SOVELLUKSIA JA ERILLISAIHEITA

Ch:1aApp

### 11.1 Homogeeniset koordinaatit

Sec:HomCoord

Tässä kappaleessa ei esitetä kattavasti homogeenisten koordinaattien teoriaa, vaan tavoite on esitellä sovellusta niin, että sen käyttökelpoisuus tietokonegrafiikan yhteydessä tulee selväksi. Erityinen tavoite on saada käsitys videokuvan tulkinnasta kuvantamisessa ja robotiikassa. Aloitamme aika yksinkertaisista asioista.

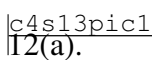
#### 11.1.1 Piirtelyharjoituksia

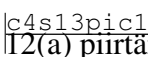
Teemme yksinkertaisen graafisen testikentän seuraavasti. Sovimme, että murtoviiva pisteiden  $P_1, P_2, \dots, P_n$  kautta esitetään  $2 \times n$  matriisina  $\mathbf{P}$ , jonka  $k$ :nnessä sarakkeessa on pisteen  $P_k$  koordinaatit. Esimerkikiksi matriisi

$$\mathbf{P} = \begin{pmatrix} 1 & 1 & 0 & -1 & -1 & 1 \\ -1 & 1 & 1.5 & 1 & -1 & -1 \end{pmatrix}$$

esittää monikulmiota, jonka kärkipisteet ovat  $P_1 = (1, -1)^T$ ,  $P_2 = (1, 1)^T$ ,  $P_3 = (0, 1.5)^T$ ,  $P_4 = (-1, 1)^T$ ,  $P_5 = (1, -1)^T$ . (Kuudes sarake matriisissa  $\mathbf{P}$  saa murtoviivan päätymään alkupisteeseen, jolloin piirtyvä kuvio on viisikulmio. Määritellään vielä toinen monikulmio

$$\mathbf{Q} = \begin{pmatrix} -0.3 & -0.3 & -0.7 & -0.7 & -0.3 \\ -0.9 & 0.0 & 0.0 & -0.9 & -0.9 \end{pmatrix}.$$

Kun kumpikin monikulmio piirretään samaan koordinaalisto, syntyy kuva  12(a).

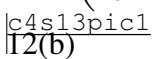
Kuvan  12(a) piirtäminen on tehty octave-komennoilla

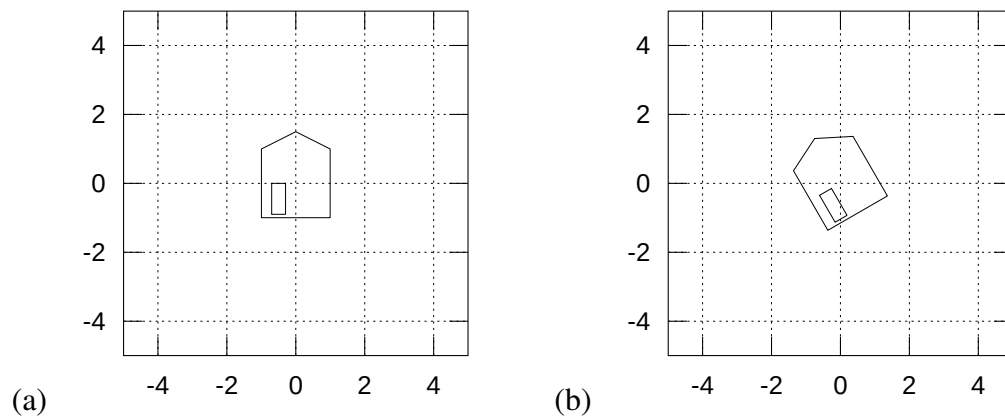
```
# Mokki
P = [1 -1; 1 1; 0 1.5; -1 1; -1 -1; 1 -1]';
Q = [-0.3 -0.9; -0.3 0.0; -0.7 0.0; -0.7 -0.9; -0.3 -0.9]';
hold off;
piirto(P); piirto(Q);
```

ja funktion `piirto` koodi löytyy kappaleen lopusta.

**Kuvan kierto.** Kuvaa saadaan kierrettyä kulman  $\alpha$  verran origon ympäri, kun jokainen kuvan pisteen paikkavektori kerrotaan matriisilla

$$\mathbf{R} = \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix}.$$

Seuraava koodi piirtää kuvan  12(b)



**Kuva 12.** (a)  $P$  ja  $Q$  piirrettynä samaan koordinaatistoon. (b)  $P$  ja  $Q$  kierrettyinä kulman  $\pi/6$  verran.

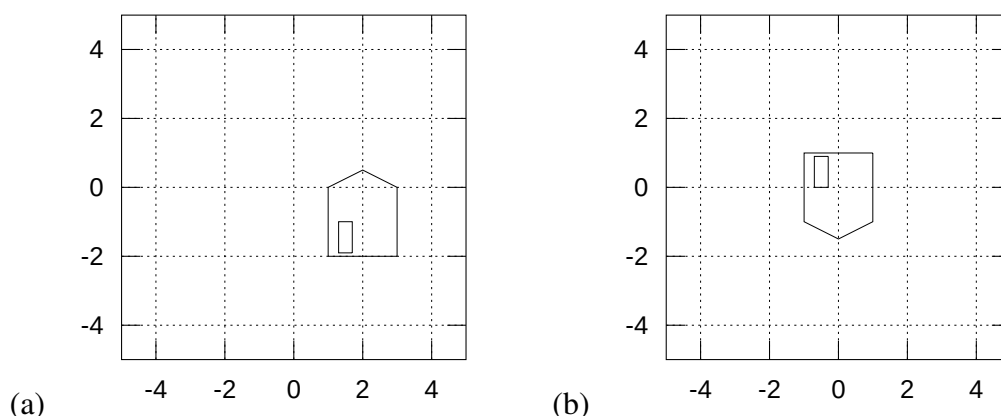
c4s13pic1

```
# Kierto
P = [1 -1; 1 1; 0 1.5; -1 1; -1 -1; 1 -1]';
Q = [-0.3 -0.9; -0.3 0.0; -0.7 0.0; -0.7 -0.9; -0.3 -0.9]';
R = [cos(pi/6) -sin(pi/6); sin(pi/6) cos(pi/6)];
hold off;
piirto(R*P); piirto(R*Q);
```

**Kuvan siirto.** Kuvan siirtäminen on kiertoa hankalampi operaatio. Jokaista kuvan pistettä on siirrettävä saman verran. Kuvan [I2](#) siirtäminen kaksi yksikköä oikealle ja yhden yksikön alaspäin vaatii vektorin  $\vec{w} = (2 \ -1)^T$  lisäämistä jokaisen nurkkapisteen koordinaatteihin. Seuraava koodi piirtää kuvan [I3\(a\)](#).

```
# Siirto
P = [1 -1; 1 1; 0 1.5; -1 1; -1 -1; 1 -1]';
Q = [-0.3 -0.9; -0.3 0.0; -0.7 0.0; -0.7 -0.9; -0.3 -0.9]';
w = [2 -1]';
[r,c] = size(P);
for j=1:c
    Puusi(:,j)=P(:,j)+w;
endfor
[r,c] = size(Q);
for j=1:c
    Quusi(:,j)=Q(:,j)+w;
endfor
hold off;
piirto(Puusi); piirto(Quusi);
```

Pian opimme helpomman tavan toteuttaa siirto, mutta se vaatii tietorakenteen uutta määrittelyä.



**Kuva 13.** (a)  $P$  ja  $Q$  siirrettynä kaksi oikealle ja yksi alas. (b)  $P$  ja  $Q$  peilattuina  $x$ -akselin suhteen.

c4s13pic2

**Kuvan peilaus.** Kuvan peilaus  $x$ -akselin suhteen onnistuu, kun data-matriisi kerrotaan matriisilla

$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Kuvan peilaus  $y$ -akselin suhteen onnistuu, kun data-matriisi kerrotaan matriisilla

$$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Seuraava koodi piirtää kuvan [c4s13pic2](#) [I3\(b\)](#), joka on kuvan [c4s13pic1](#) [I2\(a\)](#) peilikuva  $x$ -akselin suhteen.

```
# Peilaus
P = [1 -1; 1 1; 0 1.5; -1 1; -1 -1; 1 -1]';
Q = [-0.3 -0.9; -0.3 0.0; -0.7 0.0; -0.7 -0.9; -0.3 -0.9]';
R = [1 0; 0 -1];
hold off;
piirto(R*P); piirto(R*Q);
```

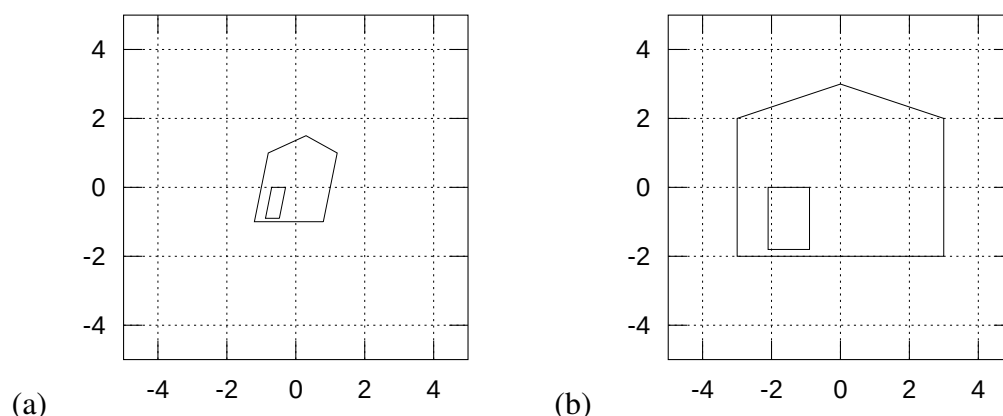
**Kuvan vääntö.** Kuvan vääntäminen on hankalammin hahmotettava operaatio. Siinä kuvan jokaisen pisteen koordinaattivektori kerrotaan matriisilla (tai sen transpoosilla)

$$\begin{pmatrix} 1 & v \\ 0 & 1 \end{pmatrix}.$$

Seuraava koodi piirtää kuvan [c4s13pic3](#) [I4\(a\)](#), joka on saatu vääntämällä kuvaa [c4s13pic1](#) [I2\(a\)](#).

```
# Vaanto
P = [1 -1; 1 1; 0 1.5; -1 1; -1 -1; 1 -1]';
Q = [-0.3 -0.9; -0.3 0.0; -0.7 0.0; -0.7 -0.9; -0.3 -0.9]';
R = [1 0.2; 0 1];
hold off;
piirto(R*P); piirto(R*Q);
```





c4s13pic3

**Kuva 14.** (a)  $P$  ja  $Q$  väännettyinä. (b)  $P$  ja  $Q$  skaalattuina kertoimin  $s_x = 3$  ja  $s_y = 2$ .

**Kuvan skaalaus (venytys).** Skaalaamisessa kuvan jokaisen pisteen  $x$ -koordinaatti kerrotaan luvulla  $s_x$  ja  $y$ -koordinaatti kerrotaan luvulla  $s_y$ . ( $x$ - ja  $y$ -koordinaatit voidaan siis skaalata eri kertoimilla.) Skaalaus saadaan aikaan kertomalla pisteiden paikkavektorit matriisilla

$$\begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix}.$$

Seuraava koodi piirtää kuvan [c4s13pic3](#) [l4\(b\)](#), joka on saatu kuvasta [c4s13pic1](#) [l2\(a\)](#) skaalaamalla  $x$ -koordinaatit kertoimella 3 ja  $y$ -koordinaatit kertoimella 2.

```
# Skaalaus
P = [1 -1; 1 1; 0 1.5; -1 1; -1 -1; 1 -1]';
Q = [-0.3 -0.9; -0.3 0.0; -0.7 0.0; -0.7 -0.9; -0.3 -0.9]';
R = [3 0; 0 2];
hold off;
piirto(R*P); piirto(R*Q);
```

**Tason affiinit kuvaukset.** Kuvaus tasolta tasolle on affiini, jos suoran pisteet kuvautuvat suoran pisteiksi. Affiini kuvaus voidaan aina esittää kiertojen, siirtojen, peilauksien, vääntöjen ja skaalauksien yhdisteenä. Tason siirto ei ole lineaarinen kuvaus, joten sitä ei voi toteuttaa matriisilla kertomisena kuten muut affiinit peruskuvaukset. Tämä saa laskemisen melko raskaaksi. Asiat yksinkertaistuvat, kun otamme käyttöön homogeeniset koordinaatit.

### 11.1.2 Tason pisteiden homogeeniset koordinaatit.

Sanomme, että  $(x, y)$ -tason pistettä  $(x, y)^T$  vastaa homogeeniset koordinaatit  $(wx, wy, w)^T$ , missä  $w \neq 0$ . Jokaista tason pistettä vastaa siis ääretön määrä homogeenisia koordinaat-

teja, mutta jokaista homogeenista koordinaattia vastaa yksikäsitteinen tason piste

$$\begin{pmatrix} x \\ y \end{pmatrix} \longrightarrow \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \sim \begin{pmatrix} wx \\ wy \\ w \end{pmatrix}, \quad \begin{pmatrix} u \\ v \\ w \end{pmatrix} \longrightarrow \begin{pmatrix} u/w \\ v/w \end{pmatrix}.$$

Homogeenisten koordinaattien avulla kaikki affiinit kuvaukset saadaan toteutettua matriisikertolaskuna:

### Rotaatio:

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} \mapsto \begin{pmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} \sim \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

### Siirto:

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} \mapsto \begin{pmatrix} 1 & 0 & u \\ 0 & 1 & v \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} = \begin{pmatrix} w(x+u) \\ w(y+v) \\ w \end{pmatrix} \sim \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} u \\ v \end{pmatrix}$$

### Peilaus:

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} \mapsto \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} = \begin{pmatrix} wx \\ -wy \\ w \end{pmatrix} \sim \begin{pmatrix} x \\ -y \end{pmatrix}$$

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} \mapsto \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} = \begin{pmatrix} -wx \\ wy \\ w \end{pmatrix} \sim \begin{pmatrix} -x \\ y \end{pmatrix}$$

### Vääntö:

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} \mapsto \begin{pmatrix} 1 & v & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} = \begin{pmatrix} w(x+vy) \\ wy \\ w \end{pmatrix} \sim \begin{pmatrix} x+vy \\ y \end{pmatrix}$$

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} \mapsto \begin{pmatrix} 1 & 0 & 0 \\ v & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} = \begin{pmatrix} wx \\ w(y+vx) \\ w \end{pmatrix} \sim \begin{pmatrix} x \\ y+vx \end{pmatrix}$$

### Skaalaus:

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} \mapsto \begin{pmatrix} u & 0 & 0 \\ 0 & v & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} = \begin{pmatrix} wux \\ wvy \\ w \end{pmatrix} \sim \begin{pmatrix} ux \\ vy \end{pmatrix}$$

Jos homogeeniset koordinaatit ovat  $(x, y, 0)^T$ , niin sanomme, että homogeeniset koordinaatit ovat ”ideaalireunan pisteen” homogeeniset koordinaatit. Ideaalireunan pisteellä ei ole valmista vastin pistettä  $(x, y)$ -tasossa. Lisäämme ideaalireunan pisteet tasoon ja lisäys alkaa saada merkitystä seuraavan ajatusleikin kautta.

Lähdetään  $(x, y)$ -tason pisteestä  $P = (u, v)^T \sim (u, v, 1)^T$ . Jos kerromme pisteen  $(x, y)$ -koordinaatit reaaliluvulla  $t > 1$ , niin saamme pisteen

$$P_t = t \cdot \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} tu \\ tv \end{pmatrix} \sim \begin{pmatrix} u \\ v \\ 1/t \end{pmatrix},$$

joka origosta katsottuna on samassa suunnassa kuin  $P$ , mutta kauempana. Rajalla, kun  $t \rightarrow \infty$ , päädyimme ideaalireunalle pisteeseen, jonka homogeeniset koordinaatit ovat  $(u, v, 0)^T$ . Kyseinen ideaalireunan piste on siis origosta katsottuna  $P$ :n suunnassa, mutta äärettömän kaukana.

**Esimerkki 53 (Kameran screen.)** Sijoitetaan videokamera  $(x, y, z)$ -koordinaatiston origoon siten, että kameran linssin optinen akseli osoittaa positiivisen  $z$ -akselin suuntaan. Kameran edessä on joukko kirkkaita valopisteitä  $P_k$  paikoissa

$$P_k = \begin{pmatrix} x_k \\ y_k \\ z_k \end{pmatrix}, z_k \geq 0.$$

Valopiste  $P_k$  sytyttää kameran screenille (kuvatason) pisteen  $(u_k, v_k)^T$ . Selvitä miten kuvataason pisteen koordinatit riippuvat valopisteen koordinaateista

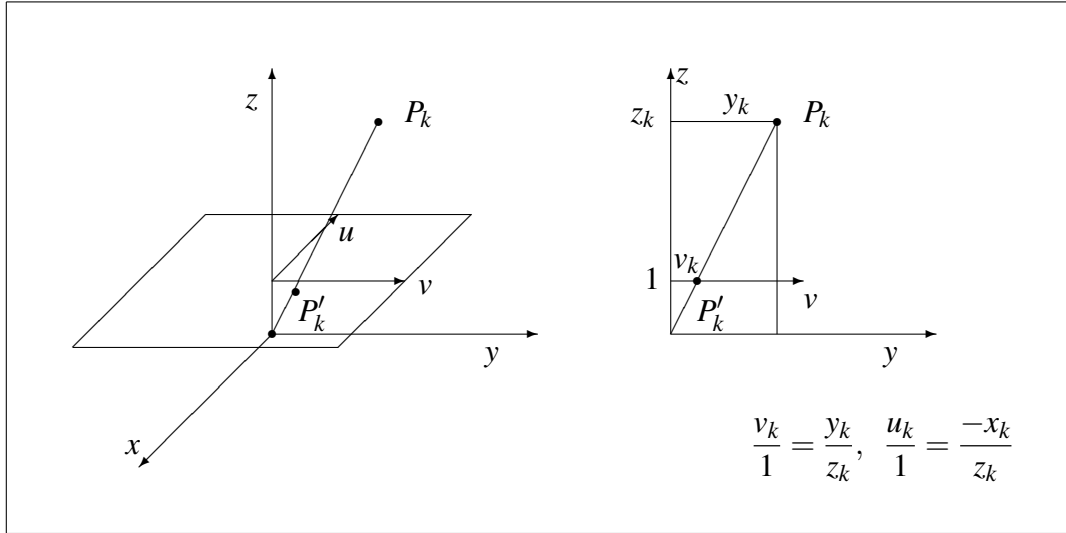
$$\begin{pmatrix} x_k \\ y_k \\ z_k \end{pmatrix} \mapsto \begin{pmatrix} u_k \\ v_k \end{pmatrix}.$$

Etsimme vastauksen, joka on oikea mahdollista rotaatiota ja skaalausta lukuunottamatta. (Tieto kameran paikasta ja optisen akselin suunnasta jättää vielä vapausasteita kameran kierron ja zoomauksen suhteen.)

Emme nyt paneudu kameran optiikkaan (emme edes camera obscuran), vaan teemme köyhän miehen ratkaisun ja pingotamme kameran eteen muovikalvon (kanvaasin), joka on kohtisuorassa optista akselia vastaan ja etäisyydellä 1 kameran linssistä. Kanvaasi yhtyy siis tasoon, jonka yhtälö on  $z = 1$ . Katsomme valopisteitä origosta ja merkitsemme valopisteet kanvaasiin tussilla. Merkintä kanvaasissa on siis pisteessä, jossa jana

valopisteestä  $P_k$  origoon leikkaa tason  $z = 1$ . Oheisesta kuvasta näkee yhdenmuotoisten kolmioiden perusteella, että kuvapisteen koordinaatit ovat

$$\begin{pmatrix} u_k \\ v_k \end{pmatrix} = \begin{pmatrix} -x_k/z_k \\ y_k/z_k \end{pmatrix} \sim \begin{pmatrix} -x_k/z_k \\ y_k/z_k \\ 1 \end{pmatrix} \sim \begin{pmatrix} -x_k \\ y_k \\ z_k \end{pmatrix}.$$



**Esimerkki 54 (Taloja kylän-raitilla.)** Piirrämme  $(x, y, z)$ -koordinaatistoon pitkän suorakulmion

$$\mathbf{T} = \begin{pmatrix} 0.5 & -0.5 & -0.5 & 0.5 & 0.5 \\ 0 & 0 & 100 & 100 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

kuvaamaan tienpätkää. Sijoitamme  $(x, y, z)$ -koordinaatistoon lisäksi kolme yksikkö-kuutiota esittämään tien varrella olevia taloja. Jokainen kuutio saadaan siirtämällä peruskuutiota

$$\begin{aligned} \mathbf{K}_1 &= \begin{pmatrix} 1 & 2 & 2 & 2 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 1 & 1 & 2 & 2 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \\ \mathbf{K}_2 &= \begin{pmatrix} 1 & 2 & 2 & 2 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 1 & 1 & 2 & 2 & 1 & 1 \\ 3 & 3 & 4 & 4 & 3 & 3 & 3 & 4 & 4 & 4 & 4 & 4 & 3 & 3 & 3 & 4 & 4 & 3 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \\ \mathbf{K}_3 &= \begin{pmatrix} 1 & 2 & 2 & 2 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 1 & 1 & 2 & 2 & 1 & 1 \\ 6 & 6 & 7 & 7 & 6 & 6 & 6 & 7 & 7 & 7 & 7 & 7 & 6 & 6 & 6 & 7 & 7 & 6 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \end{aligned}$$

Asetamme kameran pisteeseen  $C = (0, -5, 2)$ . Kameran optinen akseli (Look vector) olkoon  $\vec{a} = 10\vec{j} - \vec{k}$ . Kameran yläsuunta (Up vector) olkoon  $\vec{b} = \vec{k}$ . Millainen kuva syntyy kameran kuvatasoon?

Varustamme kameran lokaalilla koordinaatistolla  $\{O^*, \vec{i}^*, \vec{j}^*, \vec{k}^*\}$ . Uuden koordinaatiston origo  $O^*$  sijoitetaan samaan paikkaan kuin kameran linssi; siis pisteeseen jonka paikkavektori peruskoordinaatistossa on  $\overrightarrow{OO^*} = \vec{c} = (0, -5, 2)^T$ . Lokaalin koordinaatiston  $z$ -akselin suunta valitaan samaksi kuin optisen akselin suunta; siis  $\vec{k}^* = \vec{a}/\|\vec{a}\|$ . Kantavektori  $\vec{i}^*$  valitaan kohtisuoraksi optisen akselin ja yläsuunta-vektorin kanssa. Siis

$$\begin{aligned}\vec{k}^* &= \vec{a}/\|\vec{a}\| \\ \vec{i}^* &= (\vec{b} \times \vec{a})/\|\vec{b} \times \vec{a}\| \\ \vec{j}^* &= \vec{k}^* \times \vec{i}^*\end{aligned}$$

Kun paikallisen koordinaatiston kantavektorit on selvitetty, muodostamme niistä matriisin  $\mathbf{Q}$  sijoittamalla kantavektorit järjestyksessä sen sarakkeisiin,  $\mathbf{Q} = [\vec{i}^*, \vec{j}^*, \vec{k}^*]$ .

Seuraavaksi otamme käyttöön merkinnät, joiden avulla on helppo erottaa koordinaatit maailman-koordinaatistossa  $\{O, \vec{i}, \vec{j}, \vec{k}\}$  ja koordinaatit lokaalissa kamera-koordinaatistossa  $\{O^*, \vec{i}^*, \vec{j}^*, \vec{k}^*\}$ . Pisteellä  $P_k$  maailman-systeemissä paikkavektori  $\vec{x}_k$  ja kameran systeemissä paikkavektori  $\vec{\alpha}_k$ .

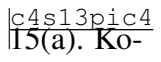
| systeemi | koordinaatit                                       | paikkavektori                                                                                                     |
|----------|----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| maailma  | $\vec{x}_k = (x_k, y_k, z_k)^T$                    | $\overrightarrow{OP} = x_k \vec{i} + y_k \vec{j} + z_k \vec{k} = \mathbf{I} \vec{x}_k$                            |
| kamera   | $\vec{\alpha}_k = (\alpha_k, \beta_k, \gamma_k)^T$ | $\overrightarrow{O^*P} = \alpha_k \vec{i}^* + \beta_k \vec{j}^* + \gamma_k \vec{k}^* = \mathbf{Q} \vec{\alpha}_k$ |

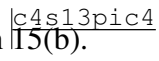
Käyttöön otettu merkintätapa (roomalaiset kirjaimet maailman-koordinaateille ja kreikkalaiset kirjaimet kameran-koordinaateille) ei ole standardi, vaan tähän hetkeen tehty ratkaisu, jolla helpotetaan seuraavien yhtälöiden hahmottamista.

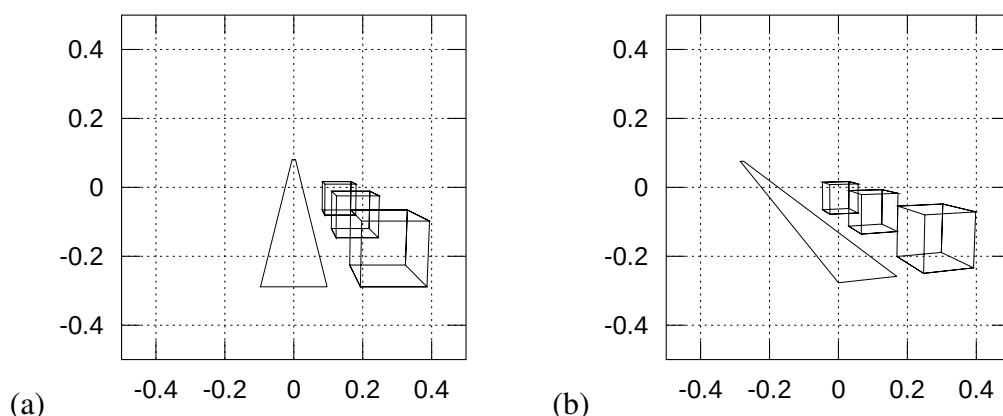
Valopisteen paikkavektorille on voimassa

$$\begin{aligned}\vec{x}_k &= \begin{pmatrix} x_k \\ y_k \\ z_k \end{pmatrix} = \overrightarrow{OP} = \overrightarrow{OO^*} + \overrightarrow{O^*P} = \vec{c} + \mathbf{Q} \vec{\alpha}_k \\ \Rightarrow \vec{\alpha}_k &= \mathbf{Q}^{-1}(\vec{x}_k - \vec{c})\end{aligned}$$

Vektori  $\vec{\alpha}_k$  on valopisteen koordinaatit kameran koordinaatistossa, ja samalla vastaavan kameran kuvatason pisteen homogeeniset koordinaatit. Ohjelma `raitti.m` laskee annetun datan perusteella jokaiselle valopisteelle kamera-koordinaatit ja systyttää pisteen screenille funktion `piirto` avulla.

Komento `raitti([0;-5;2],[0;10;-1],[0;0;1])` piirtää kuvan  15(a). Komento

`raitti([-2;-5;2],[3;10;-1],[0;0;1])` piirtää kuvan  15(b).



**Kuva 15.** (a) Kylän raitti, kun kamera on pisteessä  $(0, -5, 2)$  ja katsoo suuntaan  $10\vec{j} - \vec{k}$ . (b) Kylän raitti, kun kamera on pisteessä  $(-2, -5, 2)$  ja katsoo suuntaan  $(3\vec{i} + 10\vec{j} - \vec{k})$ .

c4s13pic4

**Esimerkki 55 (Kahden kuvan yhdistäminen (yksi valopiste).)** Olkoon kaksi kameraa sijoitettu seuraavasti

| kamera | paikka                                              | Look vector                                   | Up vector             |
|--------|-----------------------------------------------------|-----------------------------------------------|-----------------------|
| $K1$   | $\overrightarrow{OC_1} = \vec{c}_1 = (-4, -5, 1)^T$ | $\vec{a}_1 = 3\vec{i} + 10\vec{j} - \vec{k}$  | $\vec{b}_1 = \vec{k}$ |
| $K2$   | $\overrightarrow{OC_2} = \vec{c}_2 = (4, -5, 1)^T$  | $\vec{a}_2 = -3\vec{i} + 10\vec{j} - \vec{k}$ | $\vec{b}_2 = \vec{k}$ |

Yhdestä valopisteestä syntyy kameran  $K1$  screenille piste  $(\alpha_1, \beta_1)^T = (100/1024, 30/1024)^T$  ja kameran  $K2$  screenille piste  $(\alpha_2, \beta_2)^T = (25/1024, 35/1024)^T$ . Mitkä ovat valopisteen maailman-koordinaatit?

Kameran  $K1$  lokaalissa koordinaatistossa valopisteen koordinaatit ovat

$$\vec{\alpha}_1 = w_1 \begin{pmatrix} 100/1024 \\ 30/1024 \\ 1 \end{pmatrix} = t_1 \begin{pmatrix} 100 \\ 30 \\ 1024 \end{pmatrix} = t_1 \vec{u}_1,$$

Kameran  $K2$  lokaalissa koordinaatistossa valopisteen koordinaatit ovat

$$\vec{\alpha}_2 = w_2 \begin{pmatrix} 25/1024 \\ 35/1024 \\ 1 \end{pmatrix} = t_2 \begin{pmatrix} 25 \\ 35 \\ 1024 \end{pmatrix} = t_2 \vec{u}_2.$$

Edellisestä tiedämme, että valopisteen koordinaatteja maailman-koordinaatistossa ja kamera-koordinaatistoissa yhdistää yhtälöt

$$\begin{aligned} \vec{x} &= \vec{c}_1 + \mathbf{Q}_1 \vec{\alpha}_1 = \vec{c}_1 + t_1 \mathbf{Q}_1 \vec{u}_1 \\ \vec{x} &= \vec{c}_2 + \mathbf{Q}_2 \vec{\alpha}_2 = \vec{c}_2 + t_2 \mathbf{Q}_2 \vec{u}_2 \end{aligned}$$

Valopiste on siis siinä, missä kaksi suoraa  $S_1$  ja  $S_2$  leikkaavat toisiaan ( $S_1$  kulkee pisteen  $C_1$  kautta ja on vektorin  $\mathbf{Q}_1\vec{u}_1$  suuntainen ja  $S_2$  kulkee pisteen  $C_2$  kautta ja on vektorin  $\mathbf{Q}_2\vec{u}_2$  suuntainen).

### 11.1.3 Käytetyt koodit

Seuraavassa laatikossa on edellä käytetyn funktion `piirto.m` octave-koodi. Koodi on hieman mutkikas, koska se on laadittu niin, että matriisin sarakkeissa voi olla joko pisteiden tavalliset koordinaatit tai pisteiden homogeeniset koordinaatit.

```
function piirto(D, xmin=-5,xmax=5,ymin=-5,ymax=5)
[r,c] = size(P);
if r<3
    x = P(1,:);
    y = P(2,:);
else
    x = -P(1,:)./P(3,:);
    y = P(2,:)./P(3,:);
endif
subplot(2,2,1)
plot(x,y,'k-');
axis([xmin,xmax,ymin,ymax]);
set(gca,"dataaspectratio",[1,1,1]);
grid on;
endfunction
```

Funktio `raitti.m` sijoittaa kolmiulotteiseen koordinaatistoon tien ja kolme taloa sen varrelle. Funktion kutsu-parametreissä annetaan kameran paikka ja suunta. Funktio piirtää kameran kuvatasoon syntyvän kuvan.

```
function raitti(c,a,b)
# luodaan kylan-raitti
# -----
T = [0.5  -0.5  -0.5   0.5   0.5;
      0     0   100   100     0 ;
      0     0     0     0     0];
K1 = [1  2  2  2  2  2  1  1  2  2  1  1  1  1  2  2  1  1;
      0  0  1  1  0  0  0  1  1  1  1  1  0  0  0  1  1  0;
      0  0  0  1  1  0  0  0  0  1  1  0  0  1  1  1  1  1];
for j=1:18
    K2(:,j) = K1(:,j)+[0;3;0];
    K3(:,j) = K1(:,j)+[0;6;0];
endfor

# Lasketaan kameran kantavektorit Q(:,k)
```

```

# -----
Q(:,3) = a/norm(a,2);
Q(:,1) = cross(b,a)/norm(cross(b,a),2);
Q(:,2) = cross(Q(:,3),Q(:,1));

# Lasketaan kuvapisteen kameran koordinaatistossa
# -----
for j=1:5
    TC(:,j) = Q'*(T(:,j)-c);
endfor
for j=1:18
    KC1(:,j) = Q'*(K1(:,j)-c);
    KC2(:,j) = Q'*(K2(:,j)-c);
    KC3(:,j) = Q'*(K3(:,j)-c);
endfor

# Piirretään kuva
# -----
hold off;
piirto(TC,-0.5,0.5,-0.5,0.5);
piirto(KC1,-0.5,0.5,-0.5,0.5);
piirto(KC2,-0.5,0.5,-0.5,0.5);
piirto(KC3,-0.5,0.5,-0.5,0.5);
endfunction

```

## 11.2 OpenGL

### 11.3 Dynaamisen mallin tasapainon stabiilisuus

### 11.4 Lineaarista optimointia

## 11.5 Pienimmän neliösumman menetelmä

Sec:OLS

### 11.5.1 Approksimointi polynomilla

ma3f1

Olkoon lähtökohtana joukko havaintoja  $\{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$ . Muodostetaan polynomifunktio  $P(x)$  siten, että mahdollisimman tarkasti olisi voimassa

$$y_k \approx P(x_k).$$

Olkoon polynomi  $P(x)$  astetta  $n$ , eli

$$P(x) = b_0 + b_1x + b_2x^2 + \dots + b_nx^n \quad (11.1)$$



Määritetään kertoimet  $b_i$  siten, että neliösumma

$$f(\vec{b}) = \sum_{k=1}^m (P(x_k) - y_k)^2 = \sum_{k=1}^m (b_0 + b_1 x_k + \cdots + b_n x_k^n - y_k)^2 \quad (11.2)$$

saa pienimmän mahdollisen arvon.

Otamme käyttöön apuvektorit  $\vec{a}_k, k = 0, \dots, n$  siten, että

$$\vec{a}_k = \begin{pmatrix} x_1^k & x_2^k & \cdots & x_m^k \end{pmatrix}^T$$

Muodostamme näistä vektoreista  $m \times n$ -matriisin

$$\mathbf{A} = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ 1 & x_3 & x_3^2 & \cdots & x_3^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \cdots & x_m^n \end{pmatrix}, \quad (11.3)$$

Nyt

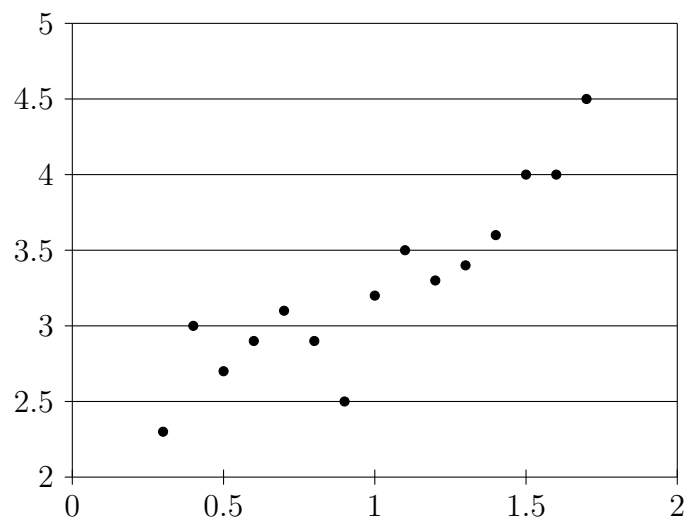
$$\begin{aligned} f(\vec{b}) &= \sum_{k=1}^m (P(x_k) - y_k)^2 = \sum_{k=1}^m ((\mathbf{A}\vec{b})_k - y_k)^2 \\ &= \|\mathbf{A}\vec{b} - \vec{y}\|_2^2 \end{aligned}$$

Etsimme siis  $\text{col}(\mathbf{A})$ :n vektoria  $\vec{A}\vec{b}$  joka on mahdollisimman lähellä vektoria  $\vec{y}$ . Lauseen [\[c4s21thdatsproj\]](#) (sivu ??) mukaan ratkaisu on

$$\begin{aligned} \vec{A}\vec{b} &= \mathbf{A}\mathbf{A}^\dagger \vec{y} \\ \Leftrightarrow \vec{b} &= \mathbf{A}^\dagger \vec{y} \end{aligned}$$

ma3fes1

**Esimerkki 56** Olemme tehneet havainnot (ks. myös kuva [\[c4s22fig1\]](#) (16))

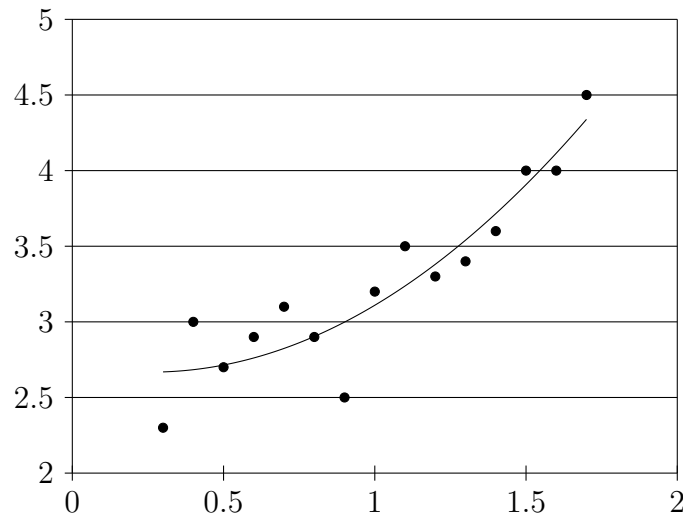


c4s22fig1

Kuva 16. Esimerkin <sup>ma3fes1</sup>(56)  $(x, y)$ -data.

| $x_k$ | $y_k$ |
|-------|-------|
| 0.3   | 2.3   |
| 0.4   | 3.0   |
| 0.5   | 2.7   |
| 0.6   | 2.9   |
| 0.7   | 3.1   |
| 0.8   | 2.9   |
| 0.9   | 2.5   |
| 1.0   | 3.2   |
| 1.1   | 3.5   |
| 1.2   | 3.3   |
| 1.3   | 3.4   |
| 1.4   | 3.6   |
| 1.5   | 4.0   |
| 1.6   | 4.0   |
| 1.7   | 4.5   |

Jos haluamme sovittaa pienimmän neliösumman menetelmällä toisen asteen polynomi-funktion pisteisiin, niin matriisit  $\mathbf{A}$  ja  $\mathbf{A}^\dagger = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$  ovat



**Kuva 17.** Esimerkin [56](#) havaintopisteet ja niihin PNS-menetelmällä sovitetun polynomifunktion kuvaaja.

ma3ffig2

$$\mathbf{A} = \begin{pmatrix} 1 & 0.3 & 0.09 \\ 1 & 0.4 & 0.16 \\ 1 & 0.5 & 0.25 \\ 1 & 0.6 & 0.36 \\ 1 & 0.7 & 0.49 \\ 1 & 0.8 & 0.64 \\ 1 & 0.9 & 0.81 \\ 1 & 1.0 & 1.00 \\ 1 & 1.1 & 1.21 \\ 1 & 1.2 & 1.44 \\ 1 & 1.3 & 1.69 \\ 1 & 1.4 & 1.96 \\ 1 & 1.5 & 2.25 \\ 1 & 1.6 & 2.56 \\ 1 & 1.7 & 2.89 \end{pmatrix} \quad \text{ja} \quad \mathbf{A}^\dagger = \begin{pmatrix} 0.915 & -1.721 & 0.735 \\ 0.623 & -1.055 & 0.420 \\ 0.370 & -0.486 & 0.154 \\ 0.157 & -0.014 & -0.065 \\ -0.017 & 0.362 & -0.234 \\ -0.151 & 0.640 & -0.356 \\ -0.246 & 0.821 & -0.428 \\ -0.301 & 0.905 & -0.452 \\ -0.317 & 0.892 & -0.428 \\ -0.294 & 0.782 & -0.356 \\ -0.231 & 0.576 & -0.234 \\ -0.129 & 0.272 & -0.065 \\ 0.013 & -0.128 & 0.154 \\ 0.194 & -0.626 & 0.420 \\ 0.415 & -1.221 & 0.735 \end{pmatrix}^T \quad (11.4)$$

Havaintopisteisiin sovitetun polynomifunktion kerroinvektoriksi saadaan

$$\vec{b} = \mathbf{A}^\dagger \vec{y} = \begin{pmatrix} 2.7217 \\ -0.4167 \\ 0.8048 \end{pmatrix} \quad (11.5)$$

Kuvaan [17](#) on piirretty havaintopisteet ja polynomin  $P(x) = b_1 + b_2x + b_3x^2$  kuvaaja.

ma3ffig2

## 11.5.2 Lineaarisen mallin sovitus

ma3f2

Yritys tuottaa eläinten rehua ja sen suurimmat asiakkaat ovat turkistarhoja ja suurkanaloita. Tulevaa neljännevuotta suunniteltaessa on käytettävissä koko turkistarha-alan liikevaihto neljännesvuosittain ( $u_k, k = 1, \dots, 8$ ) kahden viime vuoden ajalta sekä tulevan neljännesvuoden ennuste  $\tilde{u}_9$ . Vastaavasti on käytettävissä siipikarjan kasvattajien yhteenlaskettu liikevaihto neljännesvuosittain ( $w_k, k = 1, \dots, 8$ ) kahden viime vuoden ajalta sekä tulevan neljännesvuoden ennuste  $\tilde{w}_9$ . Rehun myynti ( $y_k, k = 1, \dots, 8$ ) on myös tiedossa neljännesvuosittain kahden viime vuoden ajalta. Yritys haluaa laskea mahdollisimman hyvän ennusteen tulevan neljännesvuoden myynnille  $\hat{y}_9$ .

Tietty osuus kaikista turkistarhoista on yrityksemme asiakkaita ja vastaavasti tietty osa kaikista suurkanaloista on yrityksemme asiakkaita. Siksi on luonnollista olettaa, että sopivasti valituilla kertoimilla  $x_1, x_2, x_3$ , pätee

$$x_1 + x_2 u_k + x_3 w_k = \hat{y}_k \approx y_k \quad (11.6) \quad \text{ma3f2eq1}$$

Kaava (11.6) on lineaarinen selitysmalli, jossa myynti  $y_k$  on selitettävä muuttuja ja asiakastoimialojen liikevaihdot  $u_k$  ja  $w_k$  ovat selittäviä muuttujia. Jos kertoimet  $x_1, x_2, x_3$  saadaan valittua siten, että

$$\hat{y}_k \approx y_k, \forall k \quad (11.7)$$

$$\text{eli } \|\hat{\vec{y}} - \vec{y}\| \text{ on pienin mahdollinen} \quad (11.8)$$

niin saamme hyvän ennusteen tulevan neljännesvuoden myynnille asettamalla

$$\hat{y}_9 = x_1 + x_2 \tilde{u}_9 + x_3 \tilde{w}_9. \quad (11.9)$$

Yhtälö (11.8) voidaan kirjoittaa muotoon

$$\|\mathbf{A}\vec{x} - \vec{y}\|_2 \text{ on pienin mahdollinen,} \quad (11.10) \quad \text{ma3f2eq5}$$

missä

$$\mathbf{A} = \begin{pmatrix} 1 & u_1 & w_1 \\ 1 & u_2 & w_2 \\ 1 & u_3 & w_3 \\ 1 & u_4 & w_4 \\ 1 & u_5 & w_5 \\ 1 & u_6 & w_6 \\ 1 & u_7 & w_7 \\ 1 & u_8 & w_8 \end{pmatrix} \quad (11.11)$$

Ratkaisemme siis minimointitehtävän

$$\begin{aligned} \min f(\vec{x}) &= \|\mathbf{A}\vec{x} - \vec{y}\|_2^2 \\ \Leftrightarrow \vec{x} &= (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \vec{y} = \mathbf{A}^\dagger \vec{y} \end{aligned} \quad (11.12)$$

Jos siis  $\mathbf{A}$  on ”a matrix of full rank”, niin (11.12) antaa parhaan lineaarisen selitysmallin kertoimet.

**Esimerkki 57** Olkoon alussa mainitun eläinrehua valmistavan yrityksen tilannetta kuvaavat aikasarjat  $(u_k)$ ,  $(w_k)$  ja  $(y_k)$  seuraavan taulukon mukaiset

| $k$ | $u_k$  | $w_k$  | $y_k$  |
|-----|--------|--------|--------|
| 1   | 535,53 | 96,37  | 143,62 |
| 2   | 404,93 | 118,33 | 110,84 |
| 3   | 329,68 | 132,77 | 115,94 |
| 4   | 180,52 | 212,48 | 94,34  |
| 5   | 224,70 | 237,99 | 111,78 |
| 6   | 250,57 | 236,18 | 113,44 |
| 7   | 254,89 | 284,45 | 141,12 |
| 8   | 393,06 | 349,95 | 170,14 |
| 9   | 551,26 | 321,69 |        |

Kaavasta (11.12) saamme kertoimet optimaaliselle mallille

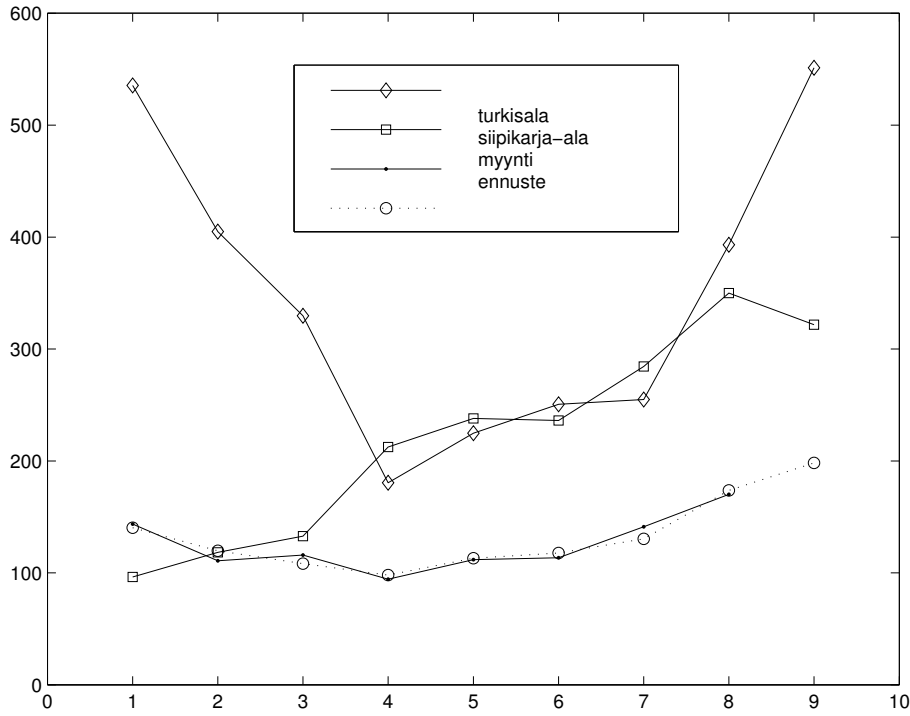
$$\hat{y}_k = 10.5293 + 0.1986u_k + 0.2432w_k \quad (11.13)$$

Mallin hyvyttä voimme nyt tutkia piirtämällä toteutuneet ja mallin ennustama myynti samaan kuvaan (ks kuva 18). Malli ei välttämättä aina onnistu ennustamisessa täsmällisesti, mutta silti se antaa melko luotettavan pohjan tulevan kauden suunnittelulle.

ma3fes2

### 11.5.3 Häiriöiden vaikutus

Seuraavaksi tutkimme, mikä vaikutus on havaintosarjaan sisältyvällä satunnaisvirheellä. Edellä myynti neljännesvuosittain  $\vec{y} = (y_1, y_2, \dots, y_8)$  sisältää satunnaista vaihtelua, joka ei johdu asiakastoimialojen kysynnästä, vaan muista syistä. (tuotannolliset häiriöt,



**Kuva 18.** Esimerkin [57](#) asiakastoimialojen liikevaihdot sekä yrityksen toteutunut myynti ja selitysmallin ennustama myynti.

saamisten siirtymiset jaksolta toiselle, yms.) Oletamme että edellä käytetty myyntiaika-sarja on ”oikea”,  $y_j^{oikea}$  plus ”häiriö”  $\varepsilon_j$ , missä häiriö on noin  $\pm 2\%$  oikeasta. Oletamme peräkkäiset häiriöt toisistaan riippumattomiksi ja normaalisti jakautuneiksi

$$y_j = y_j^{oikea} + \varepsilon_j, \quad \varepsilon \sim N(0, \sigma), \sigma \approx 2.0$$

Koska emme tunne häiriötermien tarkkoja arvoja, emme voi laskea tarkasti. Mutta kauas ei osu seuraava tarkastelu.

Saamamme arvio mallin kertoimille on

$$\begin{aligned} \vec{x} &= \mathbf{A}^\dagger \vec{y} \\ \Leftrightarrow \vec{x}^{oikea} + \vec{x}^{hairio} &= \mathbf{A}^\dagger \vec{y}^{oikea} + \mathbf{A}^\dagger \vec{\varepsilon} \\ \Leftrightarrow \vec{x}^{hairio} &= \mathbf{A}^\dagger \vec{\varepsilon} \end{aligned}$$

$$\vec{x}^{hairio} = \begin{pmatrix} -0.2888 & 0.0019 & -0.0009 \\ 0.3071 & 0.0004 & -0.0014 \\ 0.6373 & -0.0005 & -0.0017 \\ 0.9155 & -0.0018 & -0.0010 \\ 0.4712 & -0.0010 & -0.0001 \\ 0.3344 & -0.0007 & 0.0001 \\ -0.0463 & -0.0003 & 0.0012 \\ -1.3305 & 0.0020 & 0.0038 \end{pmatrix}^T \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_8 \end{pmatrix}$$

Siis

$$\begin{aligned}
 x_1^{hairio} &= -0.2888\varepsilon_1 + 0.3071\varepsilon_2 + 0.6373\varepsilon_3 + 0.9155\varepsilon_4 + 0.4712\varepsilon_5 + \\
 &\quad \dots + 0.3344\varepsilon_6 - 0.0463\varepsilon_7 - 1.3305\varepsilon_8 \\
 x_2^{hairio} &= +0.0019\varepsilon_1 + 0.0004\varepsilon_2 - 0.0005\varepsilon_3 - 0.0018\varepsilon_4 - 0.0010\varepsilon_5 + \\
 &\quad \dots - 0.0007\varepsilon_6 - 0.0003\varepsilon_7 + 0.0020\varepsilon_8 \\
 x_3^{hairio} &= -0.0009\varepsilon_1 - 0.0014\varepsilon_2 - 0.0017\varepsilon_3 - 0.0010\varepsilon_4 - 0.0001\varepsilon_5 + \\
 &\quad \dots + 0.0001\varepsilon_6 + 0.0012\varepsilon_7 + 0.0038\varepsilon_8
 \end{aligned}$$

Koska häiriötermit  $\varepsilon_j \sim N(0, \sigma)$  ovat normaalisti jakautuneita, niin niiden lineaarikombinaatiot ovat myös normaalisti jakautuneita, ja vastaavat varianssit ovat

$$\begin{aligned}
 \sigma_1^2 &= (0.2888^2 + 0.3071^2 + 0.6373^2 + 0.9155^2 + 0.4712^2 + \\
 &\quad \dots + 0.3344^2 + 0.0463^2 + 1.3305^2)\sigma^2 \\
 &= \|a_{1\bullet}^\dagger\|_2^2 \sigma^2 = 14.1135 \rightarrow \sigma_1 \approx 3.76 \\
 \sigma_2^2 &= (0.0019^2 + 0.0004^2 + 0.0005^2 + 0.0018^2 + 0.0010^2 + \\
 &\quad \dots + 0.0007^2 + 0.0003^2 + 0.0020^2)\sigma^2 \\
 &= \|a_{2\bullet}^\dagger\|_2^2 \sigma^2 = 0.000051872 \rightarrow \sigma_2 \approx 0.0072 \\
 \sigma_3^2 &= (0.0009^2 + 0.0014^2 + 0.0017^2 + 0.0010^2 + 0.0001^2 + \\
 &\quad \dots + 0.0001^2 + 0.0012^2 + 0.0038^2)\sigma^2 \\
 &= \|a_{3\bullet}^\dagger\|_2^2 \sigma^2 = 0.000093090 \rightarrow \sigma_3 \approx 0.00965
 \end{aligned}$$

Jos haluamme antaa mallin kertoimille arviot, niin että osumme oikeaan 90% varmuudella, niin kukin kolme kerrointa tulee arvioida 96.55% tarkkuudella ( $\mu \pm 2.12\sigma$ ). Siis

$$\begin{aligned}
 x_1^{oikea} &\in [x_1 - 2.12\sigma_1, x_1 + 2.12\sigma_1] \rightarrow x_1^{oikea} = 10.5 \pm 8,0 \\
 x_2^{oikea} &\in [x_2 - 2.12\sigma_2, x_2 + 2.12\sigma_2] \rightarrow x_2^{oikea} = 0.199 \pm 0,015 \\
 x_3^{oikea} &\in [x_3 - 2.12\sigma_3, x_3 + 2.12\sigma_3] \rightarrow x_3^{oikea} = 0.243 \pm 0,021
 \end{aligned}$$

Seuraavaksi pohdimme selittäviin aikasarjoihin mahdollisesti sisältyvien häiriöiden vaikutusta lineaarisen mallin kertoimiin. Olkoon nyt

$$\begin{aligned}
 u_j &= u_j^{oikea} + \eta_j \\
 w_j &= w_j^{oikea} + \xi_j
 \end{aligned}$$

Merkitsemme nyt

$$\begin{aligned}\mathbf{A} &= \begin{pmatrix} 1 & u_1 & w_1 \\ 1 & u_2 & w_2 \\ \vdots & \vdots & \vdots \\ 1 & u_8 & w_8 \end{pmatrix} = \begin{pmatrix} 1 & u_1^{oikea} & w_1^{oikea} \\ 1 & u_2^{oikea} & w_2^{oikea} \\ \vdots & \vdots & \vdots \\ 1 & u_8^{oikea} & w_8^{oikea} \end{pmatrix} + \begin{pmatrix} 0 & \eta_1 & \xi_1 \\ 0 & \eta_2 & \xi_2 \\ \vdots & \vdots & \vdots \\ 0 & \eta_8 & \xi_8 \end{pmatrix} \\ &= \mathbf{A}^{oikea} + \mathbf{E}.\end{aligned}$$

Häiriöiden takia mallin kertoimet toteuttavat nyt yhtälöt

$$\begin{aligned}\vec{x} &= \mathbf{A}^\dagger \vec{y} \\ (\mathbf{A}^T \mathbf{A}) \vec{x} &= \mathbf{A}^T \vec{y} \\ (\mathbf{A}^{oikea T} + \mathbf{E}^T)(\mathbf{A}^{oikea} + \mathbf{E}) \vec{x} &= (\mathbf{A}^{oikea T} + \mathbf{E}^T) \vec{y} \\ (\mathbf{A}^{oikea T} \mathbf{A}^{oikea} + \mathbf{A}^{oikea T} \mathbf{E} + \mathbf{E}^T \mathbf{A}^{oikea} + \mathbf{E}^T \mathbf{E}) \vec{x} &= \mathbf{A}^{oikea T} \vec{y} + \mathbf{E}^T \vec{y}\end{aligned}$$

Jos virhetermit ovat satunnaisia siten, että ne eivät yhteisvaihtelee (kovarioi) selittävien eikä selitettävän aikasarjan kanssa, niin

$$\mathbf{E}^T \mathbf{A}^{oikea} \approx \mathbf{0}, \quad \mathbf{E}^T \vec{y} \approx \vec{0}.$$

Jos aikasarjat ovat riittävän pitkät niin edellä esitetyt arviot pitävät hyvin paikkansa, ja vaikka pientä virhettä syntyisikin, niin emme tee systemaattista virhettä, jos arvoimme että

$$\begin{aligned}\mathbf{A}^T \mathbf{A} &= \mathbf{A}^{oikea T} \mathbf{A}^{oikea} + \mathbf{E}^T \mathbf{E} \\ \mathbf{A}^T \vec{y} &= \mathbf{A}^{oikea T} \vec{y}\end{aligned}$$

Siis lopulta voimme arvioida

$$\begin{aligned}\vec{x}^{oikea} &= (\mathbf{A}^{oikea T} \mathbf{A}^{oikea})^{-1} \mathbf{A}^{oikea T} \vec{y} \\ &= (\mathbf{A}^T \mathbf{A} - \mathbf{E}^T \mathbf{E})^{-1} \mathbf{A}^T \vec{y}\end{aligned} \tag{11.14}$$

Jos nyt virhe-sarjat ovat toisistaan riippumattomia,  $\eta_j \sim N(0, \sigma_1) = N(0, 1)$  ja  $\xi_j \sim N(0, \sigma_2) = N(0, 5)$ , niin vaikka emme tunne yksittäisten virhetermien arvoja, niin voimme silti arvioida

$$\mathbf{E}^T \mathbf{E} \approx \begin{pmatrix} 0 & 0 & 0 \\ 0 & n\sigma_1^2 & 0 \\ 0 & 0 & n\sigma_2^2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 200 \end{pmatrix}$$



Siis

$$\begin{aligned}
 \vec{x}^{oikea} &= (\mathbf{A}^T \mathbf{A} - \mathbf{E}^T \mathbf{E})^{-1} \mathbf{A}^T \vec{y} \\
 &= \left( \begin{pmatrix} 10 & 2570 & 1670 \\ 2570 & 924780 & 504360 \\ 1670 & 504360 & 401860 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 200 \end{pmatrix} \right)^{-1} \mathbf{A}^T \vec{y} \\
 &= \begin{pmatrix} 10.1601 \\ 0.1990 \\ 0.2443 \end{pmatrix}
 \end{aligned}$$

Korjaus, jonka saamme mallin kertoimiin, ei ole kovin suuri, mutta selvästi havaittava

|       |         |   |         |
|-------|---------|---|---------|
| $x_1$ | 10.5293 | → | 10.1601 |
| $x_2$ | 0.1986  | → | 0.1990  |
| $x_3$ | 0.2432  | → | 0.2443  |

## 11.6 Mallin nopea estimointi

Sec:FastEst

Seuraavassa tutkitaan systeemiä, jossa laitteen tila mitataan peräkkäisinä ajan hetkinä  $t_j = t_0 + j \cdot \Delta t$ . Hetkellä  $t_j$  mitattu systeemin tila esitetään vektorina

$$\vec{x}_j = \begin{pmatrix} x_j & y_j \end{pmatrix}.$$

(Käytämme nyt vaakavektoria! Syy paljastuu pian.)

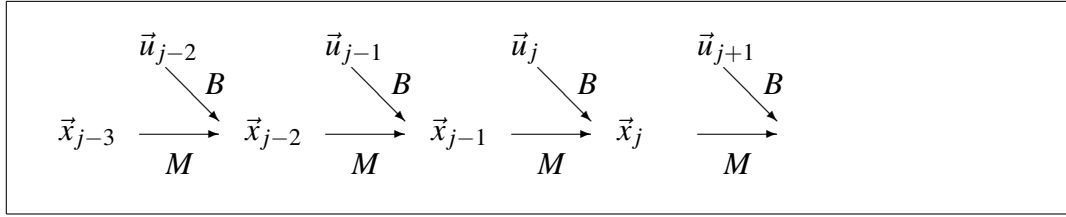
Oletamme, että tila  $\vec{x}_j$  riippuu edellisestä tilasta  $\vec{x}_{j-1}$  ja ohjausvektorista  $\vec{u}_j = (v_j \ w_j)$  seuraavasti

$$\vec{x}_j = \vec{x}_{j-1} \mathbf{M} + \vec{u}_j \mathbf{B} \quad (11.15)$$

$$\Leftrightarrow \begin{pmatrix} x_j & y_j \end{pmatrix} = \begin{pmatrix} x_{j-1} & y_{j-1} \end{pmatrix} \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} + \begin{pmatrix} v_j & w_j \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}$$

$$\Leftrightarrow \begin{cases} x_j = m_{11}x_{j-1} + m_{21}y_{j-1} + b_{11}v_j + b_{21}w_j \\ y_j = m_{12}x_{j-1} + m_{22}y_{j-1} + b_{12}v_j + b_{22}w_j \end{cases}.$$

Oletamme nyt, että matriisi  $\mathbf{B}$  tunnetaan tarkasti, mutta matriisia  $\mathbf{M}$  joudumme nyt estimoida. Toisin sanoen joka vaiheessa valitsemme matriisin  $\mathbf{M}$  niin, että malli (11.15) sopii havaittuihin arvoihin mahdollisimman tarkasti. c5slegproK



Teemme ensin sarjan mittauksia  $(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_j), (\vec{u}_1, \vec{u}_2, \dots, \vec{u}_j)$ . Näiden tietojen perusteella määritetään matriisi  $\mathbf{M}$ , jolla malli (II.15) sopii havaintoihin mahdollisimman tarkasti. Kun ohjausvektorin seuraava arvo  $\vec{u}_{j+1}$  saadaan selville, voidaan mallin avulla laskea ennuste uudeksi tilaksi  $\hat{\vec{x}}_{j+1}$ . Kun oikea arvo  $\vec{x}_{j+1}$  saadaan mitattua päivitetään malli (uusi parempi  $\mathbf{M}$ ).

Joka aika-askeleella toistuu sykli: uusi ohjaus  $\vec{u}_{j+2} \rightarrow$  uusi ennuste  $\hat{\vec{x}}_{j+2}$ , uusi mitausarvo  $\vec{x}_{j+2} \rightarrow$  päivitetty malli  $\mathbf{M}$ , jne...

*Miksi nähdä näin suuri vaiva:*

- (1) Joka aika-askeleella on käytettävissä paras mahdollinen malli.
- (2) Pienellä kehittelyllä seuraavassa kuvattu menettely saadaan adaptiiviseksi niin, että menettely sallii dynamiikan hitaan muuttumisen. Malli päivittyy perässä ja ennusteet perustuvat ajan tasalla olevaan malliin.
- (3) Edelleen pienellä kehittelyllä menettely voidaan saada häiriöitä sietäväksi niin, että ennustesarja  $(\hat{\vec{x}}_1, \hat{\vec{x}}_2, \dots, \hat{\vec{x}}_n)$  kuvaa todellista tilakehitystä paremmin kuin häiriöitä sisältävä mittaussarja  $(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_n)$ .

Aika-askeleella  $j$  mallin (II.15) matriisin  $\mathbf{M}_j$  estimointi perustuu yhtälöön

$$\underbrace{\begin{pmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_{j-1} & y_{j-1} \end{pmatrix}}_{=\mathbf{A}_j} \mathbf{M}_j = \underbrace{\begin{pmatrix} x_2 & y_2 \\ x_3 & y_3 \\ \vdots & \vdots \\ x_j & y_j \end{pmatrix}}_{=\mathbf{Y}_j} - \underbrace{\begin{pmatrix} v_2 & w_2 \\ v_3 & w_3 \\ \vdots & \vdots \\ v_j & w_j \end{pmatrix}}_{=\mathbf{U}_j} \mathbf{B}$$

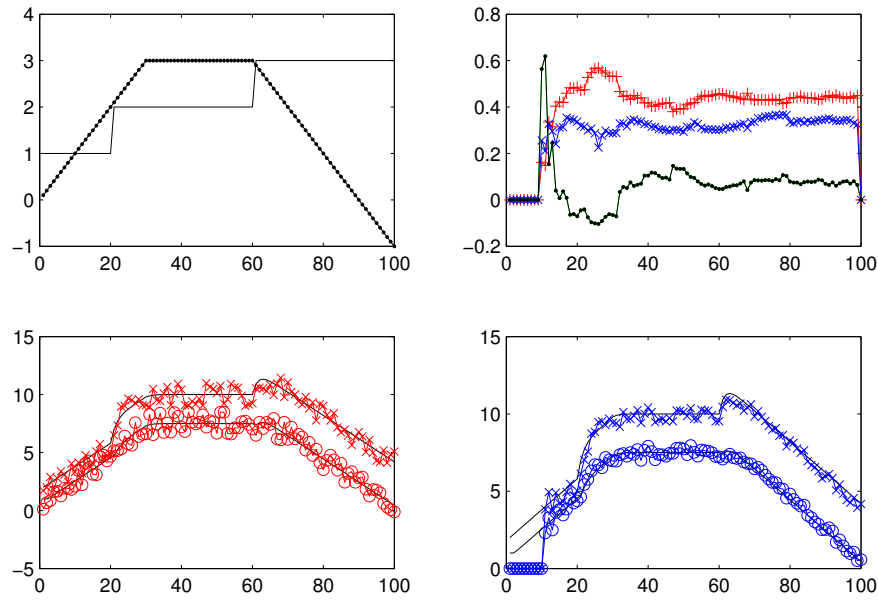
josta  $\mathbf{M}_j$  ratkaistaan yhtälöstä

$$\mathbf{A}_j^T \mathbf{A}_j \mathbf{M}_j = \mathbf{A}_j^T \mathbf{Y}_j - \mathbf{A}_j^T \mathbf{U}_j \mathbf{B} \quad (11.16)$$

$$\mathbf{M}_j = (\mathbf{A}_j^T \mathbf{A}_j)^{-1} \mathbf{A}_j^T (\mathbf{Y}_j - \mathbf{U}_j \mathbf{B}). \quad (11.17)$$

Seuraavan tilan ennuste saadaan kaavalla

$$\begin{pmatrix} \hat{x}_{j+1} & \hat{y}_{j+1} \end{pmatrix} = \begin{pmatrix} x_j & y_j \end{pmatrix} \mathbf{M}_j + \begin{pmatrix} v_{j+1} & w_{j+1} \end{pmatrix} \mathbf{B}. \quad (11.18)$$



**Kuva 19.** Esimerkin aikasarjat. (1)  $v_t$  (.),  $w_t$  (—) (2)  $m_{11}$  (+),  $m_{12}$  (×),  $m_{21}$  (·) (3)  $x_t$  (×),  $y_t$  (o) (4)  $\hat{x}_t$  (×),  $\hat{y}_t$  (o).

**Esimerkki 58** Kuvassa on erään systeemin aikasarjat. Osakuvassa (1) on ohjaussignaalit  $v_t$  ja  $w_t$ . Osassa (2) on matriisin  $\mathbf{M}$  koordinaatit estimoinnin ajan (kaksi koordinaattisarjaa ovat niin lähellä, että vain kolme on nähtävissä kuvassa). Osassa (3) ovat havaitut arvot  $x_t$  (×) ja  $y_t$  (o). Osassa (4) ovat estimoidut arvot  $\hat{x}_t$  (×) ja  $\hat{y}_t$  (o). Yllättävä seikka on se, että estimoidut arvot ovat jopa havaittuja ”siistimmät”.

Edellisen esimerkin ohjaamina teemme vielä yhden muutoksen edellä tarkasteltuun mallin estimointialgoritmiin. Merkitään

$$\begin{aligned} \mathbf{G}_j &= \mathbf{A}_j^T \mathbf{A}_j \\ \Rightarrow \mathbf{G}_{j+1} &= \mathbf{A}_{j+1}^T \mathbf{A}_{j+1} = \left( \mathbf{A}_j^T \begin{vmatrix} x_j \\ y_j \end{vmatrix} \right) \left( \begin{vmatrix} x_j & y_j \end{vmatrix} \right) \\ &= \mathbf{A}_j^T \mathbf{A}_j + \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} x_j & y_j \end{pmatrix} = \mathbf{G}_j + \begin{pmatrix} x_j^2 & x_j y_j \\ x_j y_j & y_j^2 \end{pmatrix} \end{aligned}$$

Vastaavasti

$$\begin{aligned} \mathbf{A}_{j+1}^T (\mathbf{Y}_{j+1} - \mathbf{U}_{j+1} \mathbf{B}) &= \left( \mathbf{A}_j^T \begin{vmatrix} x_j \\ y_j \end{vmatrix} \right) \left( \left( \begin{vmatrix} \mathbf{Y}_j \\ x_{j+1} & y_{j+1} \end{vmatrix} \right) - \left( \begin{vmatrix} \mathbf{U}_j \\ v_{j+1} & w_{j+1} \end{vmatrix} \right) \mathbf{B} \right) \\ &= \mathbf{A}_j^T \mathbf{Y}_j + \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} x_{j+1} & y_{j+1} \end{pmatrix} \\ &\quad \dots - \mathbf{A}_j^T \mathbf{U}_j \mathbf{B} - \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} v_{j+1} & w_{j+1} \end{pmatrix} \mathbf{B} \end{aligned}$$

Olkoon  $\mathbf{M}_{j+1} = \mathbf{M}_j + \Delta\mathbf{M}$ . Näillä merkinnöillä yhtälö (11.16) menee aika-askeleella  $j+1$  muotoon

$$\begin{aligned}
 \mathbf{A}_{j+1}^T \mathbf{A}_{j+1} (\mathbf{M}_j + \Delta\mathbf{M}) &= \mathbf{A}_{j+1}^T \mathbf{Y}_{j+1} - \mathbf{A}_{j+1}^T \mathbf{U}_{j+1} \mathbf{B} \\
 \Leftrightarrow \mathbf{A}_j^T \mathbf{A}_j \mathbf{M}_j + \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} x_j & y_j \end{pmatrix} \mathbf{M}_j + \mathbf{G}_{j+1} \Delta\mathbf{M} \\
 &= \mathbf{A}_j^T \mathbf{Y}_j + \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} x_{j+1} & y_{j+1} \end{pmatrix} - \mathbf{A}_j^T \mathbf{U}_j \mathbf{B} - \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} v_{j+1} & w_{j+1} \end{pmatrix} \mathbf{B} \\
 \Leftrightarrow \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} x_j & y_j \end{pmatrix} \mathbf{M}_j + \mathbf{G}_{j+1} \Delta\mathbf{M} &= \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} x_{j+1} & y_{j+1} \end{pmatrix} - \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} v_{j+1} & w_{j+1} \end{pmatrix} \mathbf{B} \\
 \Leftrightarrow \mathbf{G}_{j+1} \Delta\mathbf{M} &= \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} x_{j+1} - \hat{x}_{j+1} & y_{j+1} - \hat{y}_{j+1} \end{pmatrix}
 \end{aligned}$$

Mallin päivitys  $\mathbf{M}_j \rightarrow \mathbf{M}_{j+1}$  voidaan siis esittää seuraavasti:

$$\begin{aligned}
 (1) \quad \begin{pmatrix} \hat{x}_{j+1} & \hat{y}_{j+1} \end{pmatrix} &= \begin{pmatrix} x_j & y_j \end{pmatrix} \mathbf{M}_j + \begin{pmatrix} v_{j+1} & w_{j+1} \end{pmatrix} \mathbf{B}, \\
 (2) \quad \mathbf{G}_{j+1} &= \mathbf{G}_j + \begin{pmatrix} x_j^2 & x_j y_j \\ x_j y_j & y_j^2 \end{pmatrix}, \\
 (3) \quad \mathbf{M}_{j+1} &= \mathbf{M}_j + \mathbf{G}_{j+1}^{-1} \begin{pmatrix} x_j \\ y_j \end{pmatrix} \begin{pmatrix} x_{j+1} - \hat{x}_{j+1} & y_{j+1} - \hat{y}_{j+1} \end{pmatrix}.
 \end{aligned}$$

Saadut yhtälöt ovat helposti käsiteltäviä, sillä kaikki matriisit ovat enintään kokoa  $2 \times 2$ . Jopa käänteismatriisi on helppo laskea millä tahansa ohjelmoitavalla laitteella!

### Parannusyritys.

Yritämme vielä parantaa ennustetta. Häiriöiden vaikutusta saadaan vähennettyä, jos selitysmalli estimoidaan siten, että

$$\vec{x}_j = (\vec{x}_{j-2} \quad \vec{x}_{j-1}) \mathbf{M} + \vec{u}_j \mathbf{B} \quad (11.19)$$

$$\begin{aligned}
 \Leftrightarrow \begin{pmatrix} x_j & y_j \end{pmatrix} &= \begin{pmatrix} x_{j-2} & y_{j-2} & x_{j-1} & y_{j-1} \end{pmatrix} \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \\ m_{31} & m_{32} \\ m_{41} & m_{42} \end{pmatrix} \\
 &\quad \cdots + \begin{pmatrix} v_j & w_j \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}
 \end{aligned} \quad (11.20)$$

$$\Leftrightarrow \begin{cases} x_j = \vec{x}_{j-2} \begin{pmatrix} m_{11} \\ m_{21} \end{pmatrix} + \vec{x}_{j-1} \begin{pmatrix} m_{31} \\ m_{41} \end{pmatrix} + \vec{u}_j \begin{pmatrix} b_{11} \\ b_{21} \end{pmatrix} \\ y_j = \vec{x}_{j-2} \begin{pmatrix} m_{12} \\ m_{22} \end{pmatrix} + \vec{x}_{j-1} \begin{pmatrix} m_{32} \\ m_{42} \end{pmatrix} + \vec{u}_j \begin{pmatrix} b_{12} \\ b_{22} \end{pmatrix} \end{cases}.$$

Askeleella  $j$  matriisi  $\mathbf{M}$  estimoidaan yhtälöstä

$$\underbrace{\begin{pmatrix} x_1 & y_1 & x_2 & y_2 \\ x_2 & y_2 & x_3 & y_3 \\ \vdots & \vdots & \vdots & \vdots \\ x_{j-2} & y_{j-2} & x_{j-1} & y_{j-1} \end{pmatrix}}_{=\mathbf{A}_j} \mathbf{M}_j = \underbrace{\begin{pmatrix} x_3 & y_3 \\ x_4 & y_4 \\ \vdots & \vdots \\ x_j & y_j \end{pmatrix}}_{=\mathbf{Y}_j} - \underbrace{\begin{pmatrix} v_3 & w_3 \\ v_4 & w_4 \\ \vdots & \vdots \\ v_j & w_j \end{pmatrix}}_{=\mathbf{U}_j} \mathbf{B} \quad (11.21) \quad \text{c5s1eq2a}$$

josta  $\mathbf{M}_j$  ratkaistaan yhtälöstä

$$\mathbf{M}_j = (\mathbf{A}_j^T \mathbf{A}_j)^{-1} \mathbf{A}_j^T (\mathbf{Y}_j - \mathbf{U}_j \mathbf{B}). \quad (11.22)$$

Seuraavan tilan ennuste saadaan kaavalla

$$\begin{pmatrix} \hat{x}_{j+1} & \hat{y}_{j+1} \end{pmatrix} = \begin{pmatrix} x_{j-1} & y_{j-1} & x_j & y_j \end{pmatrix} \mathbf{M}_j + \begin{pmatrix} v_{j+1} & w_{j+1} \end{pmatrix} \mathbf{B}. \quad (11.23) \quad \text{c5s1eq2b}$$

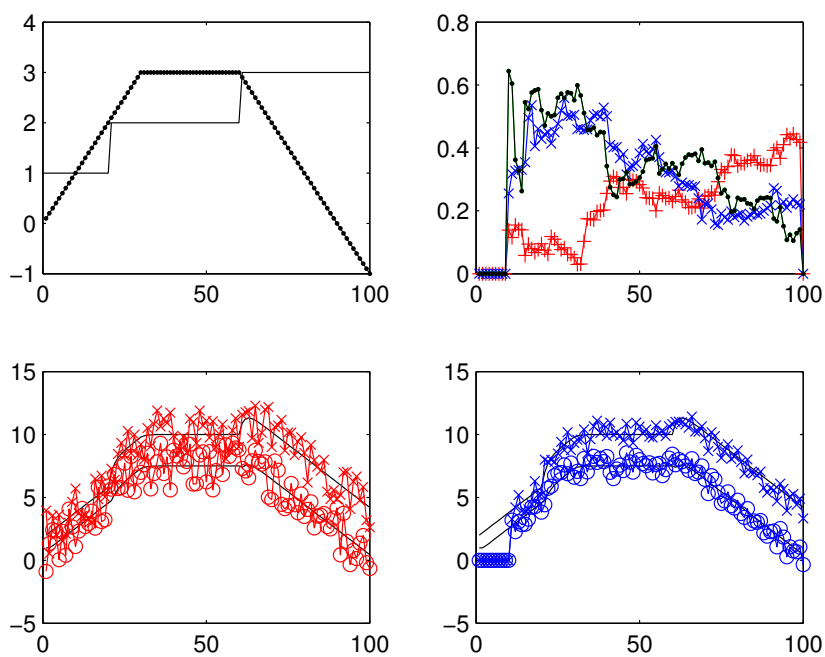
c5s1esim2

**Esimerkki 59** Kuvassa (20) on edellisessä esimerkissä (58) esitetyn systeemin aikasarjat estimoituina kaavoilla (II.21) ja (II.23). Osakuvassa (1) on ohjaussignaali  $v_t$  ja  $w_t$ . Osassa (2) on osa matriisin  $\mathbf{M}$  koordinaateista estimoinnin aikana ( $m_{11}$ ,  $m_{12}$ ,  $m_{13}$  ja  $m_{14}$ ). Osassa (3) ovat havaitut arvot  $x_t$  ( $\times$ ) ja  $y_t$  ( $\circ$ ). Osassa (4) ovat estimoidut arvot  $\hat{x}_t$  ( $\times$ ) ja  $\hat{y}_t$  ( $\circ$ ). Merkillepantavaa on nyt se, ettei malli tässäkin tapauksessa vakiintunut estimoinnin aikana. estimaatit ovat hyviä, mutta eivät ole parempia kuin esimerkissä (58). Mallien vertailu vaatisi nyt useita peräkkäisiä ajoja erilaisilla häiriöaikasarjoilla. Se malli, jonka antamissa tuloksissa on vähemmän hajontaa on parempi. Jätämme nyt tämän tekemättä. Nopea arvio kuvista (19) ja (20) on se, että parannusyritys ei johtanut yhtään parempaan tulokseen. Silloin yksinkertaisempi malli on parempi.

11.7 Sileän funktion  $f(\vec{x})$  minimikohta

11.8 Analytic Hierarchy Process (AHP)

11.9 Kalman suodin



**Kuva 20.** Esimerkin <sup>|c5s1esim2</sup>(59) aikasarjat. (1)  $v_t$  (.),  $w_t$  (—) (2)  $m_{11}$  (+),  $m_{12}$  ( $\times$ ),  $m_{21}$  ( $\cdot$ ) (3)  $x_t$  ( $\times$ ),  $y_t$  (o) (4)  $\hat{x}_t$  ( $\times$ ),  $\hat{y}_t$  (o).

c5s1fig5

## 12 LIITTEET

Ch:Liitteet

### 12.1 Liite 1. Osaamistavoitteet

Suoritettuaan opintojakson opiskelija osaa:

**Vektorit** Opiskelija osaa käyttää matematiikassa tavallista vektorimerkintää  $\mathbf{v} = \begin{pmatrix} 1 & 2 & 3 \end{pmatrix}'$  sekä fysiikassa tavallista vastaavaa merkintää  $\vec{v} = \vec{i} + 2\vec{j} + 3\vec{k}$ .

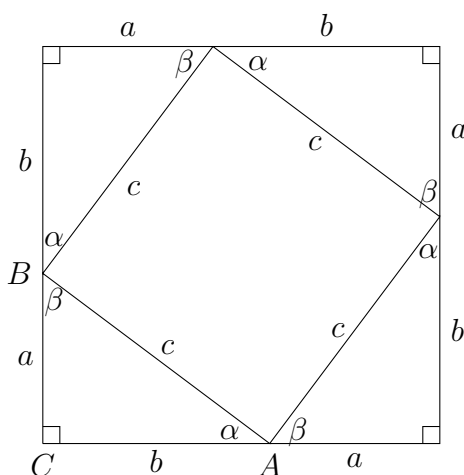
### 12.2 Liite 2. Todistuksia

Osa opetusmonisteen todistuksista on siirretty tähän liitteeseen, jotta opetusmonisteen leipäteksti olisi kevyempää. Jos toisin ei ilmoiteta, niin tämän kappaleen todistuksia ei tarvitse osata kokeessa.

#### 12.2.1 Trigonometriaa

**Lause 60 (Pythagoran lause)** Suorakulmaisessa kolmiossa hypotenuusan pituuden neliö on kateettien pituuksien neliöiden summa.

*Perustelu:* Piirretään suorakulmainen kolmio  $\triangle ABC$  ja kolme sen kanssa yhdenmuotoista kopiota oheisen kuvan mukaisesti.

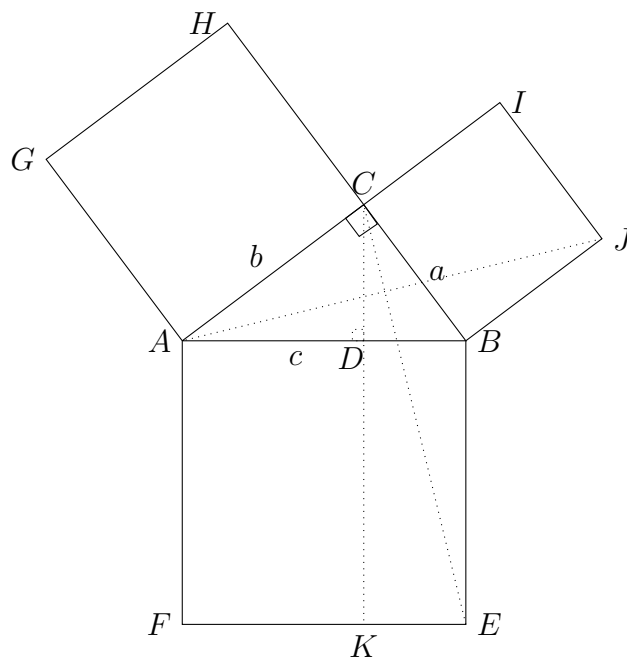


Kuvion ulkoreuna muodostaa neliön, jonka sivu on  $a + b$  ja kuvion sisään muodostuu pienempi neliö, jonka sivu on  $c$ . Nyt on selvää, että pienen neliön pinta-ala plus neljän kolmion pinta-alat ovat yhteensä ison neliön pinta-ala.

$$\begin{aligned} c^2 + 4 \cdot \frac{1}{2} \cdot a \cdot b &= (a + b)^2 \\ \Leftrightarrow c^2 + 2ab &= a^2 + 2ab + b^2 \\ \Leftrightarrow c^2 &= a^2 + b^2. \end{aligned}$$

□

*Perustelu 2:* Seuraava on klassinen (Eukleiden) perustelu ja saattaa olla koulusta tuttu. Piirretään suorakulmainen kolmio  $\triangle ABC$  niin, että hypotenuusa  $c$  on kantana. Lisätään kuvaan kolmion korkeusjana  $\overline{CD}$ . Piirretään kolmion jokaisen sivun viereen neliöt  $\square ABEF$ ,  $\square CAGH$  ja  $\square BCIJ$ . Jatketaan korkeusjanaa siten, että se jakaa ison neliön kahteen suorakulmioon. Lisätään kuvaan vielä janat  $\overline{EC}$  ja  $\overline{JA}$ .



Nyt kolmiolla  $\triangle BED$  ja kolmiolla  $\triangle BEC$  on samat pinta-alat, sillä niillä on sama kanta  $\overline{BE}$  ja sama korkeus  $\overline{BD}$ . Kolmio  $\triangle BAJ$  saadaan kolmiosta  $\triangle BED$  kiertämällä sitä pisteen  $B$  ympäri  $90^\circ$  myötäpäivään. Kolmioilla on siis samat pinta-alat. Lopulta kolmiolla  $\triangle JBA$  on sama pinta-ala kuin kolmiolla  $\triangle JBC$  koska niillä on sama kanta  $\overline{JB}$  ja sama korkeus  $\overline{BC}$ . Viimeisen kolmion pinta-ala on  $a^2/2$ . Edeltävän perusteella suorakulmion  $\square BEKD$  pinta-ala on  $a^2$ . Vastaavasti voimme osoittaa, että suorakulmion  $\square DKFA$  pinta-ala on  $b^2$ . Laskemalla osien pinta-alat yhteen, saamme

$$c^2 = a^2 + b^2.$$

□



## 12.3 Liite 3. Esimerkkien ohjelmakoodit

### 12.3.1 Ohjelmointikielten Octave, Python, Java ja C vertailua

**Octave** on tulkkaava kieli, jolla voidaan helposti tehdä nopeita kokeiluja ja laskelmia. Ohjelmiston valmiit rutiinit ovat niin hyvin suunniteltuja ja optimoituja, että käyttäjän kannalta kieli on matriisilaskuissa hyvin nopea.

- Octavella tehdyt ohjelmat toimivat pienin muutoksin MatLab-ympäristössä. MatLab on teollisuudessa eniten käytetty teknisen laskennan ohjelmisto, joten kokemus Octave-ohjelmoinnista antaa hyviä valmiuksia työelämään.
- Octaven versiosta 3.8 alkaen jakeluun liittyy graafinen käyttöliittymä (GUI). Myös Scilab tarjoaa MatLabin kaltaisen käyttöliittymän Octaveen.
- Octavella voi helposti ja tehokkaasti tuottaa hyvää tiede-grafiikkaa.
- Kurssilla käytetyt aliohjelmistot: Matplotlib.
- Octavea kehittää yhteisö, ja se on vapaa ohjelmisto (GNU General Public License).

**Python** on tulkkaava kieli. Pythonia käytetään paljon scriptaus-kielenä web-ohjelmoinnissa. Pythonin käyttäjä/kehittäjä -yhteisö on hyvin laaja. Modulikirjasto on laaja ja kehittyy edelleen.

- Pythonia varten on olemassa kehitysympäristöjä (Spyder yms.), jotka muistuttavat MatLabin käyttöliittymää.
- Pythonilla voi helposti ja tehokkaasti tuottaa hyvää tiede-grafiikkaa. Pythonilla voi tehdä pelitason liikkuvaa kuvaa (ei ole välttämättä helppoa).
- Kurssilla käytetyt työkalut: Spyder, Numpy, Scipy, Matplotlib.
- Lähellä Pythonia kehittyy tällä hetkellä kieli nimeltä Julia. Julia on pythonin muunnelma, joka on lähes yhtä nopea kuin C. Julia tai sen kaltainen muu kieli tulee jatkossa vaikuttamaan Pythonin tehostumiseen. Tällä hetkellä ”Suomen tieteellinen laskenta” CSC järjestää kursseja Pythonin käytöstä tieteellisessä teholaskennassa.
- Pythonia kehittää suuri yhteisö, ja se on vapaa ohjelmisto (Python is free to use, even for commercial products, All Python releases are Open Source (see <http://www.opensource.org/>)).

**Java** on kääntävä kieli. Java kehitys-työkalut on saatavana kaikille koneille ja käyttöjärjestelmille. Java-ohjelma toimii ilman uudelleen kääntämistä kaikissa kone-ympäristöissä. Java ohjelmoinnin alkeiden oppiminen avaa oven Android -ohjelmointiin.

- Javaa varten on olemassa kehitysympäristöjä (Eclipse, IntelliJ IDEA, yms.), jotka helpottavat suuresti Java-ohjelmointia.
- Javalla ei ole helppo tuottaa hyvää tiede-grafiikkaa, mutta sillä voi tehdä web-sivuilla toimivia Java-appletteja.
- Kurssilla käytetyt työkalut: BLAS (Lapack).
- Javaa kehittää ”Sun Microsystems”, joka ilmoittaa, että Java on vapaa ohjelmisto (GNU General Public License).

**C** on kääntävä nopea kieli, jolla voidaan viitata suoraan muuttujan muistiosoitteeseen. Tieteellisen laskennan perusohjelmat ovat 1970-luvulta alkaen olleet Fortran ja C. Viimeisin standardi on C11 (vuodelta 2011). C ohjelmoinnin luonnollinen ympäristö on Linux ja GCC-kääntäjä (GNU Compiler Collection). GCC kääntää ohjelman, jos se on kirjoitettu jollakin tunnetulla ohjelmointikielellä.

- Usein C:llä toteutetaan aliohjelmat, joiden tulee toimia nopeasti. Nämä rutiinit sitten linkitetään osaksi jollakin toisella kielellä tehtyä isoa ohjelmaa. Kääntäjiä kehitetään ja optimoidaan jatkuvasti tätä varten. Itse C kieli ei juurikaan enää muutu.
- C:llä ei ole alunperin ollut tarkoitus tuottaa grafiikkaa. Jos grafiikkaa halutaan tuottaa, niin C:n avulla luodaan tiedosto, joka sisältää piirtämiseen tarvittavat numerot ja määrittelyt ja jollain toisella ohjelmistolla luodaan kuva. Tässä monisteessa grafiikka tehdään ’gnuplot’-ohjelmistolla.
- Kurssilla käytetyt työkalut:

### 12.3.2 Esimerkiohjelmat Python kielellä

**Esimerkki** <sup>Code:es01</sup> **??(Python)**: Tee ohjelma joka tuottaa Fahrenheit/Celsius -muunnostaulukon. Jos  $t_F$  on lämpötila Farenheit asteissa ja  $t_C$  on vastaava lämpötila Celsius asteissa, niin

$$t_C = \frac{5}{9}(t_F - 32).$$

(Jotta tuloste ei ole liian pitkä, teemme taulukosta melko karkean. Voit itse muuttaa ohjelmaa siten, että se tekee tarkemman muunnostaulukon.)

```

print("Farenheit,      Celsius")
for n in range(1, 11):
    tf = 10*n
    tc = 5.0/9.0*(tf-32.0)
    print('t_F = {0:6.2f},    t_C = {1:6.2f}'.format(tf, tc
    ))

```

Ajo:

```

mla:~/tex/la$ python laEs01.py
Farenheit,      Celsius
t_F =  10.00,    t_C = -12.22
t_F =  20.00,    t_C =  -6.67
t_F =  30.00,    t_C =  -1.11
t_F =  40.00,    t_C =   4.44
t_F =  50.00,    t_C =  10.00
t_F =  60.00,    t_C =  15.56
t_F =  70.00,    t_C =  21.11
t_F =  80.00,    t_C =  26.67
t_F =  90.00,    t_C =  32.22
t_F = 100.00,    t_C =  37.78
mla:~/tex/la$

```

**Esimerkki <sup>Code:es02</sup>??(Python):** Piirrä funktion  $f(x) = x^2 - 2x$  kuvaaja välillä  $0 \leq x \leq 3$ . Tee me kuvasta hiukan karkean. Voit parantaa kuvaa.

```

from numpy import *
import matplotlib.pyplot as plt

N = 8; a = 0.0; b = 3.0;
x = zeros((N,1),float)
y = zeros((N,1),float)
for k in range(0,N):
    x[k] = a + k*(b-a)/N
    y[k] = x[k]*x[k]-2*x[k]

plt.plot(x,y,'k-o');
plt.show()

```

Ohjelman ajo tuottaa kuvan <sup>lafig\_Es02py</sup> 21

### 12.3.3 Esimerkkiohjelmat Java kielellä

**Esimerkki <sup>Code:es01</sup>??(Java):** Tee ohjelma joka tuottaa Fahrenhei/Celsius -muunnostaulukon. Jos  $t_F$  on lämpötila Farenheit asteissa ja  $t_C$  on vastaava lämpötila Celsius asteissa, niin

$$t_C = \frac{5}{9}(t_F - 32).$$



Käännös ja ajo:

```
m1a:~/tex/la$ javac laEs01.java
m1a:~/tex/la$ java laEs01
Fahrenheit,    Celsius
t_F = 10,00,   t_C = -12,22
t_F = 20,00,   t_C = -6,67
t_F = 30,00,   t_C = -1,11
t_F = 40,00,   t_C = 4,44
t_F = 50,00,   t_C = 10,00
t_F = 60,00,   t_C = 15,56
t_F = 70,00,   t_C = 21,11
t_F = 80,00,   t_C = 26,67
t_F = 90,00,   t_C = 32,22
t_F = 100,00,  t_C = 37,78
m1a:~/tex/la$
```

### 12.3.4 Esimerkkiohjelmat C kielellä

**Esimerkki <sup>Code:es01</sup>??(C):** Tee ohjelma joka tuottaa Fahrenheit/Celsius -muunnostaulukon. Jos  $t_F$  on lämpötila Farenheit asteissa ja  $t_C$  on vastaava lämpötila Celsius asteissa, niin

$$t_C = \frac{5}{9}(t_F - 32).$$

(Jotta tuloste ei ole liian pitkä, teemme taulukosta melko karkean. Voit itse muuttaa ohjelmaa siten, että se tekee tarkemman muunnostaulukon.)

```
#include<stdio.h>
main()
{
    double fahr;
    printf("Fahrenheit,    Celsius \n");
    for(fahr=10.0; fahr <= 100; fahr = fahr + 10.0)
        printf("%6.2f %6.2f\n", fahr, (5.0/9.0)*(fahr-32));
}
```

Käännös ja ajo:

```
m1a:~/tex/la$ gcc laEs01.c -o laEs01
m1a:~/tex/la$ ./laEs01
Fahrenheit,    Celsius
10.00 -12.22
20.00 -6.67
30.00 -1.11
40.00 4.44
50.00 10.00
```

```

60.00  15.56
70.00  21.11
80.00  26.67
90.00  32.22
100.00 37.78
mla~/tex/la$

```

**Esimerkki 1.1(C):** Piirrä funktion  $f(x) = x^2 - 2x$  kuvaaja välillä  $0 \leq x \leq 3$ . Teemme kuvasta hiukan karkean. Voit parantaa kuvaa.

```

1  #include <stdio.h>
2  main()
3  {
4      int N = 7, k;
5      double a = 0.0, b = 3.0;
6      double x[N], y[N];
7
8      for (k=0;k<N;k++) {
9          x[k] = a + k*(b-a)/(N-1);
10         y[k] = x[k]*x[k]-2*x[k];
11     }
12
13     FILE *fp1;
14     fp1 = fopen("job.dat", "w");
15     for (k=0;k<N;k++) {
16         fprintf(fp1, "%5.2f, %5.2f\n", x[k], y[k]); }
17     fclose(fp1);
18
19     FILE *fp2;
20     fp2 = fopen("job.gp", "w");
21     fprintf(fp2, "set title 'Esimerkki 1aEs02'\n");
22     fprintf(fp2, "set xlabel 'x'\n");
23     fprintf(fp2, "set ylabel 'y'\n");
24     fprintf(fp2, "set yrange [-2:4]\n");
25     fprintf(fp2, "unset key\n");
26     fprintf(fp2, "set grid\n");
27     fprintf(fp2, "plot 'job.dat' with linespoints\n");
28     fclose(fp2);
29
30     system("gnuplot -persist 'job.gp'");
31 }

```

Ohjelmassa on selvästi kolme osaa.

(osa 1:) Rivit 4–6 määrittelevät muuttujat.  $x$  ja  $y$  ovat vektoreita eli taulukoita (array), ja kumpikin sisältää  $N$  kappaletta liukulukuja (double).

Rivit 8–11 sisältävät toistorakenteen, jossa vektoreiden  $x$  ja  $y$  alkiot lasketaan.

(osa2:) Rivit 13–17 avaavat tiedoston 'job.dat' kirjoittamista varten ja kirjoittavat datan sinne. Lopuksi tiedosto suljetaan. Tiedoston sisältö on:

```
0.00, 0.00
0.50, -0.75
1.00, -1.00
1.50, -0.75
2.00, 0.00
2.50, 1.25
3.00, 3.00
```

(osa3:) Rivit 19–28 luovat gnuplot-komentotiedoston 'job.gp', jonka sisältö on

```
set title 'Esimerkki laEs02'
set xlabel 'x'
set ylabel 'y'
set yrange [-2:4]
unset key
set grid
plot 'job.dat' with linespoints
```

Lopullinen kuva saadaan aikaan rivin 30 komennolla, joka käynnistää gnuplot ohjelman, joka tuottaa komentotiedoston job.gp mukaisen kuvan. Vipu -persist saa aikaan sen, että kuva jää konsolille sen jälkeenkin kun pääohjelman ajo on päättynyt.

Ajo:

```
gcc laEs02.c -o laEs02
./laEs02
```

piirtää kuvan  22, s. 139. Kuvan voi kopioida leikepöydälle, josta sen saa kopioitua omiin dokumentteihin. Jos haluat kuvan tiettyyn formaattiin, esimerkiksi png-muotoon, niin lisää kolmanteen osioon rivit

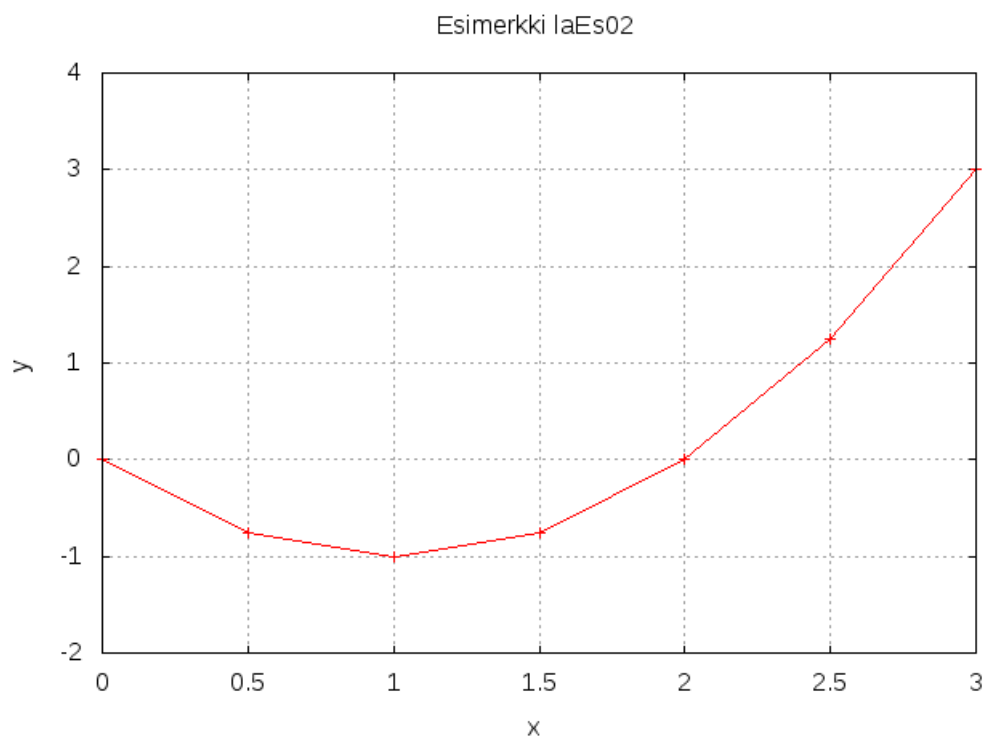
```
fprintf(fp2, "set term png\n");
fprintf(fp2, "set output 'laEs02c.png'\n");
```

Lisäyksen jälkeen voidaan -persist -vipu poistaa riviltä 30.

## 12.4 Liite 5. Lähdemateriaalia ohjelmointikielistä

Sec:CodeMat

Tässä opetusmonisteessa on melko paljon eri ohjelmointikielillä kirjoitettuja esimerkki-ohjelmia. Useimmat ohjelmat ovat ankean yksinkertaisia. Opiskelijalle voi tulla tarve tehdä niistä parempia. Tätä varten tähän kappaleeseen on kerätty linkkejä verkosta löytyviin materiaaleihin, jotka kirjoittaja itse on kokenut hyödyllisiksi.



lafig\_Es02c

**Kuva 22.** Ohjelmaesimerkin `Code:es02` piirtämä kuva.

#### 12.4.1 Octave

<https://www.gnu.org/software/octave/octave.pdf> Octave versio 3.8  
(February 2011) käsikirja. 900 sivua.

#### 12.4.2 Python

#### 12.4.3 C

#### 12.4.4 Java



lastpage